



Специфікація використання

драйвера Cashälot для BAS

v. 6.1

Версія драйвера 1.20

Зміст

Зміст

Поняття і визначення

Загальна інформація

Апаратні та програмні вимоги

Опис інтерфейсу драйвера Cashälot для BAS

(компоненти CashaLotApiDrv32.dll/CashaLotApiDrv64.dll)

Порядок виклику функцій при виконанні касових операцій

1. Отримання номера версії драйвера
2. Отримання налаштувань драйвера
3. Отримання останньої помилки
4. Встановлення налаштувань драйвера
5. Підключення фіскального реєстратора ПРРО Cashälot
6. Відключення фіскального реєстратора ПРРО Cashälot
7. Перевірка ПРРО Cashälot
8. Схема роботи з декількома ПРРО
9. Відкриття зміни
10. Реєстрація чеків
11. Відміна відкритого чека
12. Вивести візуальне відображення чека на екран
13. Отримання фіскального номера чека за його локальним номером
14. Отримання локального номера чека за його фіскальним номером
15. Отримання xml структури фіскального чека
16. Друк періодичних звітів
17. Отримання інформації про поточний стан ПРРО
18. Друк довільної текстової інформації
19. Службове внесення/видача
20. Формування X-звіту
21. Формування Z-звіту
22. Закриття зміни
23. Ручне переведення ПРРО в режим офлайн
24. Ручне повернення ПРРО в режим онлайн
25. Синхронізація залишків по товарам
26. Синхронізація товарів та залишків
27. Сторнування чека продажу/повернення
28. Відображення QR-кода зареєстрованого чека
29. Відправка чека покупцю за допомогою сервісів доставки
30. Отримання переліку налаштованих POS-терміналів в програмі
31. Встановлення пристрою POS-термінал за замовчуванням
32. Формування звіту про реалізовані товари
33. Отримання посилання для перегляду останнього зареєстрованого чека
34. Запит розширеної відповіді від Cashalot для останнього виконаного методу
35. Створення відкладеного чека продажу товарів/послуг
36. Фіскалізація відкладеного чеку продажу товарів/послуг

- 37. Отримати інформацію про відкладений чек за його ReceiptID
- 38. Отримати інформацію про відкладений чек за його номером ЕН (експрес-накладною)
- 39. Отримати інформацію про перший відкладений чек
- 40. Отримати інформацію наступний відкладений чек
- 41. Відображення візуальної форми чека за його ReceiptID
- 42. 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
- 43. Обробка JSON відповіді
- 44. Приклади використання засобів оплат у чеку (спрощений та розширений варіант)

Поняття і 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

Драйвер Cashälot використовується для інтеграції ПРРО з розрахунково-обліковими системами BAS. В цьому документі приведено специфікацію методів бібліотек Cashälot для підключення до BAS.

За допомогою інтеграції безпосередньо з розрахунково-облікової системи можна скористатися наступним функціоналом програмного забезпечення Cashälot:

- Відкриття та закриття касової зміни;
- Створення та реєстрація чеків продажу/повернення;
- Сторнування чека продажу/повернення;
- Створення та реєстрація чеків з типами оплати передоплата, післяплата, інтернет-продажа, оплата частинами;
- Створення видаткового чека на основі чека продажу та його реєстрація;
- 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.

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 2 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 Cashalot, фіскального сервера контролюючого органу та до ресурсу центру сертифікації ключів. Наприклад, якщо Ви використовуєте ключі від "АЦСК Україна", 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 Cashalot.

Опис інтерфейсу драйвера Cashälot для BAS

(компоненти CashaLotApiDrv32.dll/CashaLotApiDrv64.dll)

Драйвер Cashälot - це окремий механізм для інтеграції ПРРО Cashälot, що призначений для використання виключно в BAS. Для інтеграції використовуються бібліотеки CashaLotApiDrv32.dll та CashaLotApiDrv64.dll (далі - CashaLotApiDrv), які розміщені в кореновому каталозі ПРРО Cashälot.

В конфігураторі BAS для драйвера фіскального реєстратора вказуються методи, призначені для розробки драйверів обладнання, що підключається. Ці методи відправляють запити і отримують відповіді від драйвера CashaLotApiDrv.

CashaLotApiDrv є в поставці BAS, починаючи з версій:

- Управління виробничим підприємством (1.3.75.2)
- Управління торговим підприємством для України (1.2.64.1)
- Управління торгівлею для України (2.3.37.1)
- Роздріб для України (1.0.14.1)
- BAS Малий бізнес (Small company) PROF (1.6.15.1)
- BAS Управління торгівлею (Trade management) (3.2.23.2)
- BAS Роздрібна торгівля (Retail) (2.2.12.1)
- BAS ERP (Enterprise resource planning) (2.1.23.2)
- BAS Комплексне управління підприємством (Integrated enterprise management) (2.1.23.2)

Перед початком роботи з CashalotApiDrv потрібно:

- зареєструватися в особистому кабінеті <https://my.cashalot.org.ua/>;
- виконати всі пункти з розділу “Початок роботи” в особистому кабінеті;
- встановити ПРРО Cashälot, попередньо завантаживши дистрибутив з офіційного [сайту](#). Виконати запуск програми Cashälot (В момент першого запуску завантажуються дані з бек-офісу). Для паралельної роботи в декількох касах одночасно необхідно встановлювати ПРРО Cashälot для кожної каси окремо;
- зареєструвати бібліотеки в ОС за допомогою команди regsvr32

regsvr32 "<Повний шлях до ПРРО Cashalot>\CashalotApiDrv32.dll"

(для 32-бітної платформи)

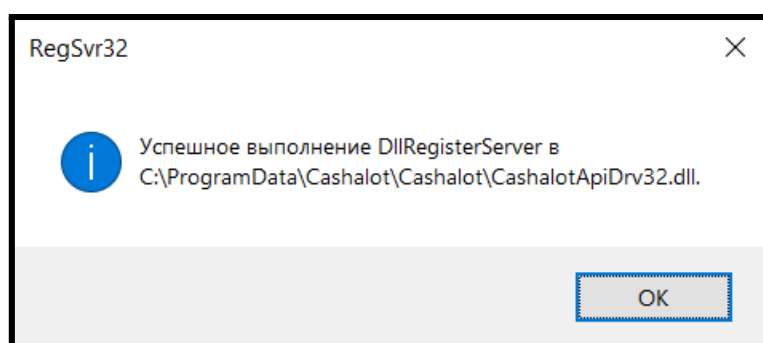
regsvr32 "<Повний шлях до ПРРО Cashalot>\CashalotApiDrv64.dll"

(для 64-бітної платформи)

Наприклад:

regsvr32 "C:\ProgramData\Cashalot\Cashalot\CashalotApiDrv32.dll"

Після успішної реєстрації бібліотеки з'явиться повідомлення зразку:



В CashalotApiDrv реалізована обробка наступних методів:

- ПолучитьНомерВерсии
- ПолучитьОшибку
- ПолучитьПараметры
- УстановитьПараметр
- Подключить
- Отключить
- ТестУстройства
- ОткрытьСмену
- ОткрытьЧек
- НапечататьФискальнуюСтроку
- НапечататьФискСтроку
- УстановитьРасширенныйПорядокОплатыЧека
- ЗаккрытьЧек
- ОтменитьЧек
- ПолучитьXMLЧека
- ОтобразитьЧек
- ПолучитьФискальныйНомерЧека
- ПолучитьЛокальныйНомерЧека
- НапечататьПериодическийОтчетПоДатам
- НапечататьЧекВнесенияВыемки
- НапечататьОтчетБезГашения
- НапечататьОтчетСГашением
- ЗаккрытьСмену
- УстановитьОфлайнРежим
- ПопыткаВийтиВОнлайнРежим
- СинхронизироватьОстаткиПоТоварам
- СинхронизироватьТоварыИОстатки
- ОтобразитьQRКодЧека
- СторнироватьЧек
- SendCheckToCustomer
- GetPOSTerminalList
- SetDefaultPOSTerminal
- PrintReportGoods
- GetLastReceiptLink
- GetLastMethodResponse
- FiscalizePostponedReceipt
- GetPostponedReceiptByID
- GetPostponedReceiptByWaybillNumber
- GetFirstPostponedReceipt
- GetNextPostponedReceipt
- ShowReceiptByID

Ініціалізація компоненти відбувається шляхом створення об'єкту відповідно до синтаксису:

```
&НаКлиенте
Перем ДрайверCshalot Экспорт;
&НаКлиенте
Процедура ИнициализацияДрайвераCshalot()

    БиблиотекаДрайвера = "Путь к библиотеке драйвера
    CshaLotApiDrv32.dll/CshaLotApiDrv64.dll";

    Попытка
        ПодключитьВнешнююКомпоненту(БиблиотекаДрайвера, "CshaLotApiDrv");
        ДрайверCshalot = Новый("AddIn.CshaLotApiDrv.CshaLotApiDrv");
        Сообщить("Драйвер Cshalot инициализирован.");
    Исключение
        Сообщить("Не удалось инициализировать драйвер Cshalot." +
            Символы.ПС +
            ОписаниеОшибки());
    КонецПопытки;

КонецПроцедуры
```

При роботі одного користувача з декількома касами необхідно встановити налаштування для фіскального реєстратора таким чином, щоб драйвер мав можливість розрізняти різні екземпляри ПРРО Cshalot, тобто для кожної каси свій фіскальний номер ([DeviceIDFnRRO](#)) та встановлений окремий екземпляр програми Cshalot, до якого вказаний шлях в параметрах драйвера ([PathToCshalotDir](#)).

Ім'я	Тип	Опис
DeviceIDFnRRO	строка	Фіскальний номер ПРРО.
PathToCshalotDir	строка	Шлях до каталогу з встановленим ПРРО Cshalot.

Порядок виклику функцій при виконанні касових операцій

При роботі з касою в системі BAS рекомендується послідовність виклику функцій ПРРО Cashalot, зазначена в таблиці:

Касова операція	Послідовність функцій драйвера
Відкриття зміни	ОткрытьСмену
Внесення грошей	НапечататьЧекВнесенияВыемки
Фіскалізація чеків	ОткрытьЧек НапечататьФискальнуюСтроку або НапечататьФискСтроку УстановитьРасширенныйПорядокОплатыЧека ЗакрытьЧек або ОтменитьЧек
Повернення товару	ОткрытьЧек НапечататьФискальнуюСтроку або НапечататьФискСтроку УстановитьРасширенныйПорядокОплатыЧека ЗакрытьЧек або ОтменитьЧек
X-звіт	НапечататьОтчетБезГашения
Видача грошей	НапечататьЧекВнесенияВыемки
Z-звіт	НапечататьОтчетСГашением
Закриття зміни	ЗакрытьСмену

1. Отримання номера версії драйвера

Для отримання номера версії драйвера використовується метод:

СТРОКА `ПолучитьНомерВерсии()`

Приклад:

Функция `ПолучитьВерсиюДрайвераCashalot()`

```
ВерсияДрайвераCashalot = ДрайверCashalot.ПолучитьНомерВерсии();
```

```
Сообщить("Версия драйвера Cashalot: " + ВерсияДрайвераCashalot);
```

```
Возврат ВерсияДрайвераCashalot;
```

КонецФункции

2. Отримання налаштувань драйвера

Для отримання списку параметрів налаштувань драйвера використовують функцію:

БУЛЕВО `ПолучитьПараметры()`.

Параметри функції для отримання параметрів драйвера:

Ім'я	Тип	Опис
XMLСтрокаПараметров	строка [out]	Опис параметрів. Таблиця списку параметрів є XML-рядком.

Увага! Функція не повертає поточні значення параметрів.

Приклад:

Процедура `ПолучитьПараметрыCshalot()`

```
XMLСтрокаПараметров = "";  
ДрайверCshalot.ПолучитьПараметры(XMLСтрокаПараметров);  
  
ЧтениеXML = Новый ЧтениеXML();  
ЧтениеXML.УстановитьСтроку(XMLСтрокаПараметров);  
Пока ЧтениеXML.Прочитать() Цикл  
    Если ЧтениеXML.ТипУзла = ТипУзлаXML.НачалоЭлемента И ЧтениеXML.Имя =  
        "Parameter"  
        Тогда  
            ИмяПараметра = ЧтениеXML.ПолучитьАтрибут("Name");  
            ОписаниеПараметра = ЧтениеXML.ПолучитьАтрибут("Caption");  
            Сообщить("Параметр драйвера "" + ИмяПараметра + """: " +  
                ОписаниеПараметра + ".");  
        КонецЕсли;  
КонецЦикла;  
  
ЧтениеXML.Закрыть();  
  
КонецПроцедуры
```

3. Отримання останньої помилки

Для отримання опису помилки використовується функція `ПолучитьОшибку()`. У функції повинен бути вказаний параметр з іменем `ТекстОшибкиCshalot` та типом строка `[out]`.

Приклад:

Функция `ПолучитьОшибкуCshalot()`

```
ТекстОшибкиCshalot = "";  
ДрайверCshalot.ПолучитьОшибку(ТекстОшибкиCshalot);  
  
Возврат ТекстОшибкиCshalot;  
  
КонецФункции
```

4. Встановлення налаштувань драйвера

Функція призначена для встановлення значень параметрів налаштувань драйвера Cashälot для BAS.

БУЛЕВО **УстановитьПараметр**(ИмяПараметра, ЗначениеПараметра)

Кожен параметр повинен містити в собі:

Ім'я	Тип	Опис
ИмяПараметра	строка	Ім'я параметра.
ЗначениеПараметра	строка	Значення параметра.

УстановитьПараметр складається з обов'язкових параметрів (виділені червоним) та необов'язкових параметрів. Параметри повинні мати послідовність, як в таблиці нижче.

При роботі з CashalotApiDrv, необхідно зазначити такі параметри:

Ім'я параметра	Значення параметра
PathToCashalotDir	Шлях до каталогу з встановленим ПРРО Cashälot.
DeviceIDFnRRO	Фіскальний номер ПРРО.
NOAUTOUPDATE	Перевірка на наявність нової версії на сервері https://load.cashalot.org.ua/update/ та запит на оновлення програми Cashälot в момент початку роботи з API: "True" – заборонити оновлення; "False" – дозволити оновлення(за замовчуванням).
NOINTERFACEMODE	Ознака використання інтерфейсу РМК для авторизації та службових операцій видачі та внесення готівки в касу. "False" – при роботі з драйвером викликається діалогове вікно. "True" – авторизація відбувається шляхом передачі параметрів авторизації PathToCertificate та PwdToCertificate. Вікна службових операцій видачі та внесення готівки не викликаються (проте, результат операції службового внесення/видачі готівки відображається).
PathToCertificate	Шлях до каталогу з файловим ключем користувача. Важливо: в папці повинні міститися ключ і сертифікат тільки одного користувача. Якщо касою користуються декілька касирів, необхідно помістити їх сертифікати і ключі в окремі каталоги.
PwdToCertificate	Пароль від ключа.
NOAUTOOPENSIFT	Перевірка стану зміни (відкрита або закрита). Якщо зміна закрита, при виклику будь-якої функції зміна автоматично відкривається. "False" – автоматично відкривати зміну, якщо вона закрита. "True" – не відкривати автоматично зміну. В цьому випадку робота функцій буде перервана і результат виконання буде повернуто у вигляді помилки.

