

2024

# SCAN MARK

РУКОВОДСТВО ПО НАСТРОЙКЕ «SCANMARK»

РЕДАКЦИЯ 1.22.1

# SCANCODE



# Содержание

---

<b>Введение</b> .....	4
<b>Состав ПО Scanmark</b> .....	6
<b>Установка на примере конфигурации Управление торговлей, ред.11.5</b> .....	7
Установка расширения конфигурации.....	7
Публикация web-сервиса .....	7
Настройка обработки Scanmark в системе «1С:Предприятие» .....	8
Установка Scanmark на ТСД.....	9
Установка SCANCODE.GeneralService на ТСД.....	10
Установка лицензии Scanmark через SCANCODE.GeneralService .....	10
Установка лицензии Scanmark через сайт .....	11
Настройка обмена данными через FTP .....	13
Установка GodexPrintService .....	17
Настройка web-сервиса Scanmark на ТСД.....	17
<b>Настройка аутентификации</b> .....	19
Простая .....	19
Авторизация при входе .....	19
<b>Пример настройки «по шагам»</b> .....	21
Задача 1: Получить документы «Заказ клиента» из «1С:Предприятие» на ТСД. ....	22
Задача 2: Настроить в документе «Заказ клиента» реквизиты как объекты метаданных + настроить интерфейс документа для ТСД. ....	25
Задача 3: Настроить данные штрихкодов.....	28
Задача 4: Добавить в настройку номенклатуры информацию об остатке и цене. ....	30
Задача 5: Добавить функционал пересчета данных в табличной части «Товары» документа «Заказ клиента» .....	31
Задача 6: Создать и заполнить данными документ «Заказ клиента» на ТСД .....	33
Задача 7: Отправка и обработка документа «Заказ клиента» с ТСД в «1С:Предприятие» .....	39
Задача 8: На ТСД на основании документа «Заказ клиента» оформить, подобрать коды маркировки DataMatrix GS1 и отправить документ «Реализация товаров и услуг» в «1С:Предприятие» .....	45
Задача 9: На ТСД создать команду для фотографирования и присоединения фотографии к номенклатуре в «1С:Предприятие» .....	56
Задача 10: Получить фото номенклатуры из «1С:Предприятие» на ТСД .....	60
Задача 11: Создать пример команды для работы со считывателем RFID .....	63
Задача 12: Создать этикетку с маркировкой «Честный знак» DataMatrix GS1 .....	66
Задача 13: Создать отчет .....	73

Задача 14: Пример работы push-уведомлениями .....	77
Задача 15: Работа с блокировками документов .....	80
Задача 16: Удалить заблокированные документы другими пользователями .....	83
Задача 17: Пример использования метода «ПолучитьДанныеИз1С» .....	86
Задача 18: Пример использования метода «ВыполнитьСканмарк» .....	89
Задача 19: Пример заполнения «по умолчанию» .....	92
Задача 20: Пример заполнения «по соответствию» .....	95
Задача 21: Пересчет табличной части документа «Заказ клиента» после выбора типа цен .....	97
Задача 22: Пример подбора номенклатуры по штрихкоду с вводом количества .....	104
Задача 23: Пример подбора номенклатуры из списка справочника с вводом количества .....	107
Задача 24: Пример поиска номенклатуры по наименованию с вводом количества и возможностью запроса номенклатуры с сервера .....	109
Задача 25: Пример настройки нажатия клавиш .....	112
Задача 26: Пример работы с синтезатором речи. ....	115
Задача 27: Пример подбора номенклатуры по штрихкоду по данным на ТСД + из 1С с возможностью установить новый штрихкод номенклатуры в 1С. ....	118
Задача 28: Пример работы с составным типом данных, преобразование типа данных. ....	127
Задача 29: Пример управления отображением элементов на форме на ТСД .....	137
<b>Условное оформление</b> .....	140
Пример условного оформления справочника «Номенклатуры» .....	140
Пример условного оформления документа «Реализация» .....	140
<b>Настройка основного меню</b> .....	142
Обычное меню (по умолчанию) .....	142
Быстрое меню .....	143
<b>Особенности объектов в «Scanmark»</b> .....	146
<b>Сохранение и восстановление данных в приложении «Scanmark»</b> .....	147
<b>Логирование</b> .....	148
Логирование на стороне 1С:Предприятие. ....	148
Логирование на стороне мобильного приложения. ....	148
<b>Процедуры и функции обработчиков «Scanmark»</b> .....	150
В помощь при настройке .....	150
IDприложения .....	151
PushУведомление .....	152
Блокировка .....	152
Видимость .....	153
ВидимостьКнопки .....	154

ВыбратьДокументы .....	154
ВыбратьЭлементы .....	155
ВыполнитьСканмарк.....	155
Данные.....	155
ДанныеШтрихкода.....	156
ДокументОснование.....	156
ЗаписатьЭтотОбъект .....	157
Источник .....	157
КОтправке.....	157
Лог .....	158
НайтиПоКоду.....	158
НайтиПоНаименованию.....	158
НайтиПоРеквизиту .....	159
НайтиПоУникальномуИдентификатору.....	159
ОбъектСканмарк .....	160
ОписаниеОшибки .....	160
ОткрытьПодбор.....	160
Отправлен.....	161
ПереместитьКнопкуНаСтраницу.....	161
ПолучитьДанныеИз1с.....	162
РазблокироватьДокумент .....	163
Редактирование .....	163
СказатьФразу.....	163
СвойНаборДанных.....	164
Событие .....	164
ТекущаяСтраница.....	165
ТекущиеДанные.....	165
ТекущийПользователь.....	165



# Введение

---

Scanmark – это специализированное программное обеспечение (ПО) предназначенное для автоматизации ведения учета в системах «1С:Предприятие» совместно с мобильными устройствами под управлением операционной системы Android, в т.ч. терминалами сбора данных (ТСД). ПО обладает широким функционалом, а возможности настройки позволяют автоматизировать практически любые пожелания пользователя, т.е. ПО является по сути конструктором мобильного приложения.

С помощью Scanmark вы сможете:

- работать с любой конфигурацией «1С:Предприятие» версии 8.3 и последующих, включая как конфигурации с управляемыми, так и с обычными формами;
- получать актуальные данные из «1С:Предприятие»;
- работать с объектами учета (документы, справочники, перечисления) в «1С:Предприятие» на ТСД, отправлять и применять результаты в «1С:Предприятие», а именно:
  - создавать документы;
  - редактировать документы;
  - создавать произвольные наборы данных на ТСД и дальнейшие передача, обработка и сохранение таких данных в «1С:Предприятие»;
  - обрабатывать полученные данные на ТСД с помощью сканера штрихкода;
  - обрабатывать полученные данные на ТСД с помощью считывателя RFID;
  - работать с кодами маркировки системы «Честный знак» DataMatrix GS1;
- печатать этикетки и ценники, в т.ч. марки с кодом маркировки системы «Честный знак» DataMatrix GS1;
- запрашивать коды маркировки системы «Честный знак» формата DataMatrix GS1, печатать этикетки с полученными кодами;
- получить отчеты из «1С:Предприятие»;
- получить фотографии из «1С:Предприятие»;
- фотографировать и отправлять фотографии в «1С:Предприятие», где эти фотографии можно «присоединить» к объектам учета, например, к элементам справочника «Номенклатура»;
- использовать push-уведомления;
- произносить текст;
- обрабатывать нажатия клавиш (для ТСД с клавиатурой).

В данном руководстве описаны состав, определения и функционал компонентов, из которых состоит ПО Scanmark. Также в руководстве вы найдете подробный пример создания настройки «с нуля» на основе типовой конфигурации «Управление торговлей 11.5». ПО поставки содержит файлы описываемых в руководстве примеров, которые можно загрузить в обработку настройки Scanmark на стороне «1С:Предприятие».

Настройку Scanmark следует производить специалисту, имеющему квалификацию *разработчик (программист) 1С*.

Scanmark использует «Web-сервис» как основной механизм обмена, работа которого возможна при установленном в операционной системе веб-сервера, например, «Apache» или «IIS» (Internet Information Server). Процедура установки и настройки веб-сервера в руководстве не рассматривается. Рекомендуется установить веб-сервер до начала внедрения Scanmark.

Если по каким то причинам использование обмена через «Web-сервис» невозможно, то альтернативным способом может быть обмен данными по FTP.

Для работы со считывателями штрихкодов и RFID из мобильного приложения Scanmark

на мобильном устройстве требуется установка приложения SCANCODE.GeneralService и лицензии. Скачать лицензию возможно с сайта [www.scancode.ru](http://www.scancode.ru) или непосредственно из приложения SCANCODE.GeneralService. SCANCODE.GeneralService гарантированно работает на следующих моделях: CipherLab, HoneyWell, NewLand, GlobalPos, Speedata.

# Состав ПО Scanmark

---

ПО Scanmark состоит из:

1. приложений для мобильного устройства:
  - a. приложение Scanmark (Scanmark\_1\_x\_x.apk) – основное приложение;
  - b. приложение GeneralService (SCANCODE.GeneralService-x.x.x.apk) – сервис для работы со сканером штрихкод;
  - c. приложение GodexPrintService (GodexPrintService\_x\_x\_x.apk) – сервис для печати на мобильные принтеры;
2. расширения конфигурации (Scanmark\_x\_x\_x.cfe) – дополнительная надстройка конфигурации для администрирования и настройки бизнес-процессов, обмена данными и пользовательского интерфейса приложения Scanmark на ТСД,

Объекты метаданных расширения являются независимыми, т.е. не содержат обращений к любым объектам метаданных и функционалу конфигурации пользователя кроме объектов текущего расширения.

Вся настройка ПО Scanmark производится непосредственно в системе «1С:Предприятие» в форме обработки «Scanmark 1.0.X».

Так как механизм работы с расширениями «1С:Предприятие» появился только в версии платформы «1С:Предприятие» 8.3.15, то для внедрения ПО Scanmark для платформ версий 8.3.14 и старше следует объекты расширения web-сервис «ScanMark» и обработку «Scanmark» внедрить непосредственно в конфигурацию.

Приложение для ТСД устанавливается непосредственно на ТСД, работающей под управлением операционной системы Android. Минимальная версия ОС Android 4.4.2. Кроме этого, для работы с драйвером сканера штрихкода необходима установка на ТСД специального программного обеспечения «SCANCODE.GeneralService» + установка лицензии (привязка по серийному номеру устройства).

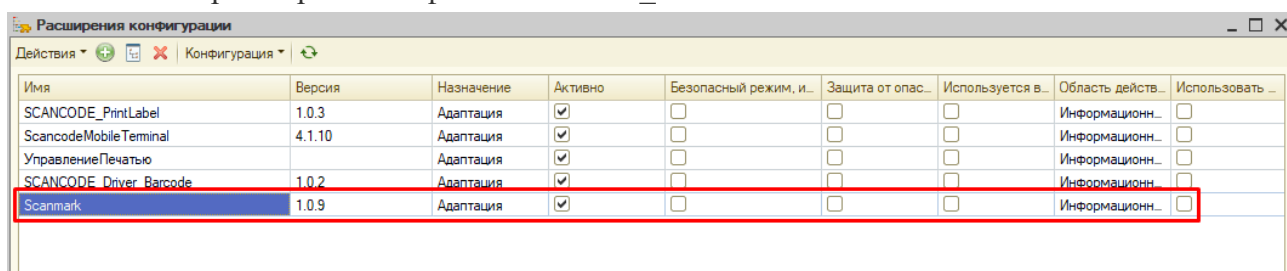
Если планируется использование функций печати на мобильные принтеры, то дополнительно необходима установка на ТСД приложения GodexPrintService.

# Установка на примере конфигурации Управление торговлей, ред.11.5

## Установка расширения конфигурации

От имени администратора операционной системы запустите конфигуратор «1С:Предприятие» с набором прав «Администрирование».

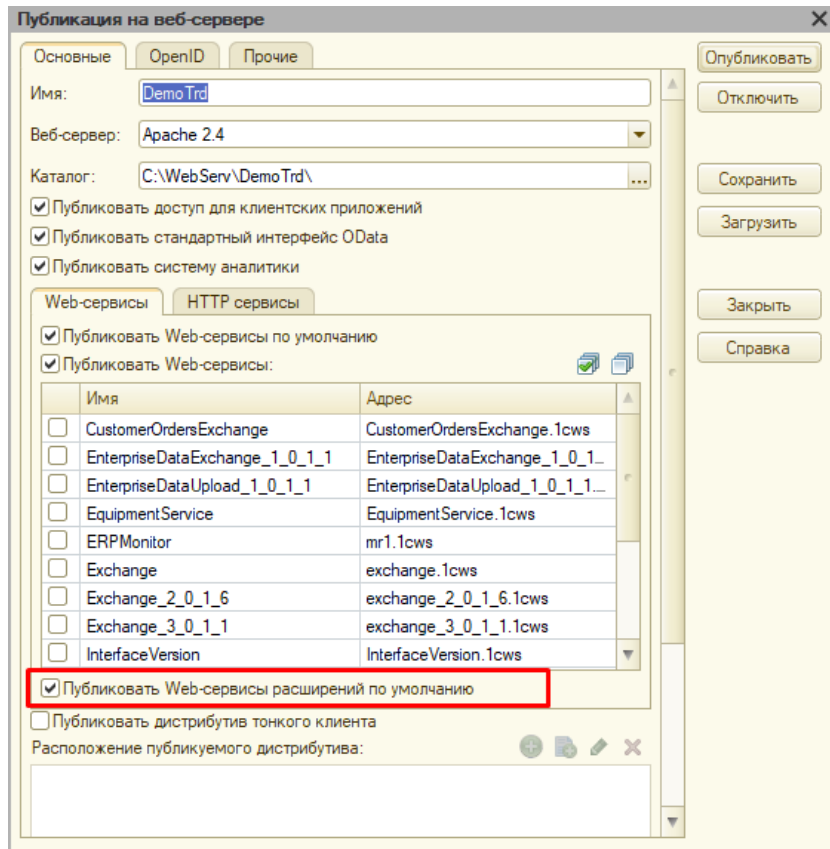
Установите расширение из файла Scanmark\_1.0.x.cfe:



В строке расширения «Scanmark» снимите все флажки кроме флажка в колонке «Активно».

## Публикация web-сервиса

Если ранее для текущей базы данных не была сделана публикация веб-сервиса, создайте такую публикацию. Убедитесь, что в окне настройки публикации установлен флажок «Публиковать Web-сервисы расширений по умолчанию».

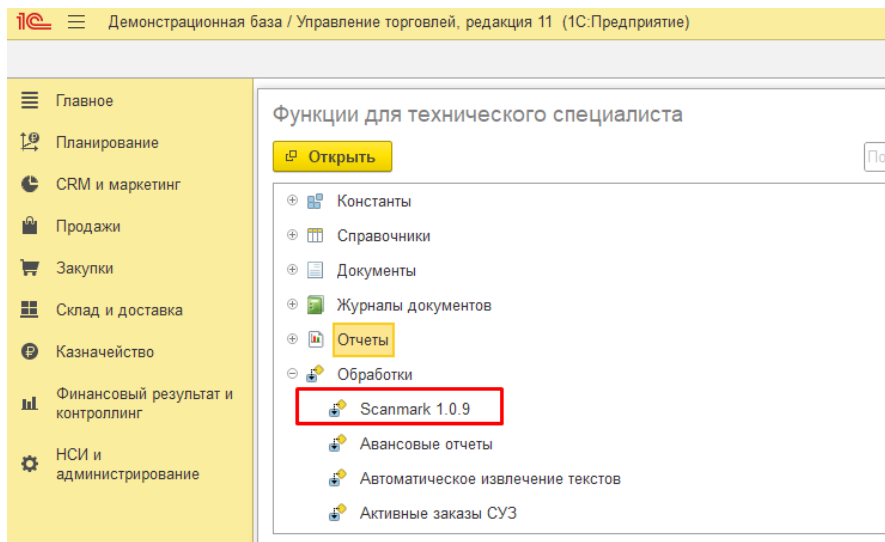


После создания/обновления публикации рекомендуется перезапустить веб-сервер (в редакция 1.22.1

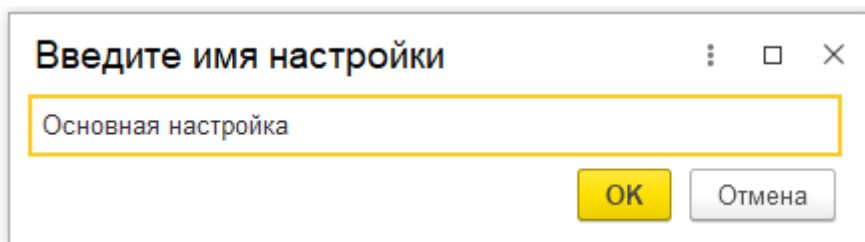
службах операционной системы).

