



Специфікація використання

**API COM Cashalot**

v.5.8

# Зміст

## Зміст

Поняття та визначення

Загальна інформація

Апаратні та програмні вимоги

API COM PPPO Cashalot

1. Встановлення налаштувань API SetParameter (Name, Value)
2. Встановлення налаштувань API SetParameterEx (FiscalNumberRRO, Name, Value)
3. Отримання номеру версії API бібліотеки GetVersion
4. Відкриття зміни OpenShift (FiscalNumberRRO)
5. Фіскалізація чека продажу товарів, послуг FiscalizeCheck (FiscalNumberRRO, JSONGoodsLstData, JSONPayData)
6. Фіскалізація чека повернення товарів, послуг FiscalizeReturnCheck (FiscalNumberRRO, JSONGoodsLstData, JSONPayData, FiscalNumReturnReceipt)
7. Отримання локального номера чека за його фіскальним номером GetReceiptLocalNumberByFiscalNumber (FiscalNumberRRO, ReceiptFiscalNumber)
8. Отримання фіскального номера чека за його локальним номером GetReceiptFiscalNumberByLocalNumber (FiscalNumberRRO, ReceiptLocalNumber)
9. Отримання xml структури фіскального чека GetReceiptXML (FiscalNumberRRO, ReceiptFiscalNumber)
10. Отримати перший чек з періоду GetFirstReceiptByPeriod (FiscalNumberRRO, dateBeg, dateEnd)
11. Отримати перший чек по зміні GetFirstReceiptByShift (FiscalNumberRRO, shiftID)
12. Отримати наступний чек з періоду GetNextReceipt (FiscalNumberRRO)
13. Отримати інформацію по поточному стану каси GetCurrentStatus (FiscalNumberRRO)
14. Отримати перший Z-звіт за період GetFirstZReportByPeriod (FiscalNumberRRO, dateBeg, dateEnd)
15. Отримати наступний Z-звіт з періоду GetNextZReport (FiscalNumberRRO)
16. Отримати Z-звіт касової зміни GetZReportByShift (FiscalNumberRRO, ShiftID)
17. Вивести візуальне відображення чека на екран ShowReceipt (FiscalNumberRRO, ReceiptFiscalNumber)
18. Формування звіту про реалізовані товари PrintReportGoods (FiscalNumberRRO)
19. Друкувати періодичний звіт по датам PrintPReportDate (FiscalNumberRRO, dateBeg, dateEnd, isShort)
20. Надрукувати довільний нефіскальний текст PrintTextDocument (FiscalNumberRRO, DocumentPackage)
21. Службове внесення грошей в касу ServiceInput (FiscalNumberRRO, Amount)
22. Службове внесення грошей в касу ServiceInputEx (FiscalNumberRRO, Amount)
23. Службова видача грошей з каси ServiceOutput (FiscalNumberRRO, Amount)
24. Службова видача грошей з каси ServiceOutputEx (FiscalNumberRRO, Amount)
25. Отримання опису останньої помилки GetLastError()
26. Формування X-звіту PrintXReport (FiscalNumberRRO)
27. Формування X-звіту GetXReport (FiscalNumberRRO, ShowPrintReport)
28. Отримати останній Z-звіт GetLastZReport (FiscalNumberRRO, ShowPrintReport)
29. Формування Z-звіту PrintZReport (FiscalNumberRRO)
30. Закриття поточної зміни CloseShift (FiscalNumberRRO)

31. Метод ручного переведення ПРРО в режим офлайн SetOfflineMode (FiscalNumberRRO, AutoExitToOnline)
32. Метод ручного повернення ПРРО в режим онлайн TryGoToOnlineMode (FiscalNumberRRO)
33. Синхронізація залишків по товарам SyncBalanceOnGoods (FiscalNumberRRO)
34. Синхронізація товарів та залишків SyncGoodsWithBalance (FiscalNumberRRO)
35. Метод сторнування чеків продажу/повернення StornoCheck (FiscalNumberRRO)
36. Метод відображення QR-коду зареєстрованого чека ShowReceiptQRCode (FiscalNumberRRO, ReceiptFiscalNumber)
37. Метод відправки чека покупцю за допомогою сервісів доставки SendCheckToCustomer (FiscalNumberRRO, ReceiptFiscalNumber, SendingAddressType, SendingAddress)
38. Отримання переліку налаштованих пристроїв POS-терміналів в програмі GetPOSTerminalList (FiscalNumberRRO)
39. Встановлення платіжного POS-терміналу пристроєм за замовчуванням SetDefaultPOSTerminal (FiscalNumberRRO, ID)
40. Проведення транзакції через POS-термінал PayByPaymentCard (FiscalNumberRRO, SumPayByCard, OtherParams)
41. Повернення транзакції (оплати) через POS-термінал ReturnPaymentByPaymentCard (FiscalNumberRRO, SumPayByCard, RRN, OtherParams)
42. Скасування транзакції (оплати) через POS-термінал CancelPaymentByPaymentCard (FiscalNumberRRO, SumPayByCard, InvoiceNumber, OtherParams)
43. Проведення транзакції через функцію "Оплата частинами" PayPartPaymentByPaymentCard (FiscalNumberRRO, SumPayByCard, CountParts, OtherParams)
44. Повернення транзакції по операції "Оплата частинами" ReturnPartPaymentByPaymentCard (FiscalNumberRRO, SumPayByCard, RRN, IsPartialReturn, ServiceDogNum, OtherParams)
45. Створення відкладеного чека продажу товарів/послуг CreatePostponedReceipt (FiscalNumberRRO, JSONGoodsLstData, JSONPayData)
46. Фіскалізація відкладеного чеку продажу товарів/послуг FiscalizePostponedReceipt (FiscalNumberRRO, ReceiptID, OtherParam)
47. Отримати інформацію про відкладений чек за його ReceiptID GetPostponedReceiptByID (FiscalNumberRRO, ReceiptID)
48. Отримати інформацію про відкладений чек за його номером ЕН (експрес-накладної) GetPostponedReceiptByWaybillNumber (FiscalNumberRRO, ExpressWaybillNumber)
49. Отримати інформацію про перший відкладений чек GetFirstPostponedReceipt (FiscalNumberRRO, StartDate, EndDate)
50. Отримати інформацію наступний відкладений чек GetNextPostponedReceipt (FiscalNumberRRO)
51. Відображення візуальної форми чека за його ReceiptID ShowReceiptByID (FiscalNumberRRO, ReceiptID)

#### Ломбардні операції

Особливості роботи з чеками типу «Інтернет-продаж» при доставці Новою поштою

Загальний опис

Створення відкладеного чека

Отримання інформації про відкладений чек

Реєстрація відкладеного чека

Особливості реєстрації відкладеного чека «Інтернет-продажу» в залежності від статусу ЕН(ТТН) та визначеного способу оплати по чеку

Приклади використання засобів оплат у чеку (спрощений та розширений варіант)

Рекомендований порядок роботи з CashalotApi

Приклад реалізації підключення COM бібліотеки на C++

Лістинг

## Поняття та визначення

**КЕП** – кваліфікований електронний підпис.

**Локальний номер чека** – номер, призначений чеку системою виписки електронних касових чеків, що використовується продавцем.

**Офлайн сесія** – сукупність документів, створених в режимі офлайн, між припиненням і відновленням зв'язку ПРРО з ФСКО.

**ПРРО** – програмний реєстратор розрахункових операцій. Система виписки електронних касових чеків, що використовується продавцем.

**Фіскальний номер чека (ФН)** – номер, призначений чеку сервером ФСКО.

**ФСКО** – фіскальний сервер контролюючого органу.

## Загальна інформація

**API COM Cashälot** використовується для інтеграції ПРРО з розрахунково-обліковими системами. В даному документі приведено специфікацію методів бібліотеки API COM Cashälot для здійснення підключення з обліковими системами.

Інтеграція надає користувачам можливість, безпосередньо з розрахунково-облікової системи, скористатися наступним функціоналом програмного забезпечення Cashälot:

- Відкриття та закриття касової зміни;
- Створення та реєстрація чеків продажу;
- Створення та реєстрація чеків зі способами оплати передоплата, післяплата, інтернет-продажа, повна післяплата, оплата частинами;
- Створення та реєстрація чеків ломбардних операцій;
- Створення видаткового чека на основі чека продажу та його реєстрація;
- Сторнування чеків;
- Формування та друк Z-звіту за даними фіскального сервера контролюючого органу (скорочено ФСКО)/X-звіту за даними ПРРО;
- Створення та реєстрація службових чеків;
- Службове внесення;
- Службова видача;
- Робота в режимі офлайн;
- Можливість відображення на екрані зареєстрованого чека на пристрої продавця з QR-кодом після успішної реєстрації;
- Друк зареєстрованого чека;
- Формування періодичних звітів.
- Відправка зареєстрованого чека покупцю за допомогою сервісів доставки Viber, SMS або E-mail.

## Апаратні та програмні вимоги

Для оптимальної роботи ПРРО Cashälot рекомендовані наступні системні та програмні вимоги:

### Операційна система

Програмний комплекс коректно функціонує на комп'ютерах з сучасними операційними системами Microsoft та компонентами, що входять до її складу. Рекомендовано використовувати останню версію ОС, наприклад, Windows 10 або Windows Server 2019 відповідно до потреб, з встановленими актуальними оновленнями.

### Апаратне забезпечення:

- процесор з мінімальною тактовою частотою від 2 ГГц;
- оперативна пам'ять від 4 GB;
- вільне місце на жорсткому диску від 1,5 GB;
- кольоровий графічний дисплей;
- маніпулятор типу миша та клавіатура;
- встановлений український або російський мовний стандарт регіональних налаштувань операційної системи;
- підтримка кирилиці в операційній системі;
- доступ до поштового сервера;
- наявний зв'язок з фіскальним сервером контролюючого органу.

### Мережа

Для роботи користувача необхідний доступ до мережі інтернет. Також, необхідно мати доступ до кабінету Cashälot, фіскального сервера контролюючого органу та до ресурсу центру сертифікації ключів. Наприклад, якщо Ви використовуєте ключі від «АЦСК Україна», то необхідно надати доступ до ресурсу: <https://uakey.com.ua/> - 443. Якщо ж ви використовуєте КЕП (ЕЦП) інших кваліфікованих надавачів (АЦСК), то необхідно надати доступ до серверів TSP, OCSP, CMP відповідних центрів.

### Загальний список ресурсів:

<https://load.cashalot.org.ua/update/> - 443 - Сервіс завантаження інсталяцій програми (дистрибутиви, оновлення);

<https://fs.tax.gov.ua:8643/fs/cmd> - Сервіс API ДПС;

<https://my.cashalot.org.ua/> - 443 - Особистий кабінет Cashälot.

<https://api.novaposhta.ua> - API Нової Пошти

# API COM ПРРО Cashalot

**API COM ПРРО Cashalot** призначений для підключення до сторонніх розрахунково - облікових систем з використанням COM-методів.

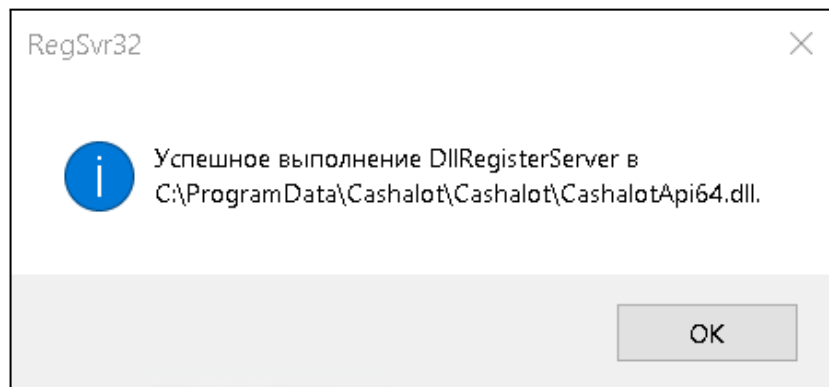
Використовуються бібліотеки **CashalotApi32.dll** та **CashalotApi64.dll** , що розміщені в кореневому каталозі ПРРО Cashalot. В даній специфікації приклади написані під VBS.

Перед початком роботи з CashalotApi потрібно:

- Зареєструватись в особистому кабінеті <https://my.cashalot.org.ua/> ;
- Виконати всі пункти з розділу «Початок роботи» в особистому кабінеті;
- Встановити ПРРО Cashalot, попередньо завантаживши дистрибутив з офіційного сайту. Виконати запуск програми Cashalot (в момент першого запуску завантажуються дані з бек-офісу). **Для паралельної роботи в декількох касах одночасно, необхідно встановлювати ПРРО Cashalot для кожної каси окремо;**
- Зареєструвати бібліотеки в ОС за допомогою команди regsvr32  
`regsvr32 "<Повний шлях до ПРРО Cashalot>\CashalotApi32.dll"`  
(для 32-бітної платформи)  
`regsvr64 "<Повний шлях до ПРРО Cashalot>\CashalotApi64.dll"`  
(для 64-бітної платформи)

## Наприклад:

В разі успішного виконання реєстрації бібліотеки командою regsvr32 "C:\ProgramData\Cashalot\Cashalot\CashalotApi64.dll" має з'явитись повідомлення наступного зразку.



## В бібліотеках CashalotApi реалізована обробка наступних методів

- SetParameter
- SetParametrEx
- GetVersion
- ServiceInput
- ServiceInputEx
- ServiceOutput
- ServiceOutputEx
- OpenShift
- FiskalizeCheck
- FiskalizeReturnCheck
- GetReceiptLocalNumberByFiscalNumber
- GetReceiptFiscalNumberByLocalNumber
- GetCurrentStatus
- GetReceiptXML
- SyncBalanceOnGoods
- GetFirstReceiptByPeriod
- SyncGoodsWithBalance
- GetFirstReceiptByShift
- GetNextReceipt
- GetFirstZReportByPeriod
- GetNextZReport
- GetZReportByShift
- ShowReceipt
- GetLastError
- PrintXReport
- GetXReport
- GetLastZReport
- PrintZReport
- CloseShift
- PrintPReportDate
- PrintTextDocument
- StornoCheck
- ShowReceiptQrCode
- SendCheckToCustomer
- GetPOSTerminalList
- SetDefaultPOSTerminal
- CreatePostponedReceipt
- FiskalizePostponedReceipt
- GetFirstPostponedReceipt
- GetNextPostponedReceipt
- GetPostponedReceiptById
- GetPostponedReceiptByWaybillNumber
- ShowReceiptById

Методи повертають відповідь у вигляді структури **CashalotApiRetVal**, в залежності від виконаного методу, можуть міститись такі дані:

| Властивість      | Тип    | Значення                                                                                                              |
|------------------|--------|-----------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Ознака успішності виконання.                                                                                          |
| Description      | string | Опис результату виконання або помилки.<br>Заповнюється у випадку коли:<br>Return = False.                             |
| JsonVal          | string | JSON-структура з переліком усіх можливих параметрів які може повернути метод (перелік параметрів залежить від метода) |
| ReceiptFiscalNum | int    | Фіскальний номер чека.                                                                                                |
| ReceiptLocalNum  | int    | Локальний номер чека.                                                                                                 |
| ShiftID          | string | Ідентифікатор касової зміни.                                                                                          |
| XML              | string | XML-структура документу представлена у Base64.                                                                        |
| Type             | int    | Тип документа.                                                                                                        |

1. Встановлення налаштувань API **SetParameter** (Name, Value)

Функція призначена для встановлення значень параметрів налаштувань API бібліотеки.  
Рекомендовано встановити всі параметри на початку роботи.

Кожен параметр повинен містити в собі:

| Ім'я  | Тип    | Опис               |
|-------|--------|--------------------|
| Name  | string | Ім'я параметра     |
| Value | string | Значення параметра |

Приклад використання **SetParameter**:

```
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
```

**SetParameter** складається з обов'язкових параметрів (виділені червоним) та не обов'язкових параметрів. Параметри повинні мати послідовність, згідно представленого опису параметрів. Тип всіх параметрів **string**, для використання параметрів з різними мовами програмування.

| Ім'я параметра (Name)    | Значення параметра (Value)                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>PathToCashalotDir</b> | Шлях до каталогу з встановленим ПРРО Cashälot.                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>DeviceIDFnRRO</b>     | Фіскальний номер ПРРО.                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>NOAUTOUPDATE</b>      | Перевірка на наявність нової версії на сервері <a href="https://load.cashalot.org.ua/update/">https://load.cashalot.org.ua/update/</a> та запит на оновлення програми Cashälot в момент початку роботи з API:<br><br>«True» – заборонити оновлення;<br>«False» – дозволити оновлення(за замовчуванням).                                                                                                                                                          |
| <b>NOINTERFACEMODE</b>   | Ознака використання інтерфейсу РМК для авторизації та службових операцій видачі та внесення готівки в касу.<br><br>«False» – при роботі з API викликається діалогові вікно.<br>«True» – авторизація відбувається шляхом передачі параметрів авторизації <b>PathToCertificate</b> та <b>PwdToCertificate</b> . Вікна службових операцій видачі та внесення готівки не викликаються (проте, результат операції службового внесення/видачі готівки відображається). |
| <b>PathToCertificate</b> | Шлях до каталогу з файловим ключем користувача.<br><br><b>Важливо!</b> В папці повинні міститися ключ і сертифікат тільки одного користувача. Якщо касою користуються декілька касирів, необхідно помістити їх сертифікати і ключі в окремі каталоги.                                                                                                                                                                                                            |
| <b>PwdToCertificate</b>  | Пароль від ключа                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

| Ім'я параметра (Name) | Значення параметра (Value)                                                                                                                                                                                                                                                                                                                                                    |
|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| USETOKEN              | <p>Чи використовується токен для авторизації.<br/> «True» – використовується.<br/> «False» – не використовується (використовується файловий ключ).</p> <p><b>Важливо!</b> Параметр USETOKEN має бути заданий після параметрів PathToCashalotDir, DeviceIDFnRRO та NOINTERFACEMODE. Для пришвидшення завантаження програми рекомендовано <b>НЕ</b> використовувати токени.</p> |
| USECLOUDKEY           | <p>Використання хмарного підпису CloudKey для авторизації.</p> <p>«True» – використовується.<br/> «False» – не використовується.</p> <p>Параметр USECLOUDKEY має бути заданий після параметрів PathToCashalotDir, DeviceIDFnRRO та NOINTERFACEMODE</p>                                                                                                                        |
| NOAUTOOPENSIFT        | <p>Перевірка стану зміни (відкрита або закрита). Якщо зміна закрита, при виклику будь-якої COM-функції зміна автоматично відкривається.</p> <p>«False» – автоматично відкривати зміну, якщо вона закрита.<br/> «True» – не відкривати автоматично зміну. В цьому випадку робота COM-функції буде перервана і результат виконання буде повернуто у вигляді помилки.</p>        |
| DEFAULTPRINTERNAME    | <p>Визначає принтер для друку при натисканні кнопки "Друк" в діалогових вікнах:</p> <p>1) «» – пустий рядок, виводиться діалог вибору принтера;<br/> 2) «DEFAULTWINDOWS» – використовувати принтер Windows за замовчуванням;<br/> 3) «ім'я принтера чеків» - в параметр передається ім'я принтера чеків, який буде використовуватись для друку чеків в ПРРО Cashalot.</p>     |
| AUTOPRINTMODE         | <p>Параметр визначає автоматичну відправку на друк чека або службового звіту без відображення інформаційного вікна:</p> <p>«True» – відправляти на друк;<br/> «False» – не відправляти на друк.</p> <p>Якщо задано принтер за замовчуванням (DEFAULTPRINTERNAME), чек друкується одразу, в іншому випадку відображається вікно вибору принтера для друку.</p>                 |
| NOPRINTMODE           | <p>Дозвіл на друк:</p> <p>«True» – заборонити друк;<br/> «False» – дозволити друк.</p>                                                                                                                                                                                                                                                                                        |

| Ім'я параметра (Name)    | Значення параметра (Value)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SHOWREPORTADDITIONALINFO | <p>Виводити додаткову інформацію в нижній частині Z-звіті. В разі встановленого значення "True", відображаються такі поля:</p> <ul style="list-style-type: none"> <li>- Залишок готівки в касі;</li> <li>- Сума знижки;</li> <li>- Сума націнки.</li> </ul> <p>«True» - відображає додаткову інформацію;<br/>«False» – не відображає додаткову інформацію.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| SPECIFICAPIMODE          | <p>Режим передбачений для інтеграції з обліковими системами, які розроблені з використанням мов програмування Pascal, Delphi та Xbase++.</p> <p>При використанні SPECIFICAPIMODE, в залежності від значення параметру, можливо використовувати режим без інтерфейсу та з інтерфейсом.</p> <p><b>Зверніть увагу!</b> Встановлення параметра SPECIFICAPIMODE та його значень <u>обов'язково</u> повинно бути виконано перед встановленням інших параметрів.</p> <p><b><u>Встановлення SPECIFICAPIMODE необхідно здійснювати один раз, перед запуском роботи з ПРРО. Зміна значень у процесі роботи не рекомендована.</u></b></p> <p>Параметр SPECIFICAPIMODE може приймати наступні значення:</p> <p>«0» - програма працює в стандартному режимі;</p> <p>«1» - переводить API COM Cashlot автоматично в режим роботи без інтерфейсу, який підтримує одночасну роботу тільки з одним екземпляром ПРРО. Встановлення параметрів підтримується тільки за допомогою SetParameter,</p> <p>«2» - переводить API COM Cashlot в оптимізований режим роботи з інтеграціями, який підтримує можливість одночасної роботи з декількома ПРРО та може працювати як в режимі без інтерфейсу, так і з увімкненим інтерфейсом. Встановлення параметрів для роботи з двома та більше ПРРО здійснюється за допомогою SetParameterEx. При необхідності використання тільки одного екземпляру ПРРО, параметри можливо встановити як за допомогою SetParameterEx так і SetParameter. Також, робота даного режиму забезпечується процесом CashlotReporter.exe, який буде автоматично запущено при виконанні операцій з ПРРО.</p> |
| USEGOODSSTORAGEMODE      | <p>Параметр USEGOODSSTORAGEMODE визначає використання ведення складського обліку по товарам та автоматичну синхронізацію залишків по ним при використанні API COM Cashlot за умови активованого налаштування в кабінеті користувача в розділі «Склад - Налаштування - Облік товарів у ПРРО». Параметр може приймати наступні значення:</p> <p>«0» - програма працює в стандартному режимі без ведення обліку товарів в API та синхронізації залишків по товарам з кабінетом користувача.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

| Ім'я параметра (Name)      | Значення параметра (Value)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                            | <p>«1» - режим передбачає використання автоматичної синхронізації залишків по товарам та інформування користувача окремим повідомленням про перевищення їх ліміту під час спроби реєстрації фіскального чека. Після появи повідомлення користувачу доступна можливість відмовитись від подальшої обробки чека або погодитись та продовжити реєстрацію, але таким чином залишки по товарам будуть мати від'ємне значення кількості.</p> <p>«2» - режим передбачає використання автоматичної синхронізації залишків по товарам та інформування користувача окремим повідомленням про перевищення їх ліміту без можливості продовжити обробку та реєстрацію чека.</p> <p>Візуалізація повідомлення про перевищення ліміту залишків з можливістю продовжити/перервати процедуру фіскалізації чека, використовуючи перший режим параметра USEGOODSSTORAGEMODE, буде виникати також при ознаці використання режиму роботи без інтерфейса (NOINTERFACEMODE=true).</p> |
| USEPRGPOSTERMINAL          | <p>Параметр відповідає за можливість проведення оплати платіжною картою через POS-термінал, який підключений в Cashalot в розділі «Налаштування - Додаткове обладнання - POS-термінал». Оплату на терміналі можливо здійснити після передачі чека на реєстрацію. В разі використання обраного параметра фіскальний чек буде містити всі необхідні обов'язкові реквізити банківського терміналу (ЕПЗ, RRN, еквайр та ін.).</p> <p>«True» - використовується POS-термінал, підключений в Cashalot для оплати під час передачі чека на фіскалізацію з обраною формою оплати Картка (SumPayByCard) .</p> <p>«False» - під час фіскалізації чека з формою оплати Картка підключений POS-термінал в Cashalot не використовується.</p> <p>В разі неуспішної транзакції на банківському терміналі операція реєстрації чека не відбувається.</p>                                                                                                                        |
| RATELETTERCREATEIFNOTEXIST | <p>Параметр відповідає за автоматичне створення податкової ставки та літерного позначення при реєстрації чеків продажу. Рекомендовано до використання, якщо в кабінеті Cashalot зайняті доступні комбінації податкової ставки та літери.</p> <p>“True” - відбувається автоматичне створення податкової ставки та літерного позначення в разі її відсутності в Cashalot</p> <p>“False” - використовуються наявні завантажені в Cashalot податкові ставки та їх літерні позначення.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| USESELTERMINALINFOWND      | <p>Параметр призначений для можливості ручного підтвердження термінальної операції користувачем за умови, якщо операція з POS-терміналом пройшла неуспішно (обірвався зв'язок термінала з Cashalot в момент проведення операції або ін.). В такому разі зв'явиться додаткове вікно за допомогою якого можливо підтвердити операцію або виконати запит результату операції з POS-термінала.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

| Ім'я параметра (Name)         | Значення параметра (Value)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                               | <p>Параметр працює за умови використання параметра <b>USEPRGPOSTERMINAL</b>. Вікно підтвердження буде з'являтися навіть якщо використовується параметр <b>NOINTERFACEMODE</b>.</p> <p>"True" - після отримання помилки від POS-терміналу при проведенні операції з'явиться додаткове вікно з можливістю ручного підтвердження операції.</p> <p>"False" - в разі отримання помилки від POS-терміналу необхідно буде повторити спробу виконання операції.</p>                                                                                                  |
| <b>AUTOCLOSETERMINALSHIFT</b> | <p>Параметр призначений для закриття термінальної зміни під час закриття зміни ПРРО.</p> <p>"0" - закриття зміни на POS-терміналі за умови закриття зміни ПРРО не відбувається.</p> <p>"1" - під час закриття зміни на ПРРО буде перевірятись наявність проведених термінальних чеків та в разі їх наявності буде здійснено закриття зміни на POS-терміналі (за умови налаштування POS-терміналу в Cashalot)</p> <p>"2" - під час закриття зміни на ПРРО буде завжди закриватись зміна на POS-терміналі (за умови налаштування POS-терміналу в Cashalot)</p> |

#### Приклад встановлення параметрів **SetParameter**:

```

Set App = CreateObject ("AddIn.CashaLotApi")

App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", "4000000000"
App.SetParameter "NOINTERFACEMODE", "True"
App.SetParameter "PathToCertificate", "C:\Key"
App.SetParameter "PwdToCertificate", "111"
App.SetParameter "USETOKEN", "False"
App.SetParameter "NOAUTOOPENSHIFT", "False"
App.SetParameter "DEFAULTPRINTINTERNAME", ""
App.SetParameter "AUTOPRINTMODE", "False"

```

Дані параметри можна змінювати в процесі роботи з API. Наприклад, для того, щоб виконувати автоматичний друк тільки Z-звіту в процесі роботи програми.

## 2. Встановлення налаштувань API **SetParameterEx** (FiscalNumberRRO, Name, Value)

Якщо використовується робота одночасно з декількома ПРРО, то встановлення параметрів здійснюється виключно через **SetParameterEx**. У такому випадку **SetParameter не використовується**.

При встановленні параметрів за допомогою **SetParameterEx** обов'язково потрібно зазначити шлях до кореневого каталогу кожного екземпляру ПРРО Cashlot та фіскальний номер каси окремо.

Кожен параметр повинен містити в собі:

| Ім'я            | Тип    | Опис                   |
|-----------------|--------|------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. |
| Name            | string | Ім'я параметра.        |
| Value           | string | Значення параметра.    |

Приклад використання **SetParameter**:

```
App.SetParameterEx FiscalNumberRRO1,"PathToCashalotDir", "C:\Cashalot1"
```

Опис параметрів вказано в даній таблиці:

| Ім'я параметра (Name)    | Значення параметра (Value)                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>PathToCashalotDir</b> | Шлях до каталогу з встановленим ПРРО Cashlot.                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>DeviceIDFnRRO</b>     | Фіскальний номер ПРРО.                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>NOAUTOUPDATE</b>      | Перевірка на наявність нової версії на сервері <a href="https://load.cashalot.org.ua/update/">https://load.cashalot.org.ua/update/</a> та запит на оновлення програми Cashlot в момент початку роботи з API:<br><br>«True» – заборонити оновлення;<br>«False» – дозволити оновлення(за замовчуванням).                                                                                                                                                           |
| <b>NOINTERFACEMODE</b>   | Ознака використання інтерфейсу РМК для авторизації та службових операцій видачі та внесення готівки в касу.<br><br>«False» – при роботі з API викликається діалогові вікно.<br>«True» – авторизація відбувається шляхом передачі параметрів авторизації <b>PathToCertificate</b> та <b>PwdToCertificate</b> . Вікна службових операцій видачі та внесення готівки не викликаються (проте, результат операції службового внесення/видачі готівки відображається). |
| <b>PathToCertificate</b> | Шлях до каталогу з файловим ключем користувача.<br><br><b>Важливо!</b> В папці повинні міститися ключ і сертифікат тільки одного користувача. Якщо касою користуються декілька касирів, необхідно помістити їх сертифікати і ключі в окремі каталоги.                                                                                                                                                                                                            |
| <b>PwdToCertificate</b>  | Пароль від ключа                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

| Ім'я параметра (Name)    | Значення параметра (Value)                                                                                                                                                                                                                                                                                                                                                                                                   |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| USETOKEN                 | <p>Чи використовується токен для авторизації.<br/> «True» – використовується.<br/> «False» – не використовується (використовується файловий ключ).</p> <p><b>Важливо!</b> Параметр USETOKEN має бути заданий після параметрів PathToCashalotDir, DeviceIDFnRRO та NOINTERFACEMODE. Для пришвидшення завантаження програми рекомендовано <b>НЕ</b> використовувати токени.</p>                                                |
| NOAUTOOPENSIFT           | <p>Перевірка стану зміни (відкрита або закрита). Якщо зміна закрита, при виклику будь-якої COM-функції зміна автоматично відкривається.<br/> "False" – автоматично відкривати зміну, якщо вона закрита.<br/> "True" – не відкривати автоматично зміну. В цьому випадку робота COM-функції буде перервана і результат виконання буде повернуто у вигляді помилки.</p>                                                         |
| DEFAULTPRINTERNAME       | <p>Визначає принтер для друку при натисканні кнопки "Друк" в діалогових вікнах:</p> <ol style="list-style-type: none"> <li>1) "" – пустий рядок, виводиться діалог вибору принтера;</li> <li>2) "DEFAULTWINDOWS" – використовувати принтер Windows за замовчуванням;</li> <li>3) "ім'я принтера чеків" - в параметр передається ім'я принтера чеків, який буде використовуватись для друку чеків в ПРРО Cashalot.</li> </ol> |
| AUTOPRINTMODE            | <p>Параметр визначає автоматичну відправку на друк чеку або службового звіту без відображення інформаційного вікна:</p> <p>"True" – відправляти на друк;<br/> "False" – не відправляти на друк.<br/> Якщо задано принтер за замовчуванням (DEFAULTPRINTERNAME), чек друкується одразу, в іншому випадку відображається вікно вибору принтера для друку.</p>                                                                  |
| NOPRINTMODE              | <p>Дозвіл на друк:</p> <p>"True" – заборонити друк;<br/> "False" – дозволити друк.</p>                                                                                                                                                                                                                                                                                                                                       |
| SHOWREPORTADDITIONALINFO | <p>Виводити додаткову інформацію в нижній частині Z-звіту.<br/> В разі встановленого значення "True", відображаються такі поля:</p> <ul style="list-style-type: none"> <li>- Залишок готівки в касі;</li> <li>- Сума знижки;</li> <li>- Сума націнки.</li> </ul> <p>"True" - відображає додаткову інформацію;<br/> "False" – не відображає додаткову інформацію.</p>                                                         |

| Ім'я параметра (Name)      | Значення параметра (Value)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>SPECIFICAPIMODE</b>     | <p>Режим передбачений для інтеграції з обліковими системами, які розроблені з використанням мов програмування Pascal, Delphi та Xbase++.</p> <p><b>Зверніть увагу!</b> Встановлення параметра <b>SPECIFICAPIMODE</b> та його значень <b>обов'язково</b> повинно бути виконано перед встановленням інших параметрів.</p> <p><b><u>Встановлення SPECIFICAPIMODE необхідно здійснювати один раз, перед запуском роботи з ПРРО. Зміна значень у процесі роботи не рекомендована.</u></b></p> <p>Параметр <b>SPECIFICAPIMODE</b> може приймати наступні значення:<br/> "0" - програма працює в стандартному режимі;<br/> "2" - переводить API COM Cashlot в оптимізований режим роботи з інтеграціями, який підтримує можливість одночасної роботи з декількома ПРРО та може працювати як в режимі без інтерфейсу, так і з увімкненим інтерфейсом. Встановлення параметрів для роботи з двома та більше ПРРО здійснюється за допомогою SetParameterEx. При необхідності використання тільки одного екземпляру ПРРО, параметри можливо встановити як за допомогою SetParameterEx так і SetParameter. Також, робота даного режиму забезпечується процесом CashlotReporter.exe, який буде автоматично запущено при виконанні операцій з ПРРО.</p>                                                                                                                                                                                                                                        |
| <b>USEGOODSSTORAGEMODE</b> | <p>Параметр <b>USEGOODSSTORAGEMODE</b> визначає використання ведення складського обліку по товарам та автоматичну синхронізацію залишків по ним при використанні API COM Cashlot за умови активованого налаштування в кабінеті користувача в розділі «Склад - Налаштування - Облік товарів у ПРРО». Параметр може приймати наступні значення:</p> <p>«0» - програма працює в стандартному режимі без ведення обліку товарів в API та синхронізації залишків по товарам з кабінетом користувача.</p> <p>«1» - режим передбачає використання автоматичної синхронізації залишків по товарам та інформування користувача окремим повідомленням про перевищення їх ліміту під час спроби реєстрації фіскального чека. Після появи повідомлення користувачу доступна можливість відмовитись від подальшої обробки чека або погодитись та продовжити реєстрацію, але таким чином залишки по товарам будуть мати від'ємне значення кількості.</p> <p>«2» - режим передбачає використання автоматичної синхронізації залишків по товарам та інформування користувача окремим повідомленням про перевищення їх ліміту без можливості продовжити обробку та реєстрацію чека.</p> <p>Візуалізація повідомлення про перевищення ліміту залишків з можливістю продовжити/перервати процедуру фіскалізації чека, використовуючи перший режим параметра <b>USEGOODSSTORAGEMODE</b>, буде виникати також при ознаці використання режиму роботи без інтерфейса (<b>NOINTERFACEMODE=true</b>).</p> |

| Ім'я параметра (Name)             | Значення параметра (Value)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>USEPRGPOSTERMINAL</b>          | <p>Параметр відповідає за можливість проведення оплати платіжною карткою через POS-термінал, який підключений в Cashalot в розділі «Налаштування - Додаткове обладнання - POS-термінал». Оплату на терміналі можливо здійснити після передачі чека на реєстрацію. В разі використання обраного параметра фіскальний чек буде містити всі необхідні обов'язкові реквізити банківського терміналу (ЕПЗ, RRN, еквайр та ін.).</p> <p>“True” - використовується POS-термінал, підключений в Cashalot для оплати під час передачі чека на фіскалізацію з обраною формою оплати Картка (<a href="#">SumPayByCard</a>) .</p> <p>“False” - під час фіскалізації чека з формою оплати Картка підключений POS-термінал в Cashalot не використовується.</p> <p>В разі неуспішної транзакції на банківському терміналі операція реєстрації чека не відбувається.</p>                    |
| <b>RATELETTERCREATEIFNOTEXIST</b> | <p>Параметр відповідає за автоматичне створення податкової ставки та літерного позначення при реєстрації чеків продажу. Рекомендовано до використання, якщо в кабінеті Cashalot зайняті доступні комбінації податкової ставки та літери.</p> <p>“True” - відбувається автоматичне створення податкової ставки та літерного позначення в разі її відсутності в Cashalot</p> <p>“False” - використовуються наявні завантажені в Cashalot податкові ставки та їх літерні позначення.</p>                                                                                                                                                                                                                                                                                                                                                                                       |
| <b>USESELTERMINALINFOWND</b>      | <p>Параметр призначений для можливості ручного підтвердження термінальної операції користувачем за умови, якщо операція з POS-терміналом пройшла неуспішно (обірвався зв'язок термінала з Cashalot в момент проведення операції або ін.). В такому разі з'явиться додаткове вікно за допомогою якого можливо підтвердити операцію або виконати запит результату операції з POS-термінала.</p> <p>Параметр працює за умови використання параметра <a href="#">USEPRGPOSTERMINAL</a>. Вікно підтвердження буде з'являтися навіть якщо використовується параметр <a href="#">NOINTERFACEMODE</a>.</p> <p>“True” - після отримання помилки від POS-термінала при проведенні операції з'явиться додаткове вікно з можливістю ручного підтвердження операції.</p> <p>“False” - в разі отримання помилки від POS-термінала необхідно буде повторити спробу виконання операції.</p> |
| <b>AUTOCLOSETERMINALSHIFT</b>     | <p>Параметр призначений для закриття термінальної зміни під час закриття зміни ПРРО.</p> <p>“0” - закриття зміни на POS-терміналі за умови закриття зміни ПРРО не відбувається.</p> <p>“1” - під час закриття зміни на ПРРО буде перевірятись наявність проведених термінальних чеків та в разі їх наявності буде</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

| Ім'я параметра (Name) | Значення параметра (Value)                                                                                                                                                                                                         |
|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                       | здійснено закриття зміни на POS-терміналі (за умови налаштування POS-терміналу в Cashalot)<br>"2" - під час закриття зміни на ПРРО буде завжди закриватись зміна на POS-терміналі (за умови налаштування POS-терміналу в Cashalot) |

#### Приклад використання **SetParameterEx**:

```
Set App = CreateObject("AddIn.CashaLotApi")
FiscalNumberRR01 = XXXXXXXXXX
FiscalNumberRR02 = YYYYYYYYYY

App.SetParameterEx FiscalNumberRR01,"PathToCashalotDir", "C:\Cashalot1"
App.SetParameterEx FiscalNumberRR01, "DeviceIDFnRR0", FiscalNumberRR01

App.SetParameterEx FiscalNumberRR02,"PathToCashalotDir", "C:\Cashalot2"
App.SetParameterEx FiscalNumberRR02, "DeviceIDFnRR0", FiscalNumberRR02
```

### 3. Отримання номеру версії API бібліотеки **GetVersion**

Функція немає вхідних параметрів. У відповіді отримаємо номер версії API бібліотеки з типом **String**.