DEFAULTPRINTERNAME	Визначає принтер для друку при натисканні кнопки "Друк" в діалогових вікнах: 1) "" – пустий рядок, виводиться діалог вибору принтера; 2) "DEFAULTWINDOWS" – використовувати принтер Windows за замовчуванням; 3) "ім'я принтера чеків" – в параметр передається ім'я принтера чеків, який буде використовуватись для друку чеків в ПРРО Cashlot.
AUTOPRINTMODE	Параметр визначає автоматичну відправку на друк чека або службового звіту без відображення інформаційного вікна: "True" – відправляти на друк; "False" – не відправляти на друк. Якщо задано принтер за замовчуванням (DEFAULTPRINTERNAME), чек друкується одразу, в іншому випадку відображається вікно вибору принтера для друку.
NOPRINTMODE	Дозвіл на друк: «True» – заборонити друк; «False» – дозволити друк.
USEGOODSSTORAGEMODE	Параметр USEGOODSSTORAGEMODE визначає використання ведення складського обліку по товарам та автоматичну синхронізацію залишків по ним при використанні драйвера Cashalot для BAS за умови активованого налаштування в кабінеті користувача в розділі «Склад - Налаштування - Облік товарів у ПРРО». Параметр може приймати наступні значення: «0» - програма працює в стандартному режимі без ведення обліку товарів та синхронізації залишків по товарам з кабінетом користувача при використанні драйвера Cashalot для BAS . «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. «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. 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). *Кроки по практичному застосуванню параметра USEGOODSSTORAGEMODE

	1. Імпортувати номенклатуру, яка використовується в драйвер Cashalot для BAS, 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 функцією СинхронизироватьТоварыИОстатки . 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 драйвері Cashalot для BAS використовується одна й та сама номенклатура, що заведена в кабінеті користувача.
USEPRGPOSTERMINAL	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.). «True» - використовується POS-термінал, підключений в Cashalot для оплати під час передачі чека на фіскалізацію з обраною формою оплати Картка (СуммаОплатыКартой) . «False» - під час фіскалізації чека з формою оплати Картка підключений POS-термінал в Cashalot не використовується. В разі неуспішної транзакції на банківському терміналі операція реєстрації чека не відбувається.
RATELETTERCREATEIFNOTEXIST	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. “True” - відбувається автоматичне створення податкової ставки та літерного позначення в разі її відсутності в Cashalot “False” - використовуються наявні завантажені в Cashalot податкові ставки та їх літерні позначення.
USESELTERMINALINFOWARNING	Параметр призначений для можливості ручного підтвердження термінальної операції користувачем за умови, якщо операція з POS-терміналом пройшла неуспішно (обірвався зв'язок терміналу 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-терміналу.

	Параметр працює за умови використання параметра USEPRGPOSTERMINAL. Вікно підтвердження буде з'являтися навіть якщо використовується параметр NOINTERFACEMODE. “True” - після отримання помилки від POS-терміналу при проведенні операції з'явиться додаткове вікно з можливістю ручного підтвердження операції. “False” - в разі отримання помилки від POS-терміналу необхідно буде повторити спробу виконання операції.
AUTOCLOSETERMINALSHIF	Параметр призначений для закриття термінальної зміни під час закриття зміни ПРРО. “False” - закриття зміни на POS-терміналі за умови закриття зміни ПРРО не відбувається. “True” - 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)
ISNEEDAPIRETRECEIPTJ PEG	Параметр використовується для отримання чека продажу або повернення у форматі JPEG, закодованого в Base64. Результат повертається у структурі відповіді в полі JsonVal, де значення доступне в параметрі Base64ReceiptJpeg. Для отримання такої інформації слід скористатись функцією GetLastMethodResponse після виклику функції Закри́тьЧек
ISNEEDAPIRETRECEIPTP DF	Параметр використовується для отримання чека продажу або повернення у форматі PDF, закодованого в Base64. Результат повертається у структурі відповіді в полі JsonVal, де значення доступне в параметрі Base64ReceiptPdf. Для отримання такої інформації слід скористатись функцією GetLastMethodResponse після виклику функції Закри́тьЧек

Приклад:

Процедура УстановитьПараметрCashalot(ИмяПараметра, ЗначениеПараметра)

```

ДрайверCashalot = ИнициализацияДрайвераCashalot();
ИмяПараметра = "DeviceIDFnRRO";
ЗначениеПараметра = "40000000000";
ПараметрУстановлен = ДрайверCashalot.УстановитьПараметр(ИмяПараметра,
ЗначениеПараметра);

```

Если НЕ ПараметрУстановлен Тогда

Сообщить("Не удалось установить параметр Cashalot "" +

ИмяПараметра +

"" из-за ошибки: " +

ПолучитьОшибкуCashalot());

КонецЕсли;

КонецПроцедуры

5. Підключення фіскального реєстратора ПРРО Cashälot

Для підключення ПРРО Cashälot з поточними значенням параметрів, встановлених функцією **УстановитьПараметр**, використовується функція:

БУЛЕВО **Подключить**(ФНПодключеннойКассы)

Параметри функції підключення пристрою:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка [out]	Фіскальний номер ПРРО. Ідентифікатор пристрою.

При успішному підключенні, функція повертає ідентифікатор підключеного ПРРО.

Приклад:

Функция **ПодключитьCashalot()**

ФНПодключеннойКассы = "";

CashalotПодключен = ДрайверCashalot.Подключить(ФНПодключеннойКассы);

Если CashalotПодключен Тогда

Сообщить("Драйвер подключен к ПРРО Cashalot и кассе с ФН: " +
ФНПодключеннойКассы);

Иначе

Сообщить("Не удалось подключить драйвер к ПРРО Cashalot из-за
ошибки: " + ПолучитьОшибкуCashalot());

КонецЕсли;

Возврат CashalotПодключен;

КонецФункции

6. Відключення фіскального реєстратора ПРРО Cashälot

Для відключення ПРРО Cashälot використовується функція:

БУЛЕВО **Отключить**(ФНПодключеннойКассы)

Параметри функції відключення пристрою:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка [out]	Фіскальний номер ПРРО. Ідентифікатор пристрою.

Приклад:

Функция ОтключитьCashalot()

```
CashalotОтключен = ДрайверCashalot.Отключить(ФНПодключеннойКассы);
```

```
Сообщить?(CashalotОтключен,
```

```
    "Драйвер отключен от ПРРО Cashalot.",
```

```
    "Не удалось отключить драйвер от ПРРО Cashalot из-за ошибки:"
```

```
    + ПолучитьОшибкуCashalot());
```

```
Возврат CashalotОтключен;
```

КонецФункции

7. Перевірка ПРРО Cashälot

Сценарій, в якому функція використовується користувачем для перевірки одного чи декількох ПРРО. Перевірка виконується шляхом пробного підключення і опитування пристрою з поточними значеннями параметрів, встановлених для кожного екземпляру окремо. Використовується функція:

БУЛЕВО **ТестУстройства** (Описание, ПризнакДемоРежима)

Параметри:

Ім'я	Тип	Опис
Описание	строка [out]	Опис результату виконання тесту.
ПризнакДемоРежима	строка [out]	Опис обмежень демонстраційного режиму, за його наявності, і пустий результат, при його відсутності.

Увага! На даному етапі розробки результат виконання функції не містить значень вихідних параметрів.

Приклад:

Процедура ТестУстройстваCashalot()

```
ТестУспешен = ДрайверCashalot.ТестУстройства("", "");
```

```
Сообщить?(ТестУспешен,
```

```
    "Тестирование подключения к ПРРО Cashalot выполнено успешно.",
```

```
    "Тестирование подключения к ПРРО Cashalot неудачно."));
```

КонецПроцедуры

8. Схема роботи з декількома ПРРО

Для організації роботи з декількома касами з одного примірника 1С/BAF перш за все необхідно встановити окремі примірники Cashalot для кожної каси. Програмно взаємодія з декількома касами реалізується через один загальний об'єкт **CashaLotApiDrv** і шляхом "переключення" між касами зі встановленням параметрів поточної (з якою будуть виконуватись фіскальні операції).

Один загальний об'єкт **CashaLotApiDrv** створюється і зберігається як змінна форми або модуля, а не конкретної процедури або функції, як наведено у прикладі ініціалізації об'єкта драйвера Cashalot.

Переключення між касами виконується за наступним алгоритмом:

- Відключення від поточної каси функцією **Отключить** (якщо попередньо певна каса була підключена, в іншому випадку даний пункт пропускається);
- Встановлення параметрів іншої каси (з якою буде вестись взаємодія) функцією **УстановитьПараметр**;
- Підключення до іншої каси функцією **Подключить**;

Для перевірки ініціалізації драйвера та підключення каси створюються змінні типу Булево, значення яких змінюються при викликах відповідних функцій драйвера.

Зверніть увагу! У випадку роботи з декількома касами при встановленні параметрів першим обов'язково має бути встановлений параметр EquipmentType:

Ім'я параметра	Тип значення	Значення параметра
EquipmentType	строка	Довільне, але фіксоване текстове значення, наприклад "ФіскальнийРеєстраторCashalot"

Інші параметри каси встановлюються за звичайним порядком, наведеним в описі функції **УстановитьПараметр**.

Приклад:

```
Перем Драйвер;  
Перем ДрайверИнициализирован = Ложь;  
Перем ДрайверПодключен = Ложь;  
Перем EquipmentType = "ФіскальнийРеєстраторCashalot";  
  
Процедура ИнициализироватьДрайверCashalot()  
  
    ОтключитьДрайверCashalot();  
  
    Если НЕ ДрайверИнициализирован Тогда  
        ПодключитьБиблиотекуДрайвера();  
    КонецЕсли;  
  
    Если ДрайверПодключен Тогда  
        Возврат;  
    КонецЕсли;  
  
    УстановитьПараметрыДрайвераCashalot();
```

КонецПроцедуры

Процедура ПодключитьБиблиотекуДрайвера()

Попытка

БиблиотекаДрайвера = "Шлях до бібліотеки драйвера Cashälot";

Если НЕ ПодключитьВнешнююКомпоненту(БиблиотекаДрайвера, "CashaLotApiDrv")

Тогда

// Опрацювання помилки...

Возврат;

КонецЕсли;

Драйвер = Новый("AddIn.CashaLotApiDrv.CashaLotApiDrv");

Исключение

// Опрацювання помилки...

Возврат;

КонецПопытки;

ДрайверИнициализирован = Истина;

КонецПроцедуры

Процедура УстановитьПараметрыДрайвераCshalot()

Если НЕ ДрайверИнициализирован ИЛИ ДрайверПодключен Тогда

Возврат;

КонецЕсли;

ПараметрыДрайвера = Новый Структура;

ПараметрыДрайвера.Вставить("EquipmentType", EquipmentType);

ПараметрыДрайвера.Вставить("PathToCshalotDir", "...");

ПараметрыДрайвера.Вставить("DeviceIdFnRro", "...");

// Інші параметри...

Для Каждого Параметр из ПараметрыДрайвера Цикл

ИмяПараметра = Параметр.Ключ;

ЗначениеПараметра = Параметр.Значение;

ПараметрУстановлен = Драйвер.

УстановитьПараметр(ИмяПараметра,
ЗначениеПараметра);

Если НЕ ПараметрУстановлен Тогда

// Опрацювання помилки...

Возврат;

КонецЕсли;

КонецЦикла;

ПодключитьДрайверCshalot();

КонецПроцедуры

Процедура ПодключитьДрайверCshalot()

Если НЕ ДрайверИнициализирован ИЛИ ДрайверПодключен Тогда

```
Возврат;  
КонецЕсли;
```

```
ФНПодключеннойКассы = "";  
ДрайверПодключен = Драйвер.Подключить(ФНПодключеннойКассы);
```

```
Если НЕ ДрайверПодключен  
ИЛИ ФНПодключеннойКассы <> "ФН необхідної каси"  
Тогда  
    // Опрацювання помилки...  
КонецЕсли;
```

```
КонецПроцедуры
```

```
Процедура ОтключитьДрайверCashalot()
```

```
Если НЕ ДрайверИнициализирован ИЛИ НЕ ДрайверПодключен Тогда  
    Возврат;  
КонецЕсли;  
ФНОтключеннойКассы = "ФН каси, від якої відключаємось";  
ДрайверПодключен = НЕ Драйвер.Отключить(ФНОтключеннойКассы);
```

```
Если ДрайверПодключен Тогда  
    // Опрацювання помилки...  
    Возврат;  
КонецЕсли;  
КонецПроцедуры
```

Вище наведений перелік функцій використовується при (пере)підключеннях кас. Виклик процедури [ИнициализироватьДрайверCashalot\(\)](#) виконає ланцюг викликів щодо ініціалізації драйвера, встановлення параметрів каси та підключення до неї. У випадку якщо підключення не первинне, процедура попередньо виконає відключення драйвера від поточної каси. При цьому, наявні перевірки (умови) не призводять до повторної ініціалізації драйвера і не дозволяють виконати відключення від каси, якщо не було попереднього підключення.

9. Відкриття зміни

Сценарій, в якому функція використовується користувачем при відкритті зміни для одного ПРРО чи для декількох екземплярів ПРРО, в залежності від налаштувань. Для кожного екземпляру окремо необхідно вказати обов'язкові значення параметра `ФНПодключеннойКассы`, щоб мати змогу застосовувати функцію

БУЛЕВО `ОткрытьСмену(ФНПодключеннойКассы)`

При виклику функції (наприклад: фіскалізації чека), якщо зміна не відкрита, відбувається автоматичне відкриття зміни з викликом функції `ОткрытьСмену`. Для запобігання автоматичного відкриття зміни за допомогою функції `УстановитьПараметр` встановіть параметр:

`NOAUTOOPENSHIFT = "True"`

При виконанні функції `ОткрытьСмену` драйвер ПРРО Cashalot викликає вікно авторизації, в якому користувач зазначає папку з файловим ключем та сертифікатом касира або старшого касира, пароль, після чого відбувається відкриття касової зміни.

Параметри функції відкриття зміни:

Ім'я	Тип	Опис
<code>ФНПодключеннойКассы</code>	строка [out]	Фіскальний номер ПРРО. Ідентифікатор пристрою.