## Настройка обработки Scanmark в системе «1С:Предприятие»

Запустите «1С:Предприятие». Через основное меню – Функции технического специалиста найдите и откройте обработку Scanmark x.x.x:

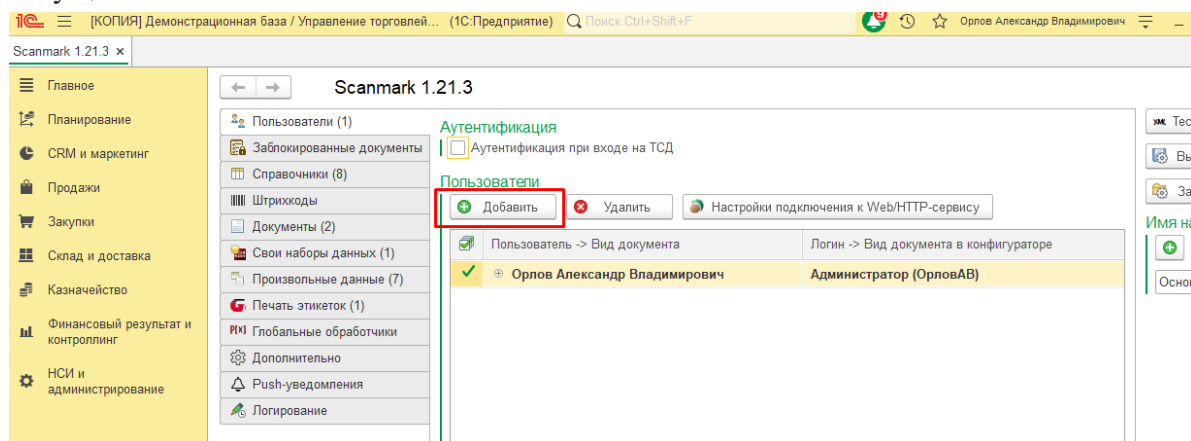


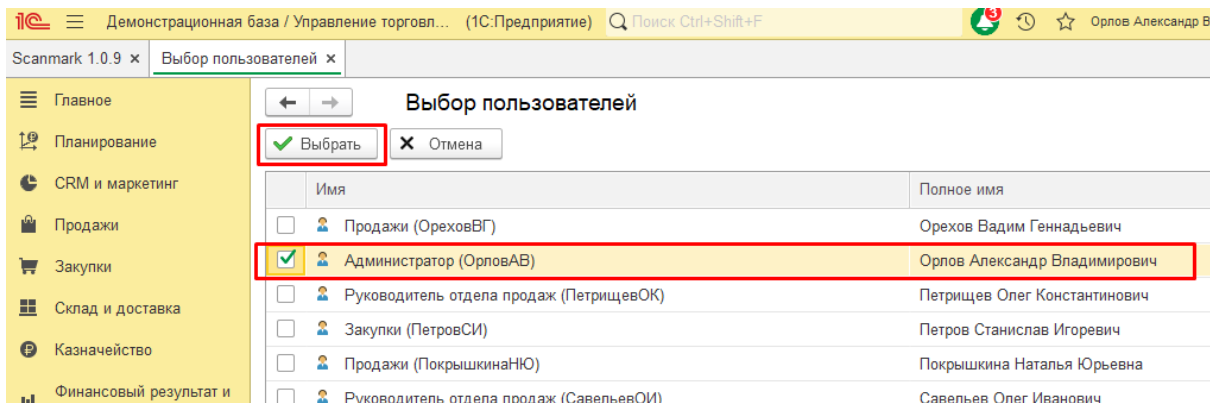
При первом открытии будет предложено задать имя настройки.



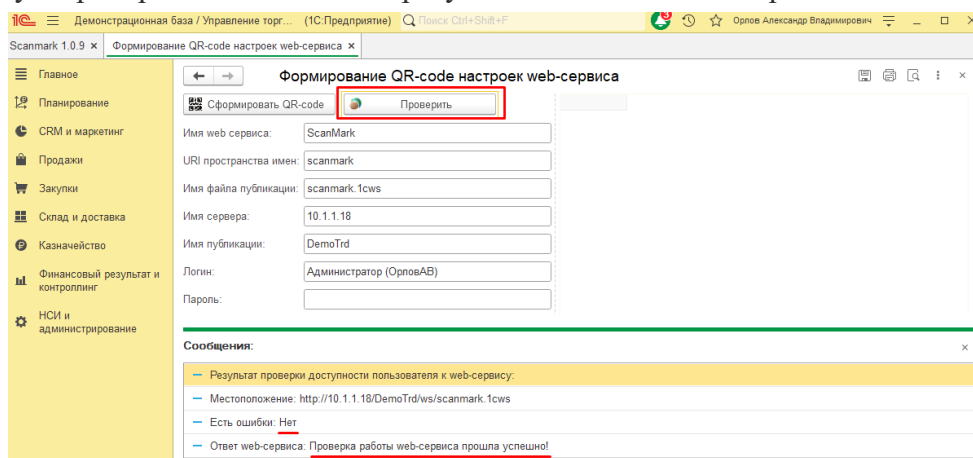
Scanmark позволяет работать одновременно с несколькими настройками. Это может быть удобно для отладки, или когда для сотрудников компании необходимо иметь различный функционал. Например, функционал кладовщиков и экспедиторов.

После открытия формы настройки Scanmark необходимо задать хотя бы одного пользователя. На закладке «Настройка пользователей» нажмите кнопку «Добавить». Выберите пользователей. Среди выбора прочих пользователей рекомендуется отметить для добавления текущего пользователя 1С.





Для проверки работоспособности веб-сервиса нажмите кнопку «Настройки подключения к Web-сервису». В открывшейся форме «Формирование QR-code настроек web-сервиса» задайте корректные значения полей «Имя сервера», «Имя публикации», «Пароль». Нажмите кнопку «Проверить» и дождитесь результата обмена с веб-сервисом.

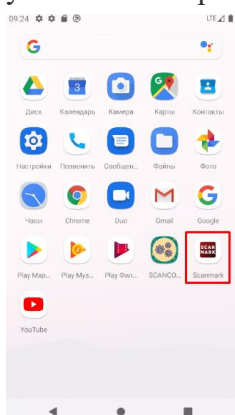


Если в результате вы получите сообщения об ошибке, то следует проверить работу веб-сервера, настройки и права веб-сервиса, корректность заполненных значений полей «Имя сервера», «Имя публикации», «Пароль». Добейтесь получения ответа об успешной проверке.

## Установка Scanmark на ТСД

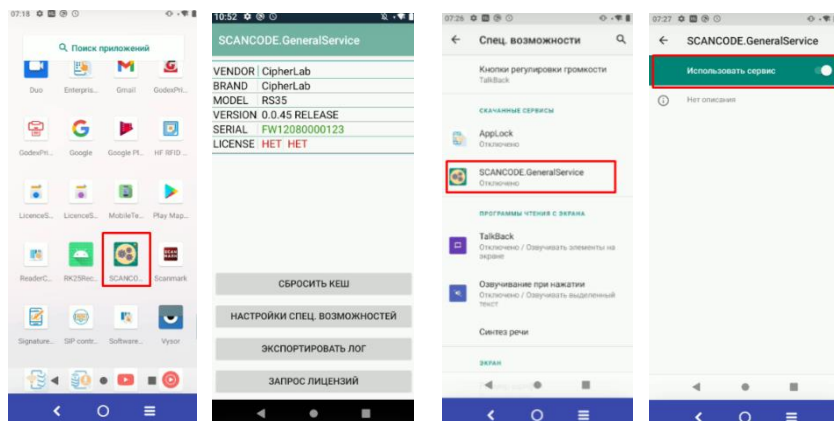
Теперь подключаем к работе ТСД.

Перепишите файл Scanmark\_1\_0\_X.apk на ТСД и запустите файл. В результате установится приложение Scanmark.



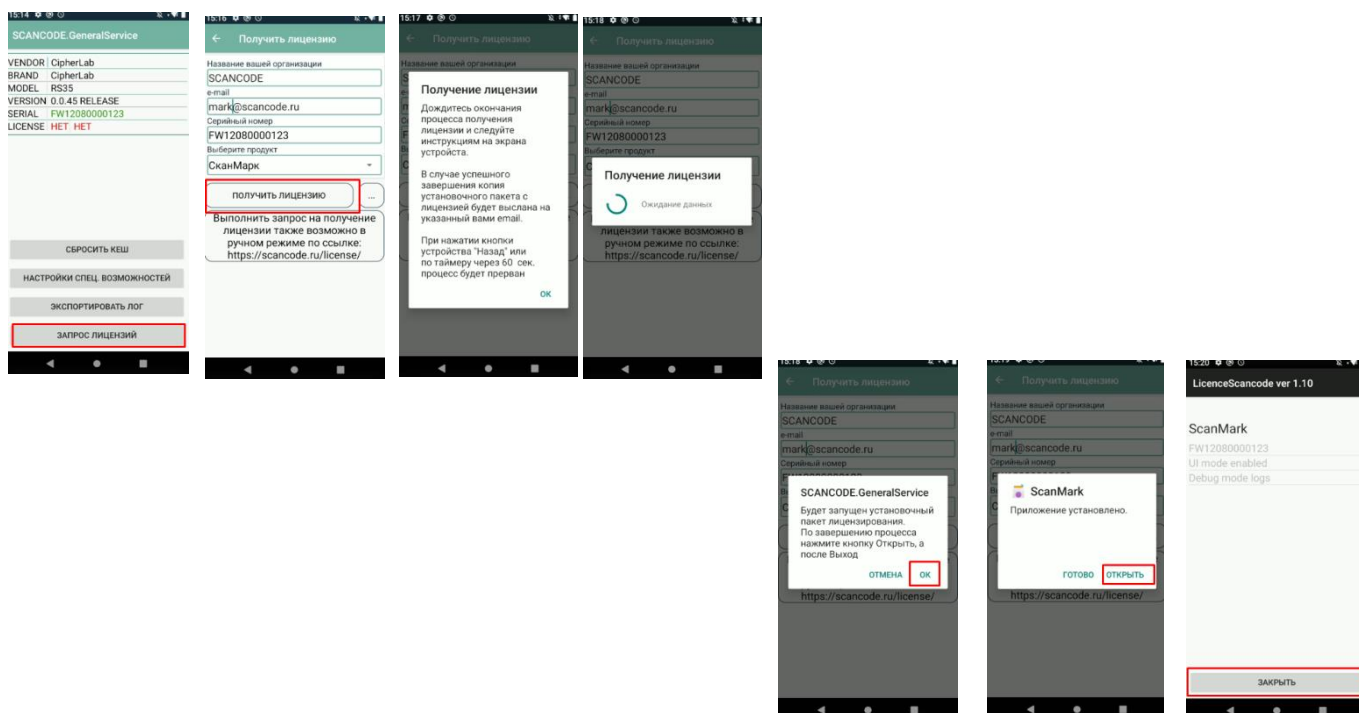
## Установка SCANCODE.GeneralService на ТСД

Для работы приложения со сканером штрихкода понадобится установка сервиса SCANCODE.GeneralService. Перепишите на мобильное устройство файл SCANCODE.GeneralService-x.x.x.apk из комплекта поставки и запустите его. В результате установки появится приложение SCANCODE.GeneralService. Далее следует запустить приложение. Через кнопку «Настройки спец.возможностей» включить сервис.



## Установка лицензии Scanmark через SCANCODE.GeneralService

Работа сканера штрихкода в приложении Scanmark возможна только после установки приложения SCANCODE.GeneralService и лицензии. Об установке SCANCODE.GeneralService написано в предыдущем пункте руководства. Получить лицензию можно двумя способами. Рекомендуемый способ – из приложения SCANCODE.GeneralService. Откройте приложение SCANCODE.GeneralService, нажмите «ЗАПРОС ЛИЦЕНЗИЙ», заполните форму получения лицензий и нажмите «Получить лицензию»

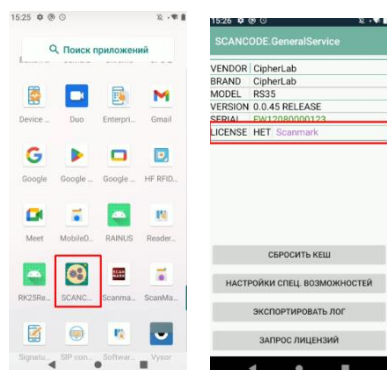


После установки закройте Лицензию и SCANCODE.GeneralService. Запустите приложение Scanmark, убедитесь что сканер штрихкода работает – просто отсканируйте в основном меню



приложения любой штрихкод.

Также можно запустить приложение SCANCODE.GeneralService и убедиться в наличии установленной лицензии Scanmark.



## Установка лицензии Scanmark через сайт

Второй способ получения и установки лицензии – это скачать лицензию с сайта по ссылке <https://scancode.ru/faq/>.

**SCANCODE** +7 (495) 984 34 99 Москва Новосибирск

Поиск Вход


Каталог Скачать Где купить? Сервис Техподдержка Форум О компании Новости Проекты Контакты YouTube канал

Главная страница > Техподдержка

## Техподдержка


### Уважаемые партнеры!

Ответы на большинство вопросов можно найти в инструкции пользователя:  
процесс подключения Терминалов Сбора Данных (ТСД) к ТС,  
загрузка прошивок в ТСД,  
установка драйвера принтера,  
печать серийных номеров  
и многое другое описано в документации на Вашу модель оборудования, доступны для скачивания на странице описания на вкладке "Файлы".



## НАШ YOUTUBE КАНАЛ

ОБЗОРЫ И ПОШАГОВЫЕ ИНСТРУКЦИИ ПО НАСТРОЙКЕ И ПОДКЛЮЧЕНИЮ ОБОРУДОВАНИЯ



На данный момент существует база знаний с ответами на часто задаваемые вопросы - F.A.Q. (Вопросы-Ответы). Ознакомиться с ними можно на Форуме, [в этом разделе](#).

Для восстановления лицензии на купленный у нас терминал на ОС андроид нажмите на кнопку:

Восстановить лицензию.

**Обращаем ваше внимание!**

Все вопросы, связанные с программированием ТСД, расширением их функционала - задавайте только на [Форуме!](#) Мы помогаем решить вопросы, связанные с оборудованием, приобретенным только [в нашей компании или у наших партнеров](#) (ведётся база серийных номеров по проданным товарам).

Свои вопросы специалистам службы поддержки можно также задавать по WhatsApp и Telegram

Заполните форму и нажмите «Отправить». Будет получен файл лицензии. Кроме того, такой файл будет выслан на заданный E-mail.

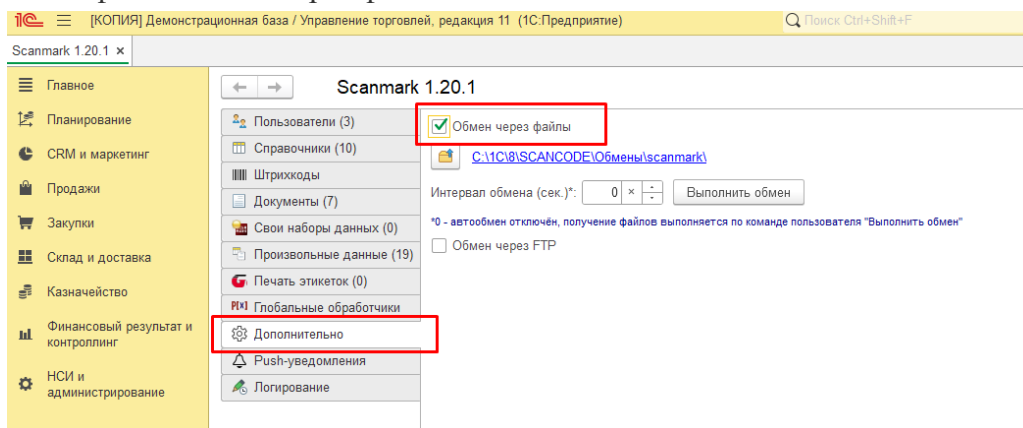


The screenshot shows two parts of an Android phone. The left part is the 'Загрузки' (Downloads) app, displaying a list of downloaded files. The file 'FW1208000123\_SCANMARK...' is highlighted with a red rectangle. The right part shows the 'ScanMark' app interface, which includes a search bar, a grid of app icons (Google, Meet, etc.), and a red rectangle around the 'ScanMark' icon. At the bottom right, there is a red rectangle around the 'ЗАКРЫТЬ' (Close) button.