#### Приклад виконання **GetVersion**:

```
Set App = CreateObject("AddIn.CashaLotApi")
libVersion = App.GetVersion()
MsgBox "Версія бібліотеки = " & libVersion
```

#### Приклад відповіді:

Версія бібліотеки = 1.14

#### 4. Відкриття зміни **OpenShift** (FiscalNumberRRO)

**OpenShift** повинен містити параметр з фіскальним номером каси, який має тип **string**.

При виклику методу (наприклад: фіскалізації чека), якщо зміна не відкрита, відбувається автоматичне відкриття зміни з викликом функції **OpenShift**. Для запобігання автоматичного відкриття зміни, за допомогою функції **SetParameter**, встановіть параметр: **NOAUTOOPENSHIFT** = "True"

В цьому випадку, поки не буде явно відкрито зміну викликом **OpenShift**, виклик інших функцій, які вимагають відкритої зміни, буде перериватися з поверненням помилки: "Операція неможлива, необхідно відкрити зміну"

##### Приклад виконання **OpenShift**:

```
Set resultOpenShift = App.OpenShift (FiscalNumberRRO)

MsgBox "Результат успішності операції = " & resultOpenShift.Return
MsgBox "Опис помилки = " & resultOpenShift.Description
MsgBox "ID зміни = " & resultOpenShift.ShiftID
MsgBox "Результат = " & resultOpenShift.JsonVal
```

При відкритті зміни, у відповіді повертається структура **CashalotApiRetVal**, що містить значення:

| Ім'я        | Тип    | Опис                                                                         |
|-------------|--------|------------------------------------------------------------------------------|
| Return      | bool   | Ознака успішності виконання.                                                 |
| Description | string | Опис помилки. Заповнюється у випадку коли: Return = False.                   |
| JsonVal     | string | JSON-структура, що містить значення успішності операції, помилки, ID зміни . |
| ShiftID     | string | Ідентифікатор поточної зміни.                                                |

##### Відповідь:

```
resultOpenShift.Description = ""
resultOpenShift.JsonVal =
{"Ret":true,"ErrorString":"","Values":{"ShiftID":"1a9eb031-b1ce-4ee7-8a11-329824d895a7"}}
resultOpenShift.Return = True
resultOpenShift.ShiftID = 1a9eb031-b1ce-4ee7-8a11-329824d895a7
```

або:

```
resultOpenShift.Description = "Авторизація відмінена користувачем"
resultOpenShift.JsonVal = {"Ret":false,"ErrorString":"Авторизація відмінена користувачем", "Values":{}}
resultOpenShift.Return = False
resultOpenShift.ShiftID =
```

## 5. Фіскалізація чека продажу товарів, послуг **FiscalizeCheck** (FiscalNumberRRO, JSONGoodsLstData, JSONPayData)

Для створення та фіскалізації чека продажу товарів та/або послуг необхідно сформувати JSON структуру [JSONGoodsLstData](#), що міститиме дані номенклатури, та структуру JSON [JSONPayData](#) з формами оплати.

Параметри, які повинен містити метод **FiscalizeCheck**:

| Ім'я             | Тип    | Опис                                                 |
|------------------|--------|------------------------------------------------------|
| FiscalNumberRRO  | string | Фіскальний номер ПРРО. Ідентифікатор пристрою.       |
| JSONGoodsLstData | string | JSON структура з описом проданих товарів або послуг. |
| JSONPayData      | string | JSON структура з підсумками оплати по чеку.          |

Структура **JSONGoodsLstData** містить масив **ReceiptLst**, кожен елемент якого містить параметри номенклатури та об'єкти **Comment**, **DocType**, **SaleOrderNumbe**

Опис **ReceiptLst**:

| Поле               | Значення                                                                                                                                                                                                                     |
|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| VendorCode*        | Артикул.                                                                                                                                                                                                                     |
| Name               | Найменування товару.                                                                                                                                                                                                         |
| DocNumber**        | Номер договору (при використанні ломбардних послуг).                                                                                                                                                                         |
| CdUKDZED           | Код УКТЗЕД.                                                                                                                                                                                                                  |
| CdDKPP             | Код ДКПП.                                                                                                                                                                                                                    |
| GoodsType          | Тип номенклатури. За замовчуванням товар, якщо параметр не задано.<br>1 – товар; 2 – послуга; 3 – ломбардна послуга **.                                                                                                      |
| Barcode            | Штрих-код.                                                                                                                                                                                                                   |
| UnitType*          | Одиниця вимірювання.                                                                                                                                                                                                         |
| Quantity*          | Кількість товару.                                                                                                                                                                                                            |
| Price*             | Ціна одиниці товару (в форматі <Гривні, Копійки>).                                                                                                                                                                           |
| Amount*            | Кінцева сума по позиції чека (з урахуванням всіх націнок та знижок, у форматі <Гривні, Копійки>).                                                                                                                            |
| DiscountPrc        | Відсоток знижки. <b>Увага!</b> При заповненні DiscountPrc, необхідно також обов'язково заповнювати та передавати DiscountSum                                                                                                 |
| DiscountSum        | Сума знижки на позицію.                                                                                                                                                                                                      |
| VATRate            | ПДВ у відсотках (від 0 до 99,99).                                                                                                                                                                                            |
| VATLetter          | Літера ставки ПДВ                                                                                                                                                                                                            |
| IsPriceIncludeVAT  | Для випадків, коли сума по позиції не включає суму ПДВ(true/false)                                                                                                                                                           |
| SumVAT             | Сума ПДВ за одиницю товару (ігнорується, якщо вказати ознаку, що враховує ПДВ "true").                                                                                                                                       |
| IsExcise           | Ознака підакцизного товару(true/false).                                                                                                                                                                                      |
| ExciseLetter       | Літера ставки акцизного податку                                                                                                                                                                                              |
| ExciseStampBarcode | Штрих-код марки акцизного податку. У випадку коли товарів більше чим 1, по одній позиції номенклатури, то всі штрих-коди заповнюються через кому. Поле ігнорується, якщо ознака підакцизного товару <b>IsExcise</b> = false. |

|             |                                            |
|-------------|--------------------------------------------|
| OtherParams | Додаткові параметри у вигляді XML таблиці. |
|-------------|--------------------------------------------|

\* обов'язкові параметри при передачі даних товару.

\*\* для [Ломбардних операцій](#).

**Зауваження!** Для передачі кількості товару з дробовою частиною (в т. ч. нульовою, наприклад, "2,00") допускається використовувати десятковий розділювач "." (крапка) або "," (кома). Якщо параметр не задано, або задано некоректно, його значення приймається рівним одиниці (1).

Параметр [OtherParams](#) передається в форматі XML з кодуванням UTF-8. Атрибути XML-рядка вказані в таблиці нижче.

Зазначимо, що параметри XML-рядка [OtherParams](#) мають пріоритет над параметрами, які передані в **ReceiptLst**.

#### Атрибути XML-рядка [OtherParams](#):

| Назва атрибута                                                             | Опис атрибута                                                                                                                                                                                                                         |
|----------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <a href="#">Nomenclature</a> *                                             | Найменування товару.                                                                                                                                                                                                                  |
| <a href="#">UKTZED</a>                                                     | Код УКТЗЕД.                                                                                                                                                                                                                           |
| <a href="#">VendorCode</a> *                                               | Артикул.                                                                                                                                                                                                                              |
| <a href="#">Barcode</a>                                                    | Штрих-код.                                                                                                                                                                                                                            |
| <a href="#">Dimension</a> *                                                | Одиниця вимірювання.                                                                                                                                                                                                                  |
| <a href="#">Discount</a>                                                   | Відсоток знижки. <b>Увага!</b> При заповненні Discount, необхідно також обов'язково заповнювати та передавати DiscountAmount                                                                                                          |
| <a href="#">DiscountAmount</a>                                             | Сума знижки.                                                                                                                                                                                                                          |
| <a href="#">VAT</a> *                                                      | Ставка ПДВ в відсотках.                                                                                                                                                                                                               |
| <a href="#">VATLetter</a>                                                  | Літера ставки ПДВ                                                                                                                                                                                                                     |
| <a href="#">PriceIncludesVAT</a><br>або<br><a href="#">PriceIncludeVAT</a> | Для випадків, коли сума по позиції не включає суму ПДВ(true/false)                                                                                                                                                                    |
| <a href="#">AmountVAT</a>                                                  | Сума ПДВ за одиницю товару, у форматі <Гривні, Копійки>.                                                                                                                                                                              |
| <a href="#">IsExcisable</a>                                                | Ознака підакцизного товару, True/False.                                                                                                                                                                                               |
| <a href="#">ExciseLetter</a>                                               | Літера ставки акцизного податку                                                                                                                                                                                                       |
| <a href="#">ExciseStampBarcode</a>                                         | Штрих-код марки акцизного податку. У випадку коли товарів більше чим 1, по одній позиції номенклатури, то всі штрих-коди заповнюються через кому. Поле ігнорується, якщо ознака підакцизного товару <a href="#">IsExcise</a> = false. |

\* обов'язкові параметри в рядку [OtherParams](#), якщо не вказані відповідні явні параметри.

Приклад OtherParameters:

```
<?xml version="1.0" encoding="UTF-8"?> <Table> <Record VAT="5"
IsExcisable="True" Nomenclature="Товар1" UKTZED="0202" VendorCode="1"
Barcode="1452" Dimension="шт" Discount="0.54" DiscountAmount="0.25"
PriceIncludeVAT="1" AmountVAT="0" ExciseStampBarcode="11111, 22222, 33333"/>
</Table>
```

Об'єкт **Comment** використовується для додавання нефіскальної інформації:

|         |                                                                                                             |
|---------|-------------------------------------------------------------------------------------------------------------|
| Comment | Опціональний рядок нефіскальної інформації, коментар.<br>Допускається використання одного коментаря на чек. |
|---------|-------------------------------------------------------------------------------------------------------------|

Об'єкт **DocType** є обов'язковим при проведенні ломбардних операцій, призначений для визначення типу документа (чека). [Ломбардні операції](#).

|         |                                                                                                                   |
|---------|-------------------------------------------------------------------------------------------------------------------|
| DocType | Тип документа:<br>0 – чек продажу;<br>4 – ломбардний чек;<br>За замовчуванням DocType = 0, якщо об'єкт не задано. |
|---------|-------------------------------------------------------------------------------------------------------------------|

Об'єкт **SaleOrderNumber** (Номер замовлення/договору) потрібен для заповнення номера договору при проведенні інтернет продажу, передоплати та післяплати. Для відображення номера замовлення в післяплаті обов'язково потрібно заповнювати в **JSONPayData** фіскальний номер чека передоплати **ParentReceiptFiscalNumber**.

**Заокруглення суми** по чеку реалізовано наступним чином:

В **JSONPayData** передається значення готівки, що надана клієнтом на касі, значення оплати картою, сертифікатом, кредитом або безготівкою, а також заокруглена сума, розрахована за певним алгоритмом користувача **SumPayCheck** (відповідно до [Постанови НБУ №115](#)):

- якщо сума закінчується від 1 до 24 копійок – здійснювати заокруглення до 00 копійок, в бік зменшення;
- якщо закінчуються від 75 до 99 копійок – здійснювати заокруглення до 00 копійок, в бік збільшення;
- якщо сума закінчується від 51 до 74 копійок - здійснювати заокруглення до 50 копійок, в бік зменшення
- якщо сума закінчується від 25 до 49 копійок - здійснювати заокруглення до 50 копійок, в бік збільшення.

В чеку буде відображено заокруглення, надану готівку та інші форми оплати, заокруглену суму до сплати, решту.

**Решта** по чеку розраховується так:

**Сума решти** = **SumCash** + **SumPayByCard** + **SumPayByBank** + **SumPayByCredit** + **SumPayByCertificate** – **SumPayCheck**

Решта розраховується програмою Cashlot автоматично, згідно вказаних користувачем даних.

Структура **JSONPayData** містить підсумки оплати по чеку, і має наступні поля:

| Поле                       | Значення                                                                                                                                                                                                                                                                         |
|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>SumCash</b>             | Сума наданої готівки.                                                                                                                                                                                                                                                            |
| <b>SumPayByCard</b>        | Сума оплати чека банківською картою.                                                                                                                                                                                                                                             |
| <b>SumPayByBank</b>        | Сума оплати безготівкою                                                                                                                                                                                                                                                          |
| <b>SumPayByCredit</b>      | Сума оплати чека в кредит.                                                                                                                                                                                                                                                       |
| <b>SumPayByCertificate</b> | Сума оплати чека сертифікатом.                                                                                                                                                                                                                                                   |
| <b>SumPayCheck</b>         | Сума чека до оплати з урахуванням всіх надбавок, знижок та округлень.                                                                                                                                                                                                            |
| <b>PaymentOrderType</b>    | Встановлення порядку сплати:<br><br>0 - звичайна;<br>1 - передплата;<br>2 - післяплата;<br>3 - інтернет продаж;<br>4 - повна післяплата;<br>5 - оплата частинами (передплата);<br>6 - оплата частинами (чергова сплата);<br>7 - оплата частинами (остаточний розрахунок).        |
| <b>SumPreparePaid</b>      | Сума попередньої оплати.                                                                                                                                                                                                                                                         |
| <b>PartPaySum</b>          | Сума попередньої оплати або чергової сплати при використанні способу оплати «Оплата частинами». Сума остаточного розрахунку не потребує заповнення, її розрахунок відбувається автоматично на основі залишків до сплати по ланцюжку чеків зі способом оплати «Оплата частинами». |

| Поле                      | Значення                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ParentReceiptFiscalNumber | Фіскальний номер чека, по якому здійснюється доплата/повернення.                                                                                                                                                                                                                                                                                                                                                                                                           |
| ParentRROFiscalNumber     | Фіскальний номер РРО, по якому здійснюється доплата/повернення.                                                                                                                                                                                                                                                                                                                                                                                                            |
| ParentReceiptOrderDate    | Дата продажу чека по якому здійснюється повернення (формат: дд.мм.рррр). Використовується тільки для чеків продажів.                                                                                                                                                                                                                                                                                                                                                       |
| CustomerEmail             | Електронна адреса покупця на яку можна відправити фіскалізований чек. Передається в поле електронної адреси при натисканні кнопки «Надіслати чек на Email» на екрані відображення чека.                                                                                                                                                                                                                                                                                    |
| CustomerSmsNumber         | Номер телефону покупця на який можна відправити фіскалізований чек у вигляді SMS повідомлення. Передається в поле номера телефону при натисканні кнопки «Надіслати на SMS».                                                                                                                                                                                                                                                                                                |
| CustomerViberNumber       | Номер телефону покупця на який можна відправити фіскалізований чек у вигляді повідомлення у Viber. Передається в поле номера телефону при натисканні кнопки «Надіслати на Viber».                                                                                                                                                                                                                                                                                          |
| AutosendCustomerEmail     | Ознака автоматичної відправки чека покупцю на E-mail після його реєстрації. Параметр може приймати значення true або false. Відправка чека буде здійснена якщо передано заповнений параметр <a href="#">CustomerEmail</a> , виконано налаштування електронної пошти в РМК в розділі «Налаштування - Загальні налаштування - Господарська одиниця - Налаштування пошти» та встановлене значення true для <a href="#">AutosendCustomerEmail</a>                              |
| AutosendCustomerSms       | Ознака автоматичної відправки чека покупцю на SMS після його реєстрації. Параметр може приймати значення true або false. Відправка чека буде здійснена якщо передано заповнений параметр <a href="#">CustomerSmsNumber</a> , виконано налаштування сервісу доставки SMS в кабінеті користувача Cashalot ( <a href="https://my.cashalot.org.ua/">https://my.cashalot.org.ua/</a> ) та встановлено значення true для <a href="#">AutosendCustomerSms</a>                     |
| AutosendCustomerViber     | Ознака автоматичної відправки чека покупцю на SMS після його реєстрації. Параметр може приймати значення true або false. Відправка чека буде здійснена якщо передано заповнений параметр <a href="#">CustomerViberNumber</a> , виконано налаштування сервісу доставки повідомлень у Viber в кабінеті користувача Cashalot ( <a href="https://my.cashalot.org.ua/">https://my.cashalot.org.ua/</a> ) та встановлено значення true для <a href="#">AutosendCustomerViber</a> |
| OtherParametrs            | Додаткові параметри у вигляді XML таблиці.                                                                                                                                                                                                                                                                                                                                                                                                                                 |

| Поле                                                                                      | Значення                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BankPayToolsNM                                                                            | Найменування безготівкового засобу оплати при використанні способу оплати 0,1,2,5, 6 та 7                                                                                                                                                                                                                                                                                                                                                                                                                              |
| PrePaymentPayToolsNM                                                                      | Найменування безготівкового засобу оплати при використанні способу оплати 3                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| AfterPaymentPayToolsNM                                                                    | Найменування безготівкового засобу оплати при використанні способу оплати 3,4                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| UseTransactionDetailsToSumPayByBank                                                       | Ознака (true/false) яка дозволяє застосувати реквізити транзакції POS-терміналу до безготівкового засобу оплати (Платіж Mono Pay, Платіж LIQPAY, Платіж WAYFORPAY) використаних в способах оплати 0,1,2,5, 6 та 7                                                                                                                                                                                                                                                                                                      |
| PaymentFormsLst                                                                           | Масив який містить інформацію про засоби оплати по чеку які передаються у вигляді окремих масивів. Масив з вказаними значеннями по засобу оплати повинен містити: код засобу оплати, суму безготівкової оплати, найменування засобу оплати та ознаку (true/false) застосування реквізитів транзакції POS-терміналу до використаного засобу оплати. Важливо зберігати саме таку послідовність заповнення. Реквізити транзакції можливо застосувати до засобів оплати: Платіж LIQPAY, Платіж Mono pay, Платіж WAYFORPAY. |
| ReceiptID                                                                                 | Параметр призначений для можливості контролю дублюючої передачі чека на реєстрацію. Значення параметру містить GUID контроль за генерацією та використанням котрого необхідно реалізувати на рівні обліково-розрахункової системи користувача. В разі повторної передачі значення GUID чека який вже був зареєстрований, API поверне відповідь на раніше сформований чек та відобразить його візуальну форму.                                                                                                          |
| ExpressWaybillNumber                                                                      | Номер експрес-накладної який зазначається для чеку зі способом оплати Інтернет-продаж (PaymentOrderType=3). В інших випадках введене значення ігнорується.                                                                                                                                                                                                                                                                                                                                                             |
| Необов'язкові параметри інформації по транзакції терміналу при безготівкових розрахунках: |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| TerminalID                                                                                | Код терміналу.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| ApprovalCode                                                                              | Код підтвердження.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| RRN                                                                                       | RRN транзакції.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| IssuerName                                                                                | Платіжна система.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| PAN                                                                                       | Номер картки.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| TransactionDate                                                                           | Дата транзакції (в форматі dd.mm.yyyy hh:mm:ss).                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| SignVerif                                                                                 | Електронний підпис.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| AcquireName                                                                               | Термінал.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| InvoiceNumber                                                                             | Номер чека.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| ParentRRN                                                                                 | RRN транзакції чека, по якому здійснено повернення.                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| AcquireID                                                                                 | Ідентифікатор еквайра                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |

**Увага!** Різниця між сумою оплати чека та SumPayCheck (сумою по всіх видах оплати) вважається значенням заокруглення. Тому розробник має самостійно реалізувати алгоритм округлення значення виданої готівки або суми оплати по чеку.

## Відповідь

У відповіді на **FiscalizeCheck** повертається структура **CashalotApiRetVal**, що містить:

|                     |        |                                                                                                                                                                           |
|---------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return              | bool   | Ознака успішності виконання.                                                                                                                                              |
| JsonVal             | string | JSON-структура, що містить значення успішності операції, тип документу, помилки, локальний та фіскальний номери чека, ID зміни та Base64 рядок XML чека та інші параметри |
| ReceiptFiscalNum    | string | Фіскальний номер чека.                                                                                                                                                    |
| ReceiptLocalNum     | string | Локальний номер чека.                                                                                                                                                     |
| OfflineMode         | bool   | Ознака перебування в режимі офлайн                                                                                                                                        |
| ShiftID             | string | Ідентифікаційний номер поточної зміни.                                                                                                                                    |
| FSKORceiptLink      | string | Посилання на перегляд чека в ДПС                                                                                                                                          |
| CashalotReceiptLink | string | Посилання на перегляд чека в кабінеті Cashalot                                                                                                                            |
| Type                | int    | <a href="#">Тип документа</a> .                                                                                                                                           |
| XML                 | string | XML-структура фіскалізованого чека.                                                                                                                                       |
| ReceiptID           | string | ID (guid) чека                                                                                                                                                            |
| ReceiptSum          | string | Сума до сплати по чеку                                                                                                                                                    |
| ReceiptDateTime     | string | Дата та час створення у форматі (дд.мм.рррр гг:хв:сс)                                                                                                                     |

Тип документа "**Type**" визначається числом, яке формується наступним чином:

1<Клас документа><Тип документа><Додатковий тип документа>

Опис "**Type**" :

| Число                      | Кількість знаків | Значення                                                                                                                                                                                                                                                                       |
|----------------------------|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <Клас документа>           | 1                | 0 - Чек<br>1 - Z-звіт<br>2 - X-звіт                                                                                                                                                                                                                                            |
| <Тип документа>            | 3                | 000 - Чек реалізації товарів/послуг<br>001 - Чек переказу коштів<br>002 - Чек операції обміну валюти<br>003 - Чек видачі готівки<br>100 - Відкриття зміни<br>101 - Закриття зміни<br>102 - Початок офлайн сесії<br>103 - Завершення офлайн сесії                               |
| <Додатковий тип документа> | 3                | 000 - Касовий чек (реалізація)<br>001 - Видатковий чек (повернення)<br>002 - Чек операції «службове внесення»/«отримання авансу»<br>003 - Чек операції «отримання підкріплення»<br>004 - Чек операції «службова видача»/«інкасація»<br>005 - Чек сторнування попереднього чека |

## Приклад використання засобів оплат в чеках

В разі потреби відображення в чеку засобів оплат таких як «Платіж Monopay», «Переказ з картки», «Платіж EasyPay» або інших, наявно два варіанти їх відображення та використання - спрощений та розширений. Засоби оплати можливо використовувати для всіх способів оплат по чеку (продаж, передплата/післяплата, інтернет-продаж, тощо).

Стандартні засоби оплат (Готівка (Готівкові кошти), Картка, Сертифікат, Кредит) заповнюються стандартно відповідно до опису встановлення порядку оплати в [JSONPayData](#).

**Примітка!** Описаний функціонал є виключно доповненням до існуючого функціоналу налаштування та застосування оплат по чеку. Таким чином новий функціонал не змінює та не порушує роботу раніше реалізованого механізму оплати, а працює як окреме, самостійне розширення. Його впровадження є повністю зворотно сумісним із поточними інтеграціями.

### **Варіант використання засобів оплат №1 (спрощений)**

Для налаштування спрощеного варіанту використання засобів оплати у чеку слід використовувати наступний перелік параметрів: [SumPayByBank](#), [BankPayToolsNM](#), [PrePaymentPayToolsNM](#), [AfterPaymentPayToolsNM](#) та [UseTransactionDetailsToSumPayByBank](#). Спрощений варіант надає змогу використовувати лише один засіб оплати у чеку.

За допомогою параметру [SumPayByBank](#) передається сума для безготівкового засобу оплати за умови, що використовується спосіб оплати 0, 1, 2, 5, 6 або 7. Залежно від обраного способу оплати по чеку, для передачі назви засобу оплати слід використовувати один з атрибутів: [BankPayToolsNM](#), [PrePaymentPayToolsNM](#) або [AfterPaymentPayToolsNM](#). Використання параметру [BankPayToolsNM](#) дозволяється у випадку застосування способів оплати 0, 1, 2, 5, 6 або 7. Параметр [PrePaymentPayToolsNM](#) можна використовувати, якщо застосовується спосіб оплати 3, а [AfterPaymentPayToolsNM](#) — якщо використовується спосіб оплати 3 або 4.

Як найменування засобів оплати радимо використовувати наступний [перелік](#) готових варіантів. Якщо в переліку відсутня потрібна назва, користувач може ввести власну. Створення власного засобу оплати відбувається під час його першого використання у чеку. Для успішного створення, на момент реєстрації такого чека необхідно мати доступ до кабінету користувача Cashalot.

За потреби додавання параметрів транзакції POS-термінала до засобу оплати, вказаного у [BankPayToolsNM](#), необхідно використовувати параметр [UseTransactionDetailsToSumPayByBank](#). Опція доступна лише за умови, що використовується один із таких засобів оплати: Платіж Mono Pay, Платіж LIQPAY або Платіж WAYFORPAY. Якщо було використано інший засіб оплати (наприклад, Інтернет-еквайринг) разом із [UseTransactionDetailsToSumPayByBank](#), то параметри транзакції POS-термінала не будуть відображені у чеку.

**Важливо!** Параметр `UseTransactionDetailsToSumPayByBank` надає пріоритет параметрам транзакції, які застосовуються до використаного засобу оплати. У такому випадку, якщо разом із засобом оплати (наприклад, Платіж LIQPAY) також використано засіб оплати Картка (SumPayByCard), то параметри транзакції буде відображено у чеку саме під засобом Платіж LIQPAY

## Варіант використання засобів оплат №2 (розширений)

Розширений варіант використання засобів оплат передбачає заповнення параметра `PaymentFormsLst` в структурі оплат `JSONPayData`. Параметр є масивом який містить необхідну кількість засобів оплат заповнених у вигляді окремих масивів. За допомогою розширеного варіанту можливо передати декілька засобів оплат одночасно.

**Приклад `JSONPayData` з заповненням `PaymentFormsLst`:**

```
JSONPayData = {"PaymentOrderType":0,"SumCash":"","SumPayCheck":"","PaymentFormsLst":[
  [
    "202",
    "50",
    "Платіж Моно Рау",
    "True"
  ],
  [
    "202",
    "50",
    "Переказ з поточного рахунку"
  ]
]}
```

де **202** - це код засобу оплати;

**50** - сума засобу оплати;

**Платіж Portmone** - назва засобу оплати.

**True** - ознака застосування реквізитів POS-термінала до використаного засобу оплати.

Перелік рекомендованих засобів оплати які доступні до використання для [способів оплати](#) 0,1,2,5,6 та 7:

| Код засобу оплати | Найменування засобу оплати |
|-------------------|----------------------------|
| 202               | Платіж Моно Рау            |
| 202               | Платіж LIQPAY              |
| 202               | Платіж WAYFORPAY           |
| 202               | Інтернет еквайринг         |
| 202               | Переказ з картки           |

|     |                                          |
|-----|------------------------------------------|
| 202 | Переказ через ННПП                       |
| 202 | Переказ через ПТКС ННПП                  |
| 202 | Переказ через ПТКС банку                 |
| 202 | Переказ через ННПП NovaPay               |
| 202 | Інтернет банкінг                         |
| 202 | Переказ з поточного рахунку              |
| 202 | Платіж mono checkout                     |
| 202 | Платіж Portmone                          |
| 202 | Платіж EasyPay                           |
| 202 | Переказ через ННФПП                      |
| 202 | Платіж через сервіс переказу ROZETKA Pay |

В рамках одного фіскального чека допускається використання засобів оплат з дублюючим значенням коду але за умови, що найменування засобу оплати заповнено відмінне від значення попереднього засобу. В разі відсутності потрібного найменування засобу оплати користувач може використати необхідну власну назву. Створення власного засобу оплати відбувається під час його першого використання у чеку. Для успішного створення, на момент реєстрації такого чека необхідно мати доступ до кабінету користувача Cashalot.

За потреби додавання параметрів транзакції POS-термінала до засобу оплати, вказаного в масиві, необхідно встановити ознаку True. Опція доступна лише за умови, що найменування засобу оплати вказано як: Платіж Mono Pay, Платіж LIQPAY або Платіж WAYFORPAY. Якщо було використано інший засіб оплати (наприклад, Інтернет-еквайринг) разом із ознакою True, то параметри транзакції POS-термінала не будуть відображені у чеку. Якщо ознака не встановлена (не використовується), то це означає, що використовується його ознака False.

**Важливо!** Ознака використання реквізитів транзакції POS-термінала надає пріоритет параметрам транзакції, які застосовуються до використаного засобу оплати. У такому випадку, якщо разом із засобом оплати (наприклад, Платіж LIQPAY) також використано засіб оплати Картка (SumPayByCard), то параметри транзакції буде відображено у чеку саме під засобом Платіж LIQPAY. Встановлення ознаки доступно тільки для засобів оплат які використовуються в способах оплати 0,1,2,5,6 та 7.