Приклад:

Процедура `ОткрытьСменуCashalot()`

```
ФНПодключеннойКассы = "40000000000";
```

```
СменаОткрыта = ДрайверCashalot.ОткрытьСмену(ФНПодключеннойКассы);
```

```
Сообщить?(СменаОткрыта,
```

```
    "Смена открыта.",
```

```
    "Не удалось открыть смену из-за ошибки: " +
```

```
    ПолучитьОшибкуCashalot()));
```

КонецПроцедуры

10. Реєстрація чеків

При реєстрації чеків дані передаються через ланцюжок функцій, призначених для друку чеків на фіскальних реєстраторах в такому порядку:

1. БУЛЕВО **ОткрытьЧек**(ФНПодключеннойКассы, ПризнакФискальногоЧека, ПризнакЧекаВозврата, АннулироватьПредыдущийОткрытыйЧек, НомерОткрытогоЧека, НомерСмены);
2. БУЛЕВО **НапечататьФискальнуюСтроку**(ФНПодключеннойКассы, НаименованиеНоменклатуры, КодУКТВЭД, Артикул, Штрихкод, ЕдиницаИзмерения, НомерСекции, Количество, Цена, Сумма, ПроцентСкидки, СуммаСкидки, СтавкаНДС, ЦенаВключаетНДС, СуммаНДС, ПризнакАкцизногоНалога, Номер строки, ДополнительныеРеквизиты);
або
БУЛЕВО **НапечататьФискСтроку**(ФНПодключеннойКассы, НаименованиеНоменклатуры, Количество, Цена, Сумма, НомерСекции, СтавкаНДС, ДополнительныеРеквизиты);
3. БУЛЕВО **НапечататьНефискСтроку**(ФНПодключеннойКассы, КомментарийКЧеку);
4. БУЛЕВО **УстановитьРасширенныйПорядокОплатыЧека** (ФНПодключеннойКассы, ТипОплаты, СуммаПредоплаты, НомерЗаказа, ФискальныйНомерЧекаПредоплатыИлиВозврата, ФискальныйНомерКассыВозврата, ДопРеквизиты);
5. БУЛЕВО **ЗакрытьЧек** (ФНПодключеннойКассы, СуммаОплатыНаличными, СуммаОплатыКартой, СуммаОплатыКредитом, СуммаОплатыСертификатом, СуммаКОплате, ФискальныйНомерЧекаДляВозврата, НомерОткрытогоЧека, ФискальныйНомерЧека);

Функція **ОткрытьЧек** формує заголовок чека.

Функції **НапечататьФискальнуюСтроку** та **НапечататьФискСтроку** призначені для заповнення чека рядками товарами, що купуються. Ці функції рівноцінні, проте, перша з них має повний перелік параметрів товару, а друга має скорочений перелік. Обидві функції мають параметр **ДополнительныеРеквизиты**, в якому передається сформований XML-рядок з переліком реквізитів товару. Реквізити, які зазначені в XML-рядку, мають пріоритет над явно заданими параметрами функції, тому рекомендується використовувати **ДополнительныеРеквизиты**, а також функцію **НапечататьФискСтроку** з використанням цього параметра. Крім того, у випадку використання функції **НапечататьФискСтроку**, в **ДополнительныеРеквизиты** необхідно обов'язково вказати значення для артикула та одиниці вимірювання товару.

Функція **ЗакрытьЧек** виконує завершення формування чека, передає дані про оплату в ПРРО Cashälot, де чек підписується і відправляється на сервер ФСКО для реєстрації, та повертає як кінцеві параметри фіскальний номер зареєстрованого чека та номер для наступного чека. Порушення послідовності функцій призведе до помилки реєстрації чеків.

Для відображення коментаря в чеку необхідно застосувати метод [НапечататьНефискСтроку](#). При побудові нефіскального рядка необхідно врахувати ширину чекової стрічки та допустиму кількість символів в одному рядку. Якщо необхідно надрукувати декілька нефіскальних рядків, необхідно заповнити параметр [КоментарийКЧеку](#) у вигляді:

```
ПереносСтроки = "\n";  
КоментарийКЧеку = "Перший рядок коментаря" + ПереносСтроки + "Другий рядок коментаря" +  
ПереносСтроки + "Останній рядок коментаря";
```

Для встановлення розширеного порядку сплати поточного чека, використовується функція [УстановитьРасширенныйПорядокОплатыЧека](#). Для різних варіантів потрібно встановити [ТипОплаты](#) = 0, 1, 2, 3, 4, 5, 6, 7.

У випадку формування чека передоплати ([ТипОплаты](#) = 1), заповнити поле суми передоплати [СуммаПредоплаты](#), сума післяплати буде перерахована автоматично, залежно від суми всіх рядків чека за вирахуванням суми передоплати.

У чеку післяплати ([ТипОплаты](#) = 2) потрібно вказати фіскальний номер чека, за яким була здійснена передплата, [ФискальныйНомерЧекаПредоплатыИлиВозврата](#) та фіскальний номер каси, на якій була передплата, якщо він відрізняється. Якщо не відрізняється, встановити значення null або порожній рядок.

При використанні способу оплати «Оплата частинами» ([ТипОплаты](#) = 5, 6, 7) для початку має бути сформовано чек передоплати ([ТипОплаты](#) = 5) з заповненою сумою в полі [СуммаПредоплаты](#). На основі чека передоплати може бути сформовано чек чергової сплати ([ТипОплаты](#) = 6) або одразу операція остаточного розрахунку по чеку передоплати ([ТипОплаты](#) = 7). Чеків чергової сплати в ланцюжку чеків зі способом оплати «Оплата частинами» може бути декілька. Для чека чергової сплати ([ТипОплаты](#) = 6) необхідно заповнити поле [ФискальныйНомерЧекаПредоплатыИлиВозврата](#) та фіскальний номер каси, на якій була передплата, якщо він відрізняється. Якщо не відрізняється, встановити значення null або порожній рядок. Додатково вказати суму чергової сплати в полі [СуммаПредоплаты](#). Якщо після чека передоплати одразу формується чек остаточного розрахунку, достатньо вказати значення в поле [ФискальныйНомерЧекаПредоплатыИлиВозврата](#). Остаточна сума розрахунку буде вирахована автоматично на основі попередніх чеків та сум проведеним по ним.

Також функцію [УстановитьРасширенныйПорядокОплатыЧека](#) можна використовувати для повернення чеків, які були сформовані в інших ПРРО та заповнення додаткової інформації в чек про термінал, при оплаті карткою, за допомогою параметра [ДопРеквизиты](#).

Заокруглення суми по чеку реалізовано наступним чином:

В функцію **Закри́тьЧек** передається значення готівки, що надана клієнтом на касі, значення оплати картою, сертифікатом, кредитом або безготівкою а також, заокруглена сума, розрахована за певним алгоритмом користувача **СуммаКОплате** (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 засоби оплати, заокруглену суму до сплати, решту. У випадку, якщо параметр **СуммаКОплате** не визначений або дорівнює нулю, то він замінюється сумою по всім засобам оплати. В такому випадку заокруглення буде дорівнювати нулю.

Решта по чеку розраховується так:

$$\text{Сума решти} = \text{СуммаОплатыНаличными} + \text{СуммаОплатыКартой} + \text{СуммаОплатыКредитом} + \text{СуммаОплатыСертификатом} - \text{СуммаКОплате}$$

Решта розраховується програмою Cashlot автоматично, згідно вказаних користувачем даних.

Повернення

Для створення та фіскалізації чека повернення товарів та/або послуг необхідно сформувати ланцюжок з функцій для реєстрації чеків.

В функції **ОткрытьЧек** необхідно встановити **ПризнакЧекаВозврата** = Істина, а в функціях **Закри́тьЧек** та, за необхідності, **УстановитьРасширенныйПорядокОплатыЧека** вказати ФН чека, на основі якого формується повернення, та ФН каси, де був сформований чек, у відповідних параметрах.

Опис параметрів функцій для формування чека вказані в таблиці:

ОткрытьЧек		
Ім'я параметра	Тип	Опис параметра
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ПризнакФискальногоЧека	булево	Ознака фіскального чека
ПризнакЧекаВозврата	булево	Ознака чека повернення
АннулироватьПредыдущийОткрытыйЧек	булево	Ознака автоматичного анулювання раніше відкритого чека
НомерОткрытогоЧека	число [out]	Номер нового чека
НомерСмены	число [out]	Номер відкритої зміни
НапечататьФискальнуюСтроку		
Ім'я параметра	Тип	Опис параметра
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

НаименованиеНоменклатуры	строка	Найменування товару. Допускається довжина до 127 символів
КодУКТВЭД	строка	Код УКТЗЕД
Артикул	строка	Артикул. Допускається довжина до 63 символів.
Штрихкод	строка	Штрих-код
ЕдиницаИзмерения	строка	Одиниця виміру
НомерСекции	строка	Номер секції (не використовується)
Количество	число	Кількість товару
Цена	число	Ціна одиниці товару
Сумма	число	Сума по позиції чека
ПроцентСкидки	число	Відсоток знижки. Увага! При заповненні ПроцентСкидки, необхідно також обов'язково заповнювати та передавати СуммаСкидки
СуммаСкидки	число	Сума знижки на позицію
СтавкаНДС	число	ПДВ у процентах (від 0 до 99,99)
ЦенаВключаетНДС	булево	Ознака, що сума містить ПДВ
СуммаНДС	число	Сума ПДВ за одиницю товару
ПризнакАкцизногоНалога	булево	Ознака підакцизного товару.
НомерСтроки	булево	Номер рядка (не використовується)
ДополнительныеРеквизиты	строка	Додаткові параметри (XML таблиця)

НапечататьФискСтроку

Ім'я параметра	Тип	Опис параметра
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
НаименованиеНоменклатуры	строка	Найменування товару. Допускається довжина до 127 символів
Количество	число	Кількість товару
Цена	число	Ціна одиниці товару
Сумма	число	Сума по позиції чека
НомерСекции	строка	Номер секції (не використовується)
СтавкаНДС	число	ПДВ у процентах (від 0 до 99,99)
ДополнительныеРеквизиты	строка	Додаткові параметри (XML таблиця)

НапечататьНефискСтроку

Ім'я параметра	Тип	Опис параметра
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
КомментарийКЧеку	строка	Нефіскальний рядок, коментар.

УстановитьРасширенныйПорядокОплатыЧека

Ім'я параметра	Тип	Опис параметра
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

ТипОплаты	число	Спосіб оплати: 0 - звичайна(продаж), 1 - передоплата, 2 - післяплата, 3 - інтернет продаж, 4 - повна післяплата, 5 - передоплата (оплата частинами), 6 - Чергова сплата (оплата частинами), 7 - остаточний розрахунок (оплата частинами).
СуммаПредоплаты	число	Сума попередньої оплати (звичайної) або попередньої оплати та чергової сплати (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».
НомерЗаказа	строка	Номер замовлення/договору.
ФискальныйНомерЧекаПредоплатыИлиВозврата	строка	Фіскальний номер чека, по якому здійснюється доплата/повернення.
ФискальныйНомерКассыВозврата	строка	Фіскальний номер РРО, по якому здійснюється доплата/повернення.
ДопРеквизиты	строка	Додаткові параметри у вигляді XML таблиці.
ЗакретьЧек		
Ім'я параметра	Тип	Опис параметра
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
СуммаОплатыНаличными	число	Сума оплати готівкою
СуммаОплатыКартой	число	Сума оплати картою
СуммаОплатыКредитом	число	Сума оплати кредитом
СуммаОплатыСертификатом	число	Сума оплати сертифікатом
СуммаКОплате	число	Сума до оплати, що враховує всі види оплати, знижки та заокруглення
ФискальныйНомерЧекаДляВозврата	строка	Фіскальний номер чека, за яким виконується повернення
НомерОткрытогоЧека	число [out]	Номер нового чека
ФискальныйНомерЧека	строка[out]	Фіскальний номер нового чека

ДополнительныеРеквизиты та ДопРеквизиты. Текст в форматі XML з кодуванням UTF-8, що передається за допомогою параметрів з типом "строка".

Увага! Параметри рядка чека, які вказані в ДополнительныеРеквизиты, мають пріоритет над параметрами, які передані в функціях НапечататьФискальнуюСтроку або НапечататьФискСтроку.

Атрибути XML-рядка параметра ДополнительныеРеквизиты:

Атрибут XML-рядка	Опис атрибута
Nomenclature *	Найменування товару. Допускається довжина до 127 символів
UKTZED	Код УКТЗЕД
VendorCode *	Артикул, допускається довжина до 63 символів
Barcode	Штрихкод
Dimension *	Одиниця вимірювання

Discount	Відсоток знижки. Увага! При заповненні Discount, необхідно також обов'язково заповнювати та передавати DiscountAmount
DiscountAmount	Сума знижки
VAT *	Ставка ПДВ у процентах
VATLetter	Літера ставки ПДВ
PriceIncludesVAT або PriceIncludeVAT	Для випадків, коли сума по позиції не містить суму ПДВ (true/false)
AmountVAT	Сума ПДВ
IsExcisable	Ознака підакцизного товару, True/False
ExciseLetter	Літера ставки акцизного податку
ExciseStampBarcode	Штрих-код марки акцизного податку. У випадку коли товарів більше ніж 1, по одній позиції номенклатури, то всі штрих-коди заповнюються через кому. Поле ігнорується, якщо ознака підакцизного товару IsExcise = false.

* обов'язкові параметри в рядку **ДополнительныеРеквизиты**, якщо не вказані відповідні явні параметри.

*Приклад формування **ДополнительныеРеквизиты**:*

Процедура СформироватьXMLДополнительныхРеквизитов(ПозицияЧека)

```

ЗаписьXML = Новый ЗаписьXML;
ЗаписьXML.УстановитьСтроку("utf-8");
ЗаписьXML.ЗаписатьОбъявлениеXML();

ЗаписьXML.ЗаписатьНачалоЭлемента("Table");
    ЗаписьXML.ЗаписатьНачалоЭлемента("Record");
        ЗаписьXML.ЗаписатьАтрибут("NOMENCLATURE",
            ПозицияЧека.НаименованиеНоменклатуры);
        ЗаписьXML.ЗаписатьАтрибут("UKTZED", ПозицияЧека.КодУКТВЭД);
        ЗаписьXML.ЗаписатьАтрибут("VENDORCODE", ПозицияЧека.Артикул);
        ЗаписьXML.ЗаписатьАтрибут("BARCODE", ПозицияЧека.Штрихкод);
        ЗаписьXML.ЗаписатьАтрибут("DIMENSION",
            ПозицияЧека.ЕдиницаИзмерения);
        ЗаписьXML.ЗаписатьАтрибут("DISCOUNT",
            Строка(ПозицияЧека.ПроцентСкидки));
        ЗаписьXML.ЗаписатьАтрибут("DISCOUNTAMOUNT",
            Строка(ПозицияЧека.СуммаСкидки));
        ЗаписьXML.ЗаписатьАтрибут("VAT",Строка(ПозицияЧека.СтавкаНДС));
        ЗаписьXML.ЗаписатьАтрибут("VATLetter",
            Строка(ПозицияЧека.БукваСтавкиНДС));
        ЗаписьXML.ЗаписатьАтрибут("PRICEINCLUDEVAT",
            ?(ПозицияЧека.ЦенаВключаетНДС, "True", "False"));
        ЗаписьXML.ЗаписатьАтрибут("AMOUNTVAT",
            Строка(ПозицияЧека.СуммаНДС));
        ЗаписьXML.ЗаписатьАтрибут("ISEXCISABLE",
            ?(ПозицияЧека.ПризнакАкцизногоНалога, "True", "False"));
        ЗаписьXML.ЗаписатьАтрибут("ExciseLetter",
            Строка(ПозицияЧека.БукваСтавкиАкциз));
    
```

```

ЗаписьXML.ЗаписатьАтрибут("EXCISESTAMPBARCODE",
ПозицияЧека.ШтрихкодАкцизнойМарки);
ЗаписьXML.ЗаписатьКонецЭлемента();
ЗаписьXML.ЗаписатьКонецЭлемента();

ПозицияЧека.ДополнительныеРеквизиты = ЗаписьXML.Закрыть();

КонецПроцедуры

```

Атрибути XML-рядка параметра *ДопРеквизиты* при використанні оплати карткою через термінал для методу *УстановитьРасширенныйПорядокОплатыЧека*:

Атрибут XML-рядка	Опис атрибута
TerminalID	Код терміналу.
ApprovalCode	Код підтвердження.
RRN	РРН 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	РРН транзакції чека, по якому здійснено повернення.
AcquireID	Ідентифікатор еквайра
ParentReceiptOrderDate	Дата продажу чека по якому здійснюється повернення (формат: dd.mm.rrrr). Використовується тільки для чеків продажів.
ParentRROFiscalNumber	Фіскальний номер РРО, по якому здійснюється доплата/повернення.
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 телефона покупця на який можна відправити фіскалізований чек у вигляді SMS повідомлення. Передається в поле номера телефону при натисканні кнопки «Надіслати на SMS».

CustomerViberNumber	Номер телефона 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 CustomerEmail , виконано налаштування електронної пошти в РМК в розділі «Налаштування - Загальні налаштування - Господарська одиниця - Налаштування пошти» та встановлене значення true для AutosendCustomerEmail
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 CustomerSmsNumber , виконано налаштування сервісу доставки SMS в кабінеті користувача Cashalot (https://my.cashalot.org.ua/) та встановлено значення true для AutosendCustomerSms
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 CustomerViberNumber , виконано налаштування сервісу доставки повідомлень у Viber в кабінеті користувача Cashalot (https://my.cashalot.org.ua/) та встановлено значення true для AutosendCustomerViber
SumPayByBank	Сума оплати безготівкою (вид оплати)
PayFormCD	Код засобу оплати (див. приклад застосування). Застосовується в складі одного Record з атрибутами PayFormNM, PayFormSum, PayFormUseTransactionDetails
PayFormNM	Найменування засобу оплати Застосовується в складі одного Record з атрибутами PayFormCD, PayFormSum, PayFormUseTransactionDetails
PayFormSum	Сума засобу оплати Застосовується в складі одного Record з атрибутами PayFormCD, PayFormNM, PayFormUseTransactionDetails

PayFormUseTransactionDetails	Ознака (true/false) яка дозволяє застосувати реквізити транзакції POS-термінала до безготівкового засобу оплати (Платіж Mono Pay, Платіж LIQPAY, Платіж WAYFORPAY) використаних в способах оплати 0,1,2,5, 6 та 7 Застосовується в складі одного Record з атрибутами PayFormCD, PayFormNM, PayFormSum
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-термінала до безготівкового засобу оплати (Платіж Mono Pay, Платіж LIQPAY, Платіж WAYFORPAY) використаних в способах оплати 0,1,2,5, 6 та 7
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, драйвер поверне відповідь на раніше сформований чек та відобразить його візуальну форму.
ExpressWaybillNumber	Номер експрес-накладної (ТТН)

Функция СформироватьТаблицуДопРеквизитовТерминала(Терминал) Экспорт

ДанныеЗагрузки = Неопределено;

Результат = Новый Массив();

ЗаписьXML = Новый ЗаписьXML;

ЗаписьXML.УстановитьСтроку("UTF-8");

ЗаписьXML.ЗаписатьОбъявлениеXML();

ЗаписьXML.ЗаписатьНачалоЭлемента("Table");

ЗаписьXML.ЗаписатьНачалоЭлемента("Record");

ЗаписьXML.ЗаписатьАтрибут("TerminalID", Строка(? (TerminalID =
неопределено, "", TerminalID)));

ЗаписьXML.ЗаписатьАтрибут("ApprovalCode", Строка(? (ApprovalCode =
неопределено, "", ApprovalCode)));

ЗаписьXML.ЗаписатьАтрибут("RRN", Строка(? (RRN = неопределено, "", RRN)));

ЗаписьXML.ЗаписатьАтрибут("IssuerName", Строка(? (IssuerName =
неопределено, "", IssuerName)));

ЗаписьXML.ЗаписатьАтрибут("PAN", ? (PAN = неопределено, "", PAN));

ЗаписьXML.ЗаписатьАтрибут("TransactionDate", ? (TransactionDate = неопределено, "",
TransactionDate));

ЗаписьXML.ЗаписатьАтрибут("AcquireName", ? (AcquireName = неопределено, "",
AcquireName));

ЗаписьXML.ЗаписатьАтрибут("InvoiceNumber", ? (InvoiceNumber = неопределено, "",
InvoiceNumber));

ЗаписьXML.ЗаписатьАтрибут("ParentRRN", ? (ParentRRN = неопределено, "", ParentRRN));

ЗаписьXML.ЗаписатьАтрибут("SignVerif", ? (SignVerif = неопределено, "", SignVerif));

ЗаписьXML.ЗаписатьКонецЭлемента();

ЗаписьXML.ЗаписатьНачалоЭлемента("Record");

ЗаписьXML.ЗаписатьАтрибут("CustomerEmail", ? (Email = неопределено, "", Email));

ЗаписьXML.ЗаписатьАтрибут("AutosendCustomerEmail", XMLСтрока(AutoSendEmail));

ЗаписьXML.ЗаписатьКонецЭлемента();

ЗаписьXML.ЗаписатьНачалоЭлемента("Record");

ЗаписьXML.ЗаписатьАтрибут("CustomerSMSNumber", ? (PhoneSMS = неопределено, "",
PhoneSMS));

ЗаписьXML.ЗаписатьАтрибут("AutosendCustomerSMS", XMLСтрока(AutoSendSMS));

ЗаписьXML.ЗаписатьКонецЭлемента();

ЗаписьXML.ЗаписатьНачалоЭлемента("Record");

ЗаписьXML.ЗаписатьАтрибут("CustomerViberNumber", ? (PhoneViber = неопределено, "",
PhoneViber));

ЗаписьXML.ЗаписатьАтрибут("AutosendCustomerViber", XMLСтрока(AutoSendViber));

ЗаписьXML.ЗаписатьКонецЭлемента();

ЗаписьXML.ЗаписатьНачалоЭлемента("Record");

ЗаписьXML.ЗаписатьАтрибут("SumPayByBank", ? (СумаБезготівковоїОплати =
неопределено, "", СумаБезготівковоїОплати));

```
ЗаписьXML.ЗаписатьКонецЭлемента();
```

```
ЗаписьXML.ЗаписатьНачалоЭлемента("Record");
```

```
    ЗаписьXML.ЗаписатьАтрибут("BankPayToolsNM", Строка(? (НазваЗасобуОплатиБезготівка =  
неопределено, "", НазваЗасобуОплатиБезготівка)));
```

```
    ЗаписьXML.ЗаписатьКонецЭлемента();
```

```
ЗаписьXML.ЗаписатьНачалоЭлемента("Record");
```

```
    ЗаписьXML.ЗаписатьАтрибут("PrePaymentPayToolsNM",  
Строка(? (НазваЗасобуОплатиПередоплатаП =  
неопределено, "", НазваЗасобуОплатиПередоплатаП)));
```

```
    ЗаписьXML.ЗаписатьКонецЭлемента();
```

```
ЗаписьXML.ЗаписатьНачалоЭлемента("Record");
```

```
    ЗаписьXML.ЗаписатьАтрибут("AfterPaymentPayToolsNM",  
Строка(? (НазваЗасобуОплатиПісляплатаП = неопределено, "", НазваЗасобуОплатиПісляплатаП)));  
    ЗаписьXML.ЗаписатьКонецЭлемента();
```

```
ЗаписьXML.ЗаписатьНачалоЭлемента("Record");
```

```
    ЗаписьXML.ЗаписатьАтрибут("UseTransactionDetailsToSumPayByBank",  
?(ВикористовуватиТермінальніРеквБезгот = неопределено, "",  
?(ВикористовуватиТермінальніРеквБезгот, "True", "False"));  
    ЗаписьXML.ЗаписатьКонецЭлемента();
```

Для Каждого ВидОплаты ИЗ ВидиСобственныхОплат Цикл

```
    ЗаписьXML.ЗаписатьНачалоЭлемента("Record");
```

```
        ЗаписьXML.ЗаписатьАтрибут("PayFormCD" , ?(ВидОплаты.Код = неопределено, "",  
ВидОплаты.Код));
```

```
        ЗаписьXML.ЗаписатьАтрибут("PayFormSum" , ?(ВидОплаты.Сумма = неопределено, "",  
ВидОплаты.Сумма));
```

```
        ЗаписьXML.ЗаписатьАтрибут("PayFormNM" , ?(ВидОплаты.Наименование = неопределено, "",  
ВидОплаты.Наименование));
```

```
        ЗаписьXML.ЗаписатьАтрибут("PayFormUseTransactionDetails" , ?(ВидОплаты.Признак =  
неопределено, ""?(ВидОплаты.Признак, "True", "False")));
```

```
        ЗаписьXML.ЗаписатьКонецЭлемента();
```

КонецЦикла;

```
ЗаписьXML.ЗаписатьКонецЭлемента();
```

```
Возврат ЗаписьXML.Закреть();
```

КонецФункции

Увага! Для передачі параметрів функцій з дробовою частиною (зокрема, нульовою, наприклад, "2,00") допускається використовувати десятковий розділювач "." (крапка) або "," (кома). Якщо параметр не задано, або задано некоректно, його значення приймається рівним одиниці (1).

Приклад використання засобів оплат в чеках:

В драйвері Cashalot для BAS наявні додаткові реквізити параметру [ДопРеквизиты](#) методу з встановленням розширеної інформації про сплату ([УстановитьРасширенныйПорядокОплатыЧека](#)) за допомогою яких можливе налаштування передачі засобів оплати у чек. Реквізити варто використовувати при необхідності відображення в чеку засобів оплат таких як Інтернет-банкінг, Переказ з картки, Платіж WAYFORPAY, тощо (див. доступний перелік нижче). Засіб оплати можливо застосовувати до всіх використовуваних [способів оплат](#) по чеку (продаж, передплата/післяплата, інтернет-продаж, повна післяплата, тощо).

Стандартні засоби оплат (Готівка (Готівкові кошти), Картка, Сертифікат, Кредит) заповнюються стандартно відповідно до опису функції [ЗакретьЧек](#).

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 реквізитів: [SumPayByBank](#), [BankPayToolsNM](#), [PrePaymentPayToolsNM](#), [AfterPaymentPayToolsNM](#) та [UseTransactionDetailsToSumPayByBank](#). Спрощений варіант надає змогу використовувати лише один засіб оплати у чеку.

За допомогою реквізиту [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 реквізиту [BankPayToolsNM](#) дозволяється у випадку застосування способів оплати 0, 1, 2, 5, 6 або 7. Реквізит [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 реквізит [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! Реквізит [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 ([СуммаОплатыКартой](#)), то параметри транзакції буде відображено у чеку саме під засобом Платіж LIQPAY.

Варіант використання засобів оплат №2 (розширений)

Розширений варіант використання засобів оплат представляє собою набір реквізитів, а саме: **PayFormCD**, **PayFormSum**, **PayFormNM** та **PayFormUseTransactionDetails**. Зазначені реквізити вказують характеристику конкретного засобу оплати та мають передаватись в складі одного Record параметра **ДопРеквизиты** в методі **УстановитьРасширенныйПорядокОплатыЧека**. За допомогою розширеного варіанту можливо передати декілька засобів оплат одночасно.

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 Mono Pay
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 (PayFormCD) але за умови, що найменування засобу оплати (PayFormNM) заповнено відмінне від значення попереднього засобу. В разі відсутності потрібного найменування засобу оплати користувач може використати необхідну власну назву заповнюючи її в PayFormNM. 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 PayFormNM в рамках Record, необхідно скористатись реквізитом PayFormUseTransactionDetails який

приймає значення True/False. Опція доступна лише за умови, що в PayFormNM заповнено значення найменування засобу оплати: Платіж Mono Pay, Платіж LIQPAY або Платіж WAYFORPAY. Якщо було використано інший засіб оплати (наприклад, Інтернет-еквайринг) разом із PayFormUseTransactionDetails = True, то параметри транзакції POS-термінала не будуть відображені у чеку. Якщо параметр не встановлено то це означає, що використовується його ознака False.

Важливо! Реквізит PayFormUseTransactionDetails 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 ([СуммаОплатыКартой](#)), то параметри транзакції буде відображено у чеку саме під засобом Платіж LIQPAY.

Увага! Інформація про засоби оплати необхідно заповнювати в окремих Record.

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 ОткрытьЧек:

Процедура ОткрытьЧекCashalot(Результат, ПризнакЧекаВозврата = Ложь)

ПризнакФискальногоЧека = Истина;

АннулироватьПредыдущийОткрытыйЧек = Истина;

НомерОткрытогоЧека = 0;

НомерСмены = 0;

ЧекОткрыт = ДрайверCashalot.ОткрытьЧек(ФНПодключеннойКассы,
ПризнакФискальногоЧека,
ПризнакЧекаВозврата,
АннулироватьПредыдущийОткрытыйЧек,
НомерОткрытогоЧека,
НомерСмены);

Сообщить?(ЧекОткрыт,
"Открыт чек №" + НомерОткрытогоЧека,
"Не удалось открыть чек из-за ошибки: " + ПолучитьОшибкуCashalot());
Результат = ЧекОткрыт;

КонецПроцедуры

Приклад НапечататьФискальнуюСтроку:

Процедура НапечататьФискальнуюСтрокуCashalot(Результат, ПозицииЧека)

Для Каждого ПозицияЧека Из ПозицииЧека Цикл

ФискальнаяСтрокаНапечатана =

ДрайверCashalot.НапечататьФискальнуюСтроку(ФНПодключеннойКассы,
ПозицияЧека.НаименованиеНоменклатуры, ПозицияЧека.КодУКТВЭД,
ПозицияЧека.Артикул,ПозицияЧека.Штрихкод, ПозицияЧека.ЕдиницаИзмерения, "",
ПозицияЧека.Количество, ПозицияЧека.Цена, ПозицияЧека.Сумма, ПозицияЧека.ПроцентСкидки,
ПозицияЧека.СуммаСкидки, ПозицияЧека.СтавкаНДС, ПозицияЧека.ЦенаВключаетНДС,
ПозицияЧека.СуммаНДС, ПозицияЧека.ПризнакАкцизногоНалога, "",
ПозицияЧека.ДополнительныеРеквизиты);

Если НЕ ФискальнаяСтрокаНапечатана Тогда

Сообщить("Не удалось заполнить номенклатуру "" +
ПозицияЧека.НаименованиеНоменклатуры +
"" в чек Cashalot из-за ошибки: " +
ПолучитьОшибкуCashalot());

Прервать;

КонецЕсли;

КонецЦикла;

Результат = ФискальнаяСтрокаНапечатана;

КонецПроцедуры

Приклад НапечататьФискСтроку:

Процедура НапечататьФискСтрокуCashalot(Результат, ПозицииЧека)

Для Каждого ПозицияЧека Из ПозицииЧека Цикл

ДополнительныеРеквизиты = СформироватьXMLДополнительныхРеквизитов(ПозицияЧека);

ФискальнаяСтрокаНапечатана =

ДрайверCashalot.НапечататьФискСтроку(ФНПодключеннойКассы,ПозицияЧека.НаименованиеНоменклатуры, ПозицияЧека.Количество, ПозицияЧека.Цена, ПозицияЧека.Сумма, НомерСекции, ПозицияЧека.СтавкаНДС, ДополнительныеРеквизиты);

Если НЕ ФискальнаяСтрокаНапечатана Тогда

Сообщить("Не удалось заполнить номенклатуру "" +

ПозицияЧека.НаименованиеНоменклатуры +

"" в чек Cashalot из-за ошибки: " +

ПолучитьОшибкуCashalot());

Прервать;

КонецЕсли;

КонецЦикла;

Результат = ФискальнаяСтрокаНапечатана;

КонецПроцедуры

Приклад НапечататьНефискСтроку:

Процедура НапечататьНефискСтрокуCashalot(Результат, КомментарийКЧеку)

КомментарийДобавлен = ДрайверCashalot.НапечататьНефискСтроку(ФНПодключеннойКассы, КомментарийКЧеку);

Если НЕ КомментарийДобавлен Тогда

Сообщить("Не удалось добавить комментарий к чеку из-за ошибки: " +

ПолучитьОшибкуCashalot());

КонецЕсли;

Результат = КомментарийДобавлен;

КонецПроцедуры

Приклад УстановитьРасширенныйПорядокОплатыЧека:

Процедура УстановитьРасширенныйПорядокОплатыЧекаCashalot(Результат, СтруктураРасширенныхПараметровОплаты)

РасширенныйПорядокОплатыУстановлен =

ДрайверCashalot.УстановитьРасширенныйПорядокОплатыЧека(ФНПодключеннойКассы, СтруктураРасширенныхПараметровОплаты.ТипОплаты, СтруктураРасширенныхПараметровОплаты.СуммаПредоплаты, СтруктураРасширенныхПараметровОплаты.НомерЗаказа, СтруктураРасширенныхПараметровОплаты.ФискальныйНомерЧекаПредоплатыИлиВозврат, СтруктураРасширенныхПараметровОплаты.ФискальныйНомерКассыВозврата, ДопРеквизиты);

Если НЕ РасширенныйПорядокОплатыУстановлен Тогда

Сообщить("Не удалось установить расширенный порядок оплаты из-за

ошибки: " + ПолучитьОшибкуCashalot());

КонецЕсли;

Результат = РасширенныйПорядокОплатыУстановлен;

КонецПроцедуры

Приклад ЗакрыватьЧек:

Процедура ЗакрыватьЧекCashalot(Результат, СтруктураОплат)

ФискальныйНомерЧека = "";

ЧекЗакрит = ДрайверCashalot.ЗакрыватьЧек(ФНПодключеннойКассы,
СтруктураОплат.СуммаОплатыНаличными, СтруктураОплат.СуммаОплатыКартой,
СтруктураОплат.СуммаОплатыКредитом, СтруктураОплат.СуммаОплатыСертификатом,
СтруктураОплат.СуммаКОплате, ФискальныйНомерЧекаВозврата, НомерОткрытогоЧека,
ФискальныйНомерЧека);

Сообщить?(ЧекЗакрит,

"Чек №" + ФискальныйНомерЧека + " успешно фискализирован.",

"Не удалось фискализировать чек из-за ошибки: " + ПолучитьОшибкуCashalot()));

Результат = ЧекЗакрит;

КонецПроцедуры

11. Відміна відкритого чека

Для відміни раніше відкритого чека використовується функція:

БУЛЕВО `ОтменитьЧек(ФНПодключеннойКассы)`

Параметри функції відміни чека:

Ім'я	Тип	Опис
<code>ФНПодключеннойКассы</code>	строка [out]	Фіскальний номер ПРРО. Ідентифікатор пристрою.

Приклад:

Процедура `ОтменитьЧекCashalot(Результат)`

```
ЧекОтменен = ДрайверCashalot.ОтменитьЧек(ФНПодключеннойКассы);
```

```
Сообщить?(ЧекОтменен,  
    "Открытый чек в Cashalot был отменен.",  
    "Не удалось отменить чек из-за ошибки: " +  
        ПолучитьОшибкуCashalot());  
Результат = ЧекОтменен;
```

КонецПроцедуры

12. Вивести візуальне відображення чека на екран

Для виводу візуального відображення чека на екран, використовується функція:

БУЛЕВО **ОтобразитьЧек** (ФНПодключеннойКассы, ФНЧекаДляОтображения)

Функція **ОтобразитьЧек** 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:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка [out]	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ФНЧекаДляОтображения	строка	Фіскальний номер чека.

Приклад:

Процедура **ОтобразитьЧек()**

```
Результат = ДрайверCashalot.ОтобразитьЧек(ФНПодключеннойКассы,  
ФНЧекаДляОтображения);
```

```
Сообщить?(Результат <> Неопределено,  
    "Чек отобразили.",  
    "Не удалось отобразить из-за ошибки: " +  
    ПолучитьОшибкуCashalot()));
```

КонецПроцедуры

13. Отримання фіскального номера чека за його локальним номером

Для отримання фіскального номера чека за його локальним номером, використовується функція:

СТРОКА **ПолучитьФискальныйНомерЧека** (ФНПодключеннойКассы,
ЛокальныйНомерЧека)

Функція **ПолучитьФискальныйНомерЧека** 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:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка [out]	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ЛокальныйНомерЧека	строка	Локальний номер чека.

Приклад:

Процедура ПолучитьФискальныйНомерЧека()

```
ПолученныйФН = ДрайверCshalot.ПолучитьФискальныйНомерЧека(ФискальныйНомерКассы,
ЛокальныйНомерЧека);
Если (ПолученныйФН = "") Тогда
    Сообщить("Чека с номером " + ЛокальныйНомерЧека + " нету в базе");
Иначе
    Сообщить("ФН чека: " + ПолученныйФН);
КонецЕсли;

КонецПроцедуры
```

14. Отримання локального номера чека за його фіскальним номером

Для отримання локального номера чека за його фіскальним номером, використовується функція:

СТРОКА **ПолучитьЛокальныйНомерЧека** (ФНПодключеннойКассы, ФНчека)

Функція **ПолучитьЛокальныйНомерЧека** 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. У випадку коли задано параметр **NOAUTOOPENSHIFT** = True, виклик функції при закритті змін переривається і відображається повідомлення "Операція неможлива, необхідно відкрити зміну"

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка [out]	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ФНчека	строка	Локальний номер чека.

Приклад:

Процедура ПолучитьЛокальныйНомерЧека()

```
ПолученныйЛокальныйНомер =
ДрайверCshalot.ПолучитьЛокальныйНомерЧека(ФискальныйНомерКассы, ФНчека);
Если (ПолученныйЛокальныйНомер = "") Тогда
    Сообщить("Чека с номером " + ФНчека + " нету в базе");
Иначе
    Сообщить("Локальный номер чека: " + ПолученныйЛокальныйНомер);
КонецЕсли;

КонецПроцедуры
```


15. Отримання xml структури фіскального чека

Для отримання XML структури чека за його фіскальним номером, використовується функція:
СТРОКА **ПолучитьXMLЧека**(ФНПодключеннойКассы, ФНчекаXML)

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

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ФНчекаXML	строка	Фіскальний номер чека.

Приклад:

Процедура **ПолучитьXMLчека()**

XMLчека = ДрайверCashalot.ПолучитьXMLЧека(ФискальныНомерКассы, ФНчекаXML);

Если (XMLчека = "") Тогда

Сообщить("Чека с номером " + ФНчекаXML + " нету в базе");

Иначе

Сообщить("XML чека: " + XMLчека);

КонецЕсли;

КонецПроцедуры

16. Друк періодичних звітів

Для друку періодичних звітів, використовується функція:

БУЛЕВО **НапечататьПериодическийОтчетПоДатам** (ФНПодключеннойКассы,
ДатаНачалаОтчета, ДатаОкончанияОтчета, ПризнакСокращенногоОтчета)

Період це проміжок часу, що встановлюється між ДатаНачалаОтчета та ДатаОкончанияОтчета.

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка [out]	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ДатаНачалаОтчета	дата	Початок періоду в форматі dd.MM.yyyy
ДатаОкончанияОтчета	дата	Кінець періоду в форматі dd.MM.yyyy
ПризнакСокращенногоОтчета	булево	Ознака скороченого звіту

Приклад:

Процедура **НапечататьПериодическийОтчетCashalot**(Результат, ДатаНачалаОтчета =
Неопределено, ДатаОкончанияОтчета = Неопределено, ПризнакСокращенногоОтчета = Ложь)

СтруктураОтвета =

ДрайверCashalot.НапечататьПериодическийОтчетПоДатам(ФНПодключеннойКассы,
?(ДатаНачалаОтчета = Неопределено, ТекущаяДата(), ДатаНачалаОтчета),
?(ДатаОкончанияОтчета = Неопределено, ТекущаяДата(), ДатаОкончанияОтчета),

ПризнакСокращенногоОтчета);

Сообщить?(СтруктураОтвета <> Неопределено,
"Периодический отчет напечатан.",
"Не удалось напечатать периодический отчет из-за ошибки: " +
ПолучитьОшибкуCashalot());

Результат = СтруктураОтвета <> Неопределено;

КонецПроцедуры

17. Отримання інформації про поточний стан ПРРО

Для отримання інформації про поточний стан використовується функція:

БУЛЕВО **ПолучитьТекущееСостояние** (ФНПодключеннойКассы, ВходныеПараметры, XMLСтрокаТекущихПараметров)

Параметри функції поточного стану ПРРО:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка [out]	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ВходныеПараметры	строка	Таблиця вхідних параметрів, параметр ігнорується.
XMLСтрокаТекущихПараметров	строка	XML-структура параметрів з поточними значеннями

Таблиця вихідних параметрів має вигляд XML-структури з переліком параметрів.

Параметри XML-структури вихідних параметрів:

Ім'я параметра	Значення
CheckNumber	Локальний номер останнього чека
CashBalance	Залишок готівки в касі
IsOfflineMode	Стан офлайн-режиму каси: 0 – каса в онлайн-режимі; 1 – каса в офлайн-режимі
NextLocalNumber	Наступний локальний номер чека
LastCheckFiscalNumber	Фіскальний номер останнього чека
ShiftNumber	Номер поточної відкритої зміни/ останньої закритої зміни
ShiftID	Ідентифікатор зміни
ShiftFiscalNumber	Фіскальний номер зміни
ShiftState	Стан зміни: 1 – закрыта. 2 – відкрита.
DateTime	Дата і час формування останнього фіскального документа у форматі (rrrr.мм.дд гг.хв.сс)
LicEndDate	Дата завершення терміну дії ліцензії ПРРО у форматі (rrrr.мм.дд гг.хв.сс)

CertEndDate	Дата завершення терміну дії сертифікату касира у форматі (pprrr.мм.дд гг.хв.сс)
-------------	---

Приклад:

Процедура ПолучитьТекущееСостояниеCashalot(Результат)

```
XMLСтрокаТекущихПараметров = "";
ПараметрыПолучены =
ДрайверCashalot.ПолучитьТекущееСостояние(ФНПодключеннойКассы, "",
XMLСтрокаТекущихПараметров);

Сообщить("Состояние кассы в XML: " + XMLСтрокаТекущихПараметров);
```

КонецПроцедуры

18. Друк довільної текстової інформації

Для друку на чекову стрічку довільного тексту використовується функція

БУЛЕВО НапечататьТекстовыйДокумент(ФНПодключеннойКассы, ТекстДляПечати)

Після виклику відображається вікно з зображенням тексту і можливістю роздрукувати.

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

Функцію можливо викликати без відкриття касової зміни.

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка [out]	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ТекстДляПечати	строка	Довільний текст, який передається на друк.

Приклад:

Процедура НапечататьТекстовыйДокументCashalot(ТекстДляПечати)

```
ПереносСтроки = "\n";
ТекстДляПечати = "Перший рядок" + ПереносСтроки + "Другий рядок";

ТекстовыйДокументНапечатан =
ДрайверCashalot.НапечататьТекстовыйДокумент(ФНПодключеннойКассы,
ТекстДляПечати);

Сообщить(?(ТекстовыйДокументНапечатан,
"Текстовый документ напечатан.",
"Не удалось напечатать текстовый документ из-за ошибки: " +
ПолучитьОшибкуCashalot()));
```

19. Службовое внесения/выдача

Для выполнения операции службового внесения или выдачи используется функция
БУЛЕВО `НапечататьЧекВнесенияВыемки` (ФНПодключеннойКассы, СуммаВнесения
или СуммаВыдачи)

Параметры функции службового внесения/выдачи:

Имя	Тип	Описание
ФНПодключеннойКассы	строка	Фискальный номер ПРРО. Идентификатор устройства.
СуммаВнесения или СуммаВыдачи	число	Сумма внесения. Сумма выдачи указывается с минусом.

Пример внесения:

Процедура `СлужбовоеВнесениеCashalot(СуммаВнесения)`

Если `СуммаВнесения < 0.01` Тогда

Сообщить("Сумма службового внесения не может быть меньше 0.01 грн.")

Возврат;

КонецЕсли;

Результат = `ДрайверCashalot.НапечататьЧекВнесенияВыемки(ФНПодключеннойКассы, СуммаВнесения)`;

Сообщить?(Результат,

`СуммаВнесения + " грн. успешно внесены в кассу."`,

`"Не удалось внести " + СуммаВнесения + " грн. в кассу из-за
ошибки: " + ПолучитьОшибкуCashalot()`);

КонецПроцедуры

Пример выдачи:

Процедура `СлужбнаяВыдачаCashalot(СуммаВыдачи)`

Если `СуммаВнесения < 0` Тогда

Сообщить("Сумма службной выемки должна быть больше или равна 0 грн.")

Возврат;

КонецЕсли;

Результат = `ДрайверCashalot.НапечататьЧекВнесенияВыемки(ФНПодключеннойКассы, -СуммаВыдачи)`;

Сообщить?(Результат,

`СуммаВыдачи + " грн. успешно изъяты с кассы."`,

`"Не удалось изъять " + СуммаВыдачи + " грн. с кассы из-за
ошибки: " + ПолучитьОшибкуCashalot()`);

КонецПроцедуры

20. Формування X-звіту

Для виконання операції формування звіту без гасіння (X-звіту) використовується функція
БУЛЕВО `НапечататьОтчетБезГашения(ФНПодключеннойКассы)`

Параметри функції для формування X-звіту:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

Приклад:

Процедура `СформироватьXОтчетCashalot()`

```
XОтчетСформирован =  
ДрайверCashalot.НапечататьОтчетБезГашения(ФНПодключеннойКассы);  
Сообщить?(XОтчетСформирован,  
    "X-отчет успешно сформирован.",  
    "Не удалось сформировать X-отчет из-за ошибки: " +  
    ПолучитьОшибкуCashalot());
```

КонецПроцедуры

21. Формування Z-звіту

Для формування Z-звіту використовується функція:

БУЛЕВО `НапечататьОтчетСГашением(ФНПодключеннойКассы)`

Увага! Якщо встановлений параметр `NOINTERFACEMODE = True`, то службову видачу, потрібно передбачити перед виконанням функції `НапечататьОтчетСГашением`.

Параметри функції для формування Z-звіту:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

Приклад:

Процедура `СформироватьZОтчетCashalot()`

```
ZОтчетСформирован =  
ДрайверCashalot.НапечататьОтчетСГашением(ФНПодключеннойКассы);  
Сообщить?(ZОтчетСформирован,  
    "Z-отчет успешно сформирован.",  
    "Не удалось сформировать Z-отчет из-за ошибки: " +  
    ПолучитьОшибкуCashalot());
```

КонецПроцедуры

22. Закриття зміни

Для виконання операції закриття зміни використовується функція:

БУЛЕВО **Закри́тьСмену**(ФНПодключеннойКассы)

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

Параметри функції закриття зміни:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

Приклад:

Процедура **Закри́тьСменуCashalot()**

СменаЗакрыта = **ДрайверCashalot.Закри́тьСмену**(ФНПодключеннойКассы);

Сообщить(?(**СменаЗакрыта**,
"Смена закрыта.",
"Не удалось закрыть смену из-за ошибки: " +
ПолучитьОшибкуCashalot()));

КонецПроцедуры

23. Ручне переведення ПРРО в режим офлайн

Для ручного переведу ПРРО в режим офлайн використовується функція:

БУЛЕВО **Установи́тьОфлайнРежим**(ФНПодключеннойКассы,
АвтоматическийВыходОнлайн)

Параметри функції переведу ПРРО в режим офлайн:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
АвтоматическийВыходОнлайн	булево	Ознака автоматичного виходу в онлайн. Истина - автоматично виходити з офлайну при наявності інтернету; Ложь - виходити з офлайну лише при виклику методу ПопыткаВыйтиВОнлайнРежим або після закриття зміни.

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

В якості відповіді повертається результат виконання методу в залежності від успішності виконання ("Истина" або "Ложь").

Приклад:

Процедура ПереходВОфлайнРежим()

Офлайн = ДрайверCashalot.УстановитьОфлайнРежим(ФНПодключеннойКассы,
АвтоматическийВыходОнлайн);

Сообщить?(Офлайн,
"Выполнен переход офлайн.",
"Не удалось выполнить переход офлайн из-за ошибки: " +
ПолучитьОшибкуCashalot()));

КонецПроцедуры

24. Ручне повернення ПРРО в режим онлайн

Для ручного переводу ПРРО в режим онлайн використовується функція:

БУЛЕВО ПопыткаВыйтиВОнлайнРежим(ФНПодключеннойКассы)

Параметри функції ручного переводу ПРРО в режим онлайн:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

В якості відповіді повертається результат виконання методу в залежності від успішності виконання ("Истина" або "Ложь").

Приклад:

Процедура ПереходВОнлайнРежим()

Онлайн = ДрайверCashalot.ПопыткаВыйтиВОнлайнРежим(ФНПодключеннойКассы);

Сообщить?(Онлайн,
"Выполнен переход онлайн.",
"Не удалось выполнить переход онлайн из-за ошибки: " +
ПолучитьОшибкуCashalot()));

КонецПроцедуры

25. Синхронізація залишків по товарам

Функція призначена 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».

БУЛЕВО СинхронизироватьОстаткиПоТоварам(ФНПодключеннойКассы)

Параметри ручної синхронізації залишків:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

В якості відповіді повертається результат виконання методу в залежності від успішності виконання ("Истина" або "Ложь").

Приклад:

Процедура СинхронизироватьОстаткиПоТоварам()

Синхронизация =

ДрайверCashalot.СинхронизироватьОстаткиПоТоварам(ФНПодключеннойКассы);

Сообщить?(Синхронизация,

"Синхронизация выполнена.",

"Не удалось выполнить синхронизацию из-за ошибки: " +

ПолучитьОшибкуCashalot());

КонецПроцедуры

26. Синхронізація товарів та залишків

Функція призначена 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».

БУЛЕВО СинхронизироватьТоварыИОстатки(ФНПодключеннойКассы)

Параметри ручної синхронізації залишків:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

В якості відповіді повертається результат виконання методу в залежності від успішності виконання ("Истина" або "Ложь").

Приклад:

Процедура СинхронизироватьТоварыИОстатки()

Синхронизация = ДрайверCashalot.СинхронизироватьТоварыИОстатки(ФНПодключеннойКассы);

Сообщить?(Синхронизация,
"Синхронизация выполнена.",
"Не удалось выполнить синхронизацию из-за ошибки: " +
ПолучитьОшибкуCashalot()));

КонецПроцедуры

27. Сторнування чека продажу/повернення

Функція призначена 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.

БУЛЕВО СторнироватьЧек(ФНПодключеннойКассы)

Параметри ручної синхронізації залишків:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

Приклад:

Процедура СторнированиеЧека()

Сторно = ДрайверCashalot.СторнироватьЧек(ФНПодключеннойКассы);

Сообщить?(Сторно,
"Сторнирование выполнено.",
"Не удалось выполнить сторнирование из-за ошибки: " +
ПолучитьОшибкуCashalot()));

КонецПроцедуры

28. Відображення QR-кода зареєстрованого чека

Функція призначена для візуалізації QR-кода з посиланням для перегляду зареєстрованого чека на ресурсі ДПС.

БУЛЕВО ОтобразитьQRКодЧека (ФНПодключеннойКассы, ФНчека)

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ФНчека	строка	Локальний номер чека.

Приклад:

Процедура ОтобразитьQRКодДляЧека()

Результат = ДрайверCashalot.ОтобразитьОтобразитьQRКодЧека(ФНПодключеннойКассы, ФНЧекаДляОтображения);

Сообщить?(Результат <> Неопределено,
"QR отобразили.",
"Не удалось отобразить QR из-за ошибки: " +
ПолучитьОшибкуCashalot()));

КонецПроцедуры

29. Відправка чека покупцю за допомогою сервісів доставки

Функція 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.

БУЛЕВО SendCheckToCustomer (ФНПодключеннойКассы, ФНчека, SendingAddressType, SendingAddress)

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ФНчека	строка	Локальний номер чека.
SendingAddressType	строка	Сервіс для відправки: 1 - Viber; 2 - SMS; 3- E-mail
SendingAddress	строка	Е-mail або номер телефону одержувача в залежності від обраного типу відправки. Номер телефону необхідно зазначати з урахуванням маски +380. Приклад: cashier@cashalot.org.ua , +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:

Процедура ОтправитьЧек(Команда)

```
Результат = Компонент.SendCheckToCustomer(ФискальныйНомерКассы, FNForSending, TypeSending, AddressForSending);  
Если (Результат) Тогда  
    Сообщить("Выполнена отправка чека");  
Иначе  
    ОписаниеОшибки = "";  
    ОписаниеИнтерфейса = Компонент.ПолучитьОшибку(ОписаниеОшибки);  
    Сообщить("Помилка виконання методу SendCheckToCustomer" + Символы.ПС + "Результат виконання  
ПолучитьОшибку: " + ОписаниеИнтерфейса);  
    Сообщить("Текст, що повернув Cashalot: " + ОписаниеОшибки);  
КонецЕсли;  
КонецПроцедуры
```

30. Отримання переліку налаштованих POS-терміналів в програмі

Функція використовується для отримання переліку налаштованих пристроїв POS-терміналів в програмі Cashalot в розділі «Налаштування - Додаткове обладнання - POS-термінали». В якості параметрів передається інформація про фіскальний номер ПРРО.

БУЛЕВО GetPOSTerminalList (ФНПодключеннойКассы)

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

Приклад:

Процедура ПолучитьСписокТерминалов()

```
Результат = ДрайверCashalot.GetPOSTerminalList(ФНПодключеннойКассы);  
  
Сообщить?(Результат <> Неопределено,  
    "Список получен.",  
    "Не удалось отобразить список из-за ошибки: " +  
    ПолучитьОшибкуCashalot());  
КонецПроцедуры
```

У відповідь на виконання функції повертається інформація про підключені POS-термінали до програми з їх параметрами.

Назва параметра	Опис
ID	Ідентифікатор налаштованого POS-терміналу в програмі
IPAddress	IP адреса POS-терміналу
Port	Порт за яким працює POS-термінал
DriverPath	Шлях до драйвера(емулятора) POS-терміналу
MerchantName	Назва банку-еквайра
MerchantID	Номер банку-еквайра (мерчанта)
ComPort	COM порт за яким працює POS-термінал
DefaultDevice	Ознака пристрою за замовчуванням
NameTerminal	Назва терміналу яка встановлена в програмі

31. Встановлення пристрою POS-термінал за замовчуванням

Функція призначена для встановлення ознаки POS-терміналу пристроєм який буде використовуватись за замовчуванням при проведенні безготівкових розрахунків за допомогою встановленого параметра [USEPRGPOSTERMINAL](#) . В якості параметрів передається інформація про фіскальний номер ПРРО та ID терміналу.

БУЛЕВО SetDefaultPOSTerminal (ФНПодключеннойКассы, ID)

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ID	строка	Ідентифікатор налаштованого POS-терміналу в програмі

Приклад:

Процедура УстановитьТерминалПоУмолчанию()

Результат = ДрайверCashalot.SetDefaultPOSTerminal(ФНПодключеннойКассы, ID);

Сообщить?(Результат <> Неопределено,
"Устройство установлено.",
"Не удалось установить устройство по умолчанию из-за ошибки: " +
ПолучитьОшибкуCashalot()));

КонецПроцедуры

32. Формування звіту про реалізовані товари

Функція призначена для формування звіту про реалізовані товари. В якості параметрів передається інформація про фіскальний номер ПРРО.

БУЛЕВО PrintReportGoods (ФНПодключеннойКассы)

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

Приклад:

Процедура СформироватьОтчетПоТовару()

```
ОтчетСформирован =  
ДрайверCashalot.PrintReportGoods(ФНПодключеннойКассы);  
Сообщить?(ОтчетСформирован,  
    "Отчет успешно сформирован.",  
    "Не удалось сформировать отчет из-за ошибки: " +  
    ПолучитьОшибкуCashalot());
```

КонецПроцедуры

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



×

Звіт про реалізовані товари

☒ За товарами

☐ За групами

Господарська одиниця

"Фортифікація"

▼

Період

28.05.2024 - 28.05.2024

▼

12

Оберіть товар/послугу

...

×

Оберіть групу номенклатури

...

×

ПРРО

ПРРО №1

▼

×

Оберіть касира

▼

×

Оберіть Z-звіт

▼

×

Оберіть додаткові параметри

▼

×

Сформувати

Відмінити

33. Отримання посилання для перегляду останнього зареєстрованого чека

Функція **GetLastReceiptLink** призначена для отримання посилання на перегляд чека який був зареєстровано останнім. У відповідь повертається посилання для перегляду чека в кабінеті користувача Cashalot та на сайті ДПС. Функцію необхідно виконувати після успішного виконання функції **ЗакритиЧек**.

Приклад:

```
Відповідь = Компонент.ЗакритиЧек(ФискальныйНомерКассы,  
СуммаОплатыНаличными,  
СуммаОплатыКартой,  
СуммаОплатыКредитом,  
СуммаОплатыСертификатом,  
СуммаКОплате,  
НомерЧекаВозврата,  
НомерСозданногоЧека,  
ФискальныйНомерСозданногоЧека );  
  
Если Відповідь И ФискальныйНомерСозданногоЧека <> "" Тогда  
    Сообщить("Чек зареєстровано. ФН чека: " + ФискальныйНомерСозданногоЧека);  
  
    ПосиланняНаПереглядЧекаФСКО = "";  
    ПосиланняНаПереглядЧекаCashalot = "";  
  
    Відповідь = Компонент.GetLastReceiptLink(ПосиланняНаПереглядЧекаФСКО,  
ПосиланняНаПереглядЧекаCashalot );  
  
    Если Відповідь Тогда  
        Сообщить("ПосиланняНаПереглядЧекаФСКО: " + ПосиланняНаПереглядЧекаФСКО  
);  
        Сообщить("ПосиланняНаПереглядЧекаCashalot : " +  
ПосиланняНаПереглядЧекаCashalot );  
        КонецЕсли;  
    КонецЕсли;
```

Приклад відповіді:

Посилання на перегляд чека в кабінеті користувача:

<https://check.cashalot.org.ua/receipt?id=4172916719&fn=4000108847>

Посилання на перегляд чека в ДПС:

<https://cabinet.tax.gov.ua/cashregs/check?date=20250410&time=09:41:13&id=4172916719&sm=63.80&fn=4000108847>

34. Запит розширеної відповіді від Cashalot для останнього виконаного методу

Функція **GetLastMethodResponse** призначена для отримання розширеної відповіді від драйвера у форматі JSON. Функцію необхідно виконувати після успішного виконання функції **ЗакричьЧек**, **ОткрычьСмену** або будь якої іншої функції драйвера Cashalot.

Приклад:

```
Відповідь = Компонент.ЗакричьЧек(ФискальньйНомерКассы,  
СуммаОплатыНаличными,  
СуммаОплатыКартой,  
СуммаОплатыКредитом,  
СуммаОплатыСертификатом,  
СуммаКОплате,  
НомерЧекаВозврата,  
НомерСозданногоЧека,  
ФискальньйНомерСозданногоЧека );  
  
Если Відповідь И ФискальньйНомерСозданногоЧека <> "" Тогда  
    Сообщить("Чек зареєстрован. ФН чека: " + ФискальньйНомерСозданногоЧека);  
  
    ВідповідьДляОстанньогоМетоду = "";  
  
    Відповідь = Компонент.GetLastMethodResponse(ВідповідьДляОстанньогоМетоду);  
  
    Если Відповідь Тогда  
        Сообщить("Відповідь для останнього методу Cashalot: " +  
ВідповідьДляОстанньогоМетоду);  
        КонецЕсли;  
    КонецЕсли;
```

У відповіді на **GetLastMethodResponse** повертається структура **JSON**, що містить:

Ret	bool	Ознака успішності виконання.
ErrorString	string	Опис помилки за наявності
Values	string	JSON-структура, що містить значення успішності операції, тип документу, помилки, локальний та фіскальний номери чека, ID зміни та Base64 рядок XML чека.
OfflineMode	bool	Ознака перебування в режимі офлайн
ReceiptFiscalNum	string	Фіскальний номер чека.
ReceiptLocalNum	string	Локальний номер чека.
ShiftID	string	Ідентифікаційний номер поточної зміни.
FSKORceiptLink	string	Посилання на перегляд чека в ДПС
CashalotReceiptLink	string	Посилання на перегляд чека в кабінеті Cashalot
ParentReceiptID	string	ID чека який повертається
ParentFiscalNumber	string	Фіскальний номер чека який повертається
Type	int	Тип документа.
Base64Str1251ReceiptXML	string	Base 64 рядок XML-структури фіскалізованого чека.

Base64ReceiptUpeg	string	JPEG структура фіскалізованого чека у форматі base64
Base64ReceiptPdf	string	PDF структура фіскалізованого чека у форматі base64
LicEndDate	string	Дата завершення терміну дії ліцензії ПРРО у форматі (pppp.мм.дд гг.хв.сс)
CertEndDate	string	Дата завершення терміну дії сертифікату касира у форматі (pppp.мм.дд 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 1 - Z-звіт 2 - X-звіт
<Тип документа>	3	000 - Чек реалізації товарів/послуг 001 - Чек переказу коштів 002 - Чек операції обміну валюти 003 - Чек видачі готівки 100 - Відкриття зміни 101 - Закриття зміни 102 - Початок офлайн сесії 103 - Завершення офлайн сесії
<Додатковий тип документа>	3	000 - Касовий чек (реалізація) 001 - Видатковий чек (повернення) 002 - Чек операції «службове внесення»/«отримання авансу» 003 - Чек операції «отримання підкріплення» 004 - Чек операції «службова видача»/«інкасація» 005 - Чек сторнування попереднього чека

35. Створення відкладеного чека продажу товарів/послуг

Для створення відкладеного чека продажу товарів/послуг необхідно виконати та передати ідентичний ланцюжок функцій як і при [реєстрації фіскального чека](#). Виключенням є ознака параметра `ПризнакФискальногоЧека` в функції `Открыть чек` яку необхідно встановити як `Ложь` (False). Створений відкладений чек зберігається на необмежений термін.

Приклад:

Процедура СтворитиВідкладенийЧек(Команда)

```
СформироватьЧекВозврата = Ложь;

ОткрытьЧек(Компонент, Ложь);

Если СформироватьXML = Истина Тогда
    НапечататьФискСтрокуМетод(Компонент);
Иначе
    НапечататьФискальнуюСтрокуМетод(Компонент);
КонецЕсли;

ДобавитьКомментарий(Компонент);

ВидОплаты = ВидиСобственныхОплат.Количество();
Если ВклИнфуТерминал = Истина или ВидОплаты <> 0 или СуммаБезналичнойОплаты <> ""
Тогда
    УстановитьРасширенныйПорядокОплатыЧекаCashalot(Компонент);
КонецЕсли;

ЗакрытьЧек(Компонент);

ReceiptID = "";
ВідповідьДляОстанньогоМетоду = "";
Ответ = Компонент.GetLastMethodResponse(ВідповідьДляОстанньогоМетоду);
Если Ответ Тогда
    Сообщить("Відповідь для методу створення відкладеного чека: " +
ВідповідьДляОстанньогоМетоду);
    ReceiptID = ПолучитьReceiptID(ВідповідьДляОстанньогоМетоду) ;
КонецЕсли;

КонецПроцедуры
```

У відповіді повертається структура **JSON**, що містить:

Ret	bool	Ознака успішності виконання.
ErrorString	string	Опис помилки за наявності
Type	string	Тип документа.
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)
ExpressWaybillNumber	string	Номер експрес-накладної (ТТН)
StatusTTN	string	Числовий статус експрес-накладної
StatusTTNDescription	string	Статус експрес-накладної

StatusTTNCanFiscalize	string	Ознака набуття чека фіналізованого статусу доставки
ReceiptPrePayOrderNum	string	Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати)

Наявна підтримка 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](#).

36. Фіскалізація відкладеного чеку продажу товарів/послуг

Функція **FiscalizePostponedReceipt** використовується для фіскалізації раніше створеного відкладеного чека. Вхідним параметром методу є ReceiptID, що повертається створенні відкладеного чека.

БУЛЕВО **FiscalizePostponedReceipt**(ФНПодключеннойКассы, ReceiptID)

Параметри::

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ReceiptID	строка	ID (guid) чека
OtherParam	строка	Зарезервований параметр (може містити null)

Приклад:

Процедура ФіскалізуватиВідкладенийЧек(Команда)

```

ВідповідьДляОстанньогоМетоду = "";
Ответ = Компонент.FiscalizePostponedReceipt(ФискальныНомерКассы, ReceiptID,"",
ВідповідьДляОстанньогоМетоду);
Если Ответ Тогда
    Сообщить("Відповідь для методу фіскалізації відкладеного чека: " +
ВідповідьДляОстанньогоМетоду);
КонецЕсли;
КонецПроцедуры

```

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

Ret	bool	Ознака успішності виконання.
ErrorString	string	Опис помилки за наявності
ShiftID	string	Ідентифікаційний номер поточної зміни.
ReceiptFiscalNumber	string	Фіскальний номер чека.
ReceiptLocalNumber	string	Локальний номер чека.
OfflineMode	string	Ознака перебування в режимі офлайн
Type	string	Тип документа.
ReceiptID	string	ID (guid) чека
ReceiptSum	string	Сума до сплати по чеку
ReceiptDateTime	string	Дата та час створення у форматі (дд.мм.рррр :хв:сс)
FSKORceiptLink	string	Посилання на перегляд чека в ДПС

CashalotReceiptLink	string	Посилання на перегляд чека в кабінеті Cashalot
ExpressWaybillNumber	string	Номер експрес-накладної (ТТН)
StatusTTN	string	Числовий статус експрес-накладної
StatusTTNDescription	string	Статус експрес-накладної
StatusTTNCanFiscalize	string	Ознака набуття чека фіналізованого статусу доставки
ReceiptPrePayOrderNum	string	Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати)
Base64Str1251ReceiptXML	string	XML-структура фіскалізованого чека у форматі base64

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

37. Отримати інформацію про відкладений чек за його ReceiptID

Функція **GetPostponedReceiptByID** призначена 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».

БУЛЕВО **GetPostponedReceiptByID**(ФНПодключеннойКассы, ReceiptID)

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ReceiptID	строка	ID (guid) чека

Приклад:

```

Процедура ОтриматиВідкладенийЧекПоІД(Команда)
    ВідповідьДляОстанньогоМетоду = "";
    Ответ = Компонент.GetPostponedReceiptByID(ФискальныйНомерКассы, ReceiptID,
    ВідповідьДляОстанньогоМетоду);
    Если Ответ Тогда

        Сообщить("Відповідь для методу отримати GetPostponedReceiptByID: " +
    ВідповідьДляОстанньогоМетоду);

    КонецЕсли;
КонецПроцедуры

```

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

Ret	bool	Ознака успішності виконання.
ErrorMessage	string	Опис помилки за наявності
Type	string	Тип документа.
ReceiptID	string	ID (guid) чека
ReceiptSum	string	Сума до сплати по чеку
ReceiptDateTime	string	Дата та час створення у форматі (дд.мм.рррр :хв:сс)
ExpressWaybillNumber	string	Номер експрес-накладної (ТТН)
StatusTTN	string	Числовий статус експрес-накладної
StatusTTNDescription	string	Статус експрес-накладної
StatusTTNCanFiscalize	string	Ознака набуття чека фіналізованого статусу доставки
ReceiptPrePayOrderNum	string	Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати)

38. Отримати інформацію про відкладений чек за його номером ЕН (експрес-накладною)

Функція **GetPostponedReceiptByWaybillNumber** призначена 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»

БУЛЕВО **GetPostponedReceiptByWaybillNumber** (ФНПодключеннойКассы, ExpressWaybillNumber)

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ExpressWaybillNumber	строка	Номер експрес-накладної (ТТН)

Приклад:

Процедура ОтриматиВідкладенийЧекПоТТН(Команда)

```
ВідповідьДляОстанньогоМетоду = "";  
Ответ = Компонент.GetPostponedReceiptByWaybillNumber(ФискальныНомерКассы,  
PostponedReceiptTTN , ВідповідьДляОстанньогоМетоду);  
Если Ответ Тогда  
  
    Сообщить("Відповідь для методу отримати ОтриматиВідкладенийЧекПоТТН: " +  
ВідповідьДляОстанньогоМетоду);  
  
КонецЕсли;  
КонецПроцедуры
```

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

Ret	bool	Ознака успішності виконання.
ErrorMessage	string	Опис помилки за наявності
Type	string	Тип документа.
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)
ExpressWaybillNumber	string	Номер експрес-накладної (ТТН)
StatusTTN	string	Числовий статус експрес-накладної
StatusTTNDescription	string	Статус експрес-накладної
StatusTTNCanFiscalize	string	Ознака набуття чека фіналізованого статусу доставки
ReceiptPrePayOrderNum	string	Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати)

39. Отримати інформацію про перший відкладений чек

Функцію **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.

БУЛЕВО **GetFirstPostponedReceipt** (ФНПодключеннойКассы, **DateBeg**, **DateEnd**)

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
DateBeg	строка	Початок періоду в форматі dd.MM.yyyy.
DateEnd	строка	Кінець періоду в форматі dd.MM.yyyy.

У відповіді на **GetFirstPostponedReceipt** повертається структура **JSON**, що містить:

Ret	bool	Ознака успішності виконання.
ErrorMessage	string	Опис помилки за наявності
Type	string	Тип документа.
ReceiptID	string	ID (guid) чека
ReceiptSum	string	Сума до сплати по чеку
ReceiptDateTime	string	Дата та час створення у форматі (дд.мм.рррр :хв:сс)
ExpressWaybillNumber	string	Номер експрес-накладної (ТТН)
StatusTTN	string	Числовий статус експрес-накладної
StatusTTNDescription	string	Статус експрес-накладної
StatusTTNCanFiscalize	string	Ознака набуття чека фіналізованого статусу доставки

ReceiptPrePayOrderNum	string	Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати)
-----------------------	--------	--

40. Отримати інформацію наступний відкладений чек

Функція **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.

БУЛЕВО **GetNextPostponedReceipt** (ФНПодключеннойКассы)

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.

Приклад виконання **GetNextPostponedReceipt** + **GetNextPostponedReceipt**:

```

Процедура ОтриматиСписокВідкладенихЧеків(Команда)
    ВідкладеніЧеки.Очистить();
    ВідповідьДляОстанньогоМетоду = "";
    Ответ = Компонент.GetFirstPostponedReceipt(ФискальныНомерКассы, "01.09.2025", "",
    ВідповідьДляОстанньогоМетоду);
    Если Ответ Тогда
        Сообщить("Відповідь для методу отримати перший відкладений чек: " +
    ВідповідьДляОстанньогоМетоду);
        ПолучитьPostponedReceiptPrm(ВідповідьДляОстанньогоМетоду);

        Пока Компонент.GetNextPostponedReceipt(ФискальныНомерКассы,
    ВідповідьДляОстанньогоМетоду) Цикл

            Сообщить("Відповідь для методу отримати наступний відкладений чек: " +
    ВідповідьДляОстанньогоМетоду);
            ПолучитьPostponedReceiptPrm(ВідповідьДляОстанньогоМетоду);
        КонецЦикла;

    КонецЕсли;
КонецПроцедуры

```

У відповіді на [GetNextPostponedReceipt](#) повертається структура **JSON**, що містить:

Ret	bool	Ознака успішності виконання.
ErrorMessage	string	Опис помилки за наявності
Type	string	Тип документа.
ReceiptID	string	ID (guid) чека
ReceiptSum	string	Сума до сплати по чеку
ReceiptDateTime	string	Дата та час створення у форматі (дд.мм.рррр :хв:сс)
ExpressWaybillNumber	string	Номер експрес-накладної (ТТН)
StatusTTN	string	Числовий статус експрес-накладної
StatusTTNDescription	string	Статус експрес-накладної
StatusTTNCanFiscalize	string	Ознака набуття чека фіналізованого статусу доставки
ReceiptPrePayOrderNum	string	Номер інтернет-замовлення (для чеків інтернет-продажу або передплати/післяплати)

41. Відображення візуальної форми чека за його ReceiptID

Функція [ShowReceiptByID](#) призначена для отримання візуальної форми чека за його ReceiptID.

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

БУЛЕВО [ShowReceiptByID](#) (ФНПодключеннойКассы, ReceiptID)

Параметри:

Ім'я	Тип	Опис
ФНПодключеннойКассы	строка	Фіскальний номер ПРРО. Ідентифікатор пристрою.
ReceiptID	строка	ID (guid) чека

Приклад:

Процедура ПоказатиЧекПоID(Команда)

Если Компонент.[ShowReceiptByID](#)(ФискальныйНомерКассы, [ReceiptID](#)) Тогда

Сообщить(" Чек відкрито");

Иначе

Сообщить(" Щось пішло не так...");

КонецЕсли;

КонецПроцедуры



ТОВ "Фортеця"

"Фортифікація"

місто Київ, вул. Сталева, буд.1

ПН 44444455555

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

42. 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). У деяких специфічних випадках дозволяється реєстрація чека за умови, якщо він не набув фінального статусу ЕН (ТТН)

Драйвер Cashalot для BAS надає інструмент у вигляді набору функцій, 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 ідентичний ланцюжок функції як і при реєстрації фіскального чека за виключенням параметра **ПризнакФискальногоЧека** в функції **Открыть чек** ознаку якого необхідно встановити як ЛОЖЬ (False).

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

- Встановити параметр **ТипОплати** = 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 функції **GetPostponedReceiptByID** та **GetPostponedReceiptByWaybillNumber**, які приймають відповідно параметри **ReceiptID** або **ExpressWaybillNumber**. Якщо ці параметри відповідають чеку, створеному з типом «Інтернет-продаж» (**ТипОплати = 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» (**ТипОплати = 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 **СуммаПредоплаты**) 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 я SumPrepare Payed	Можливість реєстрації якщо чек має StatusTTNCanFisc alize = false	Можливість реєстрації якщо чек має StatusTTNCanFiscali ze = true
-----------------------	---------------------	---------------------------------------	---	--

100% післяплата (накладений платіж)	Вид розрахунку, за яким покупець буде здійснювати оплату за товар службі доставки (експедитору) лише після огляду та отримання	Параметр не заповнено	Ні	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

43. Обробка JSON відповіді

Певна частина наявних функцій драйвера Cashalot для BAS повертає у відповідь JSON структуру з переліком певних параметрів. Для парсингу такої структури можливо використовувати наступний приклад функції.

Приклад:

Функція `ОтриматиJSONResponse(ТекстJSON)`

Попытка

// Очищення HTML-екранування

`ТекстJSON = СтрЗаменить(ТекстJSON, "&", "&");`

`ТекстJSON = СтрЗаменить(ТекстJSON, """, "\"");`

// Видалення символів переносу

`ТекстJSON = СтрЗаменить(ТекстJSON, Символ(13), "");`

`ТекстJSON = СтрЗаменить(ТекстJSON, Символ(10), "");`

// Перевірка: показати очищений JSON

`Сообщить("Очищенный JSON: " + ТекстJSON);`

// Парсинг

`Чтение = Новый ЧтениеJSON;`

`Чтение.УстановитьСтроку(ТекстJSON);`

`Возврат ПрочитатьJSON(Чтение);`

Исключение

`Сообщить("Помилка: " + ОписаниеОшибки());`

`Возврат Неопределено;`

КонецПопытки;

КонецФункции

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

Чек зі способом оплати «Продаж» (ТипОплаты=0)

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

Пояснення: В функції **УстановитьРасширенныйПорядОплатыЧека** необхідно встановити значення 0 для параметра ТипОплаты. В функції **ЗакрытьЧек** необхідно передати значення 100грн використовуючи параметр **СуммаКОплате.** В функції **УстановитьРасширенныйПорядОплатыПоЧека** транзакції за допомогою параметра **ДопРеквизиты** передається інформація у вигляді XML про реквізити транзакції POS-термінала, суму безготівкової оплати та назвою засобу.

Приклад (спрощений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record TerminalID="t123456" ApprovalCode="32456"
RRN="987654321" IssuerName="Visa Credit" PAN="XXXXXXXXXXXX4455"
TransactionDate="16.02.2023 15:07:45" AcquireName="ПриватБанк" InvoiceNumber="555777" ParentRRN=""
SignVerif="0"/>
  <Record SumPayByBank="100" />
  <Record BankPayToolsNM="Платіж LIQPAY" />
  <Record UseTransactionDetailsToSumPayByBank ="True" />
</Table>
```

Приклад (розширений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record TerminalID="t123456" ApprovalCode="32456"
RRN="987654321" IssuerName="Visa Credit" PAN="XXXXXXXXXXXX4455"
TransactionDate="16.02.2023 15:07:45" AcquireName="ПриватБанк" InvoiceNumber="555777" ParentRRN=""
SignVerif="0"/>
  <Record PAYFORMCD="202" PAYFORMSUM="100" PAYFORMNM ="Платіж LIQPAY"
PAYFORMUSETRANSACTIONDETAILS="TRUE" />
</Table>
```

ТОВ "Фортеця"
"Фортифікація"
місто Київ, вул. Сталева, буд.1
ПН 44444455555
ФН ПРРО 4000024021
КАСИР Сталевий І. І.
Чек №1519122

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

БЕЗГОТІВКОВА 100.00 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» (ТипОплати=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 функції **УстановитьРасширенныйПорядОплатыЧека** необхідно встановити значення 1 для параметра ТипОплати. В функції **ЗакретьЧек** необхідно передати значення 50 грн використовуючи параметр **СуммаКОплате**. В функції **УстановитьРасширенныйПорядОплатыПоЧека** необхідно для параметра **СуммаПредоплаты** передати значення 50грн. За допомогою параметра **ДопРеквизиты** передається інформація у вигляді ХМЛ про реквізити транзакції POS-терміналу, суму безготівкової оплати та назвою засобу.

Приклад (спрощений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record TerminalID="t123456" ApprovalCode="32456"
  RRN="987654321" IssuerName="Visa Credit" PAN="XXXXXXXXXXXX4455"
  TransactionDate="16.02.2023 15:07:45" AcquireName="ПриватБанк" InvoiceNumber="555777" ParentRRN=""
  SignVerif="0"/>
  <Record SumPayByBank="50" />
  <Record BankPayToolsNM="Платіж LIQPAY" />
  <Record UseTransactionDetailsToSumPayByBank ="True" />
</Table>
```

Приклад (розширений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record TerminalID="t123456" ApprovalCode="32456"
  RRN="987654321" IssuerName="Visa Credit" PAN="XXXXXXXXXXXX4455"
  TransactionDate="16.02.2023 15:07:45" AcquireName="ПриватБанк" InvoiceNumber="555777" ParentRRN=""
  SignVerif="0"/>
  <Record PAYFORMCD="202" PAYFORMSUM="50" PAYFORMNM ="Платіж LIQPAY"
  PAYFORMUSETRANSACTIONDETAILS="TRUE" />
</Table>
```

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 Соса-Cola сильногазований
0.5л
1,000 X 100,00 100,00 А

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» (ТипОплаты=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 функції **УстановитьРасширенныйПорядОплатыЧека** необхідно встановити значення 2 для параметра ТипОплаты. Для параметра **ФискальныйНомерЧекаПредоплатыИлиВозврата** вказати ФН чека передплати. За допомогою параметра **ДопРеквизиты** передається інформація у вигляді ХМЛ про реквізити транзакції POS-термінала, суму безготівкової оплати та назвою засобу.

Приклад (спрощений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record TerminalID="t123456" ApprovalCode="32456"
RRN="987654321" IssuerName="Visa Credit" PAN="XXXXXXXXXXXX4455"
TransactionDate="16.02.2023 15:07:45" AcquireName="ПриватБанк" InvoiceNumber="555777" ParentRRN=""
SignVerif="0"/>
  <Record SumPayByBank="50" />
  <Record BankPayToolsNM="Платіж LIQPAY" />
  <Record UseTransactionDetailsToSumPayByBank="True" />
</Table>
```

Приклад (розширений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record TerminalID="t123456" ApprovalCode="32456"
RRN="987654321" IssuerName="Visa Credit" PAN="XXXXXXXXXXXX4455"
TransactionDate="16.02.2023 15:07:45" AcquireName="ПриватБанк" InvoiceNumber="555777" ParentRRN=""
SignVerif="0"/>
  <Record PAYFORMCD="202" PAYFORMSUM="50" PAYFORMNM="Платіж LIQPAY"
PAYFORMUSETRANSACTIONDETAILS="TRUE" />
</Table>
```

ТОВ "Фортеця"
"Фортифікація"
місто Київ, вул. Сталева, буд.1
ПН 444444555555
ФН ПРРО 4000024021
КАСІР 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 А=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» (ТипОплати=3)

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

Пояснення: В функції **УстановитьРасширенныйПорядОплатыЧека** необхідно встановити значення 3 для параметра ТипОплати. В функції **УстановитьРасширенныйПорядОплатыПоЧека** необхідно для параметра **СуммаПредоплаты** передати значення 100грн. За допомогою параметра **ДопРеквизиты** передається інформація у вигляді XML про реквізити транзакції POS-термінала та назвою засобу оплати для "Передплати" по чеку.

Приклад (спрощений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record PrePaymentPayToolsNM="Власна назва" />
</Table>
```

Для розширеного варіанту необхідно для реквізитів **PayFormCd** вказати значення 2, а для **PayFormSum** значення суми яке відповідає сумі оплати по чеку. Реквізит **PayFormNM** має відповідати значенню найменування засобу оплати. Параметр [СуммаПредоплаты](#) методу **УстановитьРасширенныйПорядокОплатыЧека** має містити однакову суму зі значенням в реквізиті **PayFormSum**. Використання реквізита **PayFormUseTransactionDetails** для чеків зі способом оплати 3 та 4 не допускається

Приклад (розширений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record PAYFORMCD="2" PAYFORMSUM="100" PAYFORMNM ="Власна назва"
</Table>
```

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 100,00 100,00 А

СУМА 100.00 ГРН
ПДВ А=20% 16,67

БЕЗГОТІВКОВА Передоплата
Власна назва 100.00 ГРН

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: В функції **УстановитьРасширенныйПорядОплатыЧека** необхідно встановити значення 3 для параметра ТипОплати. В функції **УстановитьРасширенныйПорядОплатыПоЧека** необхідно для параметра **СуммаПредоплаты** передати значення 50грн або ту суму яка має була отримана по передплаті. Різниця суми буде відображена в післяплаті. За допомогою параметра **ДопРеквизиты** передається інформація у вигляді ХМЛ про реквізити транзакції POS-термінала та назвою засобу оплати для "Передплати" та "Післяплати" по чеку.

Приклад (спрощений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record PrePaymentPayToolsNM="Власна назва №1" />
  <Record AfterPaymentPayToolsNM="Власна назва №2" />
/>
</Table>
```

Для розширеного варіанту необхідно застосувати два рядки <Record>. Перший рядок має містити реквізит **PayFormCD** заповнений значенням 2, а для **PayFormSum** значення суми яке відповідає [сумі передплати](#) заповненої в методі **УстановитьРасширенныйПорядокОплатыЧека** . Другий рядок <Record> має містити заповнений реквізит **PayFormCD** зі значенням 201 та **PayFormSum** зі значенням суми яка залишається по чеку.

Приклад (розширений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record PAYFORMCD="2" PAYFORMSUM="50" PAYFORMNM="Власна назва №1"
  <Record PAYFORMCD="201" PAYFORMSUM="50" PAYFORMNM="Власна назва №2"
</Table>
```

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 Соса-Сола сильногазований
0.5л

1,000 X 100,00 100,00 A

СУМА 100.00 ГРН

ПДВ А=20% 16,67

БЕЗГОТІВКОВА Передоплата
Власна назва №2 50.00 ГРН

БЕЗГОТІВКОВА Післяплата
Власна назва №1 50.00 ГРН

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

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

Пояснення: В функції **УстановитьРасширенныйПорядОплатыЧека** необхідно встановити значення 3 для параметра **ТипОплати**. За допомогою параметра **ДопРеквизиты** передається інформація у вигляді XML про реквізити транзакції POS-термінала та назвою засобу оплати "Післяплати" по чеку. Параметр [СуммаПредоплаты](#) методу **УстановитьРасширенныйПорядокОплатыЧека** можливо не заповнювати.

Приклад (спрощений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record AfterPaymentPayToolsNM="Власна назва №2" />
</Table>
```

Для розширеного варіанту необхідно для реквізитів **PayFormCD** вказати значення 201, а для **PayFormSum** значення суми яке відповідає сумі оплати по чеку.

Приклад (розширений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record PAYFORMCD="201" PAYFORMSUM="100" PAYFORMNM ="Власна назва №2"
</Table>
```

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 Соса-Cola сильногазований
0.5л

1,000 X 100,00 100,00 A

СУМА 100.00 ГРН

ПДВ А=20% 16,67

БЕЗГОТІВКОВА Післяплата
Власна назва №1 **100.00 ГРН**

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

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

Пояснення: В функції **УстановитьРасширенныйПорядОплатыЧека** необхідно встановити значення 4 для параметра **ТипОплати**. За допомогою параметра **ДопРеквизиты** передається інформація у вигляді XML про реквізити транзакції POS-термінала та назвою засобу оплати "Післяплати" по чеку. Параметр [СуммаПредоплаты](#) методу **УстановитьРасширенныйПорядокОплатыЧека** не заповнюється.

Приклад:

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record AfterPaymentPayToolsNM="Власна назва №3" />
</Table>
```

Для розширеного варіанту необхідно для реквізитів **PayFormCD** вказати значення 201, а для **PayFormSum** значення суми яке відповідає сумі оплати по чеку.

Приклад (розширений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record PAYFORMCD="201" PAYFORMSUM="100" PAYFORMNM="Власна назва №2" />
</Table>
```

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 Соса-Сола сильногазований
0.5л

1,000 X 100,00	100,00 A
----------------	----------

СУМА	100.00 ГРН
------	------------

ПДВ А=20%	16,67
-----------	-------

БЕЗГОТІВКОВА	Післяплата
Власна назва №3	100.00 ГРН

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

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

Пояснення: В функції **УстановитьРасширенныйПорядОплатыЧека** необхідно встановити значення 5 для параметра ТипОплаты. В функції **ЗаккрытьЧек** необхідно передати значення 50 грн використовуючи параметр **СуммаКОплате.** В функції **УстановитьРасширенныйПорядОплатыПоЧека** необхідно для параметра СуммаПредоплаты передати значення 50грн. За допомогою параметра **ДопРеквизиты** передається інформація у вигляді XML про реквізити транзакції POS-терміналу та назвою засобу по чеку.

Приклад:

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record SumPayByBank="50" />
  <Record BankPayToolsNM="Інтернет-еквайринг" />
</Table>
```

Приклад (розширений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record PAYFORMCD="202" PAYFORMSUM="50" PAYFORMNM="Інтернет-еквайринг" />
</Table>
```

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» (ТипОплати=6)

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

Пояснення: В функції **УстановитьРасширенныйПорядОплатыЧека** необхідно встановити значення 6 для параметра ТипОплати. В функції **ЗакретьЧек** необхідно передати значення 25 грн використовуючи параметр **СуммаКОплате**. В функції **УстановитьРасширенныйПорядОплатыПоЧека** необхідно для параметра СуммаПредоплаты передати значення 25грн та заповнити ФН чека передплати за допомогою параметра **ФискальныйНомерЧекаПредоплатыИлиВозврата**. За допомогою параметра **ДопРеквизиты** передається інформація у вигляді XML про реквізити транзакції POS-термінала та назвою засобу по чеку

Приклад:

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record SumPayByBank="25" />
  <Record BankPayToolsNM="Інтернет-еквайринг" />
/>
</Table>
```

Приклад (розширений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record PAYFORMCD="202" PAYFORMSUM="25" PAYFORMNM="Інтернет-еквайринг" />
</Table>
```

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 A

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» (ТипОплати=7)

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

Пояснення: В функції **УстановитьРасширенныйПорядОплатыЧека** необхідно встановити значення 7 для параметра ТипОплати. В функції **УстановитьРасширенныйПорядОплатыПоЧека** необхідно заповнити ФН чека передплати за допомогою параметра **ФискальныйНомерЧекаПредоплатыИлиВозврата**. За допомогою параметра **ДопРеквизиты** передається інформація у вигляді XML про реквізити транзакції POS-термінала та назвою засобу по чеку.

Приклад:

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record SumPayByBank="25" />
  <Record BankPayToolsNM="Інтернет-еквайринг" />
</Table>
```

Приклад (розширений):

```
<?xml version="1.0" encoding="UTF-8"?>
<Table>
  <Record PAYFORMCD="202" PAYFORMSUM="25" PAYFORMNM="Інтернет-еквайринг" />
</Table>
```

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 Coca-Cola сильногазований
0.5л

1,000 X 100,00 100,00 А

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