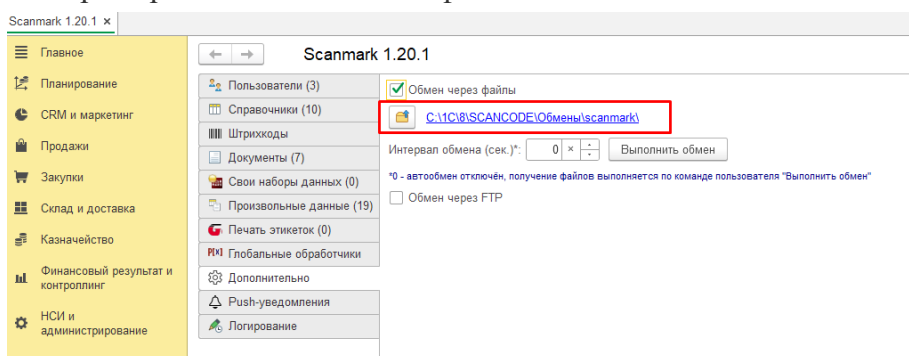
## Настройка обмена данными через FTP

Основным рекомендуемым способом обмена данными приложения Scanmark и системы «1С:Предприятие» является web-сервис. Для случаев, когда по каким либо причинам невозможно настроить обмен данными через web-сервис, в ПО Scanmark есть альтернативная возможность настройки обмена данными через FTP. Кроме того, возможно совместное использование способов обменов: через web-сервис + через FTP.

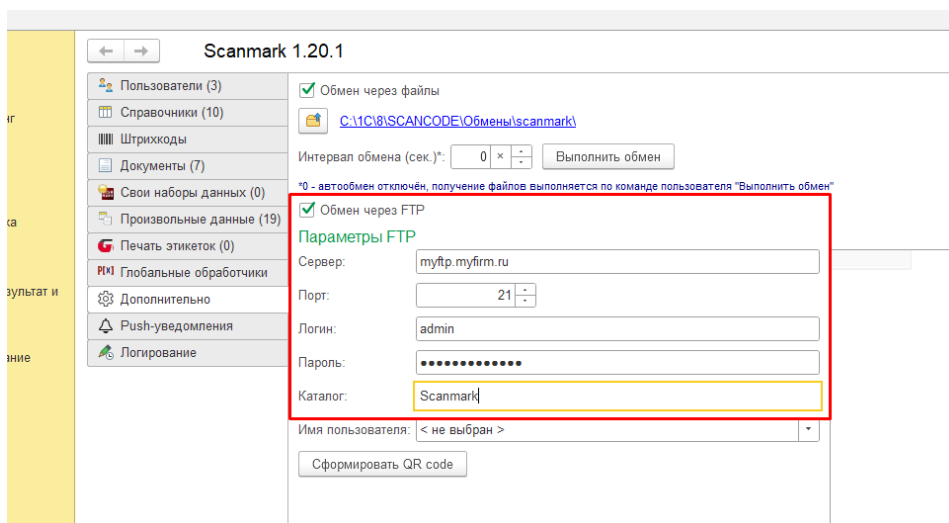
В «1С:Предприятие» в обработке Scanmark откройте закладку «Дополнительно» и установите флаг «Обмен через файлы»:



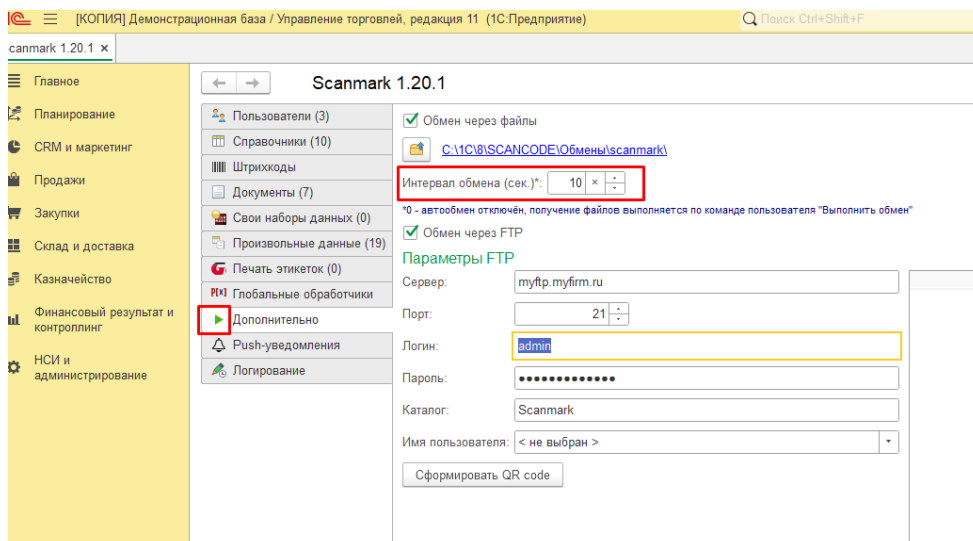
Выберите рабочий каталог для файлов обмена:



Установите флаг «Обмен через FTP». Заполните значения полей раздела «Параметры FTP» своими данными подключения к FTP:

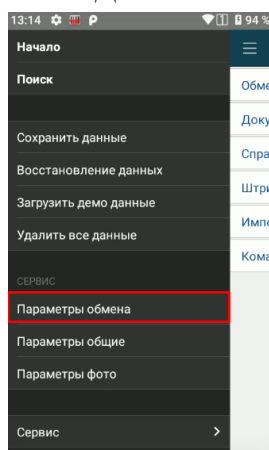


Установите желаемый интервал обмена (проверка файлов обмена в каталоге):

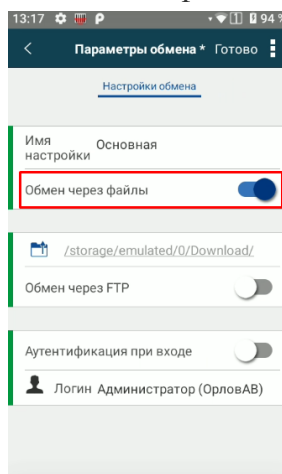


Обработка файлов происходит автоматически пока обработка открыта и интервал обмена > 0. Рекомендуется в «1С:Предприятие» создать специального рабочего пользователя с расширенным набором прав для работы обработки в режиме обмена через FTP.

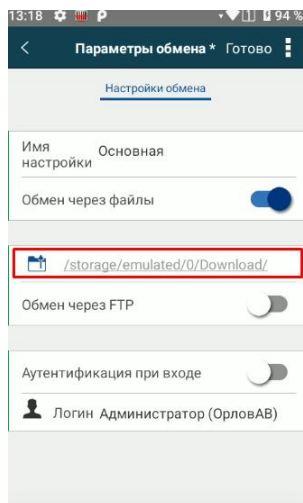
На ТСД в основном меню сдвиньте «шторку» вправо и выберите «Параметры обмена»:



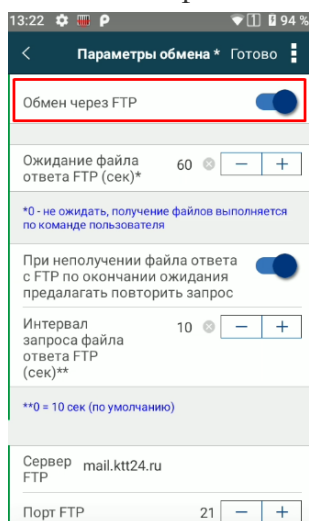
Установите флаг «Обмен через файлы»:



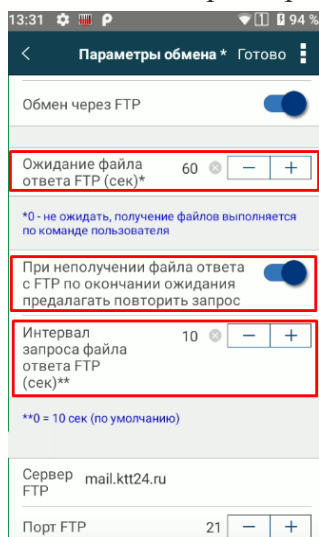
Задайте каталог обмена:



Установите флаг «Обмен через FTP»:



Установите параметры ожидания файлов:



Установите настройки подключения к FTP и имя пользователя в «1С:Предприятие»:

14:20 98 %

Параметры обмена \* Готово

запроса файла  
ответа FTP  
(сек)\*\*

\*\*0 = 10 сек (по умолчанию)

Сервер FTP myftp.myfirm.ru

Порт FTP 21

Каталог Scanmark FTP

Логин FTP admin

Пароль FTP .....

Аутентификация при входе

Логин Администратор (ОрловАВ)

Ввести данные настройки подключения к FTP и имя пользователя в «1С:Предприятие» можно с помощью сканирования подготовленного QR-кода в обработке «1С:Предприятие»:

vk 1.20.1 x

Scanmark 1.20.1

Пользователи (3)

Справочники (10)

Штрихкоды

Документы (7)

Свои наборы данных (0)

Произвольные данные (19)

Печать этикеток (0)

Глобальные обработки

Дополнительно

Push-уведомления

Логирование

Обмен через файлы

C:\1C\8\SCANCODE\Обмен\scanmark\

Интервал обмена (сек.): 0

Выполнить обмен

0 - автообмен отключён, получение файлов выполняется по команде пользователя "Выполнить обмен"

Обмен через FTP

Параметры FTP

Сервер: myftp.myfirm.ru

Порт: 21

Логин: admin

Пароль: .....

Каталог: Scanmark

Имя пользователя: Орлов Александр Владимирович

Сформировать QR code

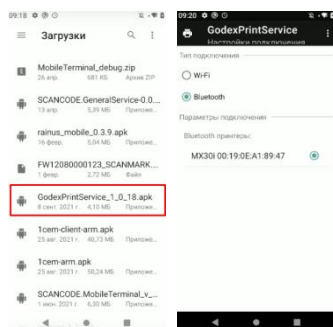
Для изменения масштаба используйте Ctrl + колесо мыши

Администратор (ОрловАВ)

На ТСД нажмите «Готово» и проверьте работоспособность обмена.

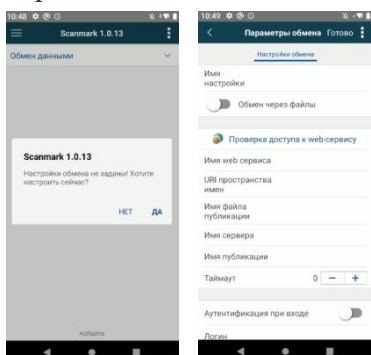
## Установка GodexPrintService

Если на ТСД предполагается производить печать на мобильном принтере GoDEX, то на ТСД необходимо установить приложение GodexPrintService. Ссылка на актуальное ПО GodexPrintService: [https://scancode.ru/catalog/po\\_skankod\\_dlya\\_markirovki/godexprintservice/](https://scancode.ru/catalog/po_skankod_dlya_markirovki/godexprintservice/). Скачайте архив с ПО, на ТСД перепишите файл GodexPrintService\_x\_x\_xx.apk, запустите установку. После установки откройте установленное приложение и сделайте необходимые настройки. Руководство по работе с сервисом печати GodexPrintService.pdf находится в архиве ПО GodexPrintService.



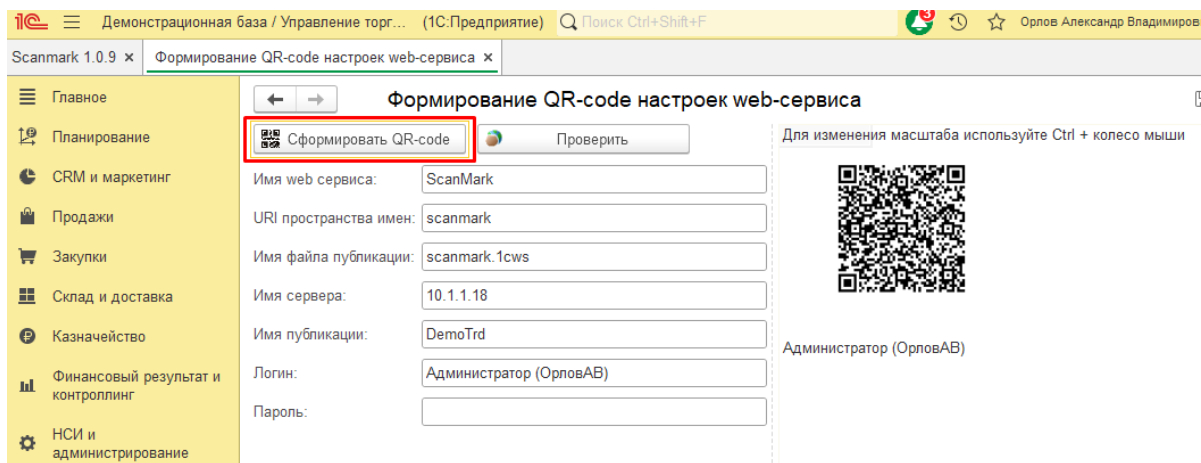
## Настройка web-сервиса Scanmark на ТСД

Запустите приложение Scanmark. При первом запуске будет предложено настроить web-сервис.

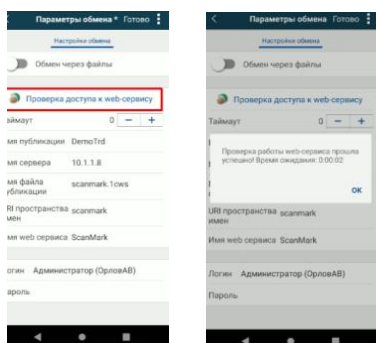


В форме Настройки обмена необходимо заполнить поля соответствующими данными. **Внимание! Параметры должны быть заданы с учетом регистра.**

Заполнение параметров web-сервиса на ТСД удобно заполнять с помощью QR-кода из системы «1С:Предприятие». В обработке «Scanmark» кнопкой «Настройки подключения к Web-сервису» откройте настройку пользователя, задайте значения полей формы. Нажмите «Сформировать QR-code». Будет сформирован QR-code.



На ТСД сканируйте с экрана QR-code. Параметры заполнятся автоматически. Нажмите кнопку «Проверка доступа к web-сервису».



При отсутствии ошибок связи и параметров соединения появится сообщение об успешной проверке. Это говорит о том, что установка и настройка подключения к веб-сервису закончены.

# Настройка аутентификации

Аутентификация входа в приложение на мобильном устройстве может быть двух видов.

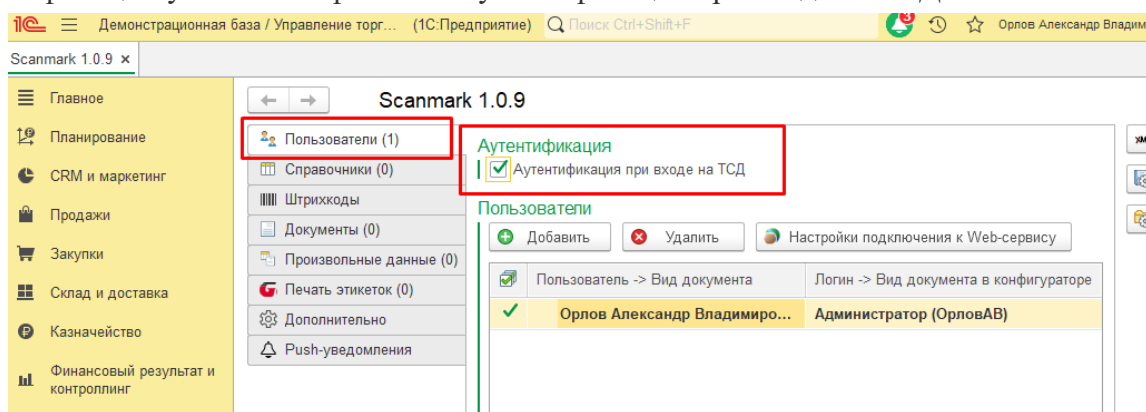
## Простая

Такой способ аутентификации применим при невысоких требованиях к безопасности для работы с информационной базой. Пример такой аутентификации был использован при настройке обмена в предыдущей главе. При таком способе аутентификации логин и пароль для подключения к web-сервису хранятся в настройках мобильного приложения, любой пользователь при работе на мобильном устройстве подключается с помощью одной и той же учетной записью.

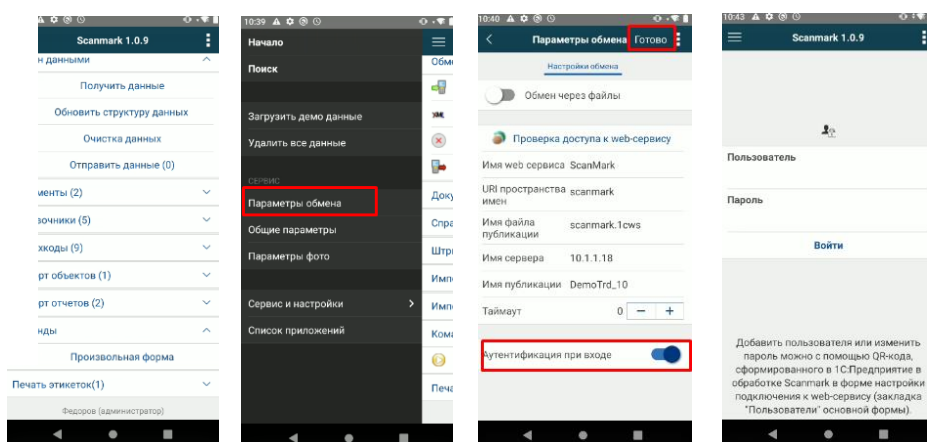
## Авторизация при входе

В отличие от предыдущего способа аутентификации, Пароль для подключения к web-сервису не сохраняется и не передается при обменах между информационной базой и данными мобильного приложения. Также позволяет нескольким пользователям работать попеременно на одном устройстве, но авторизоваться под своей учетной записью.

Переключение на авторизацию при входе настраивается в основном окне обработки Scanmark в системе 1С:Предприятие на закладке «Пользователи». Для включения режима аутентификации установите флажок «Аутентификация при входе на ТСД».



В приложении на мобильном устройстве (ТСД) в настройках обмена включите параметр «Аутентификация при входе» и нажмите «Готово». Также параметр «Аутентификация при входе» включится автоматически при сканировании QR-кода настроек пользователя.

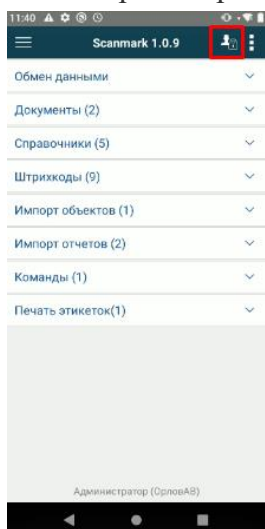




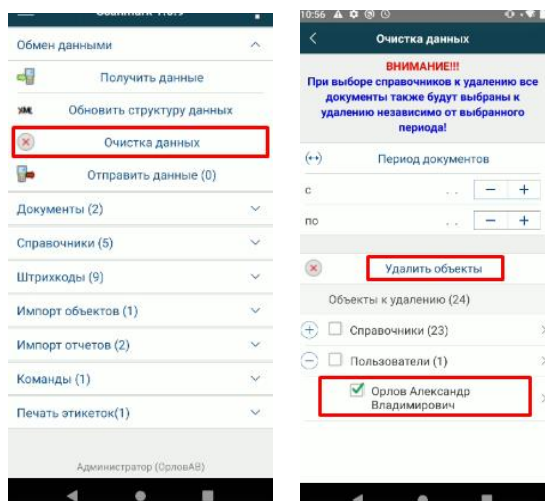
В окне авторизации также можно сканировать QR-кода настроек пользователя для добавления, обновления учетной записи пользователя.

Как было отмечено выше, при таком способе аутентификации пароль никоим образом не передается и не сохраняется в базе данных мобильного приложения. Корректность пароля проверяется с помощью сохраняемого хеш-значения пароля.

Заблокировать работу приложения возможно с помощью команды блокировки.



Для удаления «ненужных» пользователей используйте очистку данных.



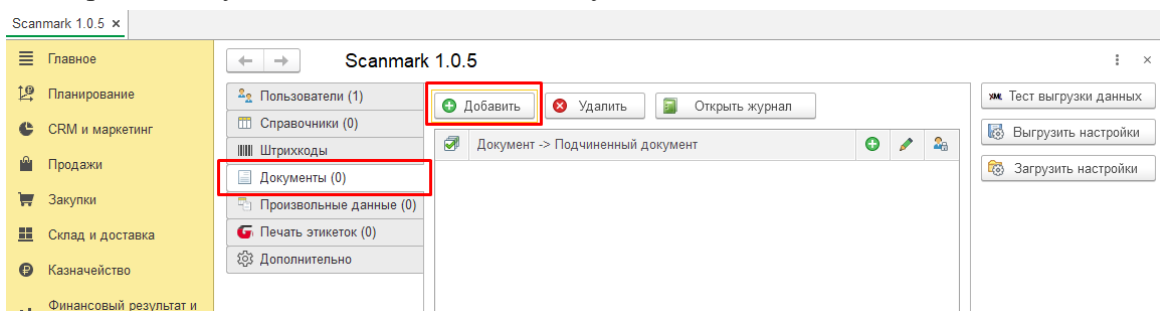
# Пример настройки «по шагам»

---

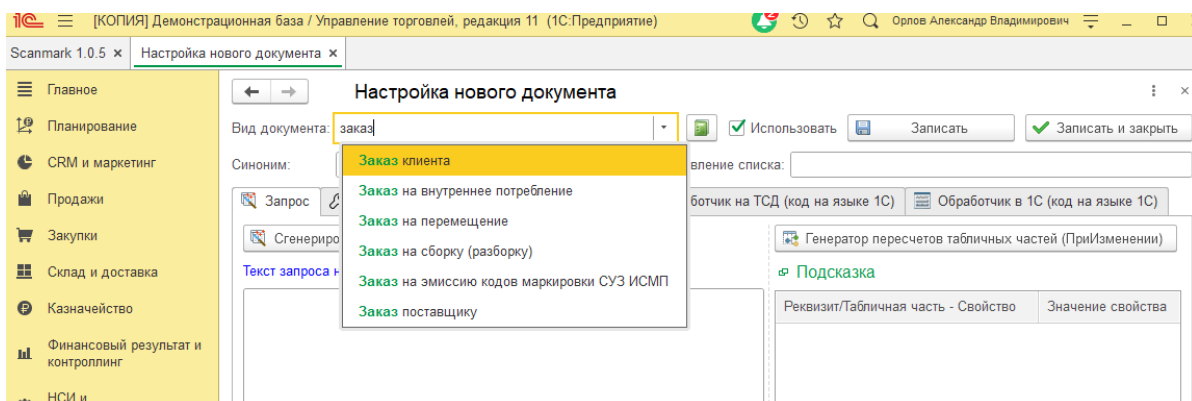
В этом разделе будет описана настройка «по шагам» на примере типовой конфигурации «1С:Предприятие» «Управление торговлей, 11.х». Пример состоит из чередующихся задач. Т.к. задачи взаимосвязаны, следует придерживаться предложенному порядку изложения задач в примере.

## Задача 1: Получить документы «Заказ клиента» из «1С:Предприятие» на ТСД.

В «1С:Предприятие» открываем обработку «Scanmark», переходим на страницу «Настройка документов», нажимаем кнопку «Добавить».

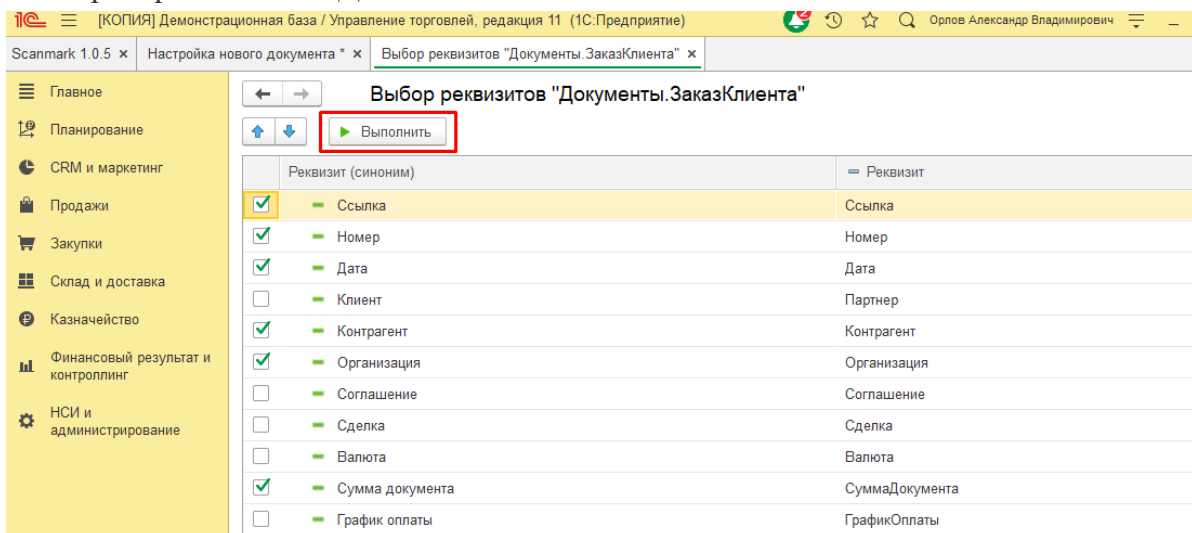


В форме «Настройка нового документа» в поле «Вид документа» выбираем «Заказ клиента».

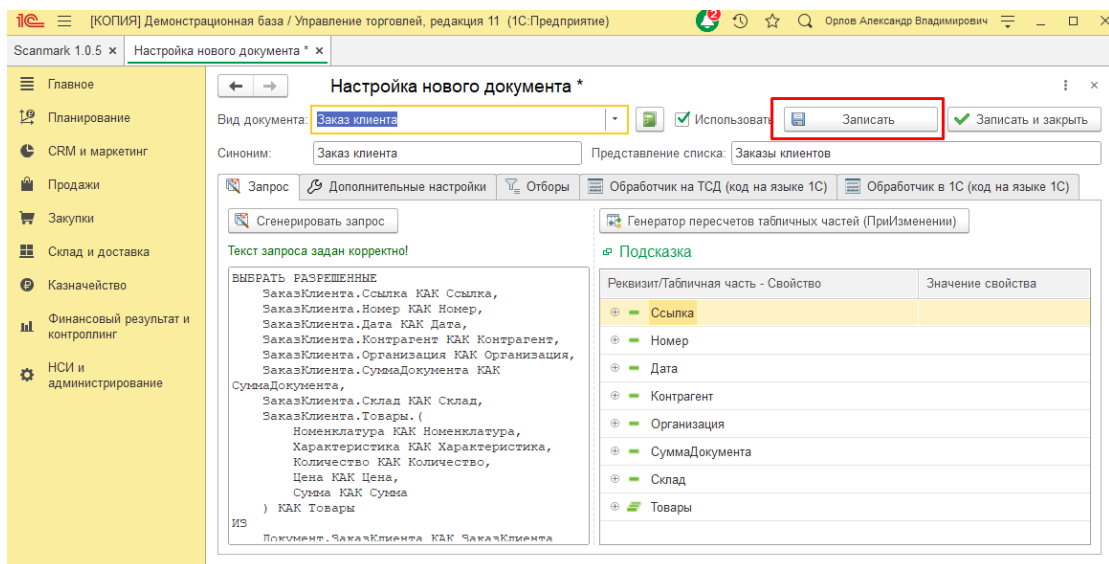


Откроется окно выбора реквизитов «Заказа клиента». Выберите поля шапки документа: Ссылка, Номер, Дата, Контрагент, Организация, Склад, СуммаДокумента, а также поля табличной части «Товары»: Номенклатура, Характеристика, Количество, Цена, Сумма. Нажмите кнопку «Выполнить».

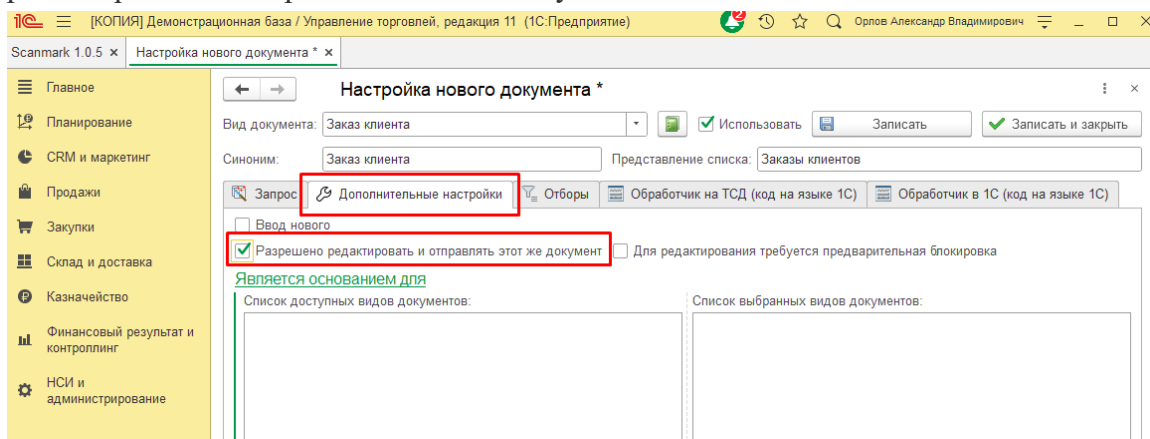
**ВАЖНО!** При настройке документа или справочника всегда выбирайте реквизит «Ссылка», т.к. по этому значению идентифицируется объект при обмене между «1С:Предприятие» и ТСД.



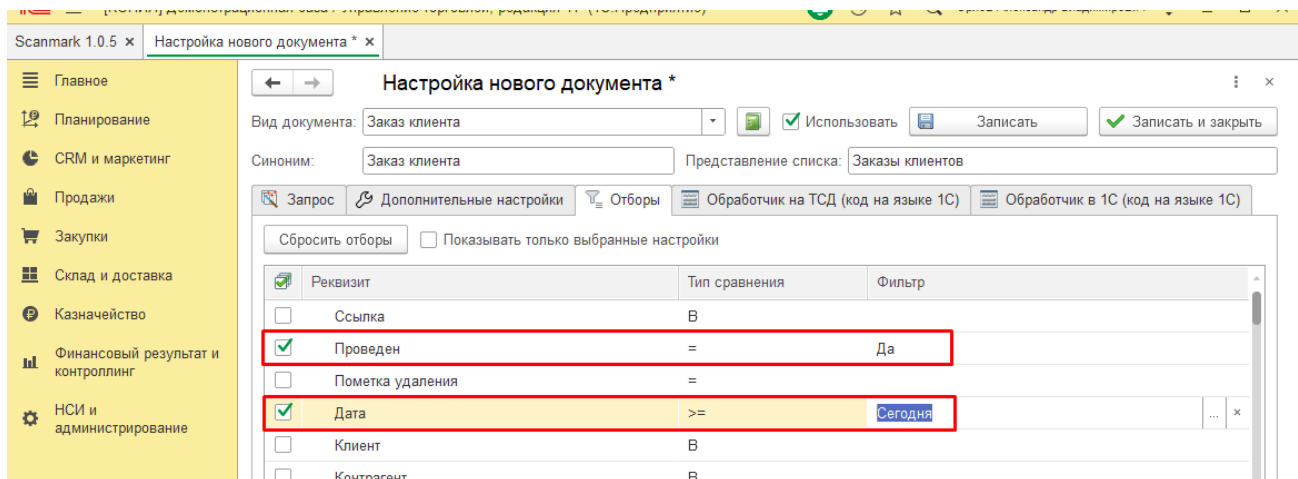
В результате будет сформирован запрос (в левой части) и появятся реквизиты (в правой части). Нажмите кнопку «Записать» для сохранения текущей настройки.



Перейдем на закладку «Дополнительные настройки». Установим флажок «Разрешено редактировать и отправлять этот же документ».

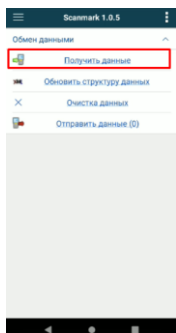


Перейдем на закладку «Отборы». Установим отбор по «Проведен» = Да и «Дата» >= Сегодня.

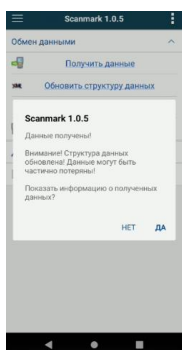


Снова запишем настройку.

Перемещаемся на ТСД. В основной форме выполняем команду Обмен данными – Получить данные.

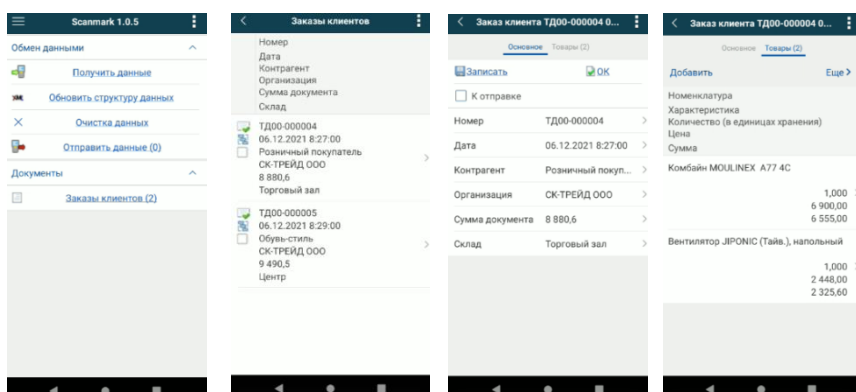


После окончания обмена должно появиться сообщение, как на картинке ниже:



**ВАЖНО!** Каждый раз, когда мы будем что-то менять в настройках обработки «Scanmark» при обмене данными на ТСД будет обновляться структура данных.

Мы получили документы, которые можно посмотреть в появившемся журнале «Заказы клиентов».

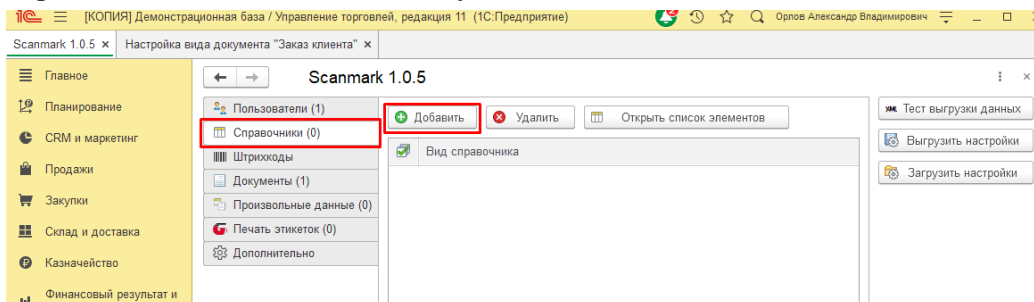


## Задача 2: Настроить в документе «Заказ клиента» реквизиты как объекты метаданных + настроить интерфейс документа для ТСД.

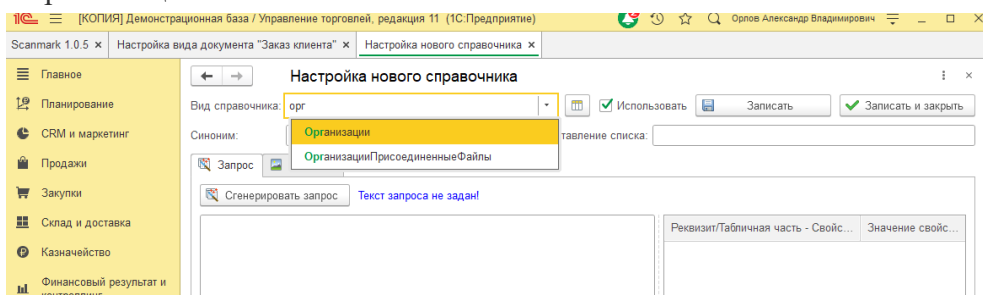
В Задаче 1 мы создали настройку документа «Заказ клиента». Эта настройка содержит реквизиты документа, но типы реквизитов такого документа на ТСД являются базовыми (строка, число, дата, булево).

В этой задаче мы добавим объекты учета «Справочники», а также сделаем пользовательскую настройку интерфейса.

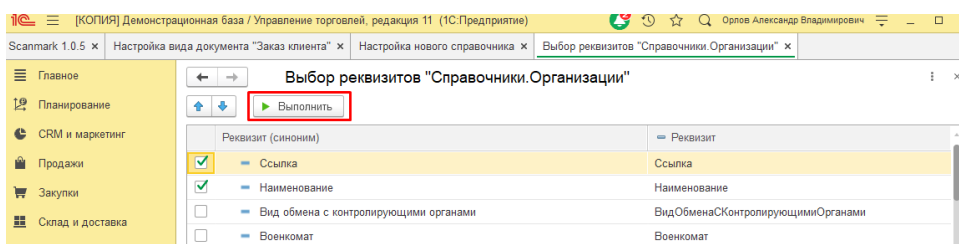
В «1С:Предприятие» в обработке «Scanmark» переходим на закладку «Настройка справочников», нажимаем кнопку «Добавить».



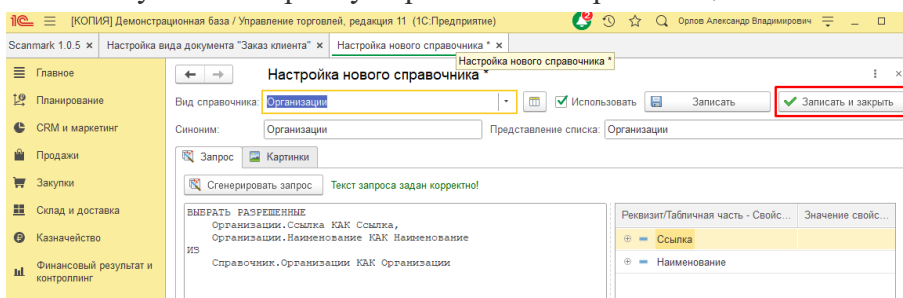
В окне «Настройка нового справочника» в поле «Вид справочника» выбираем «Организации».



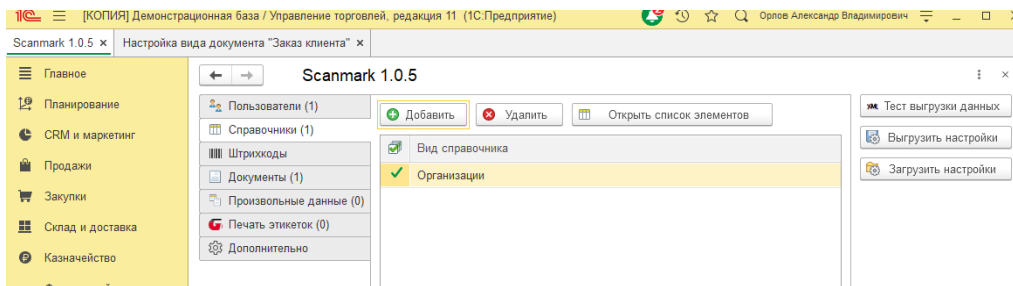
Отмечаем флажки у реквизитов Ссылка и Наименование. Нажимаем кнопку «Выполнить».



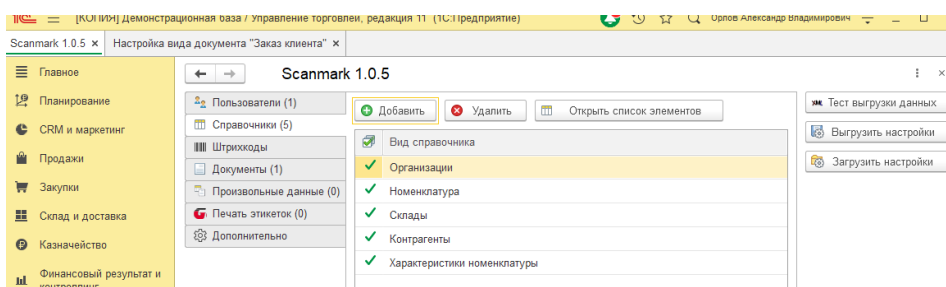
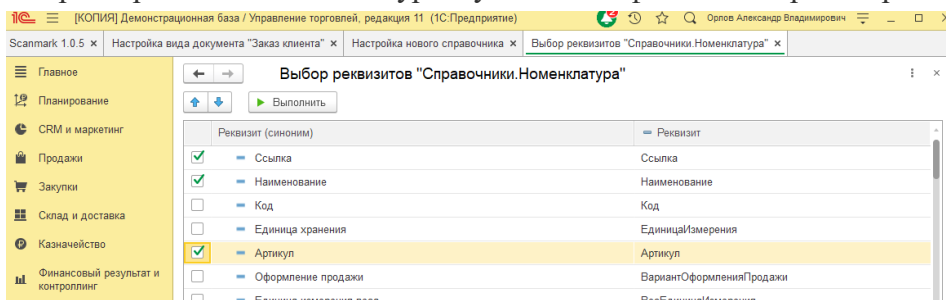
Получаем настройку справочника «Организации».



Нажимаем «Записать и закрыть».

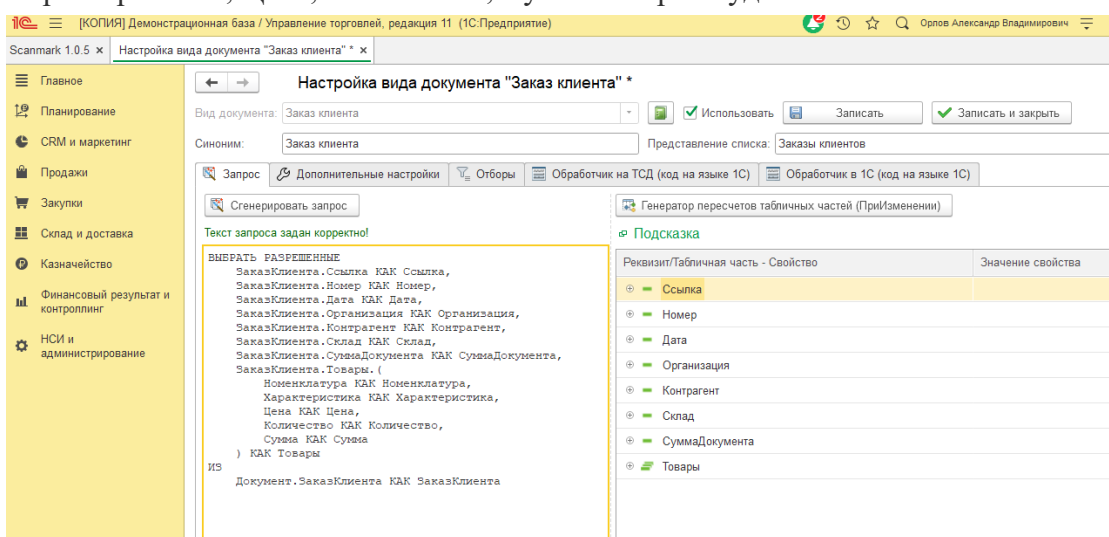


Таким же образом добавить справочники Склады, Контрагенты, Номенклатура и Характеристики номенклатуры. При настройке справочника «Номенклатура» дополнительно установим флажок в строке с реквизитом «Артикул», при настройке справочника «Характеристики номенклатуры» установим флажок в строке с реквизитом «Владелец».



Теперь вернемся к настройке документа «Заказ клиента».

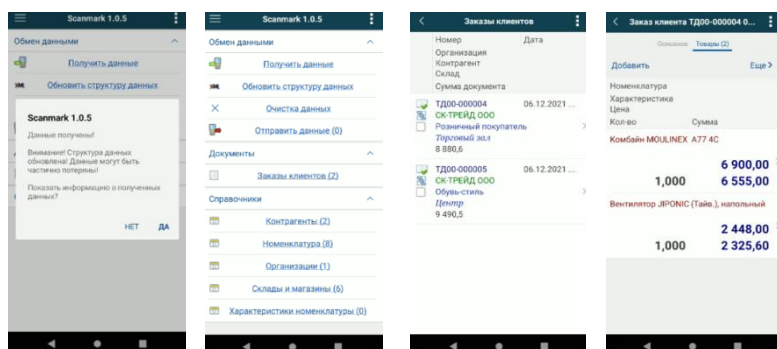
В левой части переставим строки запроса местами с порядком: Ссылка, Номер, Дата, Организация, Контрагент, Склад, СуммаДокумента. В табличной части: Номенклатура, Характеристика, Цена, Количество, Сумма. Запрос будет таким:



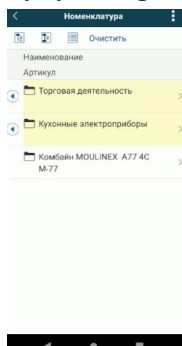
В правой части настройки, раскроем дерево и изменим значения свойств следующих реквизитов:

- «Ссылка»: Видимость = Нет;
- «Номер»: Ширина = 10, Шрифт, Цвет = 0,0,128;
- «Дата»: Ширина = 15, Положение = В той же колонке, Цвет = 0,0,128;
- «Организация»: Шрифт, Цвет = 0,128, 0;
- «Контрагент»: Шрифт, Цвет = 29,47, 190;
- «Склад»: Шрифт, Курсив = Истина, Шрифт, Цвет = 29,47, 190;
- «СуммаДокумента»: ТолькоПросмотр = Истина;
- «Товары, Номенклатура»: Шрифт, Цвет = 128,0,0;
- «Товары, Характеристика»: Шрифт, Курсив = Истина, Шрифт, Цвет = 128,0,0;
- «Товары, Цена»: ТолькоПросмотр = Истина; Шрифт, Размер = 12, Жирность = Истина, Цвет = 0,0,128;
- «Товары, Количество»: Синоним = Кол-во, Шрифт, Размер = 12, Жирность = Истина;
- «Товары, Сумма»: ТолькоПросмотр = Истина, Положение = В той же колонке, Шрифт, Размер = 12, Жирность = Истина, Цвет = 0,0,128;

Сохраним настройку. На ТСД выполним «Получить данные». В результате получим новую структуру данных и обновятся ранее полученные документы «Заказ клиента». Мы увидим новый улучшенный интерфейс.



Кроме того, теперь при попытке выбора номенклатуры, организации, склада, контрагента будут открыты формы выбора элементов справочника соответствующего вида.



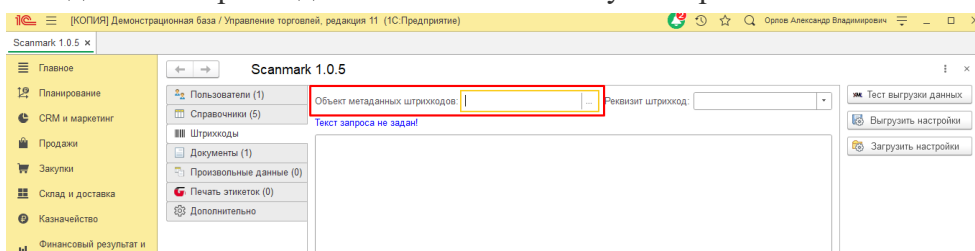


### Задача 3: Настроить данные штрихкодов.

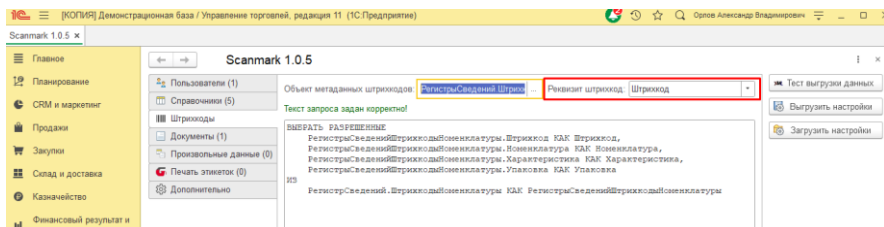
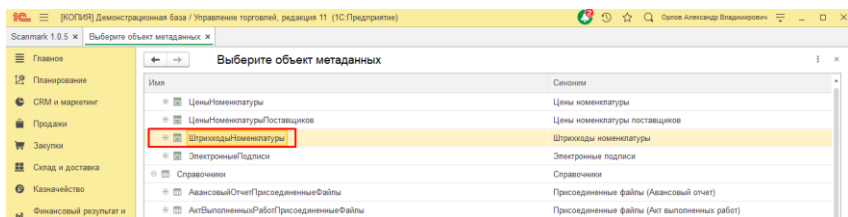
В ПО «Scanmark» имеется возможность получать дополнительный набор данных «Штрихкоды». Это набор данных, представляющий из себя таблицу, не передается независимо, а набирается при каждой передаче данных из «1С:Предприятие» в приложение на ТСД из состава объектов (документы и справочники) описанных в обработке настройки Scanmark». Структура данных таблицы «Штрихкоды» определяется текстом запроса на языке 1С. Текст не должен содержать конструкцию «ГДЕ», условия фильтра определяет само ПО «Scanmark» на основе пакета передаваемых данных.

Для получения данных по штрихкоду в настройках объектов в модулях обработчиков используйте функцию «ДанныеШтрихкодов». Описание функции см. раздел «Процедуры и функции обработчиков «Scanmark»» данного руководства.

В обработке «Scanmark» перейдем на закладку «Данные штрихкодов». В поле «Объект метаданных штрихкодов» нажмем на кнопку выбора.

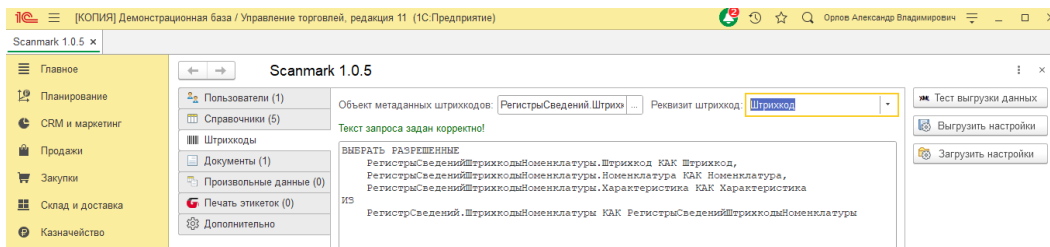


В окне «Выберите объект метаданных» выберем строку дерева Регистры сведений, ШтрихкодыНоменклатуры.

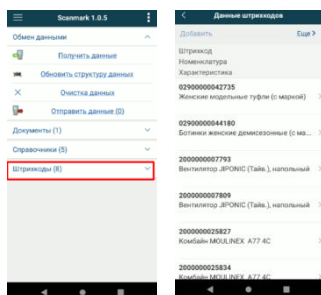


В нижней части формы появится текст запроса, поле «Реквизит штрихкод» заполнится значением «Штрихкод».

В запросе объектов штрихкода присутствует реквизит «Упаковка». Этот объект учета (Справочник «УпаковкиЕдиницыИзмерения») мы не добавляли в нашу настройку. Поэтому уберем данный реквизит. Для этого подправим текст запроса:



Теперь на ТСД еще раз получим данные:



В результате в основной форме на ТСД появился пункт с информацией о штрихкодах. В дальнейшем эти данные мы можем использовать при работе со сканером штрихкода.

**Задача 4: Добавить в настройку номенклатуры информацию об остатке и цене.**

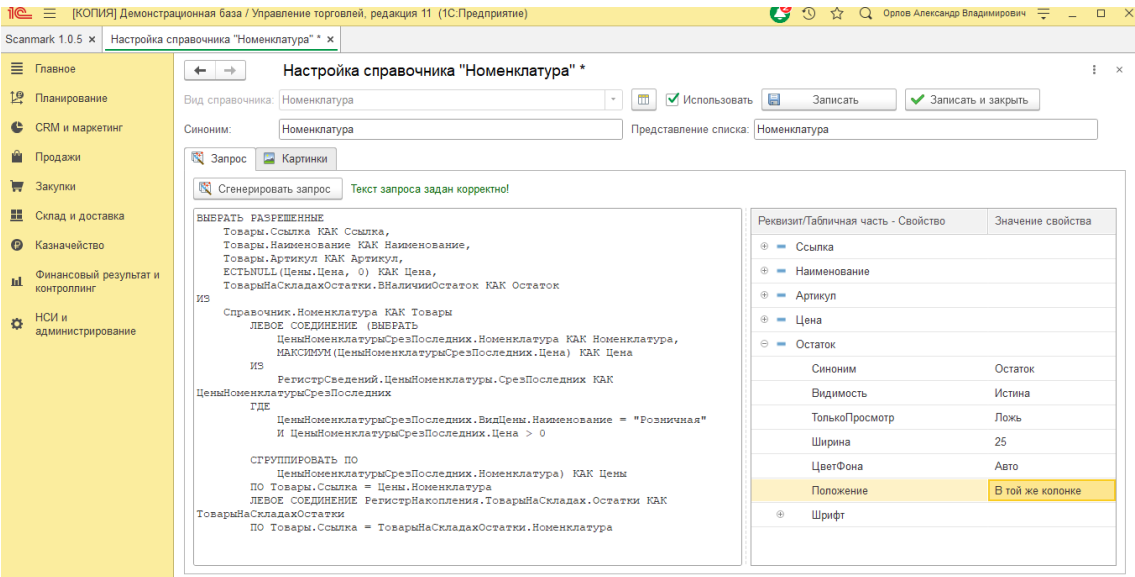
В обработке «Scanmark» откроем настройку справочника «Номенклатура» и изменим текст запроса на:

```
ВЫБРАТЬ РАЗРЕШЕННЫЕ
Товары.Ссылка КАК Ссылка,
Товары.Наименование КАК Наименование,
Товары.Артикул КАК Артикул,
ЕСТЬNULL(Цены.Цена, 0) КАК Цена,
ТоварыНаСкладахОстатки.ВНаличиОстаток КАК Остаток
ИЗ
Справочник.Номенклатура КАК Товары
ЛЕВОЕ СОЕДИНЕНИЕ (ВЫБРАТЬ
ЦеныНоменклатурыСрезПоследних.Номенклатура КАК Номенклатура,
МАКСИМУМ(ЦеныНоменклатурыСрезПоследних.Цена) КАК Цена
ИЗ
РегистрСведений.ЦеныНоменклатуры.СрезПоследних КАК ЦеныНоменклатурыСрезПоследних
ГДЕ
ЦеныНоменклатурыСрезПоследних.ВидЦены.Наименование = "Розничная"
И ЦеныНоменклатурыСрезПоследних.Цена > 0

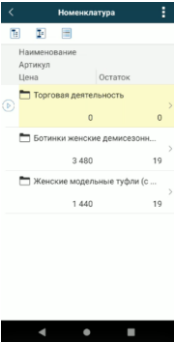
СГРУППИРОВАТЬ ПО
ЦеныНоменклатурыСрезПоследних.Номенклатура) КАК Цены
ПО Товары.Ссылка = Цены.Номенклатура
ЛЕВОЕ СОЕДИНЕНИЕ РегистрНакопления.ТоварыНаСкладах.Остатки КАК ТоварыНаСкладахОстатки
ПО Товары.Ссылка = ТоварыНаСкладахОстатки.Номенклатура
```

В правой части настройки, раскроем дерево и изменим значения свойств следующих реквизитов:

- «Остаток»: Положение = В той же колонке;



Сохраним настройку справочника. На ТСД в через команду «Получить данные» обновим данные. В результате увидим обновленный список номенклатуры, где кроме наименования и артикула есть информация о цене и остатке:

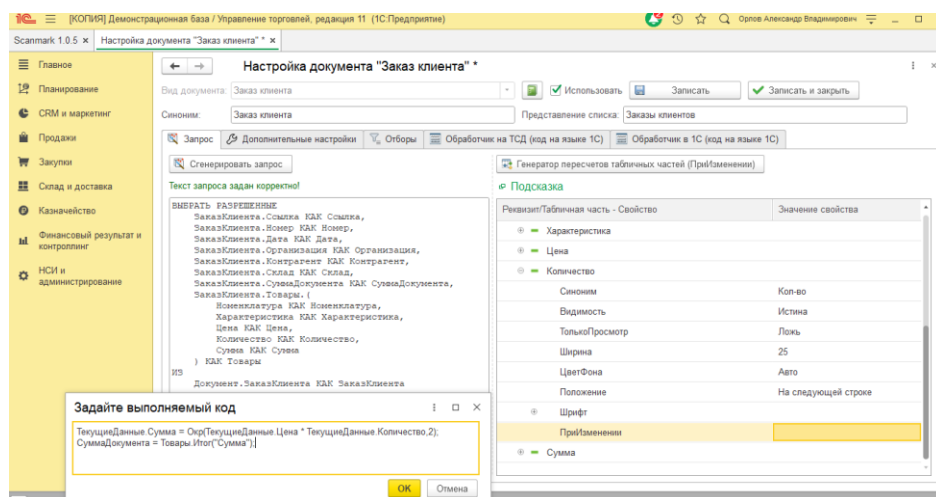


## Задача 5: Добавить функционал пересчета данных в табличной части «Товары» документа «Заказ клиента»

Текущая настройка позволяет получить документы «Заказ клиента» на ТСД. В форме документа «Заказ клиента» на ТСД мы можем менять значения данных. Но при изменении данных, например, количества или цены, не происходит пересчета суммы. В данной задаче мы выполним такую настройку.

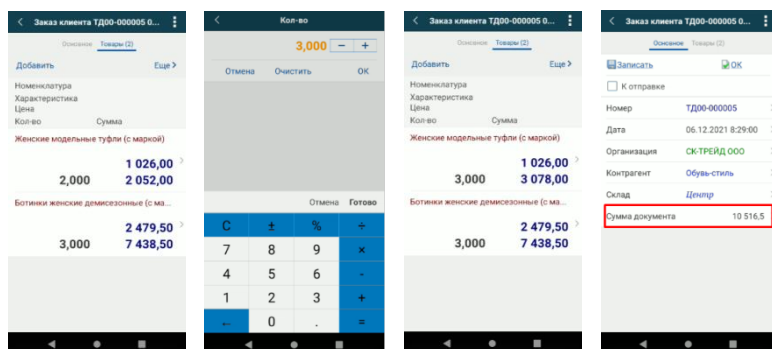
Откройте настройку документа «Заказ клиента» в обработке «Scanmark». В дереве реквизитов перейдем к реквизиту «Количество» и в свойстве «ПриИзменении» добавим код:

```
ТекущиеДанные.Сумма = Окp(ТекущиеДанные.Цена * ТекущиеДанные.Количество, 2);  
СуммаДокумента = Товары.Итог("Сумма");
```



Сохраним настройку. Обновим данные на ТСД и проверим как работает пересчет при изменении количества в строке «Товары».

Меняем количество в первой строке «Товары»:



Мы видим, что при изменении количество с 2 на 3 пересчиталась сумма по строке, а также общая сумма документа.

Как видно из текста свойства «ПриИзменении» - это код на языке 1С. Перед выполнением кода были инициализированы переменные по именам реквизитов и заполнены значениями этих реквизитов, реквизиты табличных частей доступны через обращение к таблице по имени через точку, например «Товары.Номенклатура». Кроме того, при изменении данных в табличной части инициализируется текущая строка «ТекущиеДанные» с реквизитами табличной части, например, «ТекущиеДанные.Количество» или «ТекущиеДанные.Сумма».

Код пересчета мы задали, что называется, «руками». Код такого пересчета можно задать с помощью кнопки «Генератор пересчетов табличных частей (ПриИзменении)». Будет произведен анализ структуры данных и заполнены свойства «При изменении» соответствующих логике реквизитов. После «автозаполнения» следует проверить синтаксис и

уместность кода, после чего обновить структуру данных на ТСД и проверить работу кода для выявления возможных ошибок.

Scalmark 1.0.5 x [КОПИЯ] Демонстрационная база / Управление торговлей, редакция 11 (1С:Предприятие) Орлов Александр Владимирович

Настройка вида документа "Заказ клиента" \* x

Вид документа: Заказ клиента \* ☒ Использовать

Синоним: Заказ клиента Представление списка: Заказы клиентов

Запрос ☒ Дополнительные настройки ☐ Отборы ☐ Обработка на ТСД (код на языке 1С) ☐ Обработка в 1С (код на языке 1С)

Текст запроса задан корректно!

Генератор пересчетов табличных частей (ПриИзменении)

Подсказка

Реквизит/Табличная часть - Свойство	Значение свойства
Ссылка	
Номер	
Дата	
Организация	
Контрагент	
Склад	
СуммаДокумента	
Товары	

ВЫБРАТЬ РАЗРЕШЕНИЕ  
ЗаказКлиента.Ссылка КАК Ссылка,  
ЗаказКлиента.Номер КАК Номер,  
ЗаказКлиента.Дата КАК Дата,  
ЗаказКлиента.Организация КАК Организация,  
ЗаказКлиента.Контрагент КАК Контрагент,  
ЗаказКлиента.Склад КАК Склад,  
ЗаказКлиента.СуммаДокумента КАК СуммаДокумента,  
ЗаказКлиента.Товары: (  
Номенклатура КАК Номенклатура,  
Характеристика КАК Характеристика,  
Цена КАК Цена,  
Количество КАК Количество,  
Сумма КАК Сумма  
) КАК Товары  
ИЗ  
Документ.ЗаказКлиента КАК ЗаказКлиента

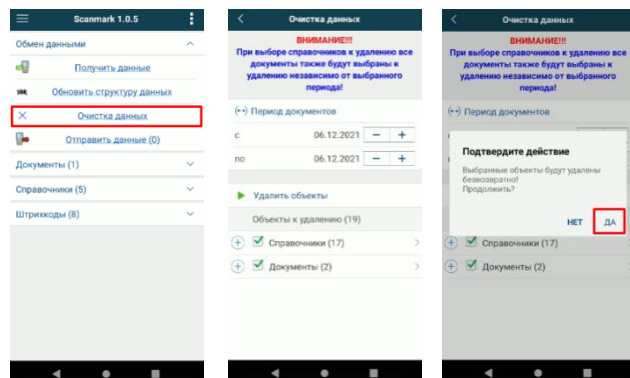
Сообщения:

— Были сгенерированы формулы пересчета реквизитов. Товары.Цена, Товары.Сумма.  
— Проверьте формулы (свойство табличной части "ПриИзменении") и исправьте при необходимости.

## Задача 6: Создать и заполнить данными документ «Заказ клиента» на ТСД

Чтобы создать и заполнить данными документ «Заказ клиента» с настроенной структурой данных в предыдущих задачах нам нужно иметь элементы объектов учета: организация, склад, контрагент, номенклатура, характеристики номенклатуры (при необходимости). Эти данные должны быть получены из «1С:Предприятие» до создания документа «Заказ клиента».

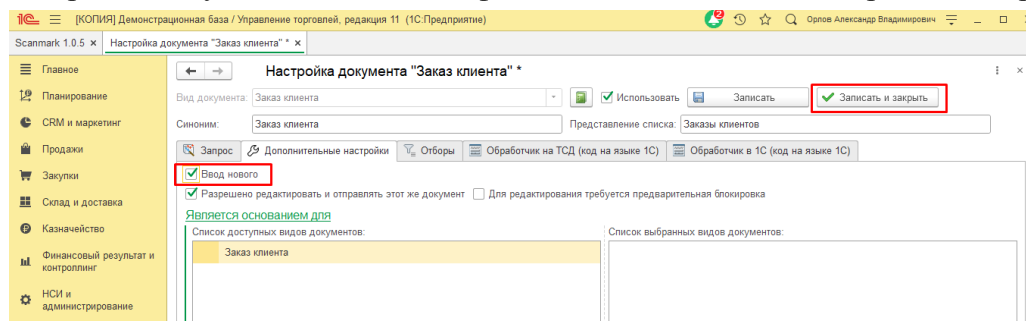
Для «чистоты» решения задачи сделаем очистку данных на ТСД:



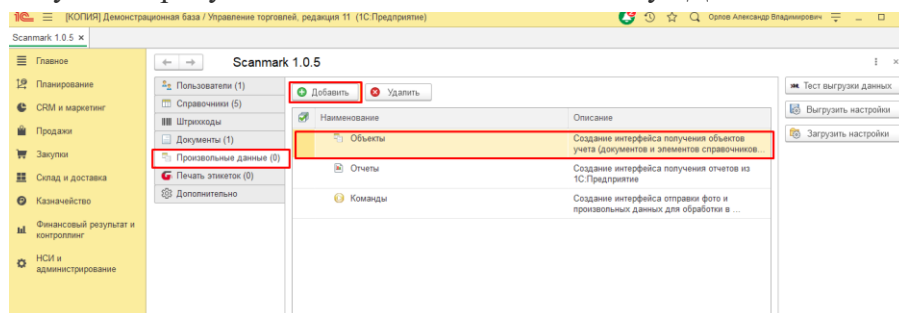
В обработке «1С:Предприятие» с помощью настройки «Произвольные данные / Объекты» мы можем настроить получение различных элементов учета, которые описаны в структуре данных на страницах «Настройка справочников» и «Настройка документов». С помощью конструктора запросов мы создадим запрос, при выполнении которого командой с ТСД будут переданы данные согласно параметров запроса.

Предположим типовую ситуацию: сотрудник в торговом зале вместе с покупателем делает подбор товаров в заказ на ТСД. При наборе такого заказа используется одна организация, один склад и один покупатель «Розничный продавец». Создадим такую настройку.

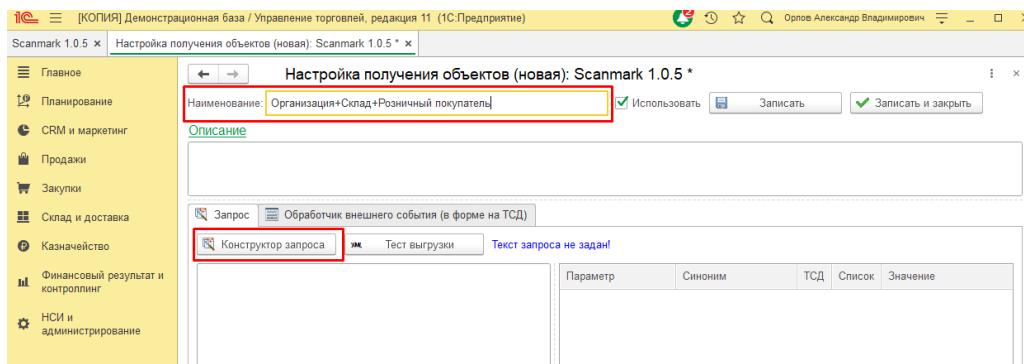
Чтобы создавать новый «Заказ клиента» на ТСД, нужно разрешить данное действие в настройке документа. Установим флажок «Ввод нового» и сохраним настройку:



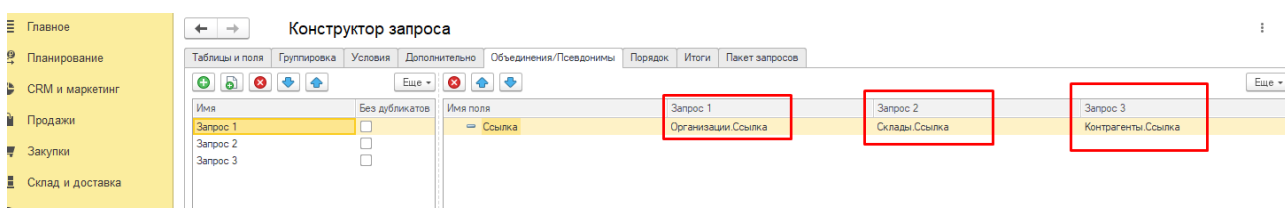
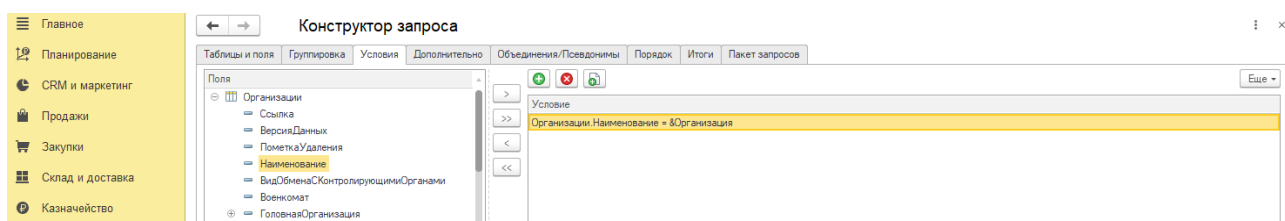
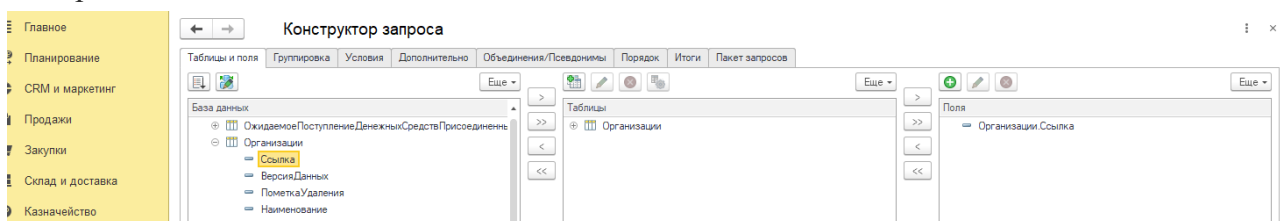
В обработке «Scanmark» переходим на страницу «Произвольные данные», устанавливаем текущую строку «Объекты» и нажимаем кнопку «Добавить»:



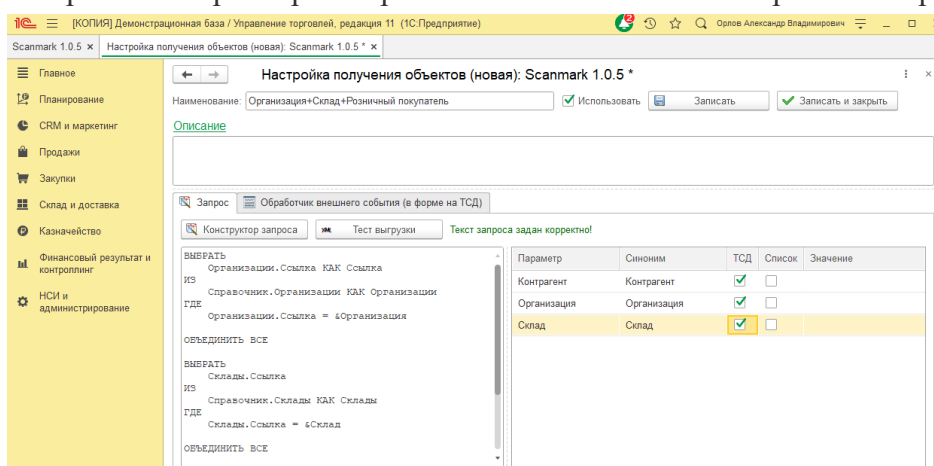
Задаем наименование «Организация+Склад+Розничный покупатель», нажимаем на кнопку «Конструктор запроса»:



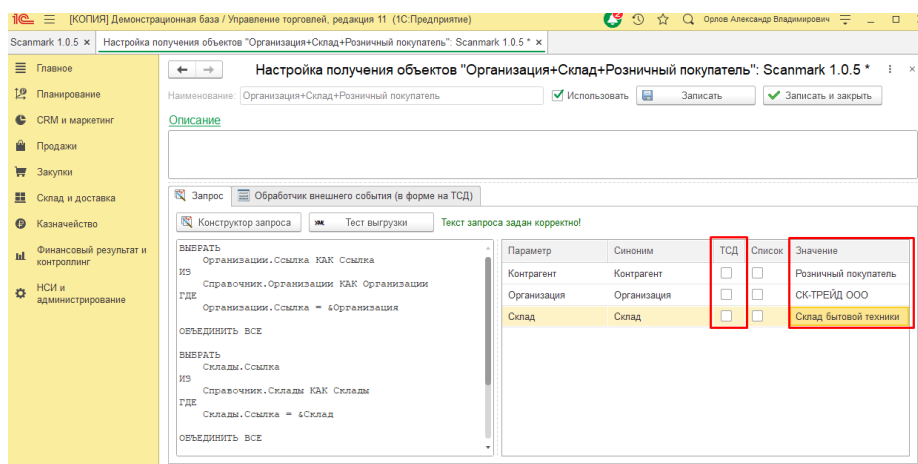
В конструкторе запросов добавляем ссылки на справочники Организация, Склад и Контрагента:



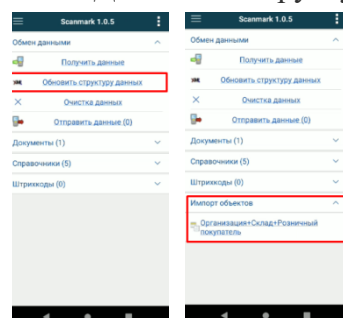
Нажимаем кнопку «Ок», в результате сформируется текст запроса в правой части окна настройки и параметры запроса в левой части окна настройки и параметры запроса:



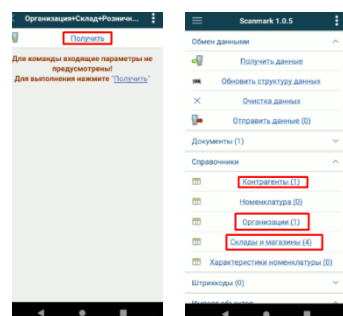
Нажимаем кнопку «Ок», в результате сформируется текст запроса в правой части окна настройки и параметры запроса в левой части окна настройки и параметры запроса. Уберем флажки из колонки «ТСД» в параметрах, а в колонке «Значение» скопируем наименования объектов, после чего сохраним настройку:



На ТСД обновим структуру данных. В результате появится новый объект:



Откроем «Организация+Склад+Розничный покупатель» и получим данные. Если проверить данные справочников, то мы увидим, что у нас есть элементы контрагентов, организаций и складов:



В обработке «Scanmark» на закладке «Произвольные данные» сделаем ещё одну настройку типа «Объекты» для получения элементов номенклатуры и характеристик номенклатуры. Зададим название «Номенклатура по остаткам», с помощью конструктора запросов создадим запрос:

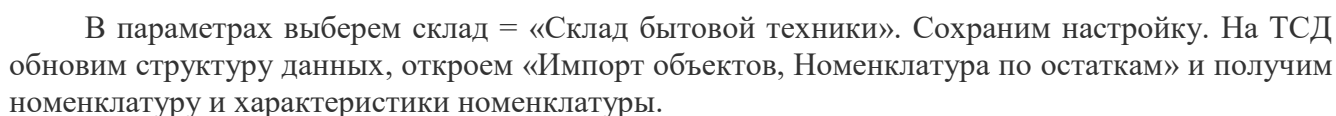
```

ВЫБРАТЬ
    ТоварыНаСкладахОстатки.Номенклатура КАК Номенклатура,
    ТоварыНаСкладахОстатки.Характеристика КАК Характеристика
ИЗ
    РегистрНакопления.ТоварыНаСкладах.Остатки КАК ТоварыНаСкладахОстатки
ГДЕ
    ТоварыНаСкладахОстатки.ВНаличииОстаток > 0
    И ТоварыНаСкладахОстатки.Склад = &Склад

СГРУППИРОВАТЬ ПО
    ТоварыНаСкладахОстатки.Номенклатура,
    ТоварыНаСкладахОстатки.Характеристика
  
```

В параметрах выберем склад = «Склад бытовой техники».





Теперь для создания нового «Заказ клиента» на ТСД у нас есть все необходимые данные. Кроме того, в задаче 5 уже настроены пересчеты документа. В этой задаче мы расширим функционал и добавим возможность обработки данных полученных от сканера штрихкода.

В обработке «Scanmark» в настройке документа «Заказ клиента» перейдем на страницу «Обработчик ТСД», в таблице слева сделаем текущим событие «ВнешнееСобытие» и нажмем кнопку «Генератор кода обработчика». В результате будет сформирован текст кода в текстовом поле правой части формы. Специалисту следует убедиться в корректности кода и при необходимости исправить или дополнить код согласно решаемой задаче обработки данных.



В сформированном коде мы видим конструкцию `ДанныеШтрихкода (Данные, Истина)`. Это специальная функция в приложении «Scanmark» на ТСД, которая вернет набор записей, как это задано в настройке «Данные штрихкодов», где параметр «Данные» - это полученный штрихкод при сканировании, а значение второго параметра «Истина» указывает на вызов окна вопроса о запросе данных из системы «1С:Предприятие», если данные по штрихкоду отсутствуют на ТСД. Данный код находит данные по товару, ищет в таблице документа «Товары» совпадение и, если не находит, то добавляет новую строку и заполняет данными. Количество увеличивается на 1 и делается пересчет. При таком коде цена никак не устанавливается. Добавим следующий код:

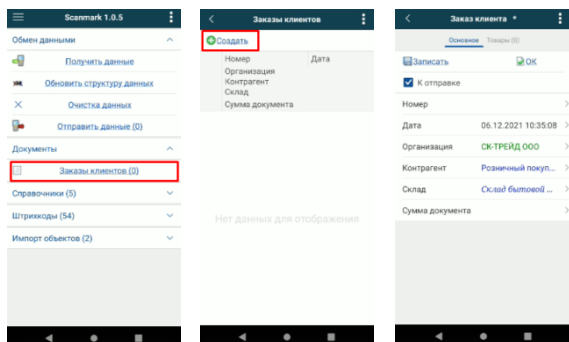
```
Товар = ОбъектСканмарк(Стр.Номенклатура);
Если ЗначениеЗаполнено(Товар) Тогда
    Стр.Цена = Товар.Цена;
КонецЕсли;
```

И снова мы видим новую конструкцию `ОбъектСканмарк(Стр.Номенклатура)`. Данные в приложении на ТСД хранятся не так, как это привычно видеть в 1С – объект с реквизитами. Мы не можем напрямую обратиться к реквизиту «Цена» номенклатуры, например, `Стр.Цена = Стр.Номенклатура.Цена;`. Для того, чтобы получить значение реквизита нужно создать объект структуры, к которому можно обратиться для получения значений реквизитов, в т.ч. и данных табличных частей. Если передаваемый параметр в функцию `ОбъектСканмарк(<Параметр>)` окажется пустым, то функция вернет значение `Неопределено`.

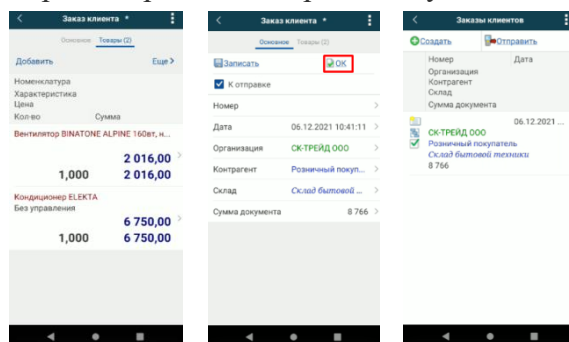
Доработанный код обработчика выглядит так:

```
Если Событие = "Barcode" Тогда
    ДанныеШтрихкода = ДанныеШтрихкода(Данные, Истина); //Данные = значение внешнего
    события
    Если ДанныеШтрихкода.Количество() > 0 Тогда
        ДанныеШтрихкода = ДанныеШтрихкода[0];
        Найден = Ложь;
        Для Каждого Стр Из Товары Цикл
            Если ДанныеШтрихкода.Номенклатура = Стр.Номенклатура
                И ДанныеШтрихкода.Характеристика = Стр.Характеристика Тогда
                    Найден = Истина;
                    Прервать;
            КонецЕсли;
        КонецЦикла;
        Если Не Найден Тогда
            Стр = Товары.Добавить();
            ЗаполнитьЗначенияСвойств(Стр, ДанныеШтрихкода);
            // добавлено++
            Товар = ОбъектСканмарк(Стр.Номенклатура);
            Если ЗначениеЗаполнено(Товар) Тогда
                Стр.Цена = Товар.Цена;
            КонецЕсли;
            // добавлено--
        КонецЕсли;
        Стр.Количество = Стр.Количество + 1;
        Стр.Сумма = Окр(Стр.Цена * Стр.Количество, 2);
        СуммаДокумента = Товары.Итог("Сумма");
        КонецЕсли;
    КонецЕсли;
```

Сохраним настройку. На ТСД обновим структуру данных и проверим работоспособность функционала, реализованного в задаче. Создадим новый «Заказ клиента», на закладке «Основное» заполним поля Организация, Склад, Контрагент.



Откроем закладку «Товары» и пробуем сканировать штрихкоды товаров. После подбора товаров сохраним и закроем документ.

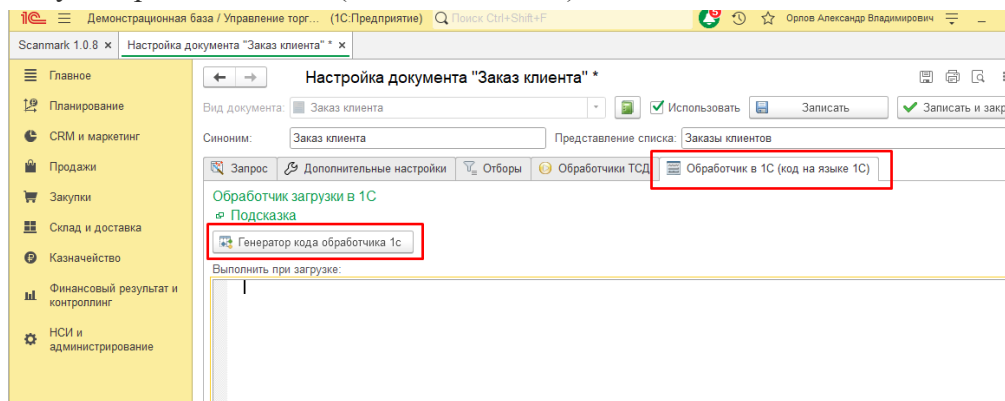


Задачу мы успешно решили. Мы создали и заполнили документ «Заказ клиента». Но передать в «1С:Предприятие» мы пока не можем, т.к. на стороне «1С:Предприятие» не задан код обработчика. Это мы реализуем в следующей задаче.

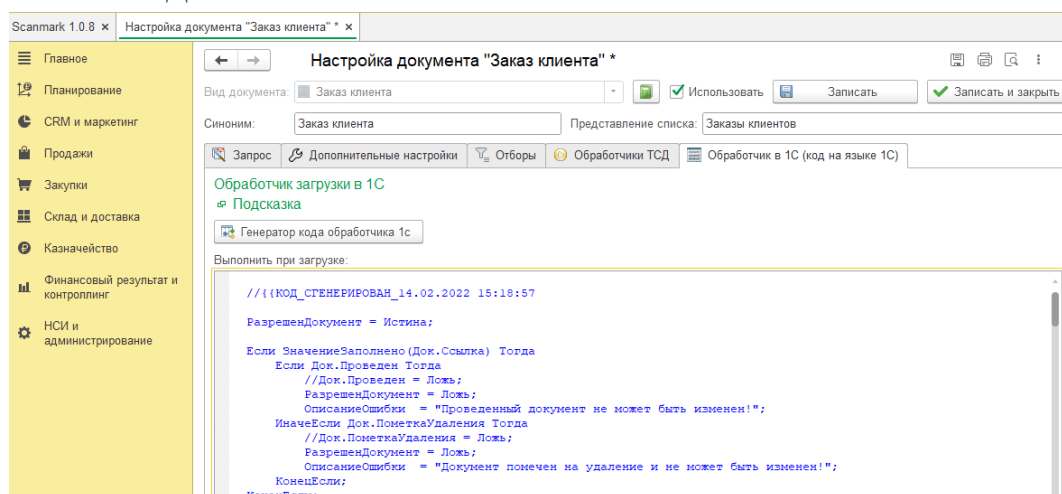
## Задача 7: Отправка и обработка документа «Заказ клиента» с ТСД в «1С:Предприятие»

В предыдущей задаче 6 мы создали и заполнили данными документ «Заказ клиента» на ТСД. Теперь нам нужно передать и обработать данные этого документа в «1С:Предприятие».

Откройте настройку документа «Заказ клиента» в обработке «Scanmark», перейдите на закладку «Обработчик в 1С (код на языке 1С)».

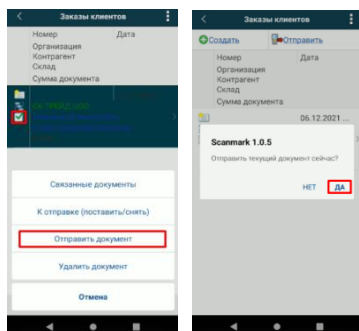


Нажмите кнопку «Генератор кода обработчика 1с». В поле «Выполнить при загрузке» будет сформирован код, который будет выполнен при получении данных по документу «Заказ клиента» с ТСД:

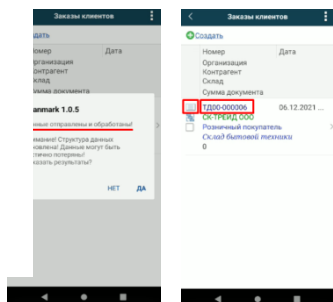


Сгенерированный таким образом код является шаблоном. При получении данных с ТСД и до выполнения кода по ссылке будет создан (или найден, например, при повторной отправке) документ «Заказ клиента», будет инициализирован Объект документа в переменную Док. Согласно заданной структуре будут инициализированы переменные и заполнены соответствующими данными. Табличные части структуры будут инициализированы данными в соответствующие имена таблиц значений.

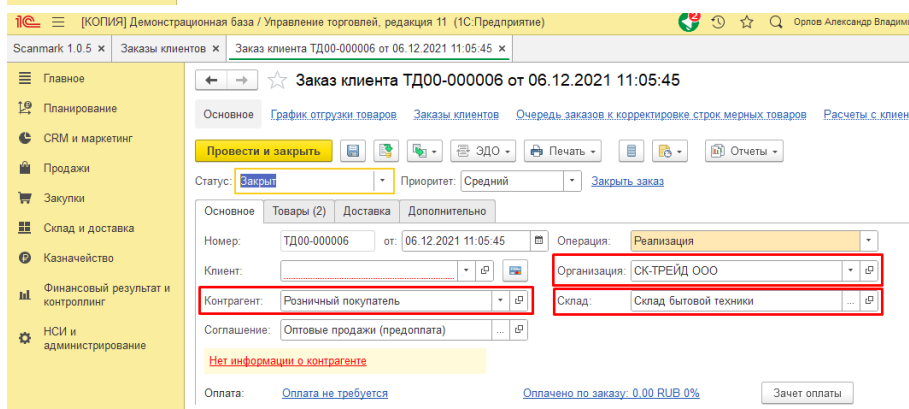
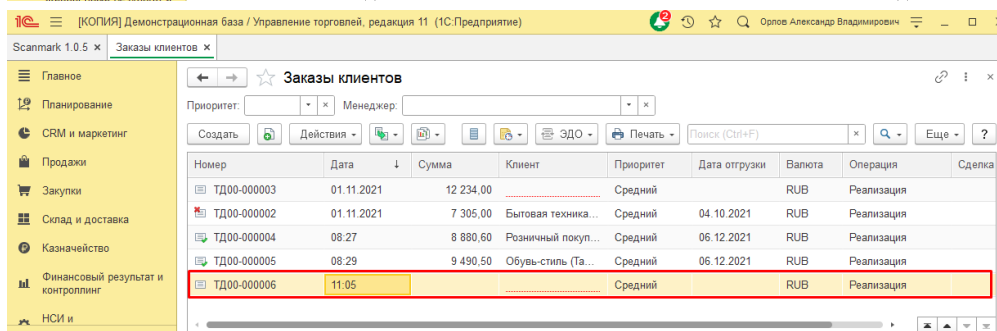
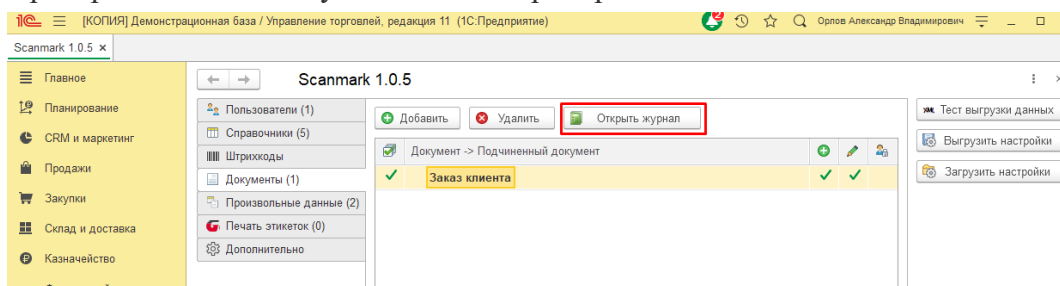
Специалист должен проверить и при необходимости внести изменения в код обработчика. После чего произвести проверку и отладку (при необходимости). Мы же оставим полученный код без изменений, сохраним настройку и проверим результат такой настройки на практике. На ТСД в журнале документов «Заказ клиента» вызовем контекстное меню (долгое нажатие на записи с документом) и выберем пункт «Отправить документ».

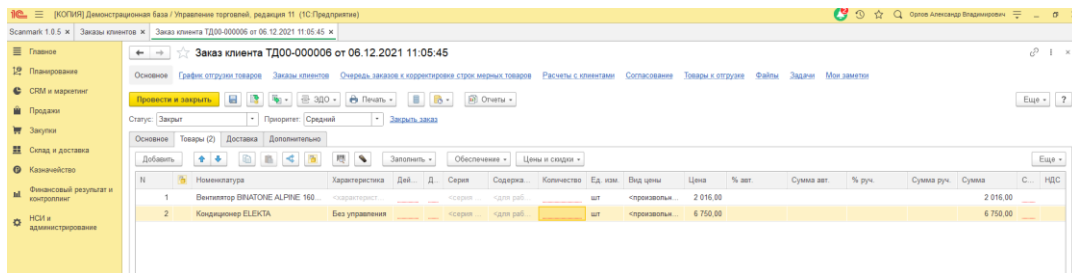


Дождемся результатов отправки. Если вы получите подобное сообщение, то это значит, что документ успешно принят и обработан в «1С:Предприятие». В документе появился номер и изменен статус.



Проверим наличие документа в «1С:Предприятие»:





Хотя мы видим, что документ заполнен данными не полностью, он присутствует в учетной системе. Т.е., некоторый промежуточный результат мы получили.

Для корректного заполнения всего документа следует внести изменения в код поля «Выполнить при загрузке». Вариант полного кода представлен ниже:

```
РазрешенДокумент = Истина;

Если ЗначениеЗаполнено (Док.Ссылка) Тогда
    Если Док.Проведен Тогда
        //Док.Проведен = Ложь;
        РазрешенДокумент = Ложь;
        ОписаниеОшибки = "Проведенный документ не может быть изменен!";
    ИначеЕсли Док.ПометкаУдаления Тогда
        //Док.ПометкаУдаления = Ложь;
        РазрешенДокумент = Ложь;
        ОписаниеОшибки = "Документ помечен на удаление и не может быть изменен!";
    КонецЕсли;
КонецЕсли;

Если РазрешенДокумент и ЗначениеЗаполнено (Док.Ссылка) Тогда
    Док.Товары.Очистить ();
    Док.ЭтапыГрафикаОплаты.Очистить ();
    Док.СкидкиНаценки.Очистить ();
    Док.НачислениеБонусныхБаллов.Очистить ();
    Док.ДополнительныеРеквизиты.Очистить ();
КонецЕсли;

Если РазрешенДокумент Тогда
    //Док.Номер =
    Если Не ЗначениеЗаполнено (Док.Дата) Тогда
        Док.Дата = ТекущаяДата ();
    КонецЕсли;

    Соглашение = Справочники.СоглашенияКлиентами.ПустаяСсылка ();
    Договор = Справочники.ДоговорыКонтрагентов.ПустаяСсылка ();

    Док.Контрагент = Контрагент;
    Док.Партнер = Док.Контрагент.Партнер;
    Док.Организация = Организация;
    //Док.Соглашение = "";
    //Док.Сделка = "";
    Док.Валюта = Справочники.Валюты.ПолучитьСсылку (Новый УникальныйИдентификатор ("26093579-c180-11e4-a7a9-000d884fd00d")); //RUB;
    Док.СуммаДокумента = СуммаДокумента;
    Док.ГрафикОплаты = Справочники.ГрафикиОплаты.ПолучитьСсылку (Новый УникальныйИдентификатор ("b49127da-752b-11df-b338-0011955cba6b"));
//Кредит;
    Док.Склад = Склад;
    Док.ЦенаВключаетНДС = Ложь;
    Док.Менеджер = Справочники.Пользователи.ПолучитьСсылку (Новый УникальныйИдентификатор ("a4212b3d-730a-11df-b338-0011955cba6b"));
//Федоров Борис Михайлович;
    //Док.ДополнительнаяИнформация = "";
    //Док.ДокументОснование = "";
    Док.НеОтгружатьЧастями = Ложь;
    Док.Статус = Перечисления.СтатусыЗаказовКлиентов.Закрыт;
    Док.МаксимальныйКодСтроки = 3;
    Док.ДатаСогласования = ТекущаяДата ();
    Док.Согласован = Ложь;
    Док.ФормаОплаты = Перечисления.ФормыОплаты.ПустаяСсылка ();
    Док.БанковскийСчет = Справочники.БанковскиеСчетаОрганизаций.ПолучитьСсылку (Новый УникальныйИдентификатор ("51ed67dd-7220-11df-b336-0011955cba6b")); //АКБ "ВПС" (ЗАО), Торговый дом "Комплексный" (RUB);
    //Док.БанковскийСчетКонтрагента = "";
    //Док.Касса = "";
    Док.СуммаАвансаДоОбеспечения = 0;
    Док.СуммаПредоплатыДоОтгрузки = 0;
    Док.ДатаОтгрузки = Док.Дата;
    //Док.АдресДоставки = "";
    Док.НалогообложениеНДС = Перечисления.ТипыНалогообложенияНДС.ПродажаОблагаетсяНДС;
    Док.СкидкиРассчитаны = Истина;
    Док.ХозяйственнаяОперация = Перечисления.ХозяйственныеОперации.РеализацияКлиенту;

    //Док.Комментарий = "";
    //Док.НомерПоДаннымКлиента = "";
    Док.ДатаПоДаннымКлиента = Дата (1, 1, 1);
    //Док.Грузоотправитель = "";
    //Док.Грузополучатель = "";
    //Док.БанковскийСчетГрузоотправителя = "";
    //Док.БанковскийСчетГрузополучателя = "";
    //Док.ГруппаФинансовогоУчета = "";
    //Док.КартаЛояльности = "";
    //Док.Договор = "";
    Док.Подразделение = Справочники.СтруктураПредприятия.ПолучитьСсылку (Новый УникальныйИдентификатор ("51ed67e1-7220-11df-b336-0011955cba6b")); //Отдел продаж торгового направления;
    Док.Автор = ПараметрыСеанса.ТекущийПользователь;
    Док.ПорядокРасчетов = Перечисления.ПорядокРасчетов.ПоЗаказам;
    //Док.Назначение = Договор.Назначение;
```

```

Док.СпособДоставки = Перечисления.СпособыДоставки.Самовывоз;
//Док.ПеревозчикПартнер = "";
//Док.ЗонаДоставки = "";
Док.ВремяДоставкиС = Дата(1,1,1);
Док.ВремяДоставкиПо = Дата(1,1,1);
//Док.АдресДоставкиПеревозчика = "";
//Док.АдресДоставкиЗначенияПолей = "";
//Док.АдресДоставкиПеревозчикаЗначенияПолей = "";
//Док.ДополнительнаяИнформацияПоДоставке = "";
//Док.КонтактноеЛицо = "";
Док.Руководитель = Справочники.ОтветственныеЛицаОрганизаций.ПолучитьСсылку(Новый УникальныйИдентификатор("bb251f0d-6b94-11e2-bfed-0015e9b8c48d")); //Исаков О.В.;
Док.ГлавныйБухгалтер = Справочники.ОтветственныеЛицаОрганизаций.ПолучитьСсылку(Новый УникальныйИдентификатор("bb251f10-6b94-11e2-bfed-0015e9b8c48d")); //Кострова В.В.;
Док.ВернутьМногооборотнуюТару = Ложь;
Док.СрокВозвратаМногооборотнойТары = 0;
Док.СостояниеЗаполненияМногооборотнойТары = Перечисления.СостоянияЗаполненияМногооборотнойТары.ПустаяСсылка();
Док.СуммаВозвратнойТары = 0;
//Док.НазначениеПлатежа = "";
Док.ТребуетсяЗалогЗаТару = Ложь;
Док.Приоритет = Справочники.Приоритеты.ПолучитьСсылку(Новый УникальныйИдентификатор("260937c5-c180-11e4-a7a9-000d884fd00d"));
//Средний;
//Док.ИдентификаторПлатежа = "";
Док.ОсобыеУсловияПеревозки = Ложь;
//Док.ОсобыеУсловияПеревозкиОписание = "";
//Док.НаправлениеДеятельности = "";
Док.ОплатаВВалюте = Ложь;
//Док.ОбъектРасчетов = "";
//Док.Сборщик = "";
//Док.Курьер = "";
//Док.АдресДоставкиЗначение = "";
//Док.АдресДоставкиПеревозчикаЗначение = "";
//Док.ЭтапГосконтрактаЕИС = "";

Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ ПЕРВЫЕ 1
|   СоглашенияСКлиентами.Ссылка КАК Ссылка
|ИЗ
|   Справочник.СоглашенияСКлиентами КАК СоглашенияСКлиентами
|ГДЕ
|   СоглашенияСКлиентами.Партнер = &Партнер
| И СоглашенияСКлиентами.Организация = &Организация
| И СоглашенияСКлиентами.Склад = &Склад
| И СоглашенияСКлиентами.Статус = &Статус
| И СоглашенияСКлиентами.ВидСоглашенияДляОграниченияИзменения = &ВидСоглашенияДляОграниченияИзменения";

Запрос.УстановитьПараметр("Контрагент", Справочники.Контрагенты.ПустаяСсылка());
Запрос.УстановитьПараметр("Партнер", Справочники.Партнеры.ПустаяСсылка());
Запрос.УстановитьПараметр("Организация", Организация);
Запрос.УстановитьПараметр("Склад", Склад);
Запрос.УстановитьПараметр("Статус", Перечисления.СтатусыСоглашенийСКлиентами.Действует);
Запрос.УстановитьПараметр("ВидСоглашенияДляОграниченияИзменения",
Перечисления.ВидыСоглашенийСКлиентамиДляОграниченияИзменения.Типовые);

РезультатЗапроса = Запрос.Выполнить();
Выборка = РезультатЗапроса.Выбрать();
Пока Выборка.Следующий() Цикл
    Соглашение = Выборка.Ссылка;
КонецЦикла;
Если Не ЗначениеЗаполнено(Соглашение) Тогда
    РазрешенДокумент = Ложь;
    ОписаниеОшибки = "Действующее соглашение с клиентом не найдено! Документ не может быть обработан!";
КонецЕсли;

КонецЕсли;

Если РазрешенДокумент Тогда
    Док.Соглашение = Соглашение;
    Док.Валюта = Соглашение.ВалютаВзаиморасчетов;
    Док.ГрафикОплаты = Соглашение.ГрафикОплаты;
    Док.ЦенаВключаетНДС = Соглашение.ЦенаВключаетНДС;
    Док.НеОтгружатьЧастями = Ложь;
    Док.Статус = Перечисления.СтатусыЗаказовКлиентов.КОбеспечению;
    Док.МаксимальныