

# Инструкции по доработке конфигурации

## Возможные способы доработки конфигурации:

1. пользовательские раскладки клавиатуры
2. [пользовательские кнопки в форме супервизор](#)
3. [расширенные отчеты](#)
4. модуль логики
5. изменение конфигурации и типовых внешних обработок

Способ №1 (использование собственных раскладок клавиатуры) рекомендуется использовать когда клиенту нужна функция (кнопка или горячая клавиша), которая выполнит типовые действия, которые отсутствуют в предопределенных раскладках клавиатуры. Например, применить определенную (или из списка) дисконтную карту, товар, продавца и т.п. Т.е. использовать набор функций формы регистрации.

Способ №2 используется когда нужно добавить дополнительную кнопку в супервизор.

Способ №3: Расширенные отчеты используются для вывода какой-то информации на экран или принтер, могут вызываться пользователем в произвольный момент времени и, опционально, автоматически при снятии отчета с гашением

Способ №4 решает большинство задач без необходимости доработок конфигурации и типовых обработок с максимальной совместимостью при обновлении конфигурации.

Вариант №5 крайне не рекомендуется использовать по следующим причинам:

- Изменение базовой версии конфигурации невозможно.
- При каждом обновлении конфигурации потребуются переносить изменения вручную.
- В новой версии конфигурации возможно изменение кода, которое потребует дополнительное время на актуализацию доработок.

## Основные реквизиты, функции, процедуры,

При доработке конфигурации рекомендуется использовать типовые методы, которые можно найти в коде конфигурации.

Описание методов в защищенной компоненты описаны в статье [API компоненты AddIn.CashContext](#).

Наиболее часто используемые методы описаны ниже.

Формат описания методов:

Имя метода - описание метода.

Параметры:

Номер параметра: Тип параметра (значение по-умолчанию) - описание параметра.

Пример кода.



```
УИД_Формы = Новый УникальныйИдентификатор();
ФормаВыбораДаты = МенеджерОбъектов.СоздатьОбщуюФорму("ОбщаяФорма.ВыборДаты",
УИД_Формы);
Результат = Неопределено;
ДатаРождения = Дата(2000, 1, 1);

Если ФормаВыбораДаты.ВыбратьДату(Результат, ДатаРождения) Тогда
    ДатаРождения = Результат;
КонецЕсли;

МенеджерОбъектов.ВыгрузитьОбщийОбъект("ОбщаяФорма.ВыборДаты", УИД_Формы);
```

## Работа с составом чека

Как получить форму регистрации в коде:

```
ФормаРегистрации = МенеджерОбъектов.ПолучитьОбщийОбъект("Регистрация.Форма");
```

Как получить состав чека формы регистрации:

```
ФормаРегистрации = МенеджерОбъектов.ПолучитьОбщийОбъект("Регистрация.Форма");
КоличествоСтрок = ФормаРегистрации.СоставЧека.Количество();
```

Как получить выделенную строку состава чека:

```
ФормаРегистрации = МенеджерОбъектов.ПолучитьОбщийОбъект("Регистрация.Форма");
ТекущаяСтрокаСоставаЧека = ФормаРегистрации.элтСоставЧека.ТекущаяСтрока;
```

Как добавить товар в состав чека по коду или артикулу (для добавления по артикулу нужно в 2-й параметр передать «Артикул»):

```
ФормаРегистрации = МенеджерОбъектов.ПолучитьОбщийОбъект("Регистрация.Форма");
ЗК = МенеджерОбъектов.ПолучитьОбщийОбъект ("AddIn.CashContext" );
ЗК.ДобавитьПозициюПоКодуАртикулу(100, "Код", "", -1, ФормаРегистрации);// добавить товар
с кодом 100, цена по-умолчанию (из БД)
ЗК.ДобавитьПозициюПоКодуАртикулу(200, "Код", "", 500, ФормаРегистрации);// добавить
товар с кодом 200 с ценой 500
ЗК.ДобавитьПозициюПоКодуАртикулу(300, "Код", "12345", -1, ФормаРегистрации);// добавить
единицу с кодом "12345" товара с кодом 300, цена по-умолчанию
```

Как добавить товар в состав чека по штрихкоду:

```
ФормаРегистрации = МенеджерОбъектов.ПолучитьОбщийОбъект("Регистрация.Форма");
ЗК = МенеджерОбъектов.ПолучитьОбщийОбъект ("AddIn.CashContext" );
ЗК.ДобавитьПозициюПоШтрихкоду("12345", -1, ФормаРегистрации, Ложь, Ложь);
ЗК.ДобавитьПозициюПоШтрихкоду("12345",// штрихкод
    -1, // цена (-1 = цена из БД)
    ФормаРегистрации,
```



```
//<CUT> - отрезка
ПС = Символ(182);
Текст = "<CENTER>Строка по центру" + ПС +
        "Строка слева" + ПС +
        "<RIGHT>Строка справа" + ПС +
        "<HR>*</HR>" + ПС +
        "<s>1234567890123<s>" + ПС +
        "<s2d>http://host<s2d>" + ПС +
        " " + ПС +
        " " + ПС +
        "<CUT>";
ЗК = МенеджерОбъектов.ПолучитьОбщийОбъект ("AddIn.CashContext");
МассивДляПечати = ЗК.КонвертироватьВМассивДляПечати(Текст);
СтрокаККМ = Кассир5_DataAccess.ПолучитьККМ();
РезультатПечати_бул = ЗК.FnККМ_PrintStringBlock(СтрокаККМ, МассивДляПечати);
```

## Работа со справочниками транзакций

В Кассир 5 есть два типа справочников с транзакциями:

1. Справочник Временные транзакции - хранит информацию о текущем чеке, при закрытии (или отмене) чека переносятся в справочник Транзакции и удаляются.
2. Справочник Транзакции - хранит информацию об основных действиях кассира, закрытых или отмененных чеках и пр.

Справочник выгружается в файл отчета по [формату Кассир 5](#).

Элементы удаляется только автоматически по регламенту.

## Работа со справочником Временные транзакции

Перед работой с временными транзакциями рекомендуется ознакомиться со статьей [Таблица временных транзакций кассира](#).

Если нужно записать произвольную транзакцию, которая относится к текущему чеку, тогда нужно записать временную транзакцию.

Все функции для работы с временными транзакциями находятся в модуле обработки Кассир5\_DataAccess.

**ПолучитьВремТранзакцию** - ф-я получает информация о временной транзакции по коду.

Если врем. транзакция не найдена, тогда в возвращенной структуре в свойство «Пустая» будет установлено значение Истина.

Параметры:

- 1: Число - код временной транзакции для поиска.

```
Транз = Кассир5_DataAccess.ПолучитьВремТранзакцию(100); // получить врем. транзакцию с кодом 100
```

**ПолучитьВремТранзакцииПоТипу** - ф-я выполняет отбор временных транзакций по типу (поле ТипТранзакции), тип возвращаемого значения Таблица значений.

Транзакции11 = Кассир5\_DataAccess.ПолучитьВремТранзакцииПоТипу(11);// в переменную Транзакции11 установится таблица со всеми транзакциями с типом 11

**ПисатьВремТранзакцию** - ф-я создаёт новый элемент справочника Временные транзакции. Параметры:

- 1: Число - код транзакции (используется для уникальности врем. транзакций и определяет порядок записи в справочник Транзакции). Если передано значение «0», тогда установится текущий максимальный код + 1.
- 2: Число - поле Тип транзакции
- 3: Дата (Неопределено) - поле Дата транзакции (если не указана - подставляется текущая дата)
- 4: Число - поле Секция
- 5: Строка (до 150 символов) - поле Код товара (поле используется для записи строковых данных)
- 6: Число - поле Цена
- 7: Число - поле Количество
- 8: Число - поле Сумма
- 9: Число - номер ККМ (определяет к какому ККМ относится транзакция, если задать несуществующий номер ККМ - транзакция не будет перенесена в справочник Транзакции)
- 10: Строка («»)- поле Скидка, поле используется для оперативной работы с длинными строками, в справочник Транзакции не переносится

Пример записи врем. транзакции с типом 777 на чек, с явным указанием кода:

```
КодВремТранзакции = 900000;
// запишем транзакцию на чек, если транзакция с переданным кодом уже существует, тогда
будет выполнено изменение транзакции
// дата транзакции = ТекущаяДата(), секция = 1, код товара = "2", цена = 3, количество = 4,
сумма = 5, номер ккм = по-умолчанию
Кассир5_DataAccess.ПисатьВремТранзакцию(КодВремТранзакции, 777, ТекущаяДата(), 1,
"2", 3, 4, 5, Кассир5_DataAccess.ИспользоватьККМ, "");
```

Пример записи врем. транзакции на чек, без явного указания кода:

```
ТипТранзакцииДопИнфо = 777;
// удаляем транзакцию с типом 777, чтобы не было дублей
Для Каждого Транз Из
Кассир5_DataAccess.ПолучитьВремТранзакцииПоТипу(ТипТранзакцииДопИнфо) Цикл
  Кассир5_DataAccess.УдалитьВремТранзакцию(Транз.Код);
КонецЦикла;
// запишем транзакцию на чек, код установится автоматически, дата транзакции =
ТекущаяДата(), секция = 1, код товара = "2", цена = 3, количество = 4, сумма = 5, номер ккм
= по-умолчанию
Кассир5_DataAccess.ПисатьВремТранзакцию(0, ТипТранзакцииДопИнфо, ТекущаяДата(), 1,
"2", 3, 4, 5, Кассир5_DataAccess.ИспользоватьККМ, "");
```

См. также в модуле обработки Кассир5\_DataAccess:

- ИзменитьВремТранзакциюПоСтруктуре**
- ИзменитьВремТранзакцию**
- УдалитьВремТранзакцию**
- ПолучитьВыборкуСкидок**

## ПолучитьВремТранзакцииЧека

### Работа со справочником Транзакции

Для записи транзакции напрямую в справочник Транзакции (транзакции, не относящиеся к конкретному чеку) используется процедура модуля обработки Кассир5\_DataAccess

#### ЗаписатьТранзакции.

Параметры:

1: Таблица значений (неопределено) - таблица значений, получаемая функцией модуля формы регистрации или модуля обработки Кассир5\_DataAccess **ДобавитьТранзакциюВТаблицу.**

Пример записи одной транзакции с типом 777:

```
Кассир5_DataAccess.ЗаписатьТранзакции(Кассир5_DataAccess.ДобавитьТранзакциюВТаблицу(Неопределено,
Неопределено, // код кассира
777, // тип транзакции
Неопределено, // дата транзакции
0, // секция
"", // код товара
0, // цена
0, // количество
0)); // сумма
```

Пример записи нескольких транзакций:

```
ТаблицаТранзакций = Неопределено;
Кассир5_DataAccess.ДобавитьТранзакциюВТаблицу(ТаблицаТранзакций, Неопределено,
777, Неопределено, 0, "", 0, 0, 0);
Кассир5_DataAccess.ДобавитьТранзакциюВТаблицу(ТаблицаТранзакций, Неопределено,
778, Неопределено, 0, "", 0, 0, 0);
Кассир5_DataAccess.ЗаписатьТранзакции(ТаблицаТранзакций);
```

### Работа с правами пользователя

**ПравоДоступно** - функции модуля обработки Кассир5\_DataAccess, выполняет проверку наличия права у пользователя, тип возвращаемого значения булево.

1: Строка - имя права.

2: Булево (Истина) - если у текущего пользователя право отсутствует, но существует пользователь с набором прав, в котором это право присутствует - тогда откроется форма для считывания карты пользователя с этим правом.

3: Булево (Истина) - выводить сообщение об отсутствии права.

4: Число (Неопределено) - если параметр указан, тогда будет проверяться пользователь с кодом, указанным в этом параметре, иначе текущий пользователь.

5: Число (Неопределено) - в этот параметр запишется код пользователя, который подтвердил право.

```
Если НЕ Кассир5_DataAccess.ПравоДоступно("Право", Ложь, Ложь) Тогда
// действие при отсутствии права без запроса карты и без вывода сообщения
```