Перелік рекомендованих засобів оплати які доступні до використання для [способів оплати](#) 3 та 4:

| Код засобу оплати | Найменування засобу оплати               |
|-------------------|------------------------------------------|
| 2 або 201*        | Платіж Mono Pay                          |
| 2 або 201*        | Платіж LIQPAY                            |
| 2 або 201*        | Платіж WAYFORPAY                         |
| 2 або 201*        | Інтернет еквайринг                       |
| 2 або 201*        | Переказ з картки                         |
| 2 або 201*        | Переказ через ННПП                       |
| 2 або 201*        | Переказ через ПТКС ННПП                  |
| 2 або 201*        | Переказ через ПТКС банку                 |
| 2 або 201*        | Переказ через ННПП NovaPay               |
| 2 або 201*        | Інтернет банкінг                         |
| 2 або 201*        | Переказ з поточного рахунку              |
| 2 або 201*        | Платіж mono checkout                     |
| 2 або 201*        | Платіж Portmone                          |
| 2 або 201*        | Платіж EasyPay                           |
| 2 або 201*        | Переказ через ННФПП                      |
| 2 або 201*        | Платіж через сервіс переказу ROZETKA Pay |

## Приклади використання та заповнення чеку

### Приклад виконання методу FiscalizeCheck:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

JSONGoodsLstData = "{""ReceiptLst"":[{""VendorCode"":""АртикулТовару1"",
""Name"": ""НазваТовару1"", ""CdUKDZED"": "", ""CdDKPP"": "",
""GoodsType"": "", ""Barcode"":""123456789"", ""UnitType"":""шт"", ""Quantity"":
""1"", ""Price"": ""183"", ""Amount"":""181,17"", ""DiscountPrc"": "",
""DiscountSum"":""1.83"", ""VATRate"":""20"", ""IsPriceIncludeVAT"":true,
""SumVAT"":""30.2"", ""IsExcise"":false, ""OtherParametrs"":null}, {""VendorCode"":
""АртикулТовару2"", ""Name"":""НазваТовару2"", ""CdUKDZED"": "", ""CdDKPP"": "",
""GoodsType"": "", ""Barcode"":""999999999"", ""UnitType"":""шт"",
""Quantity"":""1"", ""Price"":""183"", ""Amount"":""181,17"", ""DiscountPrc"": "",
""DiscountSum"":""1.83"", ""VATRate"":""20"", ""IsPriceIncludeVAT"":true,
""SumVAT"":""30.2"", ""IsExcise"": true, ""ExciseStampBarcode"":""1111"",
""OtherParametrs"":null}],""Comment"":""#####\n\r #
Тестовий коментар \n\r#####"}"

JSONPayData =
"{""SumCash"":""52"", ""SumPayByCard"":""310,34"", ""SumPayByCredit"":null, ""SumPayByC
ertificate"":null, ""SumPayCheck"":""362,34"", ""TerminalID"":
""S1K700J8"", ""ApprovalCode"": ""68014B"", ""RRN"":
""048867173620"", ""IssuerName"":""MASTERCARD"", ""PAN"":""XXXXXXXXXXXX6883"", ""Transa
ctionDate"":""07.01.2022 17:33:00"", ""AcquireName"": ""Кашалот
банк"", ""InvoiceNumber"": ""224"", ""CustomerEmail"":""test@mai.com""}"

Set Receipt = App.FiscalizeCheck(FiscalNumberRRO, JSONGoodsLstData, JSONPayData)

Wscript.Echo "JsonVal = " & Receipt.JsonVal
MsgBox "Фіскальний номер чека = " & Receipt.ReceiptFiscalNum
MsgBox "Локальний номер чека = " & Receipt.ReceiptLocalNum
MsgBox "Успіх реєстрації = " & Receipt.Return
MsgBox "Тип чека = " & Receipt.Type
MsgBox "ID зміни = " & Receipt.ShiftID
Wscript.Echo "XML зареєстрованого чека = " & Receipt.XML
```

### Приклад відповіді:

```
JsonVal =
{"Ret":true,"ErrorString":"","Values":{"ShiftID":"92b59cb8-f34b-43dd-9cec-d8b187448c
bf","ReceiptFiscalNumber":"630168","ReceiptLocalNumber":"188","OfflineMode":"False",
"Type":"1000000","Base64Str1251ReceiptXML":"'тут буде закодована структура XML
чека в стандарті Base64'"}
Фіскальний номер чека = 630168
Локальний номер чека = 188
Успіх реєстрації = true
Тип чека = 1000000
ID зміни = 92b59cb8-f34b-43dd-9cec-d8b187448cbf
XML зареєстрованого чека =
<?xml version="1.0" encoding="windows-1251"?>
```

```
<CHECK xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="check01.xsd">
  <CHECKHEAD>
    <DOCTYPE>0</DOCTYPE>
    <DOCSUBTYPE>0</DOCSUBTYPE>
    <UID>43DB4C60-50CD-4C5F-A0A2-5F72853B2A02</UID>
    <TIN>30300303</TIN>
    <ORGNM>test</ORGNM>
    <POINTNM>Магазин "ТЕСТ-ФСКО"</POINTNM>
    <POINTADDR>УКРАЇНА, м. Київ, вул. Басейна, буд. 5б</POINTADDR>
    <ORDERDATE>08012022</ORDERDATE>
    <ORDERTIME>200843</ORDERTIME>
    <ORDERNUM>188</ORDERNUM>
    <CASHDESKNUM>1007</CASHDESKNUM>
    <CASHREGISTERNUM>4000000000</CASHREGISTERNUM>
    <CASHIER>ТОВ "ТЕСТ-ФСКО" (Печатка)</CASHIER>
    <COMMENT>#####
```

# Тестовий коментар

```
#####
  </COMMENT>
  <VER>1</VER>
  <ORDERTAXNUM>630168</ORDERTAXNUM>
</CHECKHEAD>
<CHECKTOTAL>
  <SUM>362.34</SUM>
  <DISCOUNTSUM>3.66</DISCOUNTSUM>
</CHECKTOTAL>
<CHECKPAY>
  <ROW ROWNUM="1">
    <PAYFORMCD>0</PAYFORMCD>
    <PAYFORMNM>ГОТІВКА</PAYFORMNM>
    <SUM>52.00</SUM>
  </ROW>
  <ROW ROWNUM="2">
    <PAYFORMCD>1</PAYFORMCD>
    <PAYFORMNM>КАРТКА</PAYFORMNM>
    <SUM>310.34</SUM>
    <PAYSYS>
      <ROW ROWNUM="1">
        <NAME>MASTERCARD</NAME>
        <ACQUIRENM>Кашалот банк</ACQUIRENM>
        <ACQUIRETRANSID>048867173620</ACQUIRETRANSID>
        <POSTRANSDATE>07012022173300</POSTRANSDATE>
        <POSTRANSNUM>224</POSTRANSNUM>
        <EPZDETAILS>XXXXXXXXXXXX6883</EPZDETAILS>
        <AUTHCD>68014B</AUTHCD>
        <SUM>310.34</SUM>
      </ROW>
    </PAYSYS>
  </ROW>
</CHECKPAY>
<CHECKTAX>
  <ROW ROWNUM="1">
    <TYPE>0</TYPE>
    <NAME>ПДВ</NAME>
    <LETTER>A</LETTER>
```

```
<PRC>20.00</PRC>
<TURNOVER>366.00</TURNOVER>
<SOURCESUM>353.71</SOURCESUM>
<SUM>58.95</SUM>
</ROW>
<ROW ROWNUM="2">
  <TYPE>1</TYPE>
  <NAME>Акциз</NAME>
  <LETTER>Щ</LETTER>
  <PRC>5.00</PRC>
  <TURNOVER>183.00</TURNOVER>
  <SOURCESUM>181.17</SOURCESUM>
  <SUM>8.63</SUM>
</ROW>
</CHECKTAX>
<CHECKBODY>
  <ROW ROWNUM="1">
    <CODE>АртикулТовару1</CODE>
    <BARCODE>123456789</BARCODE>
    <NAME>НазваТовару1</NAME>
    <UNITCD>2009</UNITCD>
    <UNITNM>шт</UNITNM>
    <AMOUNT>1.000</AMOUNT>
    <PRICE>183.00</PRICE>
    <LETTERS>А</LETTERS>
    <COST>183.00</COST>
    <DISCOUNTTYPE>0</DISCOUNTTYPE>
    <DISCOUNTSUM>1.83</DISCOUNTSUM>
  </ROW>
  <ROW ROWNUM="2">
    <CODE>АртикулТовару2</CODE>
    <BARCODE>9999999999</BARCODE>
    <NAME>НазваТовару2</NAME>
    <UNITCD>2009</UNITCD>
    <UNITNM>шт</UNITNM>
    <AMOUNT>1.000</AMOUNT>
    <PRICE>183.00</PRICE>
    <LETTERS>АЩ</LETTERS>
    <COST>183.00</COST>
    <DISCOUNTTYPE>0</DISCOUNTTYPE>
    <DISCOUNTSUM>1.83</DISCOUNTSUM>
    <EXCISELABELS>
      <ROW ROWNUM="1">
        <EXCISELABEL>1111</EXCISELABEL>
      </ROW>
    </EXCISELABELS>
  </ROW>
</CHECKBODY>
</CHECK>
```

## 6. Фіскалізація чека повернення товарів, послуг **FiscalizeReturnCheck** (FiscalNumberRRO, JSONGoodsLstData, JSONPayData, FiscalNumReturnReceipt)

Для створення та фіскалізації чека повернення товарів та/або послуг необхідно сформувати JSON структуру [JSONGoodsLstData](#), що міститиме дані номенклатури, яка повертається, структуру JSON, структуру [JSONPayData](#) з формами повернення грошей та номер чека, за яким здійснюється повернення.

Параметри, які вказуються при використанні методу **FiscalizeReturnCheck**:

| Ім'я                   | Тип    | Опис                                                    |
|------------------------|--------|---------------------------------------------------------|
| FiscalNumberRRO        | string | Фіскальний номер ПРРО. Ідентифікатор пристрою.          |
| JSONGoodsLstData       | string | JSON структура з описом проданих товарів або послуг.    |
| JSONPayData            | string | JSON структура з підсумками оплати по чека.             |
| FiscalNumReturnReceipt | string | Фіскальний номер чека, за яким здійснюється повернення. |

Структури JSONGoodsLstData та JSONPayData визначені так само, як і в специфікації функції формування чека продажу **FiscalizeCheck**.

У відповіді повертається структура **CashalotApiRetVal**, що містить значення:

|                     |        |                                                                                                                                                          |
|---------------------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return              | bool   | Ознака успішності виконання.                                                                                                                             |
| JsonVal             | string | JSON-структура, що містить значення успішності операції, тип документу, помилки, локальний та фіскальний номери чека, ID зміни та Base64 рядок XML чека. |
| ReceiptFiscalNum    | string | Фіскальний номер чека.                                                                                                                                   |
| ReceiptLocalNum     | string | Локальний номер чека.                                                                                                                                    |
| OfflineMode         | bool   | Ознака перебування в режимі офлайн                                                                                                                       |
| ShiftID             | string | Ідентифікаційний номер поточної зміни.                                                                                                                   |
| FSKORceiptLink      | string | Посилання на перегляд чека в ДПС                                                                                                                         |
| CashalotReceiptLink | string | Посилання на перегляд чека в кабінеті Cashalot                                                                                                           |
| Type                | int    | <a href="#">Тип документа</a> .                                                                                                                          |
| ParentReceiptID     | string | ID фіскального чека який повертається                                                                                                                    |
| ParentFiscalNumber  | string | Фіскальний номер чека який повертається                                                                                                                  |
| XML                 | string | XML-структура фіскалізованого чека.                                                                                                                      |
| ReceiptID           | string | ID (guid) чека                                                                                                                                           |
| ReceiptSum          | string | Сума до сплати по чеку                                                                                                                                   |
| ReceiptDateTime     | string | Дата та час створення у форматі (дд.мм.рррр гг:хв:сс)                                                                                                    |

Приклад **FiscalizeReturnCheck**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
FiscalNumReturnReceipt = "630168"
App.SetParameter "PathToCashalotDir", "C:\ProgramData\Cashalot\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO
JSONGoodsLstData = "{"ReceiptLst":[{"VendorCode":"АртикулТовару1","Name":"НазваТовару1","CdUKDZED":"","CdDKPP":"","GoodsType":"","Barcode":"123456789","UnitType":"шт","Quantity":"1","Price":"183","Amount":"181,17","DiscountPrc":"","DiscountSum":"1.83","VATRate":"20","IsPriceIncludeVAT":true,"SumVAT":"30.2","IsExcise":false,"OtherParametr":null}, {"VendorCode":"АртикулТовару2","Name":"НазваТовару2","CdUKDZED":"","CdDKPP":"","GoodsType":"","Barcode":"999999999","UnitType":"шт","Quantity":"1","Price":"183","Amount":"181,17","DiscountPrc":"","DiscountSum":"1.83","VATRate":"20","IsPriceIncludeVAT":true,"SumVAT":"30.2","IsExcise":false,"OtherParametr":null}]}
```

```

"Quantity": "1", "Price": "183", "Amount": "181,17", "DiscountPrc": "",
"DiscountSum": "1.83", "VATRate": "20", "IsPriceIncludeVAT": true,
"SumVAT": "30.2", "IsExcise": true, "ExciseStampBarcode": "1111",
"OtherParameters": null}], "Comment": "#####\n\r #
Тестовий коментар \n\r#####"}"
JSONPayData =
{"SumCash": "52", "SumPayByCard": "310,34", "SumPayByCredit": null, "SumPayByC
ertificate": null, "SumPayCheck": "362,34", "TerminalID ":
"S1K700J8", "ApprovalCode": "68014B", "RRN":
"048867173620", "IssuerName": "MASTERCARD", "PAN": "XXXXXXXXXX6883", "Transa
ctionDate": "07.01.2022 17:33:00", "AcquireName": "Кашалот
банк", "InvoiceNumber": "224", "CustomerEmail": "test@gmail.com"}"

Set ResultReturnReceipt = App.FiscalizeReturnCheck(FiscalNumberRRO
,JSONGoodsLstData, JSONPayData, FiscalNumReturnReceipt)
MsgBox "Результат виконання = " & ResultReturnReceipt.JsonVal

```

#### Приклад відповіді:

```

Результат виконання =
{"Ret": true, "ErrorString": "", "Values": {"ShiftID": "92b59cb8-f34b-43dd-9cec-d8b187448c
bf", "ReceiptFiscalNumber": "630169", "ReceiptLocalNumber": "189", "OfflineMode": "False",
"Type": "10000001", "Base64Str1251ReceiptXML": "PD94bWwgdmVyc2lvdj0iMS4wIiB1bmNvZGluZz0
id2luZG93cy0xMjUxIj8+DQo8Q0hfFQ0sgeG1sbmM6eHNpPSJodHRwOi8vd3d3LnczLm9yZy8yMDAxL1hNTFN
jaGVtYS1pbmN0YW5jZSIgeHNpOm5vTmFtZXNwYWNlU2NoZW1hTG9jYX....."}

```

## 7. Отримання локального номера чека за його фіскальним номером

### **GetReceiptLocalNumberByFiscalNumber** (FiscalNumberRRO, ReceiptFiscalNumber)

Метод дозволяє визначити локальний номер чека за відомим фіскальним номером чека. Пошук чека відбувається в локальній базі чеків, якщо в локальній базі чек не знайдено, виконується пошук чека на сервері ФСКО.

Параметри **GetReceiptLocalNumberByFiscalNumber**:

| Ім'я                | Тип    | Опис                                          |
|---------------------|--------|-----------------------------------------------|
| FiscalNumberRRO     | string | Фіскальний номер ПРРО. Ідентифікатор пристрою |
| ReceiptFiscalNumber | number | Фіскальний номер чеку                         |

У відповіді надходить локальний номер чека. Якщо номер чека не знайдено, то повертається пустий рядок. У випадку коли задано параметр **NOAUTOOPENSHIFT** = True, виклик функції при закритті зміні переривається і відображається повідомлення "Операція неможлива, необхідно відкрити зміну"

#### Приклад виконання `GetReceiptLocalNumberByFiscalNumber`:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
ReceiptFiscalNumber = "630169"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

result = App.GetReceiptLocalNumberByFiscalNumber(FiscalNumberRRO,
ReceiptFiscalNumber)
MsgBox "Локальний номер = " & result
```

#### Приклад відповіді:

Локальний номер = 189

---

## 8. Отримання фіскального номера чека за його локальним номером **GetReceiptFiscalNumberByLocalNumber** (FiscalNumberRRO, ReceiptLocalNumber)

Метод дозволяє визначити локальний номер чека за відомим фіскальним номером чека. Пошук чека відбувається в локальній базі чеків, якщо в локальній базі чек не знайдено, виконується пошук чека на сервері ФСКО.

У відповіді надходить локальний номер чека. Якщо номер чека не знайдено, то повертається пустий рядок.

Параметри **GetReceiptFiscalNumberByLocalNumber**:

| Ім'я                | Тип    | Опис                                           |
|---------------------|--------|------------------------------------------------|
| FiscalNumberRRO     | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| ReceiptFiscalNumber | number | Фіскальний номер чека.                         |

Приклад виконання **GetReceiptLocalNumberByFiscalNumber**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
ReceiptFiscalNumber = "630169"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

result = App.GetReceiptFiscalNumberByLocalNumber(FiscalNumberRRO,
ReceiptLocalNumber)
MsgBox "Локальний номер = " & result
```

Приклад відповіді:

Локальний номер = 636465

## 9. Отримання xml структури фіскального чека **GetReceiptXML** (FiscalNumberRRO, ReceiptFiscalNumber)

Метод дозволяє отримати XML-структуру зареєстрованого чека за його відомим фіскальним номером.

Параметри **GetReceiptXML**:

| Ім'я                | Тип    | Опис                                           |
|---------------------|--------|------------------------------------------------|
| FiscalNumberRRO     | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| ReceiptFiscalNumber | number | Фіскальний номер чека.                         |

Приклад виконання **GetReceiptXML**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
ReceiptFiscalNumber = "630169"

App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

receiptXML = App.GetReceiptXML(FiscalNumberRRO, ReceiptFiscalNumber)
Wscript.Echo "XML receipt = " & receiptXML
```

У відповіді надходить XML-структура чека

## 10. Отримати перший чек з періоду **GetFirstReceiptByPeriod** (FiscalNumberRRO, dateBeg, dateEnd)

Метод **GetFirstReceiptByPeriod** рекомендовано використовувати в ланцюжку запитів для отримання чеків за певний період часу. Дана функція викликається першою для підготовки списку чеків, і для отримання першого чека з вказаного періоду часу.

Параметри методу **GetFirstReceiptByPeriod**:

| Ім'я            | Тип    | Опис                                           |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| dateBeg         | string | Початок періоду в форматі dd.MM.yyyy.          |
| dateEnd         | string | Кінець періоду в форматі dd.MM.yyyy.           |

### Приклад виконання **GetFirstReceiptByPeriod**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
dateBeg = "01.01.2022"
dateEnd = "09.01.2022"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set firstReceipt = App.GetFirstReceiptByPeriod(FiscalNumberRRO, dateBeg, dateEnd)
MsgBox "Перший чек в періоді = " & firstReceipt.JsonVal
```

У відповіді на **GetFirstReceiptByPeriod** повертається структура **CashalotApiRetVal**, що містить значення першого чека з вказаного періоду:

|                  |        |                                                                                                                                                               |
|------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.                                                                                                                                          |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, тип чека, помилки, локальний та фіскальний номери чека, ID зміни та Base64-кодований рядок XML чека. |
| ReceiptFiscalNum | string | Фіскальний номер.                                                                                                                                             |
| ReceiptLocalNum  | string | Локальний номер.                                                                                                                                              |
| ShiftID          | string | Ідентифікаційний номер поточної зміни.                                                                                                                        |
| Type             | int    | <a href="#">Тип документа.</a>                                                                                                                                |
| XML              | string | XML-структура.                                                                                                                                                |

## 11. Отримати перший чек по зміні **GetFirstReceiptByShift** (FiscalNumberRRO, shiftID)

Метод **GetFirstReceiptByShift** рекомендовано використовувати в ланцюжку запитів для отримання чеків певної касової зміни. Дана функція викликається першою для підготовки списку чеків, і для отримання першого чека з вказаної касової зміни.

Параметри **GetFirstReceiptByShift**:

|                 |        |                                                |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| shiftID         | string | ID зміни.                                      |

#### Приклад:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
shiftID = "92b59cb8-f34b-43dd-9cec-d8b187448cbf"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set firstReceiptByShift = App.GetFirstReceiptByShift(FiscalNumberRRO, shiftID)
MsgBox "Перший чек в зміні = " & firstReceiptByShift.JsonVal
```

У відповіді повертається структура **CashalotApiRetVal**, що містить значення першого чека з заданого періоду:

|                  |        |                                                                                                                                                                               |
|------------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.                                                                                                                                                          |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, тип першого чека, помилки, локальний та фіскальний номери першого чека, ID зміни та Base64 рядок XML структури чека. |
| ReceiptFiscalNum | string | Фіскальний номер першого чека.                                                                                                                                                |
| ReceiptLocalNum  | string | Локальний номер першого чека.                                                                                                                                                 |
| ShiftID          | string | Ідентифікаційний номер поточної зміни.                                                                                                                                        |
| Type             | int    | <a href="#">Тип документа</a> .                                                                                                                                               |
| XML              | string | XML-структура чека.                                                                                                                                                           |

## 12. Отримати наступний чек з періоду **GetNextReceipt** (FiscalNumberRRO)

Метод **GetNextReceipt** використовується в ланцюжку запитів для отримання чеків за певний період часу. Цей метод викликається після виконання **GetFirstReceiptByPeriod** або **GetFirstReceiptByShift**, які готують список чеків. Функцію можна викликати декілька разів, поки не буде отримано весь список чеків. Коли в списку більше немає чеків повертається **false**.

В параметрах **GetNextReceipt** має бути тільки **FiscalNumberRRO** (Фіскальний номер ПРРО. Ідентифікатор пристрою) з типом **string**

#### Приклад виконання **GetNextReceipt**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
dateBeg = "01.01.2022"
dateEnd = "09.01.2022"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set nextReceipt = App.GetFirstReceiptByPeriod(FiscalNumberRRO, dateBeg, dateEnd)
Do while nextReceipt.Return
' Обробити отримані чеки
Set nextReceipt = App.GetNextReceipt(FiscalNumberRRO)
if nextReceipt.Return <> False Then
MsgBox "Наступний чек в періоді = " & nextReceipt.JsonVal
Else
MsgBox "Чеків немає"
End if
Loop
```

У відповіді на **GetNextReceipt** повертається структура **CashalotApiRetVal**, що містить значення наступного чека з вказаного періоду:

|                  |        |                                                                                                                                                     |
|------------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.<br>Коли в списку більше немає чеків, повертається <b>false</b> .                                                               |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, тип чека, помилки, локальний та фіскальний номери чека, ID зміни та Base64 рядок XML чека. |
| ReceiptFiscalNum | string | Фіскальний номер чека.                                                                                                                              |
| ReceiptLocalNum  | string | Локальний номер чека.                                                                                                                               |
| ShiftID          | string | Ідентифікаційний номер поточної зміни.                                                                                                              |
| Type             | int    | <u>Тип документа</u> .                                                                                                                              |
| XML              | string | XML-структура поточного чека.                                                                                                                       |

### 13. Отримати інформацію по поточному стану каси **GetCurrentStatus** (FiscalNumberRRO)

Функція **GetCurrentStatus** використовується для отримання інформації по поточному стану каси. В параметрах **GetCurrentStatus** має бути тільки **FiscalNumberRRO** (Фіскальний номер ПРРО. Ідентифікатор пристрою) з типом **string**.

У відповіді повертається структура **CashalotApiRetVal**, що містить значення наступного чека з заданого періоду:

|         |        |                                                                                                                                                                                  |
|---------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return  | bool   | Успішність операції.                                                                                                                                                             |
| JsonVal | string | JSON-структура, що містить значення <b>"Ret"</b> – успішність операції, <b>"ErrorString"</b> – рядок з описом помилки (в разі невдачі), <b>"Values"</b> – масив параметрів каси. |
| ShiftID | string | Ідентифікаційний номер зміни. Якщо зміна відкрита, то відповідає ідентифікатору поточної зміни, якщо зміна закрита, то відповідає ідентифікатору останньої зміни.                |

Опис параметрів каси, які передаються в масиві **Values** структури **JsonVal**:

|                               |                                                                              |
|-------------------------------|------------------------------------------------------------------------------|
| ShiftState                    | Стан зміни: 1 – закрита, 2 – відкрита.                                       |
| ShiftStateStr                 | Стан зміни: "Closed" – закрита, "Opened" – відкрита.                         |
| ShiftLocalNumber              | Локальний номер зміни.                                                       |
| ShiftFiscalNumber             | Фіскальний номер зміни.                                                      |
| ShiftID                       | Ідентифікатор зміни.                                                         |
| LastCheckLocalNumber          | Локальний номер останнього чека.                                             |
| LastCheckFiscalNumber         | Фіскальний номер останнього чека.                                            |
| LastCheckDateTime             | Мітка часу останнього чека.                                                  |
| IsOfflineMode                 | Стан офлайн-режиму каси: 0 – каса в онлайн-режимі, 1 – каса в офлайн-режимі. |
| BusinesUnitName               | Найменування ГО ПРРО                                                         |
| BusinesUnitAddress            | Адреса ГО ПРРО                                                               |
| NextLocalNumber               | Наступний локальний номер чека.                                              |
| CashBalance                   | Значення залишків готівки в касі.                                            |
| Base64Str1251CurrentStatusXML | XML-рядок вказаних вище параметрів в форматі Base64.                         |

#### Приклад виконання **GetCurrentStatus**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set statusRRO = App.GetCurrentStatus(FiscalNumberRRO)
MsgBox "Інформація про ПРРО = " & statusRRO.JsonVal
```

#### Приклад відповіді:

```
Інформація про ПРРО =
{"Ret":true,"ErrorString":"","Values":{"ShiftState":"2","ShiftStateStr":"Opened","ShiftLocalNumber":"178","ShiftFiscalNumber":"630158","ShiftID":"92b59cb8-f34b-43dd-9cec-d8b187448cbf","LastCheckLocalNumber":"190","LastCheckFiscalNumber":"630170","LastCheckDateTime":"2022-01-09T16:16:44","IsOfflineMode":"0","NextLocalNumber":"191","CashBalance":"3055","Base64Str1251CurrentStatusXML":"PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz48RG9jdW1lbnRpdXRwdXRQYXJhbWV0ZXJzPjxQYXJhbWV0ZXJzIENoZWNrTnVtYmVyPSIxOTAiIENhc2hCYWxhbmNlPSIzMDU1IiBJc09mZmxpbmVNb2RlPSIwIiB0ZXh0TG9jYWx0dW1iZXI9IjE5MSIgTGZzdENoZWNrmlzY2FsTnVtYmVyPSI2MzAxNzAiIFNoaWZ0TnVtYmVyPSIxNzgiIFNoaWZ0SUQ9IjkyYjU5Y2I4LWYzNGItNDNkZC05Y2VjLWQ4YjE4NzQ0OGNiZiIgU2hpZnRgaXNjYWx0dW1iZXI9IjYzMDE0CIgU2hpZnRtdGF0ZT0iMiIgRGF0ZVRpbWU9IjIwMjItMDEtMDU1MTY6MTY6NDQlZ48L0RvY3VtZW50T3V0cHV0UGFyYW1ldGVyc24="}}}
```

#### 14. Отримати перший Z-звіт за період **GetFirstZReportByPeriod** (FiscalNumberRRO, dateBeg, dateEnd)

Метод **GetFirstZReportByPeriod** рекомендовано використовувати в ланцюжку запитів для отримання Z-звітів за певний період часу. Дана функція викликається першою для підготовки списку Z-звітів і для отримання першого Z-звіту з вказаного періоду часу.

#### Параметри **GetFirstZReportByPeriod**:

|                 |        |                                                |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| dateBeg         | string | Початок періоду в форматі dd.MM.yyyy.          |
| dateEnd         | string | Кінець періоду в форматі dd.MM.yyyy.           |

#### Приклад виконання **GetFirstZReportByPeriod**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
dateBeg = "01.12.2021"
dateEnd = "10.01.2022"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set firstZReport = App.GetFirstZReportByPeriod(FiscalNumberRRO, dateBeg, dateEnd)
MsgBox "Z-звіт = " & firstZReport.JsonVal
```

У відповіді повертається структура **CashalotApiRetVal**, що містить значення першого Z-звіту з заданого періоду:

|                  |        |                                                                                                                                                                                   |
|------------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.                                                                                                                                                              |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, помилки, тип документу Z-звіту, локальний та фіскальний номери Z-звіту, ID зміни та Base64 рядок XML структури Z-звіту . |
| ReceiptFiscalNum | string | Фіскальний номер першого Z-звіту.                                                                                                                                                 |
| ReceiptLocalNum  | string | Локальний номер першого Z-звіту.                                                                                                                                                  |
| ShiftID          | string | Ідентифікаційний номер поточної зміни.                                                                                                                                            |
| Type             | int    | <a href="#">.Тип документа</a> Z-звіту - 11000000.                                                                                                                                |
| XML              | string | XML-структура Z-звіту.                                                                                                                                                            |

## 15. Отримати наступний Z-звіт з періоду **GetNextZReport** (FiscalNumberRRO)

Метод **GetNextZReport** використовується в ланцюжку запитів для отримання Z-звітів за певний період часу. Цей метод викликається після виклику **GetFirstZReportByPeriod**, яка готує список Z-звітів. Можна викликати декілька разів, поки не буде отримано весь список Z-звітів. Коли в списку більше немає чеків, повертається **false**.

В параметрах **GetNextZReport** має бути тільки **FiscalNumberRRO** (Фіскальний номер ПРРО. Ідентифікатор пристрою) з типом **string**

### Приклад виконання **GetNextZReport**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
dateBeg = "01.12.2021"
dateEnd = "10.01.2022"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set nextZReport = App.GetFirstZReportByPeriod(FiscalNumberRRO, dateBeg, dateEnd)

Do while nextZReport.Return
Set nextZReport = App.GetNextZReport(FiscalNumberRRO)
if nextZReport.Return <> False Then
MsgBox "Наступний Z-звіт в періоді = " & nextZReport.ReceiptFiscalNum
Else
MsgBox "Z-звітів немає"
End if
Loop
```

У відповіді повертається структура **CashalotApiRetVal**, що містить значення наступного Z-звіту з заданого періоду:

|                  |        |                                                                                                                                                                          |
|------------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.<br>Коли в списку більше нема Z-звіту повертається <a href="#">false</a> .                                                                           |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, тип документу, помилки, локальний та фіскальний номери Z-звіту, ID зміни та Base64 рядок XML структури Z-звіту. |
| ReceiptFiscalNum | string | Фіскальний номер Z-звіту.                                                                                                                                                |
| ReceiptLocalNum  | string | Локальний номер Z-звіту.                                                                                                                                                 |
| ShiftID          | string | Ідентифікаційний номер поточної зміни.                                                                                                                                   |
| Type             | int    | <a href="#">Тип документа</a> .                                                                                                                                          |
| XML              | string | XML-структура Z-звіту.                                                                                                                                                   |

## 16. Отримати Z-звіт касової зміни **GetZReportByShift** (FiscalNumberRRO, ShiftID)

Метод **GetZReportByShift** призначений для отримання Z-звіту вказаної касової зміни.

### Параметри **GetZReportByShift**:

|                 |        |                                                |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| ShiftID         | string | Ідентифікатор касової зміни.                   |

### Приклад виконання **GetZReportByShift**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
shiftID = "92b59cb8-f34b-43dd-9cec-d8b187448cbf"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set getZReport = App.GetZReportByShift(FiscalNumberRRO, shiftID)
MsgBox "Z-звіт обраної зміни = " & getZReport.JsonVal
```

У відповіді на **GetZReportByShift** повертається структура **CashalotApiRetVal**, що містить значення Z-звіту вказаної касової зміни:

|                  |        |                                                                                                                                                                                             |
|------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.                                                                                                                                                                        |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, помилки, тип документу Z-звіту (11000000), локальний та фіскальний номери Z-звіту, ID зміни та Base64 рядок XML структури Z-звіту. |
| ReceiptFiscalNum | string | Фіскальний номер чека Z-звіту.                                                                                                                                                              |
| ReceiptLocalNum  | string | Локальний номер чека Z-звіту.                                                                                                                                                               |
| ShiftID          | string | Ідентифікаційний номер поточної зміни.                                                                                                                                                      |
| Type             | int    | <a href="#">Тип документа</a> .                                                                                                                                                             |
| XML              | string | XML-структура Z-звіту.                                                                                                                                                                      |

## 17. Вивести візуальне відображення чека на екран **ShowReceipt** (FiscalNumberRRO, ReceiptFiscalNumber)

Метод **ShowReceipt** використовується для виклику вікна з візуальним відображенням зареєстрованого чека. Вікно з відображенням чека також дозволяє використати додаткові функції друку або відправки чека на електронну пошту, кнопки яких присутні у вікні чека.

### Параметри **ShowReceipt**:

|                     |        |                                                |
|---------------------|--------|------------------------------------------------|
| FiscalNumberRRO     | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| ReceiptFiscalNumber | number | Фіскальний номер чека.                         |

### Приклад виконання **ShowReceipt**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
receiptFiscalNum = "626726"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

openReceipt = App.ShowReceipt(FiscalNumberRRO, receiptFiscalNum)
MsgBox openReceipt
```

У відповіді відкриється чек та повертається інформація про успішність виконання True або False.

## 18. Формування звіту про реалізовані товари **PrintReportGoods** (FiscalNumberRRO)

Метод PrintReportGoods використовується для формування звіту про реалізовані товари.

### Параметри **PrintReportGoods**:

|                 |        |                                                |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
|-----------------|--------|------------------------------------------------|

### Приклад виконання **PrintReportGoods**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO
Set Report = App.PrintReportGoods (FiscalNumberRRO)
```

У відповідь на PrintReportGoods відкривається діалогове вікно з можливістю обрати товар за яким необхідно сформувати звіт, за необхідності вказавши додаткові параметри.

# 19. Друкувати періодичний звіт по датам **PrintPReportDate** (FiscalNumberRRO, dateBeg, dateEnd, isShort)

Метод **PrintPReportDate** використовується для можливості друку періодичного звіту, використовуючи проміжки часу, що встановлюється між **dateBeg** та **dateEnd** у форматі "dd.mm.yyyy".

## Параметри **PrintPReportDate**:

|                        |               |                                                |
|------------------------|---------------|------------------------------------------------|
| <b>FiscalNumberRRO</b> | <b>string</b> | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| <b>dateBeg</b>         | <b>string</b> | Початок періоду в форматі dd.MM.yyyy.          |
| <b>dateEnd</b>         | <b>string</b> | Кінець періоду в форматі dd.MM.yyyy.           |
| <b>isShort</b>         | <b>string</b> | Ознака скороченого звіту True/False.           |

## Приклад виконання **PrintPReportDate**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
dateBeg = "01.12.2021"
dateEnd = "31.12.2021"
isShort = "False"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO
pReport = App.PrintPReportDate (FiscalNumberRRO, dateBeg, dateEnd, isShort)
MsgBox pReport
```

У відповіді на **PrintPReportDate** відкривається періодичний звіт (повний або скорочений) та повертається інформація про успішність виконання True або False.

## 20. Надрукувати довільний нефіскальний текст **PrintTextDocument** (FiscalNumberRRO, DocumentPackage)

Метод **PrintTextDocument** використовується для друку на чекову стрічку довільного тексту. Після виклику відображається вікно з зображенням тексту і можливістю друку. У разі потреби, можна викликати без відкриття касової зміни.

У випадку, коли задано автоматичну відправку чеків на друк і принтер для друку за замовчуванням, друк заданого тексту відбувається без попереднього перегляду:

```
App.SetParameter "AUTOPRINTMODE", "True"  
App.SetParameter "DEFAULTPRINTERNAME", "xprint80-name")
```

### Параметри **PrintTextDocument**:

|                        |               |                                                |
|------------------------|---------------|------------------------------------------------|
| <b>FiscalNumberRRO</b> | <b>string</b> | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| <b>DocumentPackage</b> | <b>string</b> | Текст для друку.                               |

### Приклад виконання **PrintTextDocument**:

```
Set App = CreateObject ("AddIn.CashaLotApi")  
FiscalNumberRRO = "4000000000"  
DocumentPackage = "З'їж цих м'яких київських перепічок і випий філіжанку львівської  
кави."  
App.SetParameter "PathToCashalotDir", "C:\Cashalot"  
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO  
  
printText = App.PrintTextDocument(FiscalNumberRRO, DocumentPackage)  
MsgBox printText
```

У відповіді на **PrintTextDocument** відкривається вікно з вказаним текстом та повертається інформація про успішність виконання True або False.

## 21. Службове внесення грошей в касу **ServiceInput** (FiscalNumberRRO, Amount)

Для виконання службового внесення коштів в касу використовують метод **ServiceInput**. Якщо задано параметр NOINTERFACEMODE = "False" (за допомогою функції **SetParameter**), то при цьому також виводиться службове вікно для підтвердження операції. Якщо задано параметр NOAUTOOPENSHIFT = "True", виклик методу при закритій зміні переривається, відображається повідомлення "Операція неможлива, необхідно відкрити зміну" і повертається **false**. При успішному виконанні відкривається чек службового внесення та повертається **true**.

### Параметри **ServiceInput**:

|                        |               |                                                      |
|------------------------|---------------|------------------------------------------------------|
| <b>FiscalNumberRRO</b> | <b>string</b> | Фіскальний номер ПРРО. Ідентифікатор пристрою.       |
| <b>Amount</b>          | <b>string</b> | Додатне значення суми внесення у форматі <Грн, Коп>. |

#### Приклад виконання **ServiceInput**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
Amount = "15.65"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO
deposit = App.ServiceInput(FiscalNumberRRO, Amount)
MsgBox deposit
```

## 22. Службове внесення грошей в касу **ServiceInputEx** (FiscalNumberRRO, Amount)

Параметри використовуються такі ж, як і при виконанні **ServiceInput**. Різниця між **ServiceInput** та **ServiceInputEx** у відповіді, яку отримуємо на виконання методу.

У відповіді на **ServiceInputEx** повертається структура **CashalotApiRetVal**, що містить значення:

|                  |        |                                                                                                                                 |
|------------------|--------|---------------------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.                                                                                                            |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, помилки, локальний та фіскальний номери документа службового внесення. |
| ShiftID          | string | Ідентифікатор зміни.                                                                                                            |
| ReceiptFiscalNum | string | Фіскальний номер чека.                                                                                                          |
| ReceiptLocalNum  | string | Локальний номер чека.                                                                                                           |
| Type             | int    | <a href="#">Тип документа.</a>                                                                                                  |

#### Приклад виконання **ServiceInputEx**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
Amount = "15.65"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set depositRet = App.ServiceInputEx(FiscalNumberRRO, Amount)
MsgBox "JSON = " & depositRet.JsonVal
```

## 23. Службова видача грошей з каси **ServiceOutput** (FiscalNumberRRO, Amount)

Для виконання службової видачі з каси використовують функцію **ServiceOutput**. Якщо задано параметр NOINTERFACEMODE = "False" (за допомогою функції **SetParameter**), то при цьому також виводиться службове вікно для підтвердження операції. Якщо задано параметр NOAUTOOPENSHIFT= "True", виклик функції при закритій зміні переривається, та з'являється повідомлення "Операція неможлива, необхідно відкрити зміну". При успішному виконанні відкривається чек службової видачі та повертається **true**.

#### Параметри **ServiceOutput**:

|                 |        |                                                      |
|-----------------|--------|------------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою.       |
| Amount          | string | Додатне значення суми внесення у форматі <Грн, Коп>. |

#### Приклад виконання **ServiceOutput**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
Amount = "15.65"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

withdrawRet = App.ServiceOutput(FiscalNumberRRO, Amount)
```

#### 24. Службова видача грошей з каси **ServiceOutputEx** (FiscalNumberRRO, Amount)

Параметри використовуються такі ж, як і при виконанні **ServiceOutput**. Різниця між **ServiceOutput** та **ServiceOutputEx** у відповіді, яку отримуємо на виконання методу. У відповіді на **ServiceOutputEx** повертається структура **CashalotApiRetVal**, що містить значення:

|                  |        |                                                                                                             |
|------------------|--------|-------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.                                                                                        |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, помилки, локальний та фіскальний номери документа. |
| ShiftID          | string | Ідентифікатор зміни.                                                                                        |
| ReceiptFiscalNum | string | Фіскальний номер чека.                                                                                      |
| ReceiptLocalNum  | string | Локальний номер чека.                                                                                       |
| Type             | int    | <a href="#">Тип документа.</a>                                                                              |

#### Приклад виконання **ServiceOutputEx**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
Amount = "15.65"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set withdrawRet = App.ServiceInputEx(FiscalNumberRRO, Amount)
MsgBox "JSON = " & withdrawRet.JsonVal
```

#### 25. Отримання опису останньої помилки **GetLastError()**

У разі неуспішної операції виклик цього методу дозволяє отримати опис останньої помилки. **GetLastError** немає вхідних параметрів. Відповідь має тип **string**

#### Приклад виконання **GetLastError**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

lastError = App.GetLastError()
MsgBox lastError
```

## 26. Формування X-звіту **PrintXReport** (FiscalNumberRRO)

Метод **PrintXReport** дозволяє викликати на екран X-звіт для перегляду та відправки на друк. В параметрах **PrintXReport** має бути тільки **FiscalNumberRRO** (Фіскальний номер ПРРО. Ідентифікатор пристрою) з типом **string**. При успішному виконанні відкривається X-звіт та повертається **true**.

### Приклад виконання **PrintXReport**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
App.SetParameter "PathToCshalotDir", "C:\Cshalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

XReport = App.PrintXReport(FiscalNumberRRO)
MsgBox XReport
```

## 27. Формування X-звіту **GetXReport** (FiscalNumberRRO, ShowPrintReport)

**GetXReport** дозволяє сформувати і програмно отримати X-звіт у вигляді JSON-структури.

### Параметри:

|                 |        |                                                                                                                                                                                                                 |
|-----------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою.                                                                                                                                                                  |
| ShowPrintReport | bool   | Ознака необхідності вивести звіт на друк.<br>Якщо "true", то на екран виводиться візуальне відображення X-звіту для перегляду та можливість вручну роздрукувати звіт (аналогічно функції <b>PrintXReport</b> ). |

### Приклад виконання **GetXReport**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
ShowPrintReport = "True"
App.SetParameter "PathToCshalotDir", "C:\Cshalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set XReport = App.GetXReport(FiscalNumberRRO, ShowPrintReport)
MsgBox XReport.JsonVal
```

У відповіді повертається структура **CshalotApiRetVal**, що містить значення:

|         |        |                                                 |
|---------|--------|-------------------------------------------------|
| Return  | bool   | Успішність операції.                            |
| JsonVal | string | JSON-структура, яка містить інформацію X-звіту. |
| ShiftId | string | Ідентифікатор зміни, в якій сформовано X-звіт.  |
| Type    | int    | <a href="#">Тип документа</a> .                 |
| XML     | string | XML-структура X-звіту.                          |

В **JsonVal**, а саме в “**Values**”, надходять наступні об'єкти:

|                        |                                                                                            |
|------------------------|--------------------------------------------------------------------------------------------|
| ServiceInput           | Загальна сума службових внесень по зміні                                                   |
| ServiceOutput          | Загальна сума службових видач по зміні                                                     |
| OrderCount             | Кількість чеків продажу                                                                    |
| OrderExpCount          | Кількість чеків повернення                                                                 |
| SumCash                | Поточний залишок готівки в касі                                                            |
| TotalSUM               | Загальна сума продажу                                                                      |
| TotalSUMCash           | Загальна сума продажу "Готівка"                                                            |
| TotalSUMCard           | Загальна сума продажу "Картка"                                                             |
| TotalTAXSUM            | Загальна сума ПДВ по продажу                                                               |
| ReturnSUM              | Загальна сума повернення                                                                   |
| ReturnSUMCash          | Загальна сума повернення "Готівка"                                                         |
| ReturnSUMCard          | Загальна сума повернення "Картка"                                                          |
| ReturnTAXSUM           | Загальна сума ПДВ по продажу                                                               |
| TotalTAX_<1>_<2>%_<3>  | Деталізація податків по фіскальним чекам                                                   |
| ReturnTAX_<1>_<2>%_<3> | Деталізація податків по чекам повернення                                                   |
| Base64Str1251ReportXML | Дані Z-звіту у вигляді XML-структури в кодуванні Windows-1251 конвертованої в Base64-рядок |

Назви об'єктів податків чеків оплати **TotalTAX** та повернення **ReturnTAX** побудовані з використанням складових, що визначаються на основі карток проданих товарів:

| Складова | Зміст складової                                  | Можливі значення | Приклад                                                                       |
|----------|--------------------------------------------------|------------------|-------------------------------------------------------------------------------|
| <1>      | Літера податку<br>зазначена в<br>картках товарів | A, Б, В, ...     | TotalTAX_A_20%_SUM<br>ReturnTAX_Б_5%_TURNOVER<br>TotalTAX_Є_10%_REM1OVEDTAXES |
| <2>      | Відсоток податку                                 | 0, 5, 20, ...    |                                                                               |
| <3>      | Сума                                             | SUM              |                                                                               |
|          | Обіг                                             | TURNOVER         |                                                                               |
|          | Знижка                                           | DISCOUNTSUM      |                                                                               |
|          | Виключені податки                                | REMOVEDTAXES     |                                                                               |

## 28. Отримати останній Z-звіт **GetLastZReport** (FiscalNumberRRO, ShowPrintReport)

Виклик функції **GetLastZReport** дозволяє отримати інформацію по останньому зареєстрованому Z-звіту у вигляді структури **CashalotApiRetVal**.

### Параметри:

|                 |        |                                                                                                                                                                                                                 |
|-----------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою.                                                                                                                                                                  |
| ShowPrintReport | bool   | Ознака необхідності вивести звіт на друк.<br>Якщо "true", то на екран виводиться візуальне відображення X-звіту для перегляду та можливість вручну роздрукувати звіт (аналогічно функції <b>PrintXReport</b> ). |

### Приклад виконання **GetLastZReport**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
ShowPrintReport = "True"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set ZReport = App.GetLastZReport(FiscalNumberRRO, ShowPrintReport)
MsgBox ZReport.JsonVal
```

У відповіді повертається структура **CashalotApiRetVal**, що містить значення:

|                  |        |                                                                                                           |
|------------------|--------|-----------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.                                                                                      |
| JsonVal          | string | JSON-структура, яка містить інформацію останнього зареєстрованого Z-звіту. <a href="#">Опис JsonVal</a> . |
| ShiftId          | string | Ідентифікатор зміни, в якій сформовано Z-звіт.                                                            |
| Type             | int    | <a href="#">Тип документа</a> .                                                                           |
| XML              | string | XML-структура Z-звіту.                                                                                    |
| ReceiptFiscalNum | string | Фіскальний номер документа.                                                                               |
| ReceiptLocalNum  | string | Локальний номер документа.                                                                                |

## 29. Формування Z-звіту **PrintZReport** (FiscalNumberRRO)

Метод **PrintZReport** призводить до виконання послідовності операцій аналогічних до виклику функції **CloseShift** в такій послідовності: службова видача з каси наявної готівки, формування Z-звіту та закриття зміни. Проте, виконання методу **PrintZReport** повертає ознаку успішності операції (**true/false**), в той час як виконання **CloseShift** повертає структуру **CashalotApiRetVal**.

В параметрах **PrintZReport** має бути тільки **FiscalNumberRRO** (Фіскальний номер ПРРО. Ідентифікатор пристрою) з типом **string**. При успішному виконанні відкривається Z-звіт та повертається **true**.

**Увага!** Якщо встановлений параметр **NOINTERFACEMODE** = **True**, то службову видачу, потрібно передбачити перед виконанням методу **PrintZReport**.

#### Приклад PrintZReport:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000015625"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

createZReport = App.PrintZReport (FiscalNumberRRO)
MsgBox createZReport
```

### 30. Закриття поточної зміни CloseShift (FiscalNumberRRO)

Метод **CloseShift** використовується для закриття поточної зміни. В параметрах **CloseShift** має бути тільки **FiscalNumberRRO** (Фіскальний номер ПРРО. Ідентифікатор пристрою) з типом **string**.

**Увага!** Якщо встановлений параметр **NOINTERFACEMODE** = **True**, то службову видачу, потрібно передбачити перед виконанням методу **CloseShift**.

#### Приклад виконання CloseShift:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "40000000000"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set finishShift = App.CloseShift(FiscalNumberRRO)
MsgBox "JSON = " & finishShift.JsonVal
```

У відповіді на **CloseShift** повертається структура **CashalotApiRetVal**, що містить значення Z-звіту закритої касової зміни:

|                  |        |                                                                                                           |
|------------------|--------|-----------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.                                                                                      |
| JsonVal          | string | JSON-структура, яка містить інформацію останнього зареєстрованого Z-звіту. <a href="#">Опис JsonVal</a> . |
| ShiftId          | string | Ідентифікатор зміни, в якій сформовано Z-звіт.                                                            |
| Type             | int    | <a href="#">Тип документа</a> .                                                                           |
| XML              | string | XML-структура Z-звіту.                                                                                    |
| ReceiptFiscalNum | string | Фіскальний номер документа.                                                                               |
| ReceiptLocalNum  | string | Локальний номер документа.                                                                                |

### 31. Метод ручного переведення ПРРО в режим офлайн SetOfflineMode (FiscalNumberRRO, AutoExitToOnline)

Метод використовується для ручного переведення ПРРО в режим офлайн.

Даний метод містить такі параметри:

|                  |        |                                                                                                                                                                                                                                                 |
|------------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| FiscalNumberRRO  | string | Фіскальний номер ПРРО                                                                                                                                                                                                                           |
| AutoExitToOnline | bool   | Ознака автоматичного повернення в онлайн.<br>"True" - автоматично повертатись в режим онлайн при наявності зв'язку з ФСКО;<br>"True" - повертатись в онлайн лише при виклику методу <a href="#">TryGoToOnlineMode</a> або після закриття зміни. |

Виклик **SetOfflineMode** не означає автоматичний перехід в офлайн одразу в момент виклику, перехід до офлайну буде здійснено при будь-якій наступній фіскальній операції(відкриття/закриття зміни, службовій видачі/внесенню або реєстрації чека) після виклику **SetOfflineMode**.

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненням параметром Return(Ознака успішності виконання).

**Приклад виконання SetOfflineMode:**

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
AutoExitToOnline = False
Amount = "15.65"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set ret = App.SetOfflineMode (FiscalNumberRRO, AutoExitToOnline)
MsgBox ret.Return
App.ServiceInput(FiscalNumberRRO, Amount)
```

---

### 32. Метод ручного повернення ПРРО в режим онлайн **TryGoToOnlineMode** (FiscalNumberRRO)

Метод використовується для ручного переведу ПРРО в режим онлайн. В якості параметру метод **TryGoToOnlineMode** містить фіскальний номер ПРРО(**FiscalNumberRRO**).

**Приклад виконання TryGoToOnlineMode:**

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set ret = App.TryGoToOnlineMode(FiscalNumberRRO)
MsgBox ret.Return
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненням параметром Return(Ознака успішності виконання).

---

### 33. Синхронізація залишків по товарам **SyncBalanceOnGoods** (FiscalNumberRRO)

Метод призначений для ручної синхронізації залишків по товарам з кабінету користувача в локальну базу даних Cashalot. Для можливості виконання методу необхідно встановити **параметр ведення обліку товарів в API** в режимі 1 або 2 та ознаку ведення обліку товарів в кабінеті користувача в розділі «Склад - Налаштування - Облік товарів у ПРРО». В якості параметру метод **SyncBalanceOnGoods** містить фіскальний номер ПРРО (**FiscalNumberRRO**).

**Приклад виконання SyncBalanceOnGoods:**

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set ret = App.SyncBalanceOnGoods(FiscalNumberRRO)
MsgBox ret.Return
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненням параметром Return(Ознака успішності виконання).

---

#### 34. Синхронізація товарів та залишків **SyncGoodsWithBalance** (FiscalNumberRRO)

Метод призначений для ручної синхронізації товарів та залишків по ним з кабінету користувача в локальну базу даних Cashalot. Для можливості виконання методу необхідно встановити **параметр ведення обліку товарів в API** в режимі 1 або 2 та ознаку ведення обліку товарів в кабінеті користувача в розділі «Склад - Налаштування - Облік товарів у ПРРО». В якості параметру метод **SyncGoodsWithBalance** містить фіскальний номер ПРРО (FiscalNumberRRO).

##### Приклад виконання **SyncGoodsWithBalance**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set ret = App.SyncGoodsWithBalance(FiscalNumberRRO)
MsgBox ret.Return
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненням параметром Return(Ознака успішності виконання).

---

#### 35. Метод сторнування чеків продажу/повернення **StornoCheck** (FiscalNumberRRO)

Метод призначений для сторнування зареєстрованого чека продажу, чека повернення, службового внесення та службової видачі за умови, що чек є зареєстрованим останнім за датою та часом у відкритій зміні. В якості параметру метод **StornoCheck** містить фіскальний номер ПРРО (FiscalNumberRRO)

##### Приклад виконання **StornoCheck**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set Storno = App.StornoCheck(FiscalNumberRRO)

Wscript.Echo "JsonVal = " & Storno.JsonVal
MsgBox "Фіскальний номер чека = " & Storno.ReceiptFiscalNum
MsgBox "Локальний номер чека = " & Storno.ReceiptLocalNum
MsgBox "Успіх реєстрації = " & Storno.Return
MsgBox "Тип чека = " & Storno.Type
MsgBox "ID зміни = " & Storno.ShiftID
Wscript.Echo "XML зареєстрованого чека = " & Storno.XML
```

У відповіді повертається структура **CashalotApiRetVal**, що містить значення:

|                  |        |                                                                                                                                                                                                             |
|------------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.                                                                                                                                                                                        |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, тип документу, помилки, локальний та фіскальний номери чека сторно, фіскальний номер чека який був сторонований, ID зміни та Base64 рядок XML чека |
| ShiftId          | string | Ідентифікатор зміни, в якій сформовано чек сторно.                                                                                                                                                          |
| Type             | int    | <a href="#">Тип документа</a> .                                                                                                                                                                             |
| XML              | string | XML-структура чека сторно.                                                                                                                                                                                  |
| ReceiptFiscalNum | string | Фіскальний номер документа.                                                                                                                                                                                 |
| ReceiptLocalNum  | string | Локальний номер документа.                                                                                                                                                                                  |

### 36. Метод відображення QR-коду зареєстрованого чека **ShowReceiptQRCode** (FiscalNumberRRO, ReceiptFiscalNumber)

Метод призначений для візуалізації QR-кода з посиланням для перегляду зареєстрованого чека на ресурсі ДПС. В якості параметрів передається інформація про фіскальний номер ПРРО та фіскальний номер чека по якому необхідно здійснити відображення QR-кода.

Параметри **ShowReceiptQRCode**

|                     |        |                                                |
|---------------------|--------|------------------------------------------------|
| FiscalNumberRRO     | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| ReceiptFiscalNumber | number | Фіскальний номер чека.                         |

#### Приклад виконання **ShowReceiptQRCode**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
receiptFiscalNum = "626726"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set ShowQrCode = App.ShowReceiptQRCode (FiscalNumberRRO, ReceiptFiscalNumber)

Wscript.Echo "JsonVal = " & ShowQrCode.JsonVal
MsgBox "Фіскальний номер чека = " & ShowQrCode.ReceiptFiscalNum
MsgBox "Локальний номер чека = " & ShowQrCode.ReceiptLocalNum
MsgBox "Успіх реєстрації = " & ShowQrCode.Return
MsgBox "Тип чека = " & ShowQrCode.Type
MsgBox "ID зміни = " & ShowQrCode.ShiftID
Wscript.Echo "XML зареєстрованого чека = " & ShowQrCode.XML
```

У відповіді повертається структура **CashalotApiRetVal**, що містить значення:

|                  |        |                                                                                                                                                         |
|------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Успішність операції.                                                                                                                                    |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, тип документу, помилки, локальний та фіскальний номери чека, ID зміни та Base64 рядок XML чека |
| ShiftId          | string | Ідентифікатор зміни, в якій сформовано чек.                                                                                                             |
| Type             | int    | <a href="#">Тип документа.</a>                                                                                                                          |
| XML              | string | XML-структура чека.                                                                                                                                     |
| ReceiptFiscalNum | string | Фіскальний номер документа.                                                                                                                             |
| ReceiptLocalNum  | string | Локальний номер документа.                                                                                                                              |

### 37. Метод відправки чека покупцю за допомогою сервісів доставки **SendCheckToCustomer** (FiscalNumberRRO, ReceiptFiscalNumber, SendingAddressType, SendingAddress)

Метод використовується для відправки покупцю зареєстрованого чека продажу/повернення за допомогою сервісів доставки SMS, Viber або e-mail. Відправка чека можлива за умови, що ПРРО перебуває в режимі онлайн. В якості параметрів передається інформація про фіскальний номер ПРРО, фіскальний номер чека який необхідно відправити, адреса отримувача (номер телефону або електронна пошта) та використовуваний сервіс для відправки.

#### Параметри **SendCheckToCustomer**:

|                     |        |                                                                                                                                                                                                                                                  |
|---------------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| FiscalNumberRRO     | string | Фіскальний номер ПРРО. Ідентифікатор пристрою.                                                                                                                                                                                                   |
| ReceiptFiscalNumber | number | Фіскальний номер чека продажу або повернення                                                                                                                                                                                                     |
| SendingAddressType  | string | Сервіс для відправки:<br>1 - Viber;<br>2 - SMS;<br>3- E-mail                                                                                                                                                                                     |
| SendingAddress      | string | Е-mail або номер телефону одержувача в залежності від обраного типу відправки. Номер телефону необхідно зазначати з урахуванням маски +380.<br><br>Приклад: <a href="mailto:cashier@cashalot.org.ua">cashier@cashalot.org.ua</a> , +380991002030 |

Для коректного функціонування метода попередньо слід виконати налаштування сервісів для відправки SMS та Viber в кабінеті користувача Cashalot ( <https://my.cashalot.org.ua/> ). Для можливості використання електронної пошти необхідно виконати її налаштування в програмі Cashalot в розділі «Налаштування - Загальні налаштування - Господарські одиниці - Електронна пошта».

#### Приклад виконання **SendCheckToCustomer**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000024021"
ReceiptFiscalNumber = "1350315"
sendingAddressType = "1"
sendingAddress = "+380976010677"

App.SetParameter "PathToCashalotDir",
"C:\Users\User\AppData\Local\Cashalot"
App.SetParameter "DeviceIDFnRRO", "4000024021"

Set SendCheck = App.SendCheckToCustomer (FiscalNumberRRO,
ReceiptFiscalNumber, SendingAddressType, SendingAddress)

Wscript.Echo "Ознака успішності виконання операції = " &
SendCheck.Return
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненим параметром Return(Ознака успішності виконання).

---

#### 38. Отримання переліку налаштованих пристроїв POS-терміналів в програмі **GetPOSTerminalList** (FiscalNumberRRO)

Метод призначений для отримання списку та ID налаштованих пристроїв платіжних терміналів в програмі в розділі «Налаштування - Додаткове обладнання - POS-термінали».

#### Параметри **GetPOSTerminalList**:

|                        |               |                                                |
|------------------------|---------------|------------------------------------------------|
| <b>FiscalNumberRRO</b> | <b>string</b> | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
|------------------------|---------------|------------------------------------------------|

#### Приклад виконання **GetPOSTerminalList**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000024021"
App.SetParameter "PathToCashalotDir",
"C:\Users\User\AppData\Local\Cashalot"
App.SetParameter "DeviceIDFnRRO", "4000024021"

Set ListPOSTerminal = App.GetPOSTerminalList (FiscalNumberRRO)

MsgBox "Ознака успішності виконання операції = " &
ListPOSTerminal.Return
MsgBox "Список налаштованих терміналів = " & ListPOSTerminal.JsonVal
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненим параметром Return(Ознака успішності виконання) та JsonVal зі списком налаштованих POS-терміналів в форматі JSON.

Опис параметрів платіжного термінала отриманих в JsonVal:

| Назва параметра | Призначення                                                         |
|-----------------|---------------------------------------------------------------------|
| ID              | ID платіжного POS-терміналу                                         |
| IPAddress       | IP адреса за яким підключений платіжний термінал                    |
| Port            | Порт за яким підключений платіжний термінал                         |
| DriverPath      | Шлях до драйвера (емулятора)                                        |
| MerchantName    | Назва банку еквайра                                                 |
| MerchantID      | Номер банку-еквайра (мерчанта)                                      |
| ComPort         | COM порт платіжного термінала                                       |
| DefaultDevice   | Ознака пристрою який використовується за замовчуванням (true/false) |
| NameTerminal    | Назва платіжного термінала                                          |

### 39. Встановлення платіжного POS-термінала пристроєм за замовчуванням **SetDefaultPOSTerminal** (FiscalNumberRRO, ID)

Метод призначений для встановлення платіжного POS-термінала пристроєм який буде використовуватись за замовчуванням при проведенні безготівкових оплат в API за умови використання параметра **USEPRGPOSTERMINAL**.

Параметри **SetDefaultPOSTerminal**:

|                 |        |                                                |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| ID              | string | ID платіжного POS- термінала                   |

Приклад виконання **SetDefaultPOSTerminal**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000024021"
App.SetParameter "PathToCashalotDir",
"C:\Users\User\AppData\Local\Cashalot"
App.SetParameter "DeviceIDFnRRO", "4000024021"

Set DefaultPOSTerminal = App.SetDefaultPOSTerminal (FiscalNumberRRO,
ID)

MsgBox "Ознака успішності виконання операції = " &
DefaultPOSTerminal.Return
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненим параметром Return(Ознака успішності виконання)

#### 40. Проведення транзакції через POS-термінал **PayByPaymentCard** (FiscalNumberRRO, SumPayByCard, OtherParamtrs)

Для проведення транзакції на POS-терміналі необхідно використовувати метод **PayByPaymentCard**.

Параметри **PayByPaymentCard**:

|                 |        |                                                |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| SumPayByCard    | string | Сума оплати                                    |
| OtherParam      | string | Додаткові параметри                            |
|                 |        | *Параметр зарезервований                       |

В результаті виконання методу буде здійснено запит на POS-термінал який використовується за замовчуванням в налаштуваннях додаткового обладнання Cashalot. Після проведення транзакції у відповідь API буде повернуто [реквізити транзакції](#). В подальшому такі реквізити можливо використати в методі реєстрації чека FiscalizeCheck.

Приклад виконання **PayByPaymentCard**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000024021"
App.SetParameter "PathToCashalotDir",
"C:\Users\User\AppData\Local\Cashalot"
App.SetParameter "DeviceIDFnRRO", "4000024021"

Set Pay = App.PayByPaymentCard (FiscalNumberRRO, "1.0", "")

MsgBox "Ознака успішності виконання операції = " &
DefaultPOSTerminal.Return
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненням параметром Return(Ознака успішності виконання) та JsonVal

|                 |        |                                                                                                                                    |
|-----------------|--------|------------------------------------------------------------------------------------------------------------------------------------|
| Ret             | bool   | Ознака успішності виконання.                                                                                                       |
| Description     | string | Опис помилки за наявності                                                                                                          |
| Values          | string | JSON-структура, що містить значення успішності операції, опис помилки та перелік реквізитів транзакції проведеної на POS-терміналі |
| InvoiceNumber   | string | Номер чека                                                                                                                         |
| RRN             | string | РРН транзакції                                                                                                                     |
| AcquireID       | string | Ідентифікатор еквайра                                                                                                              |
| AcquireName     | string | Термінал                                                                                                                           |
| ApprovalCode    | string | Код підтвердження                                                                                                                  |
| IssuerName      | string | Платіжна система                                                                                                                   |
| PAN             | string | Номер картки                                                                                                                       |
| SignVerif       | string | Електронний підпис                                                                                                                 |
| TerminalID      | string | Код терміналу                                                                                                                      |
| TransactionDate | string | Дата та час проведення транзакції                                                                                                  |

#### 41. Повернення транзакції (оплати) через POS-термінал **ReturnPaymentByPaymentCard** (FiscalNumberRRO, SumPayByCard, RRN, OtherParamtrs)

За необхідності здійснення повернення оплати по раніше проведеній транзакції слід використовувати метод **ReturnPaymentByPaymentCard**. Для виконання повернення обов'язково потрібно зазначити **RRN** транзакції.

Параметри **ReturnPaymentByPaymentCard**:

|                 |        |                                                |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| SumPayByCard    | string | Сума оплати                                    |
| RRN             | string | РРН транзакції                                 |
| OtherParam      | string | Додаткові параметри                            |
|                 |        | *Параметр зарезервований                       |

В результаті виконання методу буде здійснено запит на POS-термінал який використовується за замовчуванням в налаштуваннях додаткового обладнання Cashalot. Після проведення транзакції у відповідь API буде повернуто реквізити транзакції. Після проведення транзакції у відповідь API буде повернуто [реквізити транзакції](#). В подальшому такі реквізити можливо використати в методі реєстрації чека FiscalizeCheck з типом "Повернення"

Приклад виконання **PayByPaymentCard**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000024021"
App.SetParameter "PathToCashalotDir",
"C:\Users\User\AppData\Local\Cashalot"
App.SetParameter "DeviceIDFnRRO", "4000024021"

Set ReturnPay = App.ReturnPaymentByPaymentCard (FiscalNumberRRO,
"1.0", "096462211901", "")

MsgBox "Ознака успішності виконання операції = " &
DefaultPOSTerminal.Return
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненим параметром Return(Ознака успішності виконання) та JsonVal

|                 |        |                                                                                                                                    |
|-----------------|--------|------------------------------------------------------------------------------------------------------------------------------------|
| Ret             | bool   | Ознака успішності виконання.                                                                                                       |
| Description     | string | Опис помилки за наявності                                                                                                          |
| Values          | string | JSON-структура, що містить значення успішності операції, опис помилки та перелік реквізитів транзакції проведеної на POS-терміналі |
| InvoiceNumber   | string | Номер чека                                                                                                                         |
| RRN             | string | PPH транзакції                                                                                                                     |
| AcquireID       | string | Ідентифікатор еквайра                                                                                                              |
| AcquireName     | string | Термінал                                                                                                                           |
| ApprovalCode    | string | Код підтвердження                                                                                                                  |
| IssuerName      | string | Платіжна система                                                                                                                   |
| PAN             | string | Номер картки                                                                                                                       |
| SignVerif       | string | Електронний підпис                                                                                                                 |
| TerminalID      | string | Код терміналу                                                                                                                      |
| TransactionDate | string | Дата та час проведення транзакції                                                                                                  |

#### 42. Скасування транзакції (оплати) через POS-термінал **CancelPaymentByPaymentCard** (FiscalNumberRRO, SumPayByCard, InvoiceNumber, OtherParams)

За потреби скасування проведеної транзакції на POS-терміналі необхідно використовувати метод **CancelPaymentByPaymentCard**. Для виконання повернення обов'язково потрібно зазначити номер чеку (InvoiceNumber)

Параметри **CancelPaymentByPaymentCard**:

|                 |        |                                                |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| SumPayByCard    | string | Сума оплати                                    |
| InvoiceNumber   | string | Номер чека                                     |
| OtherParams     | string | Додаткові параметри                            |
|                 |        | *Параметр зарезервований                       |

#### Приклад виконання **CancelPaymentByPaymentCard**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000024021"
App.SetParameter "PathToCashalotDir",
"C:\Users\User\AppData\Local\Cashalot"
App.SetParameter "DeviceIDFnRRO", "4000024021"

Set Cancel = App.CancelPaymentByPaymentCard (FiscalNumberRRO, "1.0",
"03411", "")

MsgBox "Ознака успішності виконання операції = " &
DefaultPOSTerminal.Return
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненням параметром Return(Ознака успішності виконання) та JsonVal

|                 |        |                                                                                                                                    |
|-----------------|--------|------------------------------------------------------------------------------------------------------------------------------------|
| Ret             | bool   | Ознака успішності виконання.                                                                                                       |
| Description     | string | Опис помилки за наявності                                                                                                          |
| Values          | string | JSON-структура, що містить значення успішності операції, опис помилки та перелік реквізитів транзакції проведеної на POS-терміналі |
| InvoiceNumber   | string | Номер чека                                                                                                                         |
| RRN             | string | PPH транзакції                                                                                                                     |
| AcquireID       | string | Ідентифікатор еквайра                                                                                                              |
| AcquireName     | string | Термінал                                                                                                                           |
| ApprovalCode    | string | Код підтвердження                                                                                                                  |
| IssuerName      | string | Платіжна система                                                                                                                   |
| PAN             | string | Номер картки                                                                                                                       |
| SignVerif       | string | Електронний підпис                                                                                                                 |
| TerminalID      | string | Код терміналу                                                                                                                      |
| TransactionDate | string | Дата та час проведення транзакції                                                                                                  |

#### 43. Проведення транзакції через функцію “Оплата частинами” **PayPartPaymentByPaymentCard** (FiscalNumberRRO, SumPayByCard, CountParts, OtherParametrs)

Для проведення транзакції на POS-терміналі з використанням сервісу “Оплата частинами” необхідно використовувати метод **PayPartPaymentByPaymentCard**. Максимальна кількість частин може бути зазначено 12.

**Важливо!** Опція доступна лише для конфігурації обладнання POS-терміналу ПриватБанк (LAN) або ПриватБанк (USB). В налаштуваннях додаткового обладнання для POS-терміналу необхідно активувати опцію “Сервіс оплати частинами”. Окрім цього така опція має бути увімкнена в налаштуваннях безпосередньо самого POS-терміналу.

#### Параметри PayPartPaymentByPaymentCard:

|                 |        |                                                |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| SumPayByCard    | string | Сума оплати                                    |
| CountParts      | string | Кількість частин                               |
| OtherParametrs  | string | Додаткові параметри                            |
|                 |        | *Параметр зарезервований                       |

#### Приклад виконання PayPartPaymentByPaymentCard:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000024021"
App.SetParameter "PathToCshalotDir",
"C:\Users\User\AppData\Local\Cashalot"
App.SetParameter "DeviceIDFnRRO", "4000024021"

Set PartsPay = App.PayPartPaymentByPaymentCard (FiscalNumberRRO,
"1.0", "6", "")

MsgBox "Ознака успішності виконання операції = " &
DefaultPOSTerminal.Return
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненням параметром Return(Ознака успішності виконання) та JsonVal

|                 |        |                                                                                                                                    |
|-----------------|--------|------------------------------------------------------------------------------------------------------------------------------------|
| Ret             | bool   | Ознака успішності виконання.                                                                                                       |
| Description     | string | Опис помилки за наявності                                                                                                          |
| Values          | string | JSON-структура, що містить значення успішності операції, опис помилки та перелік реквізитів транзакції проведеної на POS-терміналі |
| InvoiceNumber   | string | Номер чека                                                                                                                         |
| RRN             | string | РРН транзакції                                                                                                                     |
| AcquireID       | string | Ідентифікатор еквайра                                                                                                              |
| AcquireName     | string | Термінал                                                                                                                           |
| ApprovalCode    | string | Код підтвердження                                                                                                                  |
| IssuerName      | string | Платіжна система                                                                                                                   |
| PAN             | string | Номер картки                                                                                                                       |
| SignVerif       | string | Електронний підпис                                                                                                                 |
| TerminalID      | string | Код терміналу                                                                                                                      |
| TransactionDate | string | Дата та час проведення транзакції                                                                                                  |
| ServiceDogNum   | string | Номер кредитного договору оплати частинами                                                                                         |

#### 44. Повернення транзакції по операції “Оплата частинами” **ReturnPartPaymentByPaymentCard** (FiscalNumberRRO, SumPayByCard, RRN, IsPartialReturn, ServiceDogNum, OtherParams)

За необхідності здійснення повернення оплати по раніше проведеній транзакції з використанням сервісу “Оплата частинами” слід використовувати метод **ReturnPartPaymentByPaymentCard**. Для виконання повернення обов’язково потрібно зазначити RRN транзакції чека оплати частинами та його номер кредитного договору.

**Важливо!** Опція доступна лише для конфігурації обладнання POS-термінала ПриватБанк (LAN) або ПриватБанк (USB). В налаштуваннях додаткового обладнання для POS-термінала необхідно активувати опцію “Сервіс оплати частинами”. Окрім цього така опція має бути увімкнена в налаштуваннях безпосередньо самого POS-термінала.

Параметри **ReturnPartPaymentByPaymentCard**:

|                          |        |                                                               |
|--------------------------|--------|---------------------------------------------------------------|
| FiscalNumberRRO          | string | Фіскальний номер ПРРО. Ідентифікатор пристрою.                |
| SumPayByCard             | string | Сума оплати                                                   |
| RRN                      | string | Кількість частин                                              |
| IsPartialReturn          | string | Ознака часткового повернення по оплаті частинами (TRUE/FALSE) |
| ServiceDogNum            | string | Номер кредитного договору оплати частинами                    |
| OtherParams              | string | Додаткові параметри                                           |
| *Параметр зарезервований |        |                                                               |

Приклад виконання **PayPartPaymentByPaymentCard**:

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000024021"
App.SetParameter "PathToCashalotDir",
"C:\Users\User\AppData\Local\Cashalot"
App.SetParameter "DeviceIDFnRRO", "4000024021"

Set PartsReturn = App.PayPartPaymentByPaymentCard (FiscalNumberRRO,
"1.0", "12333124124124", "false", "5122332", "")

MsgBox "Ознака успішності виконання операції = " &
DefaultPOSTerminal.Return
```

В якості відповіді повертається стандартна структура параметрів повернення **CashalotApiRetVal** з заповненим параметром Return(Ознака успішності виконання) та JsonVal

|                 |        |                                                                                                                                    |
|-----------------|--------|------------------------------------------------------------------------------------------------------------------------------------|
| Ret             | bool   | Ознака успішності виконання.                                                                                                       |
| Description     | string | Опис помилки за наявності                                                                                                          |
| Values          | string | JSON-структура, що містить значення успішності операції, опис помилки та перелік реквізитів транзакції проведеної на POS-терміналі |
| InvoiceNumber   | string | Номер чека                                                                                                                         |
| RRN             | string | РРН транзакції                                                                                                                     |
| AcquireID       | string | Ідентифікатор еквайра                                                                                                              |
| AcquireName     | string | Термінал                                                                                                                           |
| ApprovalCode    | string | Код підтвердження                                                                                                                  |
| IssuerName      | string | Платіжна система                                                                                                                   |
| PAN             | string | Номер картки                                                                                                                       |
| SignVerif       | string | Електронний підпис                                                                                                                 |
| TerminalID      | string | Код терміналу                                                                                                                      |
| TransactionDate | string | Дата та час проведення транзакції                                                                                                  |

#### 45. Створення відкладеного чека продажу товарів/послуг **CreatePostponedReceipt** (FiscalNumberRRO, JSONGoodsLstData, JSONPayData)

Для створення відкладеного чека продажу методом **CreatePostponedReceipt** потрібно підготувати дві JSON-структури: JSONGoodsLstData - з даними про товари чи послуги, та JSONPayData - з інформацією про оплату. Ці структури мають такий самий формат, як у методі **FiscalizeCheck**. Створений відкладений чек зберігається на необмежений термін.

##### Параметри, які повинен містити метод **CreatePostponedReceipt**:

| Ім'я             | Тип    | Опис                                           |
|------------------|--------|------------------------------------------------|
| FiscalNumberRRO  | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| JSONGoodsLstData | string | JSON структура з описом товарів або послуг.    |
| JSONPayData      | string | JSON структура з підсумками оплати по чеку.    |

#### Приклад виконання `CreatePostponedReceipt`:

```
Set App = CreateObject("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

JSONGoodsLstData =
"{\"ReceiptLst\": [{\"VendorCode\": \"1000003\", \"Name\": \"Цукор, ваговий\", \"CdUKDZED\": \"170112\", \"CdDKPP\": \"\", \"GoodsType\": \"1\", \"Barcode\": \"103\", \"UnitType\": \"кг\", \"Quantity\": \"1\", \"Price\": \"150\", \"Amount\": \"150\", \"DiscountPrc\": \"\", \"DiscountSum\": \"\", \"VATRate\": \"\", \"IsPriceIncludeVAT\": false, \"SumVAT\": \"\", \"IsExcise\": false, \"OtherParams\": \"\", \"VATLetter\": \"\", \"ExciseLetter\": \"\", \"ExciseStampBarcode\": \"\", \"DocNumber\": \"\", \"Comment\": \"Дякуємо за покупку!\", \"DocType\": 0, \"DocSubType\": 0, \"SaleOrderNumber\": \"\"}]}"

JSONPayData =
"{\"PaymentOrderType\": 0, \"SumCash\": \"100\", \"SumPayByCard\": \"\", \"SumPayByBank\": \"\", \"SumPayByCredit\": \"\", \"SumPayByCertificate\": \"50\", \"SumPayCheck\": \"150\", \"CustomerEmail\": \"\", \"AutoSendCustomerEmail\": \"False\", \"CustomerSMSNumber\": \"\", \"AutoSendCustomerSMS\": \"False\", \"CustomerViberNumber\": \"\", \"AutoSendCustomerViber\": \"False\", \"TerminalID\": null, \"ApprovalCode\": null, \"RRN\": null, \"IssuerName\": null, \"PAN\": null, \"TransactionDate\": null, \"SignVerif\": null, \"AcquireName\": null, \"InvoiceNumber\": null, \"ParentRRN\": null, \"SumPreparePayed\": null, \"ParentReceiptFiscalNumber\": null, \"ParentRROFiscalNumber\": null, \"OtherParams\": null, \"PartPaySum\": null, \"ReceiptID\": null, \"ExpressWaybillNumber\": null, \"UserPaymentForms\": null, \"PaymentFormsLst\": []}"

Set resultCreatePostponedReceipt = App.CreatePostponedReceipt
(FiscalNumberRRO, JSONGoodsLstData, JSONPayData)

MsgBox "Результат виконання методу = " & resultCreatePostponedReceipt.Return
MsgBox "Опис помилки = " & resultCreatePostponedReceipt.Description
MsgBox "Результат в JSON = " & resultCreatePostponedReceipt.JsonVal
```

У відповіді на `CreatePostponedReceipt` повертається структура `CashalotApiRetVal`, що містить:

|             |        |                                                                                                                   |
|-------------|--------|-------------------------------------------------------------------------------------------------------------------|
| Return      | bool   | Ознака успішності виконання операції                                                                              |
| Description | string | Опис результату виконання або текст помилки                                                                       |
| JsonVal     | string | JSON структура з переліком параметрів про тип чека, ID, дату та час створення, статус експрес-накладної та іншими |
| Type        | string | <a href="#">Тип документа.</a>                                                                                    |

Опис параметрів, які передаються в масиві `Values` структури `JsonVal`:

|                      |                                                       |
|----------------------|-------------------------------------------------------|
| Type                 | <a href="#">Тип документа.</a>                        |
| ReceiptID            | ID (guid) чека                                        |
| ReceiptSum           | Сума до сплати по чеку                                |
| ReceiptDateTime      | Дата та час створення у форматі (дд.мм.рррр гг:хв:сс) |
| ExpressWaybillNumber | Номер експрес-накладної (ТТН)                         |

|                       |                                                                                  |
|-----------------------|----------------------------------------------------------------------------------|
| StatusTTN             | Числовий статус експрес-накладної                                                |
| StatusTTNDescription  | Статус експрес-накладної                                                         |
| StatusTTNCanFiscalize | Ознака набуття чека фіналізованого статусу доставки                              |
| ReceiptPrePayOrderNum | Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати) |

Метод **CreatePostponedReceipt** підтримує створення відкладеного чека з типом «Інтернет-продаж», призначеного для реєстрації після фактичного отримання товару покупцем через Нову пошту. Вимоги до структури такого чека наведено у [розділі](#).

#### 46. Фіскалізація відкладеного чеку продажу товарів/послуг **FiscalizePostponedReceipt** (FiscalNumberRRO, ReceiptID, OtherParam)

Метод **FiscalizePostponedReceipt** використовується для фіскалізації раніше створеного відкладеного чека. Вхідним параметром методу є ReceiptID, що повертається при виклику **CreatePostponedReceipt**.

Параметри, які повинен містити метод **FiscalizePostponedReceipt**:

| Ім'я            | Тип    | Опис                                           |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| ReceiptID       | string | ID (guid) чека                                 |
| OtherParam      | string | Зарезервований параметр (може містити null)    |

Приклад виконання **FiscalizePostponedReceipt**:

```
Set App = CreateObject("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
ReceiptID = "ff0694cd-d189-407c-85c4-c5d0662dd0ec"
OtherParam = ""

App.SetParameter "PathToCashaLotDir", "C:\CashaLot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set resultFiscalizePostponedReceipt = App.FiscalizePostponedReceipt
(FiscalNumberRRO, ReceiptID, OtherParam)

MsgBox "Результат виконання методу = " & resultFiscalizePostponedReceipt.Return
MsgBox "Опис помилки = " & resultFiscalizePostponedReceipt.Description
MsgBox "Результат в JSON= " & resultFiscalizePostponedReceipt.JsonVal
MsgBox "XML = " & resultFiscalizePostponedReceipt.XML
MsgBox "Фіскальний номер = " & resultFiscalizePostponedReceipt.ReceiptFiscalNum
MsgBox "Локальний номер = " & resultFiscalizePostponedReceipt.ReceiptLocalNum
MsgBox "Ідентифікатор зміни = " & resultFiscalizePostponedReceipt.ShiftID
MsgBox "Тип документа = " & resultFiscalizePostponedReceipt.Type
```

У відповіді на **FiscalizePostponedReceipt** повертається структура **CashalotApiRetVal**, що містить:

|                  |        |                                                                                                                                                                           |
|------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Return           | bool   | Ознака успішності виконання.                                                                                                                                              |
| Description      | string | Опис результату виконання або текст помилки                                                                                                                               |
| JsonVal          | string | JSON-структура, що містить значення успішності операції, тип документу, помилки, локальний та фіскальний номери чека, ID зміни та Base64 рядок XML чека та інші параметри |
| ReceiptFiscalNum | string | Фіскальний номер чека.                                                                                                                                                    |
| ReceiptLocalNum  | string | Локальний номер чека.                                                                                                                                                     |
| ShiftID          | string | Ідентифікаційний номер поточної зміни.                                                                                                                                    |
| Type             | int    | <a href="#">Тип документа</a> .                                                                                                                                           |
| XML              | string | XML-структура фіскалізованого чека.                                                                                                                                       |

Опис параметрів, які передаються в масиві **Values** структури **JsonVal**:

|                         |                                                                                  |
|-------------------------|----------------------------------------------------------------------------------|
| ShiftID                 | Ідентифікаційний номер поточної зміни.                                           |
| ReceiptFiscalNumber     | Фіскальний номер чека.                                                           |
| ReceiptLocalNumber      | Локальний номер чека.                                                            |
| OfflineMode             | Ознака перебування в режимі офлайн                                               |
| Type                    | <a href="#">Тип документа</a> .                                                  |
| ReceiptID               | ID (guid) чека                                                                   |
| ReceiptSum              | Сума до сплати по чеку                                                           |
| ReceiptDateTime         | Дата та час створення у форматі (дд.мм.рррр гг:хв:сс)                            |
| FSKORceiptLink          | Посилання на перегляд чека в ДПС                                                 |
| CashalotReceiptLink     | Посилання на перегляд чека в кабінеті Cashalot                                   |
| ExpressWaybillNumber    | Номер експрес-накладної (ТТН)                                                    |
| StatusTTN               | Числовий статус експрес-накладної                                                |
| StatusTTNDescription    | Статус експрес-накладної                                                         |
| StatusTTNCanFiscalize   | Ознака набуття чека фіналізованого статусу доставки                              |
| ReceiptPrePayOrderNum   | Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати) |
| Base64Str1251ReceiptXML | XML-структура фіскалізованого чека у форматі base64                              |

Для реєстрації чеків типу «Інтернет-продаж», створених методом **CreatePostponedReceipt** і призначених для фіскалізації після отримання товару покупцем через Нову пошту, слід враховувати вимоги, описані в окремому [розділі](#).

#### 47. Отримати інформацію про відкладений чек за його ReceiptID **GetPostponedReceiptByID** (FiscalNumberRRO, ReceiptID)

Метод призначений для отримання інформації про відкладений чек за його ReceiptID. Якщо чек має тип «Інтернет-продаж» і містить заповнений параметр ExpressWaybillNumber, програма додатково виконує перевірку статусу експрес-накладної через API Нової пошти.

Залежно від отриманого статусу доставки, система автоматично визначає, чи набув чек фіналізованого статусу для можливості реєстрації. Відповідно до отриманого статусу доставки на API Нової Пошти, програма встановить відповідну ознаку.

**Примітка!** Для можливості перевірки статусу експрес-накладної необхідно попередньо виконати налаштування в кабінеті користувача Cashalot у розділі «Всі каси → Налаштування → API Нова Пошта».

Параметри, які повинен містити метод **GetPostponedReceiptByID**:

| Ім'я            | Тип    | Опис                                           |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| ReceiptID       | string | ID (guid) чека                                 |

Приклад виконання **GetPostponedReceiptByID**:

```
Set App = CreateObject("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
ReceiptID = "7c9d91b3-4138-4427-9ff2-7f28c724b4b2"

App.SetParameter "PathToCashaLotDir", "C:\CashaLot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set resultGetPostponedReceiptByID = App.GetPostponedReceiptByID
(FiscalNumberRRO, ReceiptID)

MsgBox "Результат успішності операції = " &
resultGetPostponedReceiptByID.Return
MsgBox "Опис помилки = " & resultGetPostponedReceiptByID.Description
MsgBox "Результат = " & resultGetPostponedReceiptByID.JsonVal
```

У відповіді на **GetPostponedReceiptByID** повертається структура **CashaLotApiRetVal**, що містить:

|             |        |                                                                                                                   |
|-------------|--------|-------------------------------------------------------------------------------------------------------------------|
| Return      | bool   | Ознака успішності виконання операції                                                                              |
| Description | string | Опис результату виконання або текст помилки                                                                       |
| JsonVal     | string | JSON структура з переліком параметрів про тип чека, ID, дату та час створення, статус експрес-накладної та іншими |
| Type        | string | <a href="#">Тип документа.</a>                                                                                    |

Опис параметрів, які передаються в масиві **Values** структури **JsonVal**:

|                       |                                                                                  |
|-----------------------|----------------------------------------------------------------------------------|
| Type                  | <a href="#">Тип документа.</a>                                                   |
| ReceiptID             | ID (guid) чека                                                                   |
| ReceiptSum            | Сума до сплати по чеку                                                           |
| ReceiptDateTime       | Дата та час створення у форматі (дд.мм.рррр гг:хв:сс)                            |
| ExpressWaybillNumber  | Номер експрес-накладної (ТТН)                                                    |
| StatusTTN             | Числовий статус експрес-накладної                                                |
| StatusTTNDescription  | Статус експрес-накладної                                                         |
| StatusTTNCanFiscalize | Ознака набуття чека фіналізованого статусу доставки                              |
| ReceiptPrePayOrderNum | Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати) |

48. Отримати інформацію про відкладений чек за його номером ЕН (експрес-накладної) **GetPostponedReceiptByWaybillNumber** (FiscalNumberRRO, ExpressWaybillNumber)

Метод призначений для отримання інформації про відкладений чек вказаним номером експрес-накладної під час створення. Додатково програма виконує перевірку статусу експрес-накладної на API Нової Пошти. В залежності від отриманого статусу доставки на API Нової Пошти, програма встановить відповідну ознаку доступності реєстрації такого чека.

**Примітка!** Для можливості перевірки статусу експрес-накладної обов'язково необхідно виконати налаштування в кабінеті користувача Cashalot в розділі «Всі каси -> Налаштування -> API Нова Пошта»

Параметри, які повинен містити метод **GetPostponedReceiptByWaybillNumber**:

| Ім'я                 | Тип    | Опис                                           |
|----------------------|--------|------------------------------------------------|
| FiscalNumberRRO      | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| ExpressWaybillNumber | string | Номер експрес-накладної (ТТН)                  |

Приклад виконання **GetPostponedReceiptByWaybillNumber**:

```
Set App = CreateObject("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
ExpressWaybillNumber = "59001034051234"

App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set resultGetPostponedReceiptByWaybillNumber =
App.GetPostponedReceiptByWaybillNumber
(FiscalNumberRRO, ExpressWaybillNumber)

MsgBox "Результат успішності операції = " &
resultGetPostponedReceiptByWaybillNumber.Return
MsgBox "Опис помилки = " &
resultGetPostponedReceiptByWaybillNumber.Description
MsgBox "Результат = " & resultGetPostponedReceiptByWaybillNumber.JsonVal
```

У відповіді на **GetPostponedReceiptByWaybillNumber** повертається структура **CashalotApiRetVal**, що містить:

|             |        |                                                                                                                   |
|-------------|--------|-------------------------------------------------------------------------------------------------------------------|
| Return      | bool   | Ознака успішності виконання операції                                                                              |
| Description | string | Опис результату виконання або текст помилки                                                                       |
| JsonVal     | string | JSON структура з переліком параметрів про тип чека, ID, дату та час створення, статус експрес-накладної та іншими |
| Type        | string | <a href="#">Тип документа.</a>                                                                                    |

Опис параметрів, які передаються в масиві **Values** структури **JsonVal**:

|                 |                                                       |
|-----------------|-------------------------------------------------------|
| Type            | <a href="#">Тип документа.</a>                        |
| ReceiptID       | ID (guid) чека                                        |
| ReceiptSum      | Сума до сплати по чеку                                |
| ReceiptDateTime | Дата та час створення у форматі (дд.мм.рррр гг:хв:сс) |

|                       |                                                                                  |
|-----------------------|----------------------------------------------------------------------------------|
| ExpressWaybillNumber  | Номер експрес-накладної (ТТН)                                                    |
| StatusTTN             | Числовий статус експрес-накладної                                                |
| StatusTTNDescription  | Статус експрес-накладної                                                         |
| StatusTTNCanFiscalize | Ознака набуття чека фіналізованого статусу доставки                              |
| ReceiptPrePayOrderNum | Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати) |

#### 49. Отримати інформацію про перший відкладений чек **GetFirstPostponedReceipt** (FiscalNumberRRO, StartDate, EndDate)

Метод **GetFirstPostponedReceipt** рекомендовано використовувати як перший крок у ланцюжку запитів для отримання відкладених чеків за певний період часу. Метод приймає два параметри: DateBeg та DateEnd, які визначають часовий інтервал для пошуку відкладених чеків. Обидва параметри є необов'язковими та можуть містити значення null. Якщо обидва параметри не задані (DateBeg = null і DateEnd = null), метод повертає усі доступні відкладені чеки без обмеження по даті їх створення. Якщо задано лише один параметр: DateBeg - метод повертає чеки від вказаної дати і далі. DateEnd - метод повертає чеки до вказаної дати включно. Якщо задано обидва параметри, метод повертає чеки в межах зазначеного періоду.

Параметри методу **GetFirstPostponedReceipt**:

| Ім'я            | Тип    | Опис                                           |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| DateBeg         | string | Початок періоду в форматі dd.MM.yyyy.          |
| DateEnd         | string | Кінець періоду в форматі dd.MM.yyyy.           |

Приклад виконання **GetFirstPostponedReceipt**:

```
Set App = CreateObject ("AddIn.CashalotApi")
FiscalNumberRRO = "4000000000"
dateBeg = "01.01.2022"
dateEnd = "09.01.2022"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set firstPostponedReceipt = App.GetFirstPostponedReceipt(FiscalNumberRRO, dateBeg, dateEnd)
MsgBox "Перший чек в періоді = " & firstPostponedReceipt.JsonVal
```

У відповіді на **GetFirstPostponedReceipt** повертається структура **CashalotApiRetVal**, що містить значення першого чека з вказаного періоду:

|             |        |                                                                                                                   |
|-------------|--------|-------------------------------------------------------------------------------------------------------------------|
| Return      | bool   | Ознака успішності виконання операції                                                                              |
| Description | string | Опис результату виконання або текст помилки                                                                       |
| JsonVal     | string | JSON структура з переліком параметрів про тип чека, ID, дату та час створення, статус експрес-накладної та іншими |
| Type        | string | <u>Тип документа.</u>                                                                                             |

Опис параметрів, які передаються в масиві [Values](#) структури [JsonVal](#):

|                       |                                                                                  |
|-----------------------|----------------------------------------------------------------------------------|
| Type                  | <a href="#">Тип документа.</a>                                                   |
| ReceiptID             | ID (guid) чека                                                                   |
| ReceiptSum            | Сума до сплати по чеку                                                           |
| ReceiptDateTime       | Дата та час створення у форматі (дд.мм.рррр гг:хв:сс)                            |
| ExpressWaybillNumber  | Номер експрес-накладної (ТТН)                                                    |
| StatusTTN             | Числовий статус експрес-накладної                                                |
| StatusTTNDescription  | Статус експрес-накладної                                                         |
| StatusTTNCanFiscalize | Ознака набуття чека фіналізованого статусу доставки                              |
| ReceiptPrePayOrderNum | Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати) |

## 50. Отримати інформацію наступний відкладений чек [GetNextPostponedReceipt](#) (FiscalNumberRRO)

Метод [GetNextPostponedReceipt](#) використовується після виклику [GetFirstPostponedReceipt](#) у ланцюжку запитів для отримання списку відкладених чеків за певний період часу. Після того як [GetFirstPostponedReceipt](#) підготує список чеків, метод [GetNextPostponedReceipt](#) дозволяє поетапно отримувати кожен наступний чек зі списку. Метод може викликатися багаторазово, доки не буде отримано всі чеки зі сформованого списку. Коли всі чеки вже отримано, метод повертає false, що сигналізує про завершення вибірки. Метод приймає єдиний обов'язковий параметр FiscalNumberRRO

**Примітка!** перед використанням [GetNextPostponedReceipt](#) необхідно обов'язково викликати [GetFirstPostponedReceipt](#), інакше список чеків не буде сформовано.

### Приклад виконання [GetNextPostponedReceipt](#):

```
Set App = CreateObject ("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
dateBeg = "01.01.2022"
dateEnd = "09.01.2022"
App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

Set nextPostponedReceipt = App.GetNextPostponedReceipt(FiscalNumberRRO, dateBeg,
dateEnd)
Do while nextPostponedReceipt.Return
'Обробити отримані чеки
Set nextPostponedReceipt = App.GetNextPostponedReceipt(FiscalNumberRRO)
if nextPostponedReceipt.Return <> False Then
MsgBox "Наступний чек в періоді = " & nextPostponedReceipt.JsonVal
Else
MsgBox "Чеків більше немає"
End if
Loop
```

У відповіді на [GetNextPostponedReceipt](#) повертається структура [CashalotApiRetVal](#), що містить значення наступного чека з вказаного періоду:

|             |        |                                                                                                                   |
|-------------|--------|-------------------------------------------------------------------------------------------------------------------|
| Return      | bool   | Ознака успішності виконання операції                                                                              |
| Description | string | Опис результату виконання або текст помилки                                                                       |
| JsonVal     | string | JSON структура з переліком параметрів про тип чека, ID, дату та час створення, статус експрес-накладної та іншими |
| Type        | string | <a href="#">Тип документа.</a>                                                                                    |

Опис параметрів, які передаються в масиві [Values](#) структури [JsonVal](#):

|                       |                                                                                  |
|-----------------------|----------------------------------------------------------------------------------|
| Type                  | <a href="#">Тип документа.</a>                                                   |
| ReceiptID             | ID (guid) чека                                                                   |
| ReceiptSum            | Сума до сплати по чеку                                                           |
| ReceiptDateTime       | Дата та час створення у форматі (дд.мм.рррр гг:хв:сс)                            |
| ExpressWaybillNumber  | Номер експрес-накладної (ТТН)                                                    |
| StatusTTN             | Числовий статус експрес-накладної                                                |
| StatusTTNDescription  | Статус експрес-накладної                                                         |
| StatusTTNCanFiscalize | Ознака набуття чека фіналізованого статусу доставки                              |
| ReceiptPrePayOrderNum | Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати) |

## 51. Відображення візуальної форми чека за його ReceiptID [ShowReceiptByID](#) (FiscalNumberRRO, ReceiptID)

Метод [ShowReceiptByID](#) призначений для отримання візуальної форми чека за його ReceiptID.

Для виклику методу необхідно передати два параметри: FiscalNumberRRO та ReceiptID.

За допомогою цього методу можна отримати візуальне представлення як зареєстрованого, так і відкладеного чека продажу, а також видаткового чека (у випадку повернення).

Параметри, які повинен містити метод [ShowReceiptByID](#):

| Ім'я            | Тип    | Опис                                           |
|-----------------|--------|------------------------------------------------|
| FiscalNumberRRO | string | Фіскальний номер ПРРО. Ідентифікатор пристрою. |
| ReceiptID       | string | ID (guid) чека                                 |

Приклад виконання [ShowReceiptByID](#):

```
Set App = CreateObject("AddIn.CashaLotApi")
FiscalNumberRRO = "4000000000"
ReceiptID = "6a307ee1-07f8-4c5d-bbb9-d543a872d32e"

App.SetParameter "PathToCashalotDir", "C:\Cashalot"
App.SetParameter "DeviceIDFnRRO", FiscalNumberRRO

resultShowReceiptByID = App.ShowReceiptByID(FiscalNumberRRO, ReceiptID)

MsgBox resultShowReceiptByID
MsgBox App.GetLastError()
```

У відповідь на виклик методу відкривається візуальна форма з чеком, а також повертається значення True або False, що вказує на успішність виконання.



ТОВ "Фортеця"  
"Фортифікація"  
місто Київ, вул. Сталева, буд.1  
ПН 444444555555  
ФН ПРРО 4000024425  
КАСІР ТОВ "Фортеця"(Печатка)  
**Чек №1566232**

-----  
Повернення по чеку ФН 1566231  
-----

УКТ ЗЕД 170112  
Штрихкод товару: 103  
Цукор, ваговий  
1,000 X 10,00 10,00 ю  
-----  
СУМА 10.00 ГРН  
ПДВ ю=20% 1,67  
-----  
**ГОТІВКА 10.00 ГРН**  
-----

Дякуємо за покупку!  
-----

ЧЕК №1566232 17.09.2025 17:25:47



ВИДАТКОВИЙ ЧЕК ОНЛАЙН  
Cashalot

Друк

Надіслати на Email

Зберегти чек

Готово

## Ломбардні операції

Для фіскалізації ломбардних послуг введено системні номенклатури, які необхідно використовувати при виклику методу фіскалізації чека **FiscalizeCheck**. Повернення товару при ломбардних операціях не підтримується.

Також необхідно вказати тип документа **DocType = 4** в JSON-структурі параметрів чека. У разі потреби можна вказати номер договору **DocNumber**, за яким надається ломбардна послуга.

Напрямок руху необхідно враховувати при обчисленні кінцевого підсумку в чеку: від позицій типу 2 віднімаються позиції типу 3.

Системні номенклатури для ломбардних операцій:

| Артикул    | Назва                                           | Тип послуги | Напрямок руху* |
|------------|-------------------------------------------------|-------------|----------------|
| "ЛО-00001" | "Повернення страхування"                        | "3"         | -              |
| "ЛО-00002" | "Надання кредиту"                               | "3"         | -              |
| "ЛО-00003" | "Добір кредиту"                                 | "3"         | -              |
| "ЛО-00004" | "Надання додаткового кредиту"                   | "3"         | -              |
| "ЛО-00005" | "Зберігання"                                    | "2"         | +              |
| "ЛО-00006" | "Страховий платіж"                              | "2"         | +              |
| "ЛО-00007" | "Часткове повернення кредиту (Частковий викуп)" | "2"         | +              |
| "ЛО-00008" | "Повне повернення кредиту (Повний викуп)"       | "2"         | +              |
| "ЛО-00009" | "Повернення кредиту"                            | "2"         | +              |
| "ЛО-00010" | "Користування кредитом"                         | "2"         | +              |
| "ЛО-00011" | "Сплата відсотків по кредиту"                   | "2"         | +              |
| "ЛО-00012" | "Сплата пені"                                   | "2"         | +              |

\* **Напрямок руху** "-" видача грошей з каси; "+" повернення грошей в касу.

# Особливості роботи з чеками типу «Інтернет-продаж» при доставці Новою поштою

## Загальний опис

Чеки типу «Інтернет-продаж» застосовуються для фіскалізації операцій, у яких товар доставляється покупцю через Нову пошту. Реєстрація таких чеків має виконуватись після фактичного отримання товару покупцем, що підтверджується статусом експрес-накладної (ТТН), отриманим через API Нової пошти ([api.novaposhta.ua](https://api.novaposhta.ua)). У деяких специфічних випадках дозволяється реєстрація чека за умови, якщо він не набув фінального статусу ЕН (ТТН)

API Cashalot надає інструмент у вигляді набору методів, які дозволяють реалізувати повний цикл взаємодії між продавцем і покупцем у рамках доставки:

- Створення відкладеного чека з номером ЕН (ТТН);
- Перевірка статусу доставки (отримання інформації про чек);
- Фіскалізація чека після підтвердження отримання товару.

Цей механізм забезпечує відповідність фіскальним вимогам та автоматизує процес обробки онлайн-продажів із доставкою. Використання механізму можливо за умови виконаних налаштувань в кабінеті користувача Cashalot в розділі «Всі каси → Налаштування → API Нова Пошта».

## Створення відкладеного чека

Для створення відкладеного чека використовується метод **CreatePostponedReceipt**. Під час виклику методу необхідно передати дві JSON-структури **JSONGoodsLstData** та **JSONPayData**.

У випадку створення чека типу «Інтернет-продаж», в структурі **JSONPayData** обов'язково потрібно:

- Встановити параметр **PaymentOrderType** = 3, що вказує на тип операції «Інтернет-продаж».
- Заповнити параметр **ExpressWaybillNumber**, який відповідає номеру експрес-накладної (ТТН) Нової пошти.

Ці параметри дозволяють системі автоматично виконати запит до API Нової пошти після виконання методу для отримання актуального статусу доставки, що буде використано на наступних етапах обробки чека.

Можливі статуси доставки (ТТН) для відкладених чеків типу «Інтернет-продаж»

| StatusTTNDescription                        | Пояснення до статусу                                                              | StatusTTNCanFiscalize |
|---------------------------------------------|-----------------------------------------------------------------------------------|-----------------------|
| Потребує повторної перевірки стану доставки | Статус не вдалося визначити через відсутність зв'язку з API сервісом Нової Пошти. | False                 |
| Номер ЕН не знайдено                        | ЕН (ТТН) з таким номером не знайдено. Перевірте, чи правильно введено номер.      | False                 |
| Відмова одержувача                          | Доставка не виконана - одержувач відмовився від посилки.                          | False                 |
| Очікується доставка одержувачу              | Відправлення ще не доставлено - воно перебуває в дорозі.                          | False                 |

|                       |                                        |      |
|-----------------------|----------------------------------------|------|
| Відправлення отримано | Посилка успішно доставлена отримувачу. | True |
|-----------------------|----------------------------------------|------|

**Примітка!** У разі неналаштованої інтеграції в кабінеті користувача Cashalot, перевірка статусу експрес-накладної (ТТН) буде недоступна, і створення відкладеного чека типу «Інтернет-продаж» не відбудеться.

### Отримання інформації про відкладений чек

Для перевірки статусу доставки товару за номером експрес-накладної (ТТН) використовуються методи **GetPostponedReceiptById** та **GetPostponedReceiptByWaybillNumber**, які приймають відповідно параметри **ReceiptID** або **ExpressWaybillNumber**. Якщо ці параметри відповідають чеку, створеному з типом «Інтернет-продаж» (**PaymentOrderType = 3**) та заповненим номером ТТН, система виконує запит до API Нової пошти для отримання актуального статусу доставки. Перелік можливих статусів доставки для відкладеного чека наведений в таблиці вище.

### Реєстрація відкладеного чека

У результаті виклику методів створення або отримання інформації про відкладений чек, які передбачають звернення до API Нової пошти, у відповіді повертається параметр **StatusTTNCanFiscalize**. Це логічне значення (true або false), що вказує на можливість фіскалізації чека залежно від статусу доставки. Якщо значення true, це означає, що посилка успішно доставлена, і чек може бути зареєстрований. Якщо false, доставка ще не завершена, але в окремих випадках реєстрація може бути дозволена згідно з внутрішньою логікою системи.

Для реєстрації відкладеного чека необхідно скористатися методом **FiscalizePostponedReceipt**, який приймає параметр **ReceiptID** - ідентифікатор створеного чека. Під час виклику цього методу здійснюється додаткове звернення до сервісів API Нової пошти з метою отримання актуального статусу ЕН (ТТН). Якщо статус вказує, що відправлення отримано, чек реєструється, і у відповіді повертається оновлене значення **StatusTTNCanFiscalize = true**, що свідчить про те, що ТТН набула фіналізованого статусу.

**Примітка!** Реєстрація відкладеного чека неможлива, якщо статус ЕН (ТТН) - "Номер ЕН не знайдено" або "Відмова одержувача".

### Особливості реєстрації відкладеного чека «Інтернет-продажу» в залежності від статусу ЕН(ТТН) та визначеного способу оплати по чеку

Особливості реєстрації відкладеного чека типу «Інтернет-продаж» (**PaymentOrderType = 3**) із заповненим номером експрес-накладної (**ExpressWaybillNumber**) залежать від статусу доставки ЕН (ТТН) та способу оплати, визначеного по чеку. Основним орієнтиром є параметр **StatusTTNCanFiscalize**, який вказує, що ТТН набула фіналізованого статусу «Відправлення отримано» (**StatusTTNCanFiscalize = true**). У випадках, коли по чеку передбачена комбінована модель оплати - тобто частина суми вже сплачена наперед (визначається заповненістю параметра **SumPreparePayed**) або виконана 100% передплата засобами інтернет-еквайрингу, допускається можливість реєстрації чека навіть за умови, що статус доставки ще не фіналізований (**StatusTTNCanFiscalize = false**).

| Спосіб оплати по чеку                  | Додаткові пояснення                                                                                                            | Заповненн<br>я<br>SumPrepare<br>Payed                             | Можливість<br>реєстрації якщо<br>чек має<br>StatusTTNCanFisc<br>alize = false | Можливість<br>реєстрації якщо чек<br>має<br>StatusTTNCanFiscali<br>ze = true |
|----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------|-------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 100% післяплата<br>(накладений платіж) | Вид розрахунку, за яким покупець буде здійснювати оплату за товар службі доставки (експедитору) лише після огляду та отримання | Параметр не заповнено                                             | Hi                                                                            | Так                                                                          |
| 100% передплата                        | Вид розрахунку, за яким покупець сплачує повну суму (або її частину) ще до отримання товарів.                                  | Значення параметру відповідає 100% сумі оплати по чеку            | Так                                                                           | Так                                                                          |
| Оплата "50/50"                         | Часткова передплата та післяплата                                                                                              | Параметр заповнено на суму яка не відповідає повній сумі по чеку. | Так                                                                           | Так                                                                          |

## Приклади використання засобів оплат у чеку (спрощений та розширений варіант)

### Чек зі способом оплати «Продаж» (PaymentOrderType=0)

Умова: Чек на суму 100грн з засобом оплати “Платіж LIQPAY” до якого застосовуються реквізити транзакції.

Пояснення: в структурі оплат для спрощеного варіанту застосування засобів оплат заповнюється SumPayByBank значенням суми безготівкової оплати. В параметрі SumPayCheck заповнюється загальна сума по чеку. За допомогою параметра BankPayToolsNm вказується необхідна назва засобу оплат, а ознака True встановлена в параметрі UseTransactionDetailsToSumPayByBank визначає те, що до обраного засобу оплати будуть додані реквізити транзакції POS-терміналу. У розширеному варіанті застосування суму безготівкової оплати та інші характеристики засобу необхідно заповнювати в складі параметра PaymentFormsLst у вигляді масиву з чіткою послідовністю заповнення параметрів.

Приклад заповнення структури оплат (спрощений):

```
{
  "PaymentOrderType": 0,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "100",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "100",
  "TerminalID": "S1K700J8",
  "ApprovalCode": "68014B",
  "RRN": "048867173620",
  "IssuerName": "MASTERCARD",
  "PAN": "XXXXXXXXXX6883",
  "TransactionDate": "03.04.2025 10:49:27",
  "SignVerif": "12334444",
  "AcquireName": "ПриватБанк",
  "InvoiceNumber": "224",
  "BankPayToolsNm": "Платіж LIQPAY",
  "UseTransactionDetailsToSumPayByBank": "true"
}
```

Приклад заповнення структури оплат (розширений):

```
{
  "PaymentOrderType": 0,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "100",
  "TerminalID": "S1K700J8",
  "ApprovalCode": "68014B",
  "RRN": "048867173620",
  "IssuerName": "MASTERCARD",
  "PAN": "XXXXXXXXXX6883",
  "TransactionDate": "03.04.2025 10:49:27",
  "SignVerif": "12334444",
  "AcquireName": "ПриватБанк",
  "InvoiceNumber": "224",
  "PaymentFormsLst": [
    [
      "202",
      "100",
      "Платіж LIQPAY",
      "True"
    ]
  ]
}
```

ТОВ "Фортеця"

"Фортифікація"

місто Київ, вул. Сталева, буд.1

ПН 444444555555

ФН ПРРО 4000024021

КАСИР Сталевий І. І.

Чек №1519122

Штрихкод товару: 2562374154874

Напій Coca-Cola сильногазований

0.5л

1,000 X 100,00100,00 A

-----

СУМА100.00 ГРН

ПДВ А=20%16,67

-----

БЕЗГОТІВКОВА100.00 ГРН

Платіж LIQPAY

Еквайр:ПриватБанк

Термінал:t123456

Вид операції:ОПЛАТА

Сума:100.00 грн

Платіжна система:Visa Credit

ЕПЗ:XXXXXXXXXXXX4455

Код авторизації:32456

RRN:987654321

Дата та час:10.04.2025 10:57:16

Чек №:555777

### Чек зі способом оплати «Передплата» (PaymentOrderType=1)

Умова: Чек на загальну суму 100грн по якому відбувається передплата 50грн з засобом оплати "Платіж LIQPAY". До засобу оплати застосовуються реквізити транзакції.

Пояснення: в структурі оплат для спрощеного варіанту застосування засобів оплат заповнюється SumPayByBank значенням суми безготівкової оплати яка має дорівнювати сумі передплати. В параметрі SumPayCheck заповнюється загальна сума по чеку передплати. В параметрі SumPreparePayed зазначається сума передплати по чеку. За допомогою параметра BankPayToolsNm вказується необхідна назва засобу оплати, а ознака True встановлена в параметрі UseTransactionDetailsToSumPayByBank визначає те, що до обраного засобу оплати будуть додані реквізити транзакції POS-терміналу. У розширеному варіанті застосування суму безготівкової оплати та інші характеристики засобу необхідно заповнювати в складі параметра PaymentFormsLst у вигляді масиву з чіткою послідовністю заповнення..

Приклад заповнення структури оплат (спрощений):

```
{
  "PaymentOrderType": 1,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "50",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "50",
  "SumPreparePayed": "50",
  "TerminalID": "S1K700J8",
  "ApprovalCode": "68014B",
  "RRN": "048867173620",
  "IssuerName": "MASTERCARD",
  "PAN": "XXXXXXXXXX6883",
  "TransactionDate": "03.04.2025 10:49:27",
  "SignVerif": "12334444",
  "AcquireName": "ПриватБанк",
  "InvoiceNumber": "224",
  "BankPayToolsNm": "Платіж LIQPAY",
  "UseTransactionDetailsToSumPayByBank": "true"
}
```

Приклад заповнення структури оплат (розширений):

```
{
  "PaymentOrderType": 1,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
```

```
"SumPayByCertificate": "",
"SumPayCheck": "50",
"SumPreparePayed": "50",
"TerminalID": "S1K700J8",
"ApprovalCode": "68014B",
"RRN": "048867173620",
"IssuerName": "MASTERCARD",
"PAN": "XXXXXXXXXXXX6883",
"TransactionDate": "03.04.2025 10:49:27",
"SignVerif": "12334444",
"AcquireName": "ПриватБанк",
"InvoiceNumber": "224",
"PaymentFormsLst": [
  [
    "202",
    "50",
    "Платіж LIQPAY",
    "True"
  ]
]
}
```

ТОВ "Фортеця"  
"Фортифікація"  
місто Київ, вул. Сталева, буд.1  
ПН 444444555555  
ФН ПРРО 4000024021  
КАСІР Сталевий І. І.

**Чек №1519127**

**ПЕРЕДПЛАТА**

Штрихкод товару: 2562374154874

Напій Coca-Cola сильногазований  
0.5л

1,000 X 100,00 100,00 A

**Сума по чеку** 100.00 грн

Передплата 50.00 грн

**СУМА** 50.00 ГРН

ПДВ А=20% 8,33

**БЕЗГОТІВКОВА 50.00 ГРН**

**Платіж LIQPAY**

Еквайр: ПриватБанк

Термінал: t123456

Вид операції: ОПЛАТА

Сума: 50.00 грн

Платіжна система: Visa Credit

ЕПЗ: XXXXXXXXXXXX4455

Код авторизації: 32456

RRN: 987654321

Дата та час: 10.04.2025 10:57:16

Чек №: 555777

**Залишок до сплати 50.00 ГРН**

Чек зі способом оплати «Післяплата» (PaymentOrderType=2)

Умова: Чек на загальну суму 100грн по якому відбувається доплата 50грн з засобом оплати "Платіж WAYFORPAY". До засобу оплати застосовуються реквізити транзакції.

Пояснення: в структурі оплат для спрощеного варіанту застосування засобів оплат заповнюється SumPayByBank значенням суми безготівкової оплати яка має дорівнювати сумі післяплати. За допомогою параметра BankPayToolsNm вказується необхідна назва засобу оплати, а ознака True встановлена в параметрі UseTransactionDetailsToSumPayByBank визначає те, що до обраного засобу оплати будуть додані реквізити транзакції POS-терміналу. У розширеному варіанті застосування суму безготівкової оплати та інші характеристики засобу необхідно заповнювати в складі параметра PaymentFormsLst у вигляді масиву з чіткою послідовністю заповнення. Обов'язково потрібно вказати за допомогою параметра ParentReceiptFiscalNumber ФН чека по якому була здійснена передплата.

Приклад заповнення структури оплат (спрощений):

```
{
  "PaymentOrderType": 2,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "50",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "",
  "ParentReceiptFiscalNumber": "1519246",
  "TerminalID": "S1K700J8",
  "ApprovalCode": "68014B",
  "RRN": "048867173620",
  "IssuerName": "MASTERCARD",
  "PAN": "XXXXXXXXXXXX6883",
  "TransactionDate": "03.04.2025 10:49:27",
  "SignVerif": "12334444",
  "AcquireName": "ПриватБанк",
  "InvoiceNumber": "224",
  "BankPayToolsNm": "Платіж LIQPAY",
  "UseTransactionDetailsToSumPayByBank": "true"
}
```

Приклад заповнення структури оплат (розширений):

```
{
  "PaymentOrderType": 2,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "",
  "ParentReceiptFiscalNumber": "1519246",
  "TerminalID": "S1K700J8",
  "ApprovalCode": "68014B",
  "RRN": "048867173620",
  "IssuerName": "MASTERCARD",
  "PAN": "XXXXXXXXXXXX6883",
  "TransactionDate": "03.04.2025 10:49:27",
  "SignVerif": "12334444",
  "AcquireName": "ПриватБанк",
  "InvoiceNumber": "224",
  "PaymentFormsLst": [
    [
      "202",
      "50",
      "Платіж LIQPAY",
      "True"
    ]
  ]
}
```

ТОВ "Фортеця"  
"Фортифікація"  
місто Київ, вул. Сталева, буд.1  
ПН 444444555555  
ФН ПРРО 4000024021  
КАСИР Сталевий І. І.

**Чек №1519134**

ФН чека передплати 1519127

Штрихкод товару: 2562374154874

Напій Coca-Cola сильногазований  
0.5л

1,000 X 100,00 100,00 A

-----  
**Сума по чеку** 100.00 грн

Передплата 50.00 грн

Остаточний розрахунок 50.00 грн  
-----

СУМА 50.00 ГРН

ПДВ A=20% 8,33  
-----

**БЕЗГОТІВКОВА 50.00 ГРН**

**Платіж WAYFORPAY**

Еквайр: ПриватБанк

Термінал: t123456

Вид операції: ОПЛАТА

Сума: 50.00 грн

Платіжна система: Visa Credit

ЕПЗ: XXXXXXXXXXXX4455

Код авторизації: 32456

RRN: 987654321

Дата та час: 10.04.2025 10:57:16

Чек №: 555777

Чек зі способом оплати «Інтернет-продаж» (PaymentOrderType=3)

Умова: Чек на загальну суму 100грн по якому відбувається 100% передплата з власним засобом оплати "Власна назва".

Пояснення: для спрощеного варіанту застосування засобів оплат достатньо вказати найменування засобу який буде відображено в передплаті. В параметрі SumPreparePayed зазначається 100% сума передплати. Для розширеного варіанту необхідно в якості коду засобу оплати вказати значення 2, а для суми вказати значення яке відповідає сумі оплати по чеку. Параметр SumPreparePayed структури jSONPayData має містити однакову суму зі значенням в реквізиті суми. Ознака використання реквізитів транзакції POS-терміналу для способу оплати "Інтернет-продаж" не допускається.

Приклад заповнення структури оплат (спрощений):

```
{
  "PaymentOrderType": 3,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "",
  "SumPreparePayed": "100",
  "PrePaymentPayToolsNM": "Власна назва"
}
```

Приклад заповнення структури оплат (розширений):

```
{
  "PaymentOrderType": 3,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "",
  "SumPreparePayed": "100",
  "PaymentFormsLst": [
    [
      "2",
      "100",
      "Власна назва"
    ]
  ]
}
```

ТОВ "Фортеця"

"Фортифікація"

місто Київ, вул. Сталева, буд.1

ПН 444444555555

ФН ПРРО 4000024021

КАСІР Сталевий І. І.

**Чек №1519149**

Номер інтернет-замовлення 31123

Штрихкод товару: 2562374154874

Напій Coca-Cola сильногазований

0.5л

1,000 X 100,00100,00 A

-----

СУМА100.00 ГРН

ПДВ A=20%16,67

-----

Передоплата

**БЕЗГОТІВКОВА100.00 ГРН**

**Власна назва**

### Чек зі способом оплати «Інтернет-продаж» (ТипОплати=3)

Умова: Чек на загальну суму 100грн по якому відбувається комбінована оплата з заповненням засобів оплат для “Передплати” та “Післяплати” по чеку.

Пояснення: для спрощеного варіанту застосування засобів оплат необхідно вказати найменування засобу який буде відображено в передплаті та післяплаті за допомогою параметрів PrePaymentPayToolsNM та AfterPaymentPayToolsNM відповідно. В параметрі SumPreparePayed зазначається потрібна сума передплати. Різниця суми буде відображена в післяплаті. Для розширеного варіанту необхідно в якості коду засобу оплати вказати значення 2 та 201 заповнених в окремих масивах. В масиві засобу для передплати для суми вказати значення яке відповідає сумі передплати по чеку. В масиві засобу післяплати необхідно вказати суму післяплати. Ознака використання реквізитів транзакції POS-терміналу для способу оплати “Інтернет-продаж” не допускається.

Приклад заповнення структури оплат (спрощений):

```
{
  "PaymentOrderType": 3,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "",
  "SumPreparePayed": "50",
  "PrePaymentPayToolsNM": "Власна назва №2",
  "AfterPaymentPayToolsNM": "Власна назва №1"
}
```

Приклад заповнення структури оплат (розширений):

```
{
  "PaymentOrderType": 3,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "",
  "SumPreparePayed": "50",
  "PaymentFormsLst": [
    [
      "2",
      "50",
      "Власна назва №2",
    ],
    [
      "201",
      "50",
      "Власна назва №1",
    ]
  ]
}
```

ТОВ "Фортеця"  
"Фортифікація"  
місто Київ, вул. Сталева, буд.1  
ПН 444444555555  
ФН ПРРО 4000024021  
КАСИР Сталевий І. І.

**Чек №1519152**

Номер інтернет-замовлення 31123

Штрихкод товару: 2562374154874

Напій Coca-Cola сильногазований  
0.5л

1,000 X 100,00 100,00 A

СУМА 100.00 ГРН

ПДВ A=20% 16,67

**БЕЗГОТІВКОВА** Передплата  
**50.00 ГРН**

Власна назва №2

**БЕЗГОТІВКОВА** Післяплата  
**50.00 ГРН**

Власна назва №1

Чек зі способом оплати «Інтернет-продаж» (PaymentOrderType=3)

Умова: Чек на загальну суму 100грн по якому відбувається 100% післяплата з заповненням засобів оплат для "Післяплати" по чеку.

Пояснення: для спрощеного варіанту використання засобів оплати достатньо вказати найменування для післяплати по чеку в параметрі AfterPaymentPayToolsNM. Для розширеного варіанту в масиві засобу потрібно вказати код 201, суму яка дорівнює 100% післяплаті та потрібну назву оплати. Ознака використання реквізитів транзакції POS-терміналу для способу оплати "Інтернет-продаж" не допускається.

Приклад заповнення структури оплат (спрощений):

```
{  
  "PaymentOrderType": 3,  
  "SumCash": "",  
  "SumPayByCard": "",  
  "SumPayByBank": "",  
  "SumPayByCredit": "",  
  "SumPayByCertificate": "",  
  "SumPayCheck": "",  
  "AfterPaymentPayToolsNM": "Власна назва №1"  
}
```

Приклад заповнення структури оплат (розширений):

```
{
  "PaymentOrderType": 3,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "",
  "PaymentFormsLst": [
    [
      "201",
      "100",
      "Власна назва №1"
    ]
  ]
}
```

ТОВ "Фортеця"  
"Фортифікація"  
місто Київ, вул. Сталева, буд.1  
ПН 444444555555  
ФН ПРРО 4000024021  
КАСІР Сталевий І. І.  
**Чек №1519156**  
Номер інтернет-замовлення 31123  
Штрихкод товару: 2562374154874  
Напій Coca-Cola сильногазований  
0.5л  
1,000 X 100,00 100,00 A  
-----  
СУМА 100.00 ГРН  
ПДВ А=20% 16,67  
-----  
Післяплата  
**БЕЗГОТІВКОВА 100.00 ГРН**  
**Власна назва №1**

#### Чек зі способом оплати «Повна післяплата» (PaymentOrderType=4)

Умова: Чек на загальну суму 100грн по якому відбувається 100% післяплата з заповненням засобів оплат для "Передплати" та "Післяплати" по чеку.

Пояснення: для спрощеного варіанту використання засобів оплати достатньо вказати найменування для післяплати по чеку в параметрі AfterPaymentPayToolsNM. Для розширеного варіанту в масиві засобу потрібно вказати код 201, суму яка дорівнює 100% післяплаті та потрібну назву оплати. Ознака використання реквізитів транзакції POS-терміналу для способу оплати "Повна післяплата" не допускається.

Приклад заповнення структури оплат (спрощений):

```
{
  "PaymentOrderType": 4,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "AfterPaymentPayToolsNM": "Власна назва №3"
}
```

Приклад заповнення структури оплат (розширений):

```
{
  "PaymentOrderType": 4,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "",
  "PaymentFormsLst": [
    [
      "201",
      "100",
      "Власна назва №3"
    ]
  ]
}
```

ТОВ "Фортеця"  
"Фортифікація"  
місто Київ, вул. Сталева, буд.1  
ПН 444444555555  
ФН ПРРО 4000024021  
КАСІР Сталевий І. І.

**Чек №1519159**

Номер замовлення 31123

Штрихкод товару: 2562374154874  
Напій Coca-Cola сильногазований  
0.5л  
1,000 X 100,00 100,00 A  
-----  
СУМА 100.00 ГРН  
ПДВ A=20% 16,67  
-----  
**БЕЗГОТІВКОВА** Післяплата  
**Власна назва №3** **100.00 ГРН**

Чек зі способом оплати «Передплата (оплата частинами)» (PaymentOrderType=5)

Умова: Чек на загальну суму 100грн по якому відбувається частковий розрахунок розділений на 3 платежі. Перший платіж складає 50грн.

Пояснення: в спрощеному варіанті використання засобів оплат необхідно встановити суму безготівкової оплати яка дорівнює першому платежу за допомогою параметра SumPayByBank. В параметрі PartPaySum зазначається сума цього першого платежу, а в SumPayCheck сума чека першої сплати. Параметром BankPayToolsNm визначається назва безготівкового засобу оплати до чека. В розширеному варіанті використання інформація про безготівкову оплату, найменування та код засобу зазначається в масиві.

Приклад заповнення структури оплат (спрощений):

```
{  
  "PaymentOrderType": 5,  
  "SumCash": "",  
  "SumPayByCard": "",  
  "SumPayByBank": "50",  
  "SumPayByCredit": "",  
  "SumPayByCertificate": "",  
  "SumPayCheck": "50",  
  "TerminalID": "S1K700J8",  
  "ApprovalCode": "68014B",  
  "RRN": "048867173620",  
  "IssuerName": "MASTERCARD",  
  "PAN": "XXXXXXXXXX6883",  
  "TransactionDate": "03.04.2025 10:49:27",  
  "SignVerif": "12334444",  
  "AcquireName": "ПриватБанк",  
}
```

```
"InvoiceNumber": "224",
"PartPaySum": "50",
"BankPayToolsNm": "Інтернет-еквайринг",
"UseTransactionDetailsToSumPayByBank": "False"
}
```

Приклад заповнення структури оплат (розширений):

```
{
  "PaymentOrderType": 5,
  "SumCash": "",
  "SumPayByCard": "",
  "SumPayByBank": "",
  "SumPayByCredit": "",
  "SumPayByCertificate": "",
  "SumPayCheck": "50",
  "TerminalID": "S1K700J8",
  "ApprovalCode": "68014B",
  "RRN": "048867173620",
  "IssuerName": "MASTERCARD",
  "PAN": "XXXXXXXXXXXX6883",
  "TransactionDate": "03.04.2025 10:49:27",
  "SignVerif": "12334444",
  "AcquireName": "ПриватБанк",
  "InvoiceNumber": "224",
  "PartPaySum": "50",
  "PaymentFormsLst": [
    [
      "202",
      "50",
      "Інтернет-еквайринг"
    ]
  ]
}
```

ТОВ "Фортеця"  
"Фортифікація"  
місто Київ, вул. Сталева, буд.1  
ПН 444444555555  
ФН ПРРО 4000024021  
КАСІР Сталевий І. І.

**Чек №1519162**

**ПЕРЕДПЛАТА**

Штрихкод товару: 2562374154874  
Напій Соса-Cola сильногазований  
0.5л  
1,000 X 100,00 100,00 A  
-----  
**Сума по чеку** 100.00 грн  
Передплата 50.00 грн  
-----  
СУМА 50.00 ГРН  
ПДВ А=20% 8,33  
-----  
**БЕЗГОТІВКОВА 50.00 ГРН**  
**Інтернет-еквайринг**  
-----  
Залишок до сплати 50.00 ГРН

Чек зі способом оплати «Чергова сплата» (PaymentOrderType=6)

Умова: Чек на загальну суму 100грн по якому відбувається черговий розрахунок в 25 грн.  
Перший платіж складав 50грн.

Пояснення: спрощений та розширений варіант застосування безготівкового засобу не відрізняється від прикладу формування чека першої сплати. За виключенням необхідно додати параметр ParentReceiptFiscalNumber з вказаним ФН чека першої сплати.

Приклад заповнення структури оплат (спрощений):

```
{  
  "PaymentOrderType": 6,  
  "SumCash": "",  
  "SumPayByCard": "",  
  "SumPayByBank": "25",  
  "SumPayByCredit": "",  
  "SumPayByCertificate": "",  
  "SumPayCheck": "25",  
  "TerminalID": "S1K700J8",  
  "ApprovalCode": "68014B",  
  "RRN": "048867173620",  
  "IssuerName": "MASTERCARD",  
  "PAN": "XXXXXXXXXX6883",  
  "TransactionDate": "03.04.2025 10:49:27",  
  "SignVerif": "12334444",  
  "AcquireName": "ПриватБанк",  
  "InvoiceNumber": "224",  
  "PartPaySum": "25",  
}
```

```
"BankPayToolsNm": "Інтернет-еквайринг",  
"ParentReceiptFiscalNumber": "1515457",  
"UseTransactionDetailsToSumPayByBank": "False"  
}
```

Приклад заповнення структури оплат (розширений):

```
{  
  "PaymentOrderType": 6,  
  "SumCash": "",  
  "SumPayByCard": "",  
  "SumPayByBank": "",  
  "SumPayByCredit": "",  
  "SumPayByCertificate": "",  
  "SumPayCheck": "25",  
  "TerminalID": "S1K700J8",  
  "ApprovalCode": "68014B",  
  "RRN": "048867173620",  
  "IssuerName": "MASTERCARD",  
  "PAN": "XXXXXXXXXXXX6883",  
  "TransactionDate": "03.04.2025 10:49:27",  
  "SignVerif": "12334444",  
  "AcquireName": "ПриватБанк",  
  "InvoiceNumber": "224",  
  "PartPaySum": "25",  
  "ParentReceiptFiscalNumber": "1519254",  
  "PaymentFormsLst": [  
    [  
      "202",  
      "25",  
      "Інтернет-еквайринг"  
    ]  
  ]  
}
```

ТОВ "Фортеця"  
"Фортифікація"  
місто Київ, вул. Сталева, буд.1  
ПН 444444555555  
ФН ПРРО 4000024021  
КАСІР Сталевий І. І.

**Чек №1519163**

**ЧЕРГОВА СПЛАТА**

ФН чека передплати 1519162

Штрихкод товару: 2562374154874

Напій Coca-Cola сильногазований  
0.5л

1,000 X 100,00 100,00 А

-----  
**Сума по чеку** 100.00 грн

Чергова сплата 25.00 грн

Сума передплат 50.00 грн

-----  
СУМА 25.00 ГРН

ПДВ А=20% 4,17

-----  
**БЕЗГОТІВКОВА 25.00 ГРН**

**Інтернет-еквайринг**

-----  
Залишок до сплати 25.00 ГРН

#### Чек зі способом оплати «Остаточного розрахунку» (PaymentOrderType=7)

Умова: Чек на загальну суму 100грн по якому відбувається остаточний розрахунок в 25 грн.  
Загальна передплата по чеку складала 75 грн.

Пояснення: в спрощеному та розширеному варіанті застосування засобів оплати для чека остаточного розрахунку не потрібно зазначати суму передплат та загальну суму по чеку в параметрі SumPayCheck, сума буде розрахована автоматично. Обов'язково необхідно зазначати в параметрі ParentReceiptFiscalNumber ФН чека першої сплати.

Приклад заповнення структури оплат (спрощений):

```
{  
  "PaymentOrderType": 7,  
  "SumCash": "",  
  "SumPayByCard": "",  
  "SumPayByBank": "25",  
  "SumPayByCredit": "",  
  "SumPayByCertificate": "",  
  "SumPayCheck": "",  
  "TerminalID": "S1K700J8",  
  "ApprovalCode": "68014B",  
  "RRN": "048867173620",  
  "IssuerName": "MASTERCARD",  
  "PAN": "XXXXXXXXXX6883",  
  "TransactionDate": "03.04.2025 10:49:27",  
  "SignVerif": "12334444",  
  "AcquireName": "ПриватБанк",  
  "InvoiceNumber": "224",  
}
```

```
"BankPayToolsNm": "Інтернет-еквайринг",  
"ParentReceiptFiscalNumber": "1515370",  
"UseTransactionDetailsToSumPayByBank": "False"  
}
```

Приклад заповнення структури оплат (розширений):

```
{  
  "PaymentOrderType": 7,  
  "SumCash": "",  
  "SumPayByCard": "",  
  "SumPayByBank": "",  
  "SumPayByCredit": "",  
  "SumPayByCertificate": "",  
  "SumPayCheck": "",  
  "TerminalID": "S1K700J8",  
  "ApprovalCode": "68014B",  
  "RRN": "048867173620",  
  "IssuerName": "MASTERCARD",  
  "PAN": "XXXXXXXXXXXX6883",  
  "TransactionDate": "03.04.2025 10:49:27",  
  "SignVerif": "12334444",  
  "AcquireName": "ПриватБанк",  
  "InvoiceNumber": "224",  
  "ParentReceiptFiscalNumber": "1519254",  
  "UseTransactionDetailsToSumPayByBank": "False",  
  "PaymentFormsLst": [  
    [  
      "202",  
      "25",  
      "Інтернет-еквайринг"  
    ]  
  ]  
}
```

ТОВ "Фортеця"  
"Фортифікація"  
місто Київ, вул. Сталева, буд.1  
ПН 444444555555  
ФН ПРРО 4000024021  
КАСІР Сталевий І. І.

**Чек №1519164**

ФН чека передплати 1519162

Штрихкод товару: 2562374154874

Напій Соса-Cola сильногазований  
0.5л

1,000 X 100,00 100,00 A

-----  
**Сума по чеку** 100.00 грн

Остаточний розрахунок 25.00 грн

Сума передплат 75.00 грн

-----  
СУМА 25.00 ГРН

ПДВ А=20% 4,17

-----  
**БЕЗГОТІВКОВА 25.00 ГРН**

**Інтернет-еквайринг**

## Рекомендований порядок роботи з CashalotApi

Для коректної роботи розрахунково-облікової системи рекомендується описана послідовність виклику функцій при виконанні касових операцій.

### Встановлення параметрів

Перед початком роботи рекомендовано в [SetParameter](#) або [SetParameterEx](#) встановити параметри в такій послідовності:

[SPECIFICAPIMODE](#) (за необхідності)

[PathToCashalotDir](#) (обов'язково)

[DeviceIDFnRRO](#) (обов'язково)

[NOAUTOUPDATE](#)

[NOINTERFACEMODE](#)

[PathToCertificate](#)

[PwdToCertificate](#)

[USETOKEN](#)

[NOAUTOOPENSIFT](#)

[AUTOPRINTMODE](#)

[NOPRINTMODE](#)

[DEFAULTPRINTERNAME](#)

[SHOWREPORTADDITIONALINFO](#)

[USEGOODSSTORAGEMODE](#) (за необхідністю)

[USEPRGPOSTERMINAL](#) (за необхідністю)

## **Відкриття зміни**

Для відкриття касової зміни необхідно скористатися методом [OpenShift](#).

## **Продаж товарів або послуг**

Для формування чека та його фіскалізації створюються JSON-структури з масивом товарів та підсумками оплати по чеку, які передаються як параметри для виконання методу фіскалізації чека [FiscalizeCheck](#).

У чеку необхідно сформувати щонайменше 1 рядок товарів та встановити розширений порядок оплати для поточного чека [PaymentOrderType](#).

Спосіб оплати (0-звичайна | 1 - передоплата | 2 - післяплата | 3 - інтернет продаж | 4 - повна післяплата | 5 - передоплата (оплата частинами) | 6 - чергова оплата (оплата частинами) | 7 - остаточний розрахунок (оплата частинами)).

## **Повернення товарів або рекомпенсація послуг**

Для повернення товару або рекомпенсації послуги необхідно скористатися методом [FiscalizeReturnCheck](#), в якому потрібно вказати фіскальний номер чека, за яким було продано товар або послугу.

## **Використання POS-термінала в API COM Cashalot при проведенні безготівкових розрахунків**

Перед початком роботи необхідно встановити параметр **USEPRGPOSTERMINAL** за допомогою SetParameter або SetParameterEx. Для виклику POS-термінала під час проведення безготівкового розрахунку слід виконати його налаштування РМК Cashalot в розділі «Налаштування - Додаткове обладнання - POS-термінали». В подальшому при виклику метода FiscalizeCheck або FiscalizeReturnCheck якщо в масив підсумків оплат зазначена форма оплати картка (SumPayByCard)то перед відправкою чека на ФСКО попередньо відбувається запит на POS-термінал з сумою коштів яка була зазначена для параметра SumPayByCard. В разі успішної транзакції на терміналі відбувається реєстрація фіскального чека. В разі неуспішної транзакції на банківському терміналі операція реєстрації чека не відбувається.

## **Використання декількох платіжних POS-терміналів при проведенні безготівкових розрахунків**

Першочергово для можливості використання декількох платіжних терміналів в API COM Cashalot слід виконати їх налаштування в програмі в розділі «Налаштування - Додаткове обладнання - POS-термінали» та призначити ознаку необхідному пристрою який буде використовуватись за замовчуванням в програмі. Саме до такого термінала буде відбуватись звернення при виконанні безготівкової оплати в API COM Cashalot за умови використання параметра **USEPRGPOSTERMINAL**.

За необхідності зміни пристрою який буде використовуватись в API слід викликати метод **SetDafeultPOSTerminal** передавши значення ID терміналу який буде встановлено пристроєм за замовчуванням.

Для отримання ID підключених терміналів до програми необхідно скористатися методом **GetListPOSTerminal**.

### **Застосування засобів оплати при формуванні чеку**

Для застосування засобів оплати у чеку

### **Пошук фіскального та локального номера чека**

Для визначення фіскального номера чека за його локальним номером використовується метод **GetReceiptFiscaNumberByLocalNumber**.

Для визначення локального номера чека за його фіскальним номером використовується функція **GetReceiptLocalNumberByFiscalNumber**.

### **Відправка зареєстрованого чека покупцю у Viber, SMS або на E-mail.**

Для відправки чека покупцю за допомогою сервісів доставки слід використовувати метод **SendCheckToSutomer**

### **Отримання XML структури чека**

Метод **GetReceiptXML** повертає XML-структуру чека за його вказаним фіскальним номером.

### **Відображення зареєстрованого чека**

Метод [ShowReceipt](#) відкриває вікно з візуальним відображенням чека. У вікні є кнопки "Друк", "Надіслати на email", "Надіслати чек в SMS" та "Надіслати чек в Viber" які дозволяють роздрукувати або відправити чек з використанням сервісів доставки Viber, SMS або E-Mail.

### **Друкувати періодичний звіт по датам**

Метод [PrintPReportDate](#) використовується для друку періодичного звіту за проміжок часу, що фіксується значеннями (дата початку періоду - у форматі "dd.mm.yyyy" та дата закінчення періоду - у форматі "dd.mm.yyyy").

### **Налаштування електронної пошти**

Електронну пошту можна налаштувати в РМК Cashälot, а саме в розділі "Загальні налаштування - Господарська одиниця - Налаштування електронної пошти". Для налаштування необхідно зазначити електронну адресу, з якої будуть відправлятися повідомлення, а також, налаштування з'єднання з поштовою скринькою: вказати SMTP-сервер, порт, ім'я для входу в поштову скриньку, пароль поштової скриньки.

Відправляти чек на пошту можна після фіскалізації чека, виклику функції відображення чека ([ShowReceipt](#)) або використовувачи ознаку автоматичної доставки чека на електронну пошту.

### **Службове внесення грошей в касу**

Для виконання службового внесення грошей в касу використовується методи [ServiceInput](#) або [ServiceInputEx](#).

### **Службова видача грошей з каси**

Для виконання службової видачі з каси використовується методи [ServiceOutput](#) або [ServiceOutputEx](#).

### **Формування Х-звіту**

Х-звіт формується в результаті виконання методів [PrintXReport](#) або [GetXReport](#).

### **Отримання чеків**

Для отримання чеків з історії операцій реалізовано функції, що передбачають завантаження чеків за певний період часу або за певну касову зміну. Дані завантажуються з бази даних Cashälot.

Для формування списку чеків за заданий період часу скористайтесь: [GetFirstReceiptByPeriod](#)

Для формування списку чеків за задану касову зміну скористайтесь: [GetFirstReceiptByShift](#)

Обидві функції формують список чеків і повертають перший чек з сформованого списку. Для отримання наступних чеків скористайтесь: [GetNextReceipt](#) необхідну кількість разів, або поки не буде вичерпано список чеків.

## **Отримання документів Z-звітів закритих касових змін**

Для отримання Z-звітів з історії операцій реалізовано методи, що передбачають завантаження документів за певний період часу або для певної касової зміни. Дані завантажуються з бази даних Cashälot.

Для формування списку документів за заданий період часу скористайтеся:

**GetFirstZReportByPeriod**. Функція формує список чеків і повертає перший документ із сформованого списку.

Для отримання наступних документів скористайтеся: **GetNextZReport** необхідну кількість разів, або поки не буде вичерпано список документів.

Для отримання Z-звіту заданої касової зміни скористайтеся: **GetZReportByShift**

Якщо звіт для вказаної касової зміни відсутній (наприклад, поточна касова зміна), метод поверне ознаку неуспішності операції.

## **Отримання останнього Z-звіту**

Для отримання і перегляду останнього зареєстрованого Z-звіту використовується метод **GetLastZReport**.

## **Закриття зміни, формування Z-звіту**

Формування Z-звіту відбувається при закритті зміни, тому виконання операції закриття зміни та формування Z-звіту рівнозначні за результатом. Для цього виконується один з методів: **PrintZReport** або **CloseShift**.

Виклик зазначених функцій призводить до виконання послідовності операцій: службова видача з каси наявної готівки, формування Z-звіту та закриття зміни. Проте, виконання функції **PrintZReport** повертає ознаку успішності операції, в той час як виконання **CloseShift** повертає структуру **CashalotApiRetVal**. Якщо встановлений параметр **NOINTERFACEMODE** = **True**, то службову видачу потрібно передбачити перед виконанням методу **CloseShift**.

## **Використання складського обліку в API Cashalot**

За необхідністю можливо використовувати ведення складського обліку при роботі з API COM Cashälot якщо подібний функціонал відсутній в розрахунково-обліковій системі користувача яка використовує взаємодію з API. Для цього необхідно перед початком роботи встановити параметр **USEGOODSSTORAGEMODE** за допомогою **SetParameter** або **SetParameterEx** в необхідному режимі роботи

## **Кроки по практичному застосуванню параметра USEGOODSSTORAGEMODE**

1. Імпортувати номенклатуру, яка використовується в API COM Cashälot, до кабінету користувача. Після імпорту необхідно перенести номенклатуру до еталонної групи.
2. В кабінеті користувача в модулі «Склад» сформувати та провести документ «Надходження» по необхідним товарним позиціям.
3. Налаштувати та встановити використання параметра **USEGOODSSTORAGEMODE** у необхідному режимі.
4. Виконати синхронізацію даних по товарам (номенклатурі) та залишкам товарів методом **SyncGoodsWithBalance**. Подальше оновлення інформації по залишкам відбувається в автоматичному

режимі при відкриті зміни та реєстрації фіскального чека, а також в ручному режимі, з використанням методів синхронізації товарів та залишків по ним або окремо, виконавши синхронізацію залишків.

5. Сформувати та заповнити фіскальний чек даними по номенклатурі, для якої доступні залишки по товарам.

**Зверніть увагу!** Коректна робота обраного параметра відбувається лише за умови, що в API COM Cashalot використовується одна й та сама номенклатура, що заведена в кабінеті користувача.

## Приклад реалізації підключення COM бібліотеки на C++

Створення файлів заголовків \*.h

Якщо відсутні файли заголовків, або потрібно їх оновити:

1. Генеруємо через імпорт, вказавши шлях до COM бібліотеки, наприклад:

```
#import "C:\\Users\\User1\\AppData\\Local\\Cashalot\\CashalotApi64.dll"
```

2. Компілюємо solution з рядком import (після компіляції цей рядок прибираємо).

3. Далі в папці debug\ вашого solution з'являються файли CashalotApi64.tlh та CashalotApi64.tli.

4. Переносимо їх в каталог solution, перейменовуємо в CashalotApi.tlh, CashalotApi.tli (за бажанням, можна зібрати два x32 та x64, та підключати директивами).

5. Всередині CashalotApi.tlh прибираємо повний шлях до CashalotApi64.tli і також міняємо ім'я на CashalotApi.tli.

6. Заголовні файли готові до використання, підключаємо їх в solution:

```
#include "CashalotApi.tlh" using namespace CashalotApi.
```

## Зауваження

1. При старті програми додаємо ініціалізацію роботи з COM об'єктами: CoInitialize(NULL).
2. При закритті програми припиняємо роботу з COM об'єктами: CoUninitialize().
3. Перед початком роботи з COM-об'єктом ініціалізуємо його викликом InitCashaLotApiComInterface(), перед виходом з програми звільняємо ресурси викликом функції TermCashaLotApiComInterface().
4. Функції повертають структуру ICashalotApiRetVal або BOOL.

## Лістинг

```
#include "stdafx.h"
#include "SampleUseCashaLotCOMAPI_CPP.h"
#include "SampleUseCashaLotCOMAPI_CPPDlg.h"
#include "afxdialogex.h"
#include <string>           // std::string
#include <sstream>          // std::ostringstream
#include <algorithm>
#include <list>

#ifdef _DEBUG
#define new DEBUG_NEW
#endif

#include "CashaLotApi.tlh"
using namespace CashaLotApi;

extern LPCTSTR pAppName;

CSampleUseCashaLotCOMAPICPPDlg::CSampleUseCashaLotCOMAPICPPDlg(CWnd* pParent
/*=nullptr*/)
: CDialogEx(IDD_SAMPLEUSECASHALOTCOMAPI_CPP_DIALOG, pParent)
, m_FiscalNumberRRO(_T(""))
, m_PathToCashaLot(_T(""))
, m_IsNoInterfaceMode(FALSE)
, m_PathCert(_T(""))
, m_pwdCert(_T(""))
, m_PrinterName(_T(""))
, m_useToken(FALSE)
, m_UseDefCshalotPrinter(FALSE)
, m_PrintUseDefWindowsPrinter(FALSE)
, m_PrintUseNamedPrinter(FALSE)
, m_DateBeg(COleDateTime::GetCurrentTime())
, m_DateEnd(COleDateTime::GetCurrentTime())
{
    m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
}

void CSampleUseCashaLotCOMAPICPPDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialogEx::DoDataExchange(pDX);
    DDX_Control(pDX, IDC_EDPATH, m_edPathToCashaLot);
    DDX_Control(pDX, IDC_EDFISKNUMRCP, m_edFiscNumRcp);
    DDX_Text(pDX, IDC_EDPATH2, m_FiscalNumberRRO);
    DDX_Text(pDX, IDC_EDPATH, m_PathToCashaLot);
    DDX_Control(pDX, IDC_EDPATH2, m_edFiscalNumberRRO);
}
```

```

DDX_Control(pDX, IDC_BTGETPATH, m_btnPath);
DDX_Check(pDX, IDC_CHECKNOINTERFACEMODE, m_IsNoInterfaceMode);
DDX_Text(pDX, IDC_EDPATHHTOCERT, m_PathCert);
DDX_Text(pDX, IDC_EDPWD, m_pwdCert);
DDX_Control(pDX, IDC_RADIO1, m_rdbtPrintUseNamedPrinter);
DDX_Control(pDX, IDC_RADIO2, m_rdbtPrintUseDefPrinter);

DDX_Text(pDX, IDC_EDPATH5, m_PrinterName);
DDX_Control(pDX, IDC_EDPATHHTOCERT, m_edPathCert);
DDX_Control(pDX, IDC_EDPWD, m_edPwdCert);
DDX_Control(pDX, IDC_CHECKNOINTERFACEMODE, m_chNoInterfaceMode);
DDX_Check(pDX, IDC_CHECKUSETOKEN, m_useToken);
DDX_Control(pDX, IDC_CHECKUSETOKEN, m_chUseToken);
DDX_Check(pDX, IDC_CHECKUSECASHALOTPRINTERSETTINGS,
m_UseDefCashalotPrinter);
DDX_Control(pDX, IDC_CHECKUSECASHALOTPRINTERSETTINGS,
m_chUseDefCashalotPrinter);
DDX_Control(pDX, IDC_EDPATH5, m_edPrinterName);
DDX_Radio(pDX, IDC_RADIO2, m_PrintUseDefWindowsPrinter);
DDX_Radio(pDX, IDC_RADIO1, m_PrintUseNamedPrinter);
DDX_DateTimeCtrl(pDX, IDC_DATETIMEPICKER1, m_DateBeg);
DDX_DateTimeCtrl(pDX, IDC_DATETIMEPICKER2, m_DateEnd);
}

BEGIN_MESSAGE_MAP(CSampleUseCashaLotCOMAPICPPDlg, CDialogEx)
    ON_WM_SYSCOMMAND()
    ON_WM_PAINT()
    ON_WM_QUERYDRAGICON()
    ON_BN_CLICKED(IDC_BTOPENSHIFT,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtopenshift)
    ON_BN_CLICKED(IDC_BTcloseShift,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtcloseshift)
    ON_BN_CLICKED(IDC_BTfiscalizeReceipt,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtfiscalizereceipt)
    ON_BN_CLICKED(IDC_BTReturnReceipt,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtreturnreceipt)
    ON_BN_CLICKED(IDC_BTServiceInput,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtserviceinput)
    ON_BN_CLICKED(IDC_BTServiceOutput,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtserviceoutput)
    ON_BN_CLICKED(IDC_BTServiceOutput2,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedPrintXRep)
    ON_WM_DESTROY()
    ON_BN_CLICKED(IDC_BTGETPATH,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtgetpath)
    ON_BN_CLICKED(IDC_BTGETPATHHTOCERT,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtgetpathtocert)
    ON_BN_CLICKED(IDC_BTPrintXRepEx,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtprintxrepex)
    ON_BN_CLICKED(IDC_CHECKUSECASHALOTPRINTERSETTINGS,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedCheckusecshalotprintersettings)
    ON_BN_CLICKED(IDC_RADIO2,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedRadio2)
    ON_BN_CLICKED(IDC_RADIO1,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedRadio1)
    ON_BN_CLICKED(IDC_BTGETPRINTERNAME,
&CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtgetprintername)

```

```

    ON_BN_CLICKED(IDC_BTPrintLastZRep,
&CSampleUseCashaLotCOMAPICPPDLg::OnBnClickedBtprintlastzrep)
    ON_BN_CLICKED(IDC_BTGETRECEIPTANDREPFROMPERIOD,
&CSampleUseCashaLotCOMAPICPPDLg::OnBnClickedGetRcpByPeriod)
END_MESSAGE_MAP()

// CSampleUseCashaLotCOMAPICPPDLg message handlers

BOOL CSampleUseCashaLotCOMAPICPPDLg::OnInitDialog()
{
    CDialogEx::OnInitDialog();

    // Set the icon for this dialog. The framework does this automatically
    // when the application's main window is not a dialog
    SetIcon(m_hIcon, TRUE);        // Set big icon
    SetIcon(m_hIcon, FALSE);       // Set small icon

    // TODO: Add extra initialization here
    CWinApp* pApp = AfxGetApp();
    m_PathToCashaLot = pApp->GetProfileString(pAppName, _T("PathToCashaLot"));
    m_FiscalNumberRRO = pApp->GetProfileString(pAppName,
_T("FiscalNumberRRO"));
    m_PathCert = pApp->GetProfileString(pAppName, _T("PathToCert"));
    m_pwdCert = pApp->GetProfileString(pAppName, _T("PwdToCert"));
    m_IsNoInterfaceMode = pApp->GetProfileString(pAppName,
_T("NoInterfaceMode")) == L"1";
    m_useToken = pApp->GetProfileString(pAppName, _T("UseToken")) == L"1";

    m_PrintUseDefWindowsPrinter = (pApp->GetProfileString(pAppName,
_T("PrintUseDefWindowsPrinter")) == L"0")? 0 : -1;
    m_PrintUseNamedPrinter = (pApp->GetProfileString(pAppName,
_T("PrintUseNamedPrinter")) == L"0")? 0 : -1;;
    m_PrinterName = pApp->GetProfileString(pAppName, _T("PrinterName"));

    m_edPrinterName.SetWindowText(m_PrinterName);

    CString defPrndPrm = pApp->GetProfileString(pAppName,
_T("UseDefCashalotPrinter"));
    if (defPrndPrm == L"") defPrndPrm = L"1";
    m_UseDefCashalotPrinter = defPrndPrm == L"1";

    m_chUseDefCashalotPrinter.SetCheck(m_UseDefCashalotPrinter);

    m_chNoInterfaceMode.SetCheck(m_IsNoInterfaceMode);
    m_chUseToken.SetCheck(m_useToken);

    m_edFiscalNumberRRO.SetWindowText(m_FiscalNumberRRO);
    m_edPathToCashaLot.SetWindowText(m_PathToCashaLot);
    m_edPathCert.SetWindowText(m_PathCert);
    m_edPwdCert.SetWindowText(m_pwdCert);
    UpdateData(0);

    OnBnClickedCheckusecashalotprintersettings();
    OnBnClickedRadio2();

    return TRUE; // return TRUE unless you set the focus to a control
}

void CSampleUseCashaLotCOMAPICPPDLg::OnDestroy()

```

```

{
    UpdateData();
    CDialogEx::OnDestroy();

    CWinApp* pApp = AfxGetApp();
    pApp->WriteProfileString(pAppName, _T("PathToCashaLot"), m_PathToCashaLot);
    pApp->WriteProfileString(pAppName, _T("FiscalNumberRRO"),
m_FiscalNumberRRO);
    pApp->WriteProfileString(pAppName, _T("PathToCert"), m_PathCert);
    pApp->WriteProfileString(pAppName, _T("PwdToCert"), m_pwdCert);
    pApp->WriteProfileString(pAppName, _T("NoInterfaceMode"),
((m_IsNoInterfaceMode == TRUE)? L"1" : L"0"));
    pApp->WriteProfileString(pAppName, _T("UseToken"), ((m_useToken == TRUE)?
L"1" : L"0"));
    pApp->WriteProfileString(pAppName, _T("UseDefCashalotPrinter"),
((m_UseDefCashalotPrinter == TRUE)? L"1" : L"0"));

    pApp->WriteProfileString(pAppName, _T("PrintUseDefWindowsPrinter"),
(m_PrintUseDefWindowsPrinter == 0 ? L"0" : L"-1"));
    pApp->WriteProfileString(pAppName, _T("PrintUseNamedPrinter"),
(m_PrintUseNamedPrinter == 0 ? L"0" : L"-1"));
    pApp->WriteProfileString(pAppName, _T("PrinterName"), m_PrinterName );
}

void CSampleUseCashaLotCOMAPICPPDlg::OnSysCommand(UINT nID, LPARAM lParam)
{
    CDialogEx::OnSysCommand(nID, lParam);
}

// If you add a minimize button to your dialog, you will need the code below
// to draw the icon. For MFC applications using the document/view model,
// this is automatically done for you by the framework.

void CSampleUseCashaLotCOMAPICPPDlg::OnPaint()
{
    if (IsIconic())
    {
        CPaintDC dc(this); // device context for painting

        SendMessage(WM_ICONERASEBKGND,
reinterpret_cast<WPARAM>(dc.GetSafeHdc()), 0);

        // Center icon in client rectangle
        int cxIcon = GetSystemMetrics(SM_CXICON);
        int cyIcon = GetSystemMetrics(SM_CYICON);
        CRect rect;
        GetClientRect(&rect);
        int x = (rect.Width() - cxIcon + 1) / 2;
        int y = (rect.Height() - cyIcon + 1) / 2;

        // Draw the icon
        dc.DrawIcon(x, y, m_hIcon);
    }
    else
    {
        CDialogEx::OnPaint();
    }
}

```

```

}

// The system calls this function to obtain the cursor to display while the
// user drags
// the minimized window.
HCURSOR CSampleUseCashaLotCOMAPICPPDlg::OnQueryDragIcon()
{
    return static_cast<HCURSOR>(m_hIcon);
}

static int CALLBACK BrowseCallbackProc(HWND hwnd, UINT uMsg, LPARAM lParam,
LPARAM lpData)
{
    if (uMsg == BFFM_INITIALIZED)
    {
        std::string tmp = (const char *)lpData;

        SendMessage(hwnd, BFFM_SETSELECTION, TRUE, lpData);
    }

    return 0;
}

CString BrowseFolder()
{
    TCHAR path[MAX_PATH] = { 0 };

    const char * path_param = "";

    BROWSEINFO bi = { 0 };
    bi.lpszTitle = _T("Вибір каталогу з програмою CashaLot");
    bi.ulFlags = BIF_RETURNONLYFSDIRS | BIF_NEWDIALOGSTYLE;
    bi.lpfnc = BrowseCallbackProc;
    bi.lParam = (LPARAM)path_param;

    LPITEMIDLIST pidl = SHBrowseForFolder(&bi);

    if (pidl != 0)
    {
        //get the name of the folder and put it in path
        SHGetPathFromIDList(pidl, path);

        //free memory used
        IMalloc * imalloc = 0;
        if (SUCCEEDED(SHGetMalloc(&imalloc)))
        {
            imalloc->Free(pidl);
            imalloc->Release();
        }

        return CString(path);
    }

    return CString("");
}

```

```

void
CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedCheckusecshalotprintersettings()
{
    UpdateData();

    m_rdbtPrintUseDefPrinter.EnableWindow(!m_UseDefCshalotPrinter);
    m_rdbtPrintUseNamedPrinter.EnableWindow(!m_UseDefCshalotPrinter);
    m_edPrinterName.EnableWindow(!m_UseDefCshalotPrinter);
}

```

```

void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedRadio2()
{
    UpdateData();
    m_rdbtPrintUseDefPrinter.SetCheck(m_PrintUseDefWindowsPrinter == 0);
    m_rdbtPrintUseNamedPrinter.SetCheck(m_PrintUseDefWindowsPrinter == -1);
}

```

```

void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedRadio1()
{
    UpdateData();
    m_rdbtPrintUseDefPrinter.SetCheck(m_PrintUseDefWindowsPrinter == -1);
    m_rdbtPrintUseNamedPrinter.SetCheck(m_PrintUseDefWindowsPrinter == 0);
}

```

```

void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtgetprintername()
{
    CPrintDialog dlg(true, PD_NOPAGENUMS | PD_HIDEPRINTTOFILE |
PD_NOCURRENTPAGE | PD_NONETWORKBUTTON);
    if (dlg.DoModal() == IDOK)
    {
        m_PrinterName = dlg.GetDeviceName();
        UpdateData(0);
    }
}

```

```

void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtgetpath()
{
    CString ret = BrowseFolder();
    if (ret.GetLength())
    {
        m_PathToCashaLot = ret;
        m_edPathToCashaLot.SetWindowText(m_PathToCashaLot);
    }
}

```

/////////////////////////////////ПРИКЛАД ВИКОРИСТАННЯ/////////////////////////////////

```

struct CashaLotApiRsp
{
    void Clear()
    {
        Ret = false;
        ErrorString = "";
        Values.clear();
    }
}

```

```

CashaLotApiRsp()
{
    Clear();
}
bool Ret ;
std::string ErrorString ;
std::list< std::pair<std::string, std::string> > Values;
};

```

```

CashaLotApiRsp rspEx;
CashaLotApiRsp DeserializeResp(std::string response)
{
    rspEx.Clear();
    return rspEx;
}

```

*//Структура інформації по позиції товару в чеці(артикул, назва сумма, кількість, податки і тд.)*

```

struct ReceiptItemInfo
{
    ReceiptItemInfo()
    {
        VendorCode = "";
        Name = "";
        CdUKDZED = "";
        CdDKPP = "";
        Barcode = "";
        GoodsType = 1; //1- товар, 2- услуга
        UnitType = "шт";
        Quantity = 0;
        Price = 0;
        Amount = 0;
        DiscountPrc = 0;
        DiscountSum = 0;
        VATRate = 0;
        IsPriceIncludeVAT = true;
        SumVAT = 0;
        IsExcise = false;
        OtherParametrs = "";
    }
}

```

```

std::string VendorCode;
std::string Name;
std::string CdUKDZED;
std::string CdDKPP;
std::string Barcode;
int GoodsType; //1- товар, 2- услуга
std::string UnitType;
double Quantity;
double Price;
double Amount;
double DiscountPrc;
double DiscountSum;
double VATRate;
bool IsPriceIncludeVAT;
double SumVAT;
bool IsExcise;

```

```

std::string OtherParams;

//Повертає JSON рядок з інформацією по позиції товару
std::string ToJSON()
{
    std::ostringstream ret;

    ret << "{" <<
        "\"VendorCode\": \"" << (VendorCode.c_str()) <<
        "\", \"Name\": \"" << (Name.c_str()) <<
        "\", \"CdUKDZED\": \"" << (CdUKDZED.c_str()) <<
        "\", \"CdDKPP\": \"" << (CdDKPP.c_str()) <<
        "\", \"Barcode\": \"" << (Barcode.c_str()) <<
        "\", \"GoodsType\": \"" << GoodsType <<
        "\", \"UnitType\": \"" << (UnitType.c_str()) <<
        "\", \"Quantity\": \"" << Quantity <<
        "\", \"Price\": \"" << Price <<
        "\", \"Amount\": \"" << Amount <<
        "\", \"DiscountPrc\": \"" << DiscountPrc <<
        "\", \"DiscountSum\": \"" << DiscountSum <<
        "\", \"VATRate\": \"" << VATRate <<
        "\", \"IsPriceIncludeVAT\": \"" << ((!IsPriceIncludeVAT) ? "false" :
"true") <<
        "\", \"SumVAT\": \"" << SumVAT <<
        "\", \"IsExcise\": \"" << ((!IsExcise)? "false": "true") <<
        "\", \"OtherParams\": \"" << (OtherParams.c_str()) << "\""
        << "}"
        ;

    return ret.str();
}
};

struct ReceiptInfoLst
{
private:
    std::list<ReceiptItemInfo*> lstReceiptItemsInfo;
    std::string Comment;
public:
    ReceiptInfoLst()
    {
        Clear();
    };
    ~ReceiptInfoLst()
    {
        Clear();
    };

    std::string ToJSON()
    {
        std::ostringstream ret;
        std::ostringstream comment;
        long cnt=0;

        if(Comment.length() > 0)
            comment << ", \"Comment\": \"" << Comment.c_str() << "\"";

        ret << "{\"ReceiptLst\":[";
    }

```

```

        for(std::list<ReceiptItemInfo*>::iterator x=
1stReceiptItemsInfo.begin(); x!= 1stReceiptItemsInfo.end(); ++x)
        {
            if(cnt++!=0)
                ret << ",";
            ret << ((*x)->ToJSON());
        }
        ret << "]"<< (comment.str().c_str()) <<"}";

        return ret.str();
    }

    _bstr_t ToBSTRJSON()
    {
        return _bstr_t(ToJSON().c_str());;
    }
    void Clear()
    {
        Comment.clear();
        for (std::list<ReceiptItemInfo*>::iterator x =
1stReceiptItemsInfo.begin(); x != 1stReceiptItemsInfo.end(); ++x)
            delete (*x);
        1stReceiptItemsInfo.clear();
    }

    void InsertItem( ReceiptItemInfo* check )
    {
        1stReceiptItemsInfo.push_back(check);
    }

    void SetComment(std::string strComment)
    {
        Comment = strComment;
    }
};

//Структура оплати по чеку
struct PayInfoRsp
{
    //Сума готівки наданої покупцем(може бути більшою за сумму до сплати, з
неї вираховується решта/здача)
    double SumCash;
    //Сума оплати картою
    double SumPayByCard;
    //Сума оплати за рахунок кредитних коштів
    double SumPayByCredit;
    //Сума оплати сертифікатом
    double SumPayByCertificate;
    // Сума фактичної оплати всіма видами оплати, з урахуванням заокруглення
    double SumPayCheck;
    // Email платника
    std::string CustomerEmail;

    PayInfoRsp()
    {
        //Сума готівки наданої покупцем(може бути більшою за сумму до сплати,
з неї вираховується решта/здача)

```

```

    SumCash =
    SumPayByCard =
    SumPayByCredit =
    SumPayByCertificate =
    SumPayCheck = 0;
    CustomerEmail = "";
}

PayInfoRsp()
//До структури оплати PayInfoRsp, додані поля:
//Тип виду оплати (0-звичайна | 1 - передоплата | 2 - післяплата | 3 -
інтернет продаж )
    public int PaymentOrderType;
    //Сума попередньої оплати
    public String SumPreparePayed { get; set; }
    //Фіскальний номер чека по якому здійснюється доплата/повернення
    public String ParentReceiptFiscalNumber { get; set; }
    //Фіскальний номер РРО по якому здійснюється доплата/повернення
    public String ParentRROFiscalNumber { get; set; }
    // OtherParametrs - Додаткові параметри ХМЛ(зарезервовано - не
використовується)
    public String OtherParametrs { get; set; }

В структуру інформації о чеке LstCheckInfoRsp, добавлено поле:
    //Номер замовлення/договору
    public String SaleOrderNumber { get; set; } = String.Empty;

    //Отримати JSON рядок з інформацією про оплату по чеку(готівка, карта,
знижки та заокруглення)
    std::string ToJSON()
    {
        std::ostringstream ret;
        long cnt = 0;
        ret << "{\"SumCash\":\"" << SumCash <<
            "\",\"SumPayByCard\":\"" << SumPayByCard <<
            "\",\"SumPayByCredit\":\"" << SumPayByCredit <<
            "\",\"SumPayByCertificate\":\"" << SumPayByCertificate <<
            "\",\"SumPayCheck\":\"" << SumPayCheck <<
            "\",\"CustomerEmail\":\"" << CustomerEmail << "\"}";

        return ret.str();
    }

    _bstr_t ToBSTRJSON()
    {
        return _bstr_t(ToJSON().c_str());
    }
};

std::ostringstream strRet;
ReceiptInfoLst gsLstCheckItems;
ICashalotApiRetValPtr retVal = NULL;
ICashalotApiAddinPtr gsCashaLotApi = NULL;
BSTR fiscalNumRRO = NULL;
BSTR pathToCashaLot = NULL;
BSTR pathToCert = NULL;
BSTR pwdToCert = NULL;
bool IsNoInterfaceMode = false;

```

```

bool useToken = false;
int  setupPrinterMode = 0;
BSTR printerName = NULL;

void SetPrinterParametr()
{
    if (gsCashaLotApi != NULL)
    {
        gsCashaLotApi->SetParameter( L"DEFAULTPRINTERNAME", (setupPrinterMode
== 0)? BSTR(L"") : ((setupPrinterMode == 1)? BSTR(L"DEFAULTWINDOWS") :
printerName) );
    }
}

//Ініціалізація COM об'єкта CashaLot
bool InitCashaLotApiComInterface()
{
    if ( gsCashaLotApi == NULL )
    {
        // Вказуємо ІД COM об'єкта AddIn.CashaLotApi, який має бути попередньо
зареєстрований в реєстрі(regsvr32 D:\prro\CashLot\CashaLotApi64.dll)
        char* szProgID = "AddIn.CashaLotApi";
        WCHAR  szWideProgID[128];
        CLSID  clsid;
        long lLen = MultiByteToWideChar(CP_ACP,
            0,
            szProgID,
            strlen(szProgID),
            szWideProgID,
            sizeof(szWideProgID));

        szWideProgID[lLen] = '\\0';
        HRESULT hr = ::CLSIDFromProgID(szWideProgID, &clsid);
        if (FAILED(hr))
        {
            MessageBox(0, L"Unable to get CLSID from ProgID.
AddIn.CashaLotApi", 0, 0);
            return false;
        }

        IClassFactory* pCF;
        // Отримати фабрику класів
        hr = CoGetClassObject(clsid,
            CLSCTX_INPROC,
            NULL,
            IID_IClassFactory,
            (void**)&pCF);
        if (FAILED(hr))
        {
            MessageBox(0, L"Failed to GetClassObject server instance.", 0, 0);
            return false;
        }

        // за допомогою фабрики класів створити екземпляр
        // компонента та отримати інтерфейс IUnknown.
        IUnknown* pUnk;
        hr = pCF->CreateInstance(NULL, IID_IUnknown, (void**)&pUnk);
    }
}

```

```

// Release the class factory
pCF->Release();

if (FAILED(hr))
{
    MessageBox(0, L"Failed to create server instance.", 0, 0);
    return false;
}

hr = pUnk->QueryInterface(__uuidof(ICashaLotApiAddin),
(LPVOID*)&gsCashaLotApi);
pUnk->Release();

if (FAILED(hr))
{
    MessageBox(0, L"Failed to create ICashaLotApiAddin", 0, 0);
    return false;
}

//-----Задаємо обов'язкові параметри(шлях до каталогу з програмою
Кашалот, та фіскальний номер РРО з якою будемо працювати)
-----

gsCashaLotApi->SetParameter(L"PathToCashalotDir", pathToCashaLot);
gsCashaLotApi->SetParameter(L"DeviceIDFnRRO", fiscalNumRRO);
gsCashaLotApi->SetParameter(L"NOINTERFACEMODE", IsNoInterfaceMode ?
L"true" : L"false");
gsCashaLotApi->SetParameter(L"NOAUTOOPENSHIFT", L"true");
//Ознака використанні захищеного носія(має бути після PathToCashaLotDir
та DeviceIDFnRRO та NOINTERFACEMODE, якщо необхідно пришвидшити
завантаження, та немає необхідності використовувати токени. Інакше токени
будуть ініціалізуватися)
gsCashaLotApi->SetParameter(L"USETOKEN", useToken ? L"true" :
L"false");

SetPrinterParametr();

gsCashaLotApi->SetParameter(L"PathToCertificate", pathToCert);
gsCashaLotApi->SetParameter(L"PwdToCertificate", pwdToCert);

}

return true;
}

bool CSampleUseCashaLotCOMAPICPPDlg::InitCashaLotApiComInterfaceEx()
{
    UpdateData();

    if (useToken != m_useToken && gsCashaLotApi != NULL)
        gsCashaLotApi->SetParameter(L"USETOKEN", m_useToken ? L"true" :
L"false");

    useToken = m_useToken;

    if ( pathToCashaLot == NULL )
    {
        pathToCashaLot = ::SysAllocString(m_PathToCashaLot);
    }
}

```

```

}

if (fiscalNumRRO == NULL )
{
    fiscalNumRRO = ::SysAllocString(m_FiscalNumberRRO);
}

if (pathToCert != NULL && pathToCert != m_PathCert)
{
    ::SysFreeString(pathToCert);
    pathToCert = ::SysAllocString(m_PathCert);
    if (gsCashaLotApi != NULL)
        gsCashaLotApi->SetParameter(L"PathToCertificate", pathToCert);
}
else
if (pathToCert == NULL )
{
    pathToCert = ::SysAllocString(m_PathCert);
}

if (pwdToCert != NULL && pwdToCert != m_pwdCert)
{
    ::SysFreeString(pwdToCert);
    pwdToCert = ::SysAllocString(m_pwdCert);
    if (gsCashaLotApi != NULL)
        gsCashaLotApi->SetParameter(L"PwdToCertificate", pwdToCert);
}
else
if (pwdToCert == NULL)
{
    pwdToCert = ::SysAllocString(m_pwdCert);
}
int oldSetupPrinterMode = setupPrinterMode;
setupPrinterMode = (m_UseDefCshalotPrinter)? 0 :
m_PrintUseDefWindowsPrinter!=-1? 1 : 2 ;

if (printerName != NULL && (printerName != m_PrinterName ||
oldSetupPrinterMode != setupPrinterMode) )
{
    ::SysFreeString(printerName);
    printerName = ::SysAllocString(m_PrinterName);
    SetPrinterParametr();
}
else
if(printerName == NULL)
    printerName = ::SysAllocString(m_PrinterName);

if(IsNoInterfaceMode != m_IsNoInterfaceMode && gsCashaLotApi != NULL )
    gsCashaLotApi->SetParameter(L"NOINTERFACEMODE", m_IsNoInterfaceMode ?
L"true" : L"false");

IsNoInterfaceMode = m_IsNoInterfaceMode;

return InitCashaLotApiComInterface();
}

```

*//Очистка всіх ресурсів по роботі з COM об'єктом CashaLot*

```
bool TermCashaLotApiComInterface()
{
    try
    {
        if (gsCashaLotApi != NULL)
        {
            gsCashaLotApi->Release();
            gsCashaLotApi = NULL;

            if (pathToCashaLot != NULL)
            {
                ::SysFreeString(pathToCashaLot);
                pathToCashaLot = NULL;
            }
            if (fiscalNumRRO != NULL)
            {
                ::SysFreeString(fiscalNumRRO);
                fiscalNumRRO = NULL;
            }
            if (pathToCert != NULL)
            {
                ::SysFreeString(pathToCert);
                pathToCert = NULL;
            }
            if (pwdToCert != NULL)
            {
                ::SysFreeString(pwdToCert);
                pwdToCert = NULL;
            }
            return true;
        }
    }
    catch (const char* msg)
    {
        MessageBox(0, BSTR(msg), L"Exception", MB_OK);
    }
    catch (...)
    {
        MessageBox(0, L"", L"Exception", MB_OK);
    }

    return false;
}
```

*//Приклад відкриття зміни*

```
void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtopenshift()
{
    if(!InitCashaLotApiComInterfaceEx())
        return;

    gsCashaLotApi->SetParameter(L"NOINTERFACEMODE", IsNoInterfaceMode ? L"true"
: L"false");

    if (pathToCert != NULL)
    {
        ::SysFreeString(pathToCert);
        pathToCert = NULL;
    }
}
```

```

    }
    if (pwdToCert != NULL)
    {
        ::SysFreeString(pwdToCert);
        pwdToCert = NULL;
    }

    pathToCert = ::SysAllocString(m_PathCert);
    pwdToCert = ::SysAllocString(m_pwdCert);

    gsCashaLotApi->SetParameter(L"PathToCertificate", pathToCert);
    gsCashaLotApi->SetParameter(L"PwdToCertificate", pwdToCert);

    _variant_t ret = gsCashaLotApi->OpenShift(fiscalNumRRO);

    if (V_VT(&ret) == VT_DISPATCH)
    {
        retVal = (ICashalotApiRetVal*)ret.pdispVal;
        if (retVal != NULL && retVal->Return != FALSE)
        {
            m_edPathToCashaLot.EnableWindow(false);
            m_btnPath.EnableWindow(false);
            m_edFiscalNumberRRO.EnableWindow(false);

            strRet.str("");
            strRet << "Номер відкритої зміни: " << retVal->ShiftID;
            MessageBox(CString(strRet.str().c_str()));
        }
    }
}

//Приклад закриття зміни
void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtcloseshift()
{
    if (!InitCashaLotApiComInterfaceEx())
        return;

    _variant_t ret = gsCashaLotApi->CloseShift(fiscalNumRRO);

    if (V_VT(&ret) == VT_DISPATCH)
    {
        retVal = (ICashalotApiRetVal*)ret.pdispVal;
        if (retVal != NULL && retVal->Return != FALSE)
        {
            strRet.str("");
            strRet << "Номер закритої зміни: " << retVal->ShiftID<<"\n"
                << "Результат у вигляді JSON\n" << retVal->JsonVal;
            MessageBox(CString(strRet.str().c_str()));
        }
    }
}

//Приклад фіскалізації чека на дві позиції
void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtfiscalizereceipt()
{
    if (!InitCashaLotApiComInterfaceEx())
        return;

```

```

gsLstCheckItems.Clear();

ReceiptItemInfo* inf = new ReceiptItemInfo();
inf->VendorCode = "АртикулТовару1";
inf->Name = "НазваТовару1";
inf->Barcode = "123456789";
inf->UnitType = "шт";
inf->Quantity = 1;
inf->GoodsType = 1; /*товар*/
inf->Price = 183;
inf->Amount = 181.17;
inf->DiscountSum = 1.83;
inf->VATRate = 20;
inf->SumVAT = 30.2;
inf->IsExcise = false;
gsLstCheckItems.InsertItem(inf);

inf = new ReceiptItemInfo();
inf->VendorCode = "АртикулПослуги1";
inf->Name = "НазваПослуги1";
inf->Barcode = "923456780";
inf->UnitType = "шт";
inf->Quantity = 1;
inf->GoodsType = 2; /*послуга*/
inf->Price = 183;
inf->Amount = 181.17;
inf->DiscountSum = 1.83;
inf->VATRate = 20;
inf->SumVAT = 30.2;
inf->IsExcise = false;
gsLstCheckItems.InsertItem( inf );
gsLstCheckItems.SetComment("Тестовий рядок1\r\nТестовий рядок2\r\nТестовий
рядок3");

_bstr_t jsonChkData = gsLstCheckItems.ToBSTRJSON();

PayInfoRsp payInfo;
payInfo.SumCash = 366;
payInfo.SumPayCheck = 362.34;
payInfo.SumPayByCredit = 100;
payInfo.SumPayByCertificate = 50;
_bstr_t jsonPayData = payInfo.ToBSTRJSON();

_variant_t ret = gsCashaLotApi->FiscalizeCheck(fiscalNumRRO, jsonChkData,
jsonPayData);

if (V_VT(&ret) == VT_DISPATCH)
{
    retVal = (ICashalotApiRetVal*)ret.pdispVal;
    if (retVal != NULL && retVal->Return != FALSE)
    {
        m_edFiscNumRcp.SetWindowTextW(retVal->ReceiptFiscalNum);

        strRet.str("");
        strRet << "Фіскальний номер створеного чека: " <<
retVal->ReceiptFiscalNum;
        strRet << "\nXML створеного чека: \n" << retVal->XML;
        MessageBox(CString(strRet.str().c_str()));
    }
}

```

```

    }
}

gsLstCheckItems.Clear();
}

//Приклад фіскалізації чека повернення на дві позиції, з вказанням чека по якому
здійснюється повернення
void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtreturnreceipt()
{
    CString strfiscNumRcpToRet;
    m_edFiscNumRcp.GetWindowText(strfiscNumRcpToRet);

    if (strfiscNumRcpToRet.GetLength() == 0)
    {
        MessageBox(L"Не заданий фіскальний номер чека по якому здійснюється
повернення!");
        return;
    }

    if (!InitCashaLotApiComInterfaceEx())
        return;

    gsLstCheckItems.Clear();
    ReceiptItemInfo* inf = new ReceiptItemInfo();
    inf->VendorCode = "АртикулТовару1";
    inf->Name = "НазваТовару1";
    inf->Barcode = "123456789";
    inf->UnitType = "шт";
    inf->Quantity = 1;
    inf->GoodsType = 1; /*товар*/
    inf->Price = 183;
    inf->Amount = 181.17;
    inf->DiscountSum = 1.83;
    inf->VATRate = 20;
    inf->SumVAT = 30.2;
    inf->IsExcise = false;
    gsLstCheckItems.InsertItem(inf);

    inf = new ReceiptItemInfo();
    inf->VendorCode = "АртикулПослуги1";
    inf->Name = "НазваПослуги1";
    inf->Barcode = "923456780";
    inf->UnitType = "шт";
    inf->Quantity = 1;
    inf->GoodsType = 2; /*послуга*/
    inf->Price = 183;
    inf->Amount = 181.17;
    inf->DiscountSum = 1.83;
    inf->VATRate = 20;
    inf->SumVAT = 30.2;
    inf->IsExcise = false;
    gsLstCheckItems.InsertItem(inf);
    _bstr_t jsonChkData = gsLstCheckItems.ToBSTRJSON();

    PayInfoRsp payInfo;
    payInfo.SumCash = 366;
    payInfo.SumPayCheck = 362.34;
}

```

```

_bstr_t jsonPayData = payInfo.ToBSTRJSON();

_bstr_t retFiscNum = ::SysAllocString(strfiscNumRcpToRet);

_variant_t ret = gsCashaLotApi->FiscalizeReturnCheck(fiscalNumRRO,
jsonChkData, jsonPayData, retFiscNum );

if ( V_VT(&ret) == VT_DISPATCH )
{
    retVal = (ICashalotApiRetVal*)ret.pdispVal;
    if (retVal != NULL && retVal->Return != FALSE)
    {
        strRet.str("");
        strRet << "Фіскальний номер створеного видаткового чека: " <<
retVal->ReceiptFiscalNum;
        //strRet << "\nXML створеного видаткового чека: \n" << retVal->XML;
        strRet << "\nТип чека: \n" << retVal->Type;
        MessageBox(CString(strRet.str().c_str()));
    }
}

::SysFreeString(retFiscNum);
}

//Приклад проведення чека ПЕРЕДОПЛАТИ
private void buttonPrepayment_Click( object sender, EventArgs e )
{
    LstCheckInfoRsp lstGoodsInCheck = new LstCheckInfoRsp();
    lstGoodsInCheck.ReceiptLst = new List<CheckInfo>();

    //Заповнюємо дві позиції товарів в чек
    CheckInfo chInfo = new CheckInfo();// структура з параметрами
першої позиції товару
    chInfo.Name = "ТоварПредоплатыПослеплаты1";
    chInfo.UnitType = "шт";
    chInfo.Quantity = "2";
    chInfo.Price = "1000";
    chInfo.Amount = "2000";
    //додати рядок чека з товаром №1
    lstGoodsInCheck.ReceiptLst.Add( chInfo );

    CheckInfo chInfo2 = new CheckInfo( chInfo );//Структура з
параметрами другої позиції товару
    chInfo2.Name = "ТоварПредоплатыПослеплаты2";
    chInfo2.Quantity = "1";
    chInfo2.Price = "1000";
    chInfo2.Amount = "1000";
    //додати рядок чека з товаром №2
    lstGoodsInCheck.ReceiptLst.Add( chInfo2 );

    //присвоїти номер договору/замовлення передоплати(друкується в
чеці)
    lstGoodsInCheck.SaleOrderNumber = "555444333";

    //Заповнюємо інформацію про оплату
    PayInfoRsp payInfo = new PayInfoRsp();

```

```

payInfo.SumCash = "1500"; //Сума готівки наданої покупцем( може
бути більшою за суму до сплати, з неї вираховується решта/здача)

//встановлюємо порядок оплати "ПЕРЕДОПЛАТА"
payInfo.PaymentOrderType = 1; //Тип виду сплати (0-звичайна | 1 -
передоплата | 2 - післяплата | 3 - інтернет продаж )
//вказуємо суму передоплати
payInfo.SumPreparePayed = "1500";

//вказуємо загальну суму по чеку
payInfo.SumPayCheck = "1500"; //Сума оплати загальна з урахуванням
заокруглення

//Конвертація структур до формату джейсон строки
string strJSONData = JsonConvert.SerializeObject( lstGoodsInCheck
);

string payDataJSON = JsonConvert.SerializeObject( payInfo );

//Параметри функції
Object[] parameters =
{
    edFiscalNumberRRO.Text.Trim(), // фіскальний номер РРО
    strJSONData, // структура інформації зі строками чека
    payDataJSON // структура інформації про оплату по чеку
};

var ret = CallCashaLotMethod( "FiscalizeCheck", parameters );
if ( ret?.Ret != true )
{
    MessageBox.Show( String.IsNullOrEmpty( ret?.ErrorString ) ? "Не
вдалося фіскалізувати чек" : ret?.ErrorString );
}
else if ( ret?.Values != null )
{
    //отримання параметра в результаті виконання функції
    var vals = ret.Values.ToArray();
    //перебираємо параметри
    for ( int i = 0; i < vals.Length; i++ )
    {
        // отримуємо значення по ключу
        if ( vals[i].Key.ToLower() == "receiptlocalnumber" )
        {
            edLocalCheckNum.Text = vals[i].Value;
        }
        else
        if ( vals[i].Key.ToLower() == "receiptfiscalnumber" )
        {
            //Отримуємо фіскальний номер створеного чека
            ПЕРЕДОПЛАТИ, та зберігаємо його для можливості здійснити "післяплату"
            edFNRcpPrepayment.Text = vals[i].Value;
        }
    }
    MessageBox.Show( $"Чек було успішно фіскалізовано.\n" +
        $"Чеку присвоєно локальний номер:
{edLocalCheckNum.Text},\n" +
        $"та фіскальний номер:
{edFiscalCheckNum.Text} "
    );
}

```

```

    }
}

//Приклад проведення чека "ПІСЛЯПЛАТИ"
private void buttonAfterpayment_Click( object sender, EventArgs e )
{
    LstCheckInfoRsp lstGoodsInCheck = new LstCheckInfoRsp();
    lstGoodsInCheck.ReceiptLst = new List<CheckInfo>();

    //присвоїти номер договору/замовлення передоплати(друкується в чеці)
    lstGoodsInCheck.SaleOrderNumber = "555444333";

    PayInfoRsp payInfo = new PayInfoRsp();
    payInfo.SumCash = "1600"; //Сума готівки наданої покупцем( може бути більшою за суму до сплати, з неї вираховується решта/здача)

    //встановлюємо порядок оплати "ПІСЛЯПЛАТА"
    payInfo.PaymentOrderType = 2; //Тип виду сплати (0-звичайна | 1 - передоплата | 2 - післяплата | 3 - інтернет продаж )

    //Вказуємо фіскальний номер чека "ПЕРЕДОПЛАТИ", на основі якого буде післяплата(поля чека будуть автоматично зачитані з нього, передані користувачем будуть проігноровані)
    payInfo.ParentReceiptFiscalNumber = edFNRcpPrepayment.Text;
    //вказуємо суму передоплати
    payInfo.SumPreparePaid = "1500";
    //вказуємо загальну суму по чеку
    payInfo.SumPayCheck = "1500"; //Сума оплати загальна з урахуванням заокруглення

    //Конвертація структур до формату джейсон строки
    string strJSONData = JsonConvert.SerializeObject( lstGoodsInCheck );

    string payDataJSON = JsonConvert.SerializeObject( payInfo );

    //Параметри функції
    Object[] parameters =
    {
        edFiscalNumberRRO.Text.Trim(), // фіскальний номер РРО
        strJSONData, // структура інформації зі строками чека
        payDataJSON // структура інформації про оплату по чеку
    };

    var ret = CallCashaLotMethod( "FiscalizeCheck", parameters );
    if ( ret?.Ret != true )
    {
        MessageBox.Show( String.IsNullOrEmpty( ret?.ErrorString ) ? "Не вдалося фіскалізувати чек" : ret?.ErrorString );
    }
    else if ( ret?.Values != null )
    {
        //отримання параметра в результаті виконання функції
        var vals = ret.Values.ToArray();
        //перебираємо параметри
        for ( int i = 0; i < vals.Length; i++ )
        {
            // отримуємо значення по ключу

```

```

        if ( vals[i].Key.ToLower() == "receiptlocalnumber" )
        {
            edLocalCheckNum.Text = vals[i].Value;
        }
        else
        if ( vals[i].Key.ToLower() == "receiptfiscalnumber" )
        {
            edFiscalCheckNum.Text = vals[i].Value;
        }
    }
    MessageBox.Show( $"Чек було успішно фіскалізовано.\n" +
        $"Чеку присвоєно локальний номер:
{edLocalCheckNum.Text},\n" +
        $"та фіскальний номер:
{edFiscalCheckNum.Text} "
        );
    }
}

//Приклад проведення чека ПЕРЕДОПЛАТИ
private void buttonPrepayment_Click( object sender, EventArgs e )
{
    LstCheckInfoRsp lstGoodsInCheck = new LstCheckInfoRsp();
    lstGoodsInCheck.ReceiptLst = new List<CheckInfo>();

    //Заповнюємо дві позиції товарів в чек
    CheckInfo chInfo = new CheckInfo();// структура з параметрами
першої позиції товару
    chInfo.Name = "ТоварПредоплатыПослеплаты1";
    chInfo.UnitType = "шт";
    chInfo.Quantity = "2";
    chInfo.Price = "1000";
    chInfo.Amount = "2000";
    //додати рядок чека з товаром №1
    lstGoodsInCheck.ReceiptLst.Add( chInfo );

    CheckInfo chInfo2 = new CheckInfo( chInfo );//Структура з
параметрами другої позиції товару
    chInfo2.Name = "ТоварПредоплатыПослеплаты2";
    chInfo2.Quantity = "1";
    chInfo2.Price = "1000";
    chInfo2.Amount = "1000";
    //додати рядок чека з товаром №2
    lstGoodsInCheck.ReceiptLst.Add( chInfo2 );

    //присвоїти номер договору/замовлення передоплати(друкується в
чеці)
    lstGoodsInCheck.SaleOrderNumber = "555444333";

    //Заповнюємо інформацію про оплату
    PayInfoRsp payInfo = new PayInfoRsp();
    payInfo.SumCash = "1500"; //Сума готівки наданої покупцем( може
бути більшою за сумму до сплати, з неї вираховується решта/здача)

    //встановлюємо порядок оплати "ПЕРЕДОПЛАТА"

```

```

payInfo.PaymentOrderType = 1; //Тип виду оплати (0-звичайна | 1 -
передоплата | 2 - післяплата | 3 - інтернет продаж )
//вказуємо суму передоплати
payInfo.SumPreparePayed = "1500";

//вказуємо загальну суму по чеку
payInfo.SumPayCheck = "1500"; //Сума оплати загальна з урахуванням
заокруглення

//Конвертація структур до формату джейсон строки
string strJSONData = JsonConvert.SerializeObject( lstGoodsInCheck
);

string payDataJSON = JsonConvert.SerializeObject( payInfo );

//Параметри функції
Object[] parameters =
{
    edFiscalNumberRRO.Text.Trim(), // фіскальний номер РРО
    strJSONData, // структура інформації зі строками чека
    payDataJSON // структура інформації про оплату по чеку
};

var ret = CallCashaLotMethod( "FiscalizeCheck", parameters );
if ( ret?.Ret != true )
{
    MessageBox.Show( String.IsNullOrEmpty( ret?.ErrorString ) ? "Не
вдалося фіскалізувати чек" : ret?.ErrorString );
}
else if ( ret?.Values != null )
{
    //отримання параметра в результаті виконання функції
    var vals = ret.Values.ToArray();
    //перебираємо параметри
    for ( int i = 0; i < vals.Length; i++ )
    {
        // отримуємо значення по ключу
        if ( vals[i].Key.ToLower() == "receiptlocalnumber" )
        {
            edLocalCheckNum.Text = vals[i].Value;
        }
        else
        if ( vals[i].Key.ToLower() == "receiptfiscalnumber" )
        {
            //Отримуємо фіскальний номер створеного чека
            ПЕРЕДОПЛАТИ, та зберігаємо його для можливості здійснити "післяплату"
            edFNRcpPrepayment.Text = vals[i].Value;
        }
    }
    MessageBox.Show( $"Чек було успішно фіскалізовано.\n" +
        $"Чеку присвоєно локальний номер:
{edLocalCheckNum.Text},\n" +
        $"та фіскальний номер:
{edFiscalCheckNum.Text} "
        );
}
}

//Приклад проведення чека "ПІСЛЯПЛАТИ"

```

```

private void buttonAfterpayment_Click( object sender, EventArgs e )
{
    LstCheckInfoRsp lstGoodsInCheck = new LstCheckInfoRsp();
    lstGoodsInCheck.ReceiptLst = new List<CheckInfo>();

    //присвоїти номер договору/замовлення передоплати(друкується в чеці)
    lstGoodsInCheck.SaleOrderNumber = "555444333";

    PayInfoRsp payInfo = new PayInfoRsp();
    payInfo.SumCash = "1600"; //Сума готівки наданої покупцем( може бути більшою за сумму до сплати, з неї вираховується решта/здача)

    //встановлюємо порядок оплати "ПІСЛЯПЛАТА"
    payInfo.PaymentOrderType = 2; //Тип виду оплати (0-звичайна | 1 - передоплата | 2 - післяплата | 3 - інтернет продаж )

    //Вказуємо фіскальний номер чека "ПЕРЕДОПЛАТИ", на основі якого буде післяплата(поля чека будуть автоматично зачитані з нього, передані користувачем будуть проігноровані)
    payInfo.ParentReceiptFiscalNumber = edFNRcpPrepayment.Text;
    //вказуємо сумму передоплати
    payInfo.SumPreparePaid = "1500";
    //вказуємо загальну суму по чеку
    payInfo.SumPayCheck = "1500"; //Сума оплати загальна з урахуванням заокруглення

    //Конвертація структур до формату джейсон строки
    string strJSONData = JsonConvert.SerializeObject( lstGoodsInCheck );
    string payDataJSON = JsonConvert.SerializeObject( payInfo );

    //Параметри функції
    Object[] parameters =
    {
        edFiscalNumberRRO.Text.Trim(), // фіскальний номер РРО
        strJSONData, // структура інформації зі строками чека
        payDataJSON // структура інформації про оплату по чеку
    };

    var ret = CallCashaLotMethod( "FiscalizeCheck", parameters );
    if ( ret?.Ret != true )
    {
        MessageBox.Show( String.IsNullOrEmpty( ret?.ErrorString ) ? "Не вдалося фіскалізувати чек" : ret?.ErrorString );
    }
    else if ( ret?.Values != null )
    {
        //отримання параметра в результаті виконання функції
        var vals = ret.Values.ToArray();
        //перебираємо параметри
        for ( int i = 0; i < vals.Length; i++ )
        {
            // отримуємо значення по ключу
            if ( vals[i].Key.ToLower() == "receiptlocalnumber" )
            {
                edLocalCheckNum.Text = vals[i].Value;
            }
        }
    }
}

```

```

        else
        {
            if ( vals[i].Key.ToLower() == "receiptfiscalnumber" )
            {
                edFiscalCheckNum.Text = vals[i].Value;
            }
        }
        MessageBox.Show( $"Чек було успішно фіскалізовано.\n" +
            $"Чеку присвоєно локальний номер:
{edLocalCheckNum.Text},\n" +
            $"та фіскальний номер:
{edFiscalCheckNum.Text} "
            );
    }
}

//Приклад чека "Інтернет-продаж"
private void btnInternetPayment_Click(object sender, EventArgs e)
{
    LstCheckInfoRsp lstGoodsInCheck = new LstCheckInfoRsp();
    lstGoodsInCheck.ReceiptLst = new List<CheckInfo>();

    //Заповнюємо дві позиції товарів в чек
    CheckInfo chInfo = new CheckInfo(); // структура з параметрами
    першої позиції товару
    chInfo.Name = "ТоварІнтернетПродажу1";
    chInfo.UnitType = "шт";
    chInfo.Quantity = "2";
    chInfo.Price = "1000";
    chInfo.VATRate = "20";
    chInfo.Amount = "2000";

    //додати рядок чека з товаром №1
    lstGoodsInCheck.ReceiptLst.Add(chInfo);

    CheckInfo chInfo2 = new CheckInfo(chInfo); // структура з
    параметрами другої позиції товару
    chInfo2.Name = "ТоварІнтернетПродажу2";
    chInfo2.Quantity = "1";
    chInfo2.Price = "1000";
    chInfo2.Amount = "1000";

    //додати рядок чека з товаром №2
    lstGoodsInCheck.ReceiptLst.Add(chInfo2);

    //присвоїти номер договору/замовлення (друкується в чеці)
    lstGoodsInCheck.SaleOrderNumber = "555444333555";

    PayInfoRsp payInfo = new PayInfoRsp();
    //вказуємо порядок оплати "Інтернет-продаж"
    payInfo.PaymentOrderType = 3; //Тип виду сплати (0-звичайна | 1 -
    передоплата | 2 - післяплата | 3 - інтернет-продаж )

    payInfo.SumPayByCard = "1500"; //Сума оплати картою або на
    рахунок
    //сума передоплати, якщо вона менша за суму всіх позицій чека то
    решта додається в рядок ПІСЛЯПЛАТА
    payInfo.SumPreparePayed = "1500";
}

```

```

payInfo.SumPayCheck = "3000";//Сума оплати по чеку загальна

//Конвертація структур до формату джейсон строки
string strJSONData = JsonConvert.SerializeObject(lstGoodsInCheck);
string payDataJSON = JsonConvert.SerializeObject(payInfo);

//Параметри функції
Object[] parameters =
{
    edFiscalNumberRRO.Text.Trim(), // фіскальний номер РРО
    strJSONData, // структура інформації зі строками чека
    payDataJSON // структура інформації про оплату по чеку
};

var ret = CallCashaLotMethod("FiscalizeCheck", parameters);
if (ret?.Ret != true)
{
    MessageBox.Show(String.IsNullOrEmpty(ret?.ErrorString) ? "Не
вдалося фіскалізувати чек" : ret?.ErrorString);
}
else if (ret?.Values != null)
{
    //отримання параметра в результаті виконання функції
    var vals = ret.Values.ToArray();
    //перебираємо параметри
    for (int i = 0; i < vals.Length; i++)
    {
        // отримуємо значення по ключу
        if (vals[i].Key.ToLower() == "receiptlocalnumber")
        {
            edLocalCheckNum.Text = vals[i].Value;
        }
        else
        if (vals[i].Key.ToLower() == "receiptfiscalnumber")
        {
            edFNRcpPrepayment.Text = vals[i].Value;
        }
    }
    MessageBox.Show($"Чек було успішно фіскалізовано.\n" +
        $"Чеку присвоєно локальний номер:
{edLocalCheckNum.Text},\n" +
        $"та фіскальний номер:
{edFiscalCheckNum.Text} "
        );
}
}

//Приклад чека службового внесення
void CSampleUseCashaLotCOMAPICPPDllg::OnBnClickedBtserviceinput()
{
    if (!InitCashaLotApiComInterfaceEx())
        return;
    BSTR strSm = ::SysAllocString(L"100");

    VARIANT vSm;
    ::VariantInit(&vSm);
    V_VT(&vSm) = VT_BSTR;

```

```

V_BSTR(&vSm) = strSm;

_variant_t ret = gsCashaLotApi->ServiceInput(fiscalNumRRO, vSm);

if (V_VT(&ret) == VT_BOOL && V_BOOL(&ret) != FALSE )
    MessageBox(L"Службове внесення виконано успішно");
else
    MessageBox(L"Не вдалося виконати службове внесення");

::SysFreeString(strSm);
}

//Приклад чека службової видачі
void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtserviceoutput()
{
    if (!InitCashaLotApiComInterfaceEx())
        return;
    BSTR strSm = ::SysAllocString(L"100");

    VARIANT vSm;
    ::VariantInit(&vSm);
    V_VT(&vSm) = VT_BSTR;
    V_BSTR(&vSm) = strSm;
    _variant_t ret = gsCashaLotApi->ServiceOutput(fiscalNumRRO, vSm);

    if (V_VT(&ret) == VT_BOOL && V_BOOL(&ret) != FALSE)
        MessageBox(L"Службова видача виконана успішно");
    else
        MessageBox(L"Не вдалося виконати службову видачу");

    ::SysFreeString(strSm);
}

//Приклад чека друку X-Звіту(без гасіння)
void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedPrintXRep()
{
    if (!InitCashaLotApiComInterfaceEx())
        return;

    _variant_t ret = gsCashaLotApi->PrintXReport(fiscalNumRRO);

    if (V_VT(&ret) == VT_BOOL && V_BOOL(&ret) != FALSE)
        MessageBox(L"X-звіт сформований успішно");
    else
        MessageBox(L"Не вдалося сформувати X-звіт");
}

void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtgetpathtocert()
{
    CString ret = BrowseFolder();
    if (ret.GetLength())
    {
        m_PathCert = ret;
        m_edPathCert.SetWindowText(m_PathCert);
    }
}

```

```

void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtprintxrepex()
{
    if (!InitCashaLotApiComInterfaceEx())
        return;

    _variant_t ret = gsCashaLotApi->GetXReport(fiscalNumRRO, VARIANT_FALSE );

    if (V_VT(&ret) == VT_DISPATCH)
    {
        retVal = (ICashalotApiRetVal*)ret.pdispVal;
        if (retVal != NULL && retVal->Return != FALSE)
        {
            strRet.str("");
            strRet << "Результат у вигляді JSON:\n" << retVal->JsonVal;
            MessageBox(CString(strRet.str().c_str()));
        }
    }
}

void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedBtprintlastzrep()
{
    if (!InitCashaLotApiComInterfaceEx())
        return;

    _variant_t ret = gsCashaLotApi->GetLastZReport(fiscalNumRRO, VARIANT_TRUE);

    if (V_VT(&ret) == VT_DISPATCH)
    {
        retVal = (ICashalotApiRetVal*)ret.pdispVal;
        if (retVal != NULL && retVal->Return != FALSE)
        {
            strRet.str("");
            strRet << "Результат у вигляді JSON:\n" << retVal->JsonVal;
            MessageBox(CString(strRet.str().c_str()));
        }
    }
}

void CSampleUseCashaLotCOMAPICPPDlg::OnBnClickedGetRcpByPeriod()
{
    if (!InitCashaLotApiComInterfaceEx())
        return;

    strRet.str("");
    BSTR dtBeg = ::SysAllocString(m_DateBeg.Format(CString("%d.%m.%Y")));
    BSTR dtEnd = ::SysAllocString(m_DateEnd.Format(CString("%d.%m.%Y")));
    //зачитуємо перший чек за період
    _variant_t ret = gsCashaLotApi->GetFirstReceiptByPeriod(fiscalNumRRO,
dtBeg, dtEnd);

    if (V_VT(&ret) == VT_DISPATCH)
    {
        retVal = (ICashalotApiRetVal*)ret.pdispVal;
        if (retVal != NULL && retVal->Return != FALSE)
        {
            strRet << "Фіскальні номери чеків за період:\n";
            strRet << retVal->ReceiptFiscalNum;
        }
    }
}

```

```

        bool retN = false;
        do
        {
            retN = false;
            //зачитуємо наступний чек за період
            ret = gsCashaLotApi->GetNextReceipt(fiscalNumRRO);
            if (V_VT(&ret) == VT_DISPATCH)
            {
                retVal = (ICashalotApiRetVal*)ret.pdispVal;
                if (retVal != NULL && retVal->Return != FALSE)
                {
                    retN = true;
                    strRet << ",";
                    strRet << retVal->ReceiptFiscalNum;
                }
            }
            //читаємо поки будуть наступні
        } while (retN);
    }

    ret = gsCashaLotApi->GetFirstZReportByPeriod(fiscalNumRRO, dtBeg, dtEnd);

    if (V_VT(&ret) == VT_DISPATCH)
    {
        retVal = (ICashalotApiRetVal*)ret.pdispVal;
        if (retVal != NULL && retVal->Return != FALSE)
        {
            strRet << "\nФіскальні номери Z-звітів за період:\n";
            strRet << retVal->ReceiptFiscalNum;

            bool retN = false;
            do
            {
                retN = false;
                //зачитуємо наступний чек за період
                ret = gsCashaLotApi->GetNextZReport(fiscalNumRRO);
                if (V_VT(&ret) == VT_DISPATCH)
                {
                    retVal = (ICashalotApiRetVal*)ret.pdispVal;
                    if (retVal != NULL && retVal->Return != FALSE)
                    {
                        retN = true;
                        strRet << ",";
                        strRet << retVal->ReceiptFiscalNum;
                    }
                }
                //читаємо поки будуть наступні
            } while (retN);
        }
    }

    MessageBox(CString(strRet.str().c_str()));

    ::SysFreeString(dtBeg);
    ::SysFreeString(dtEnd);
}

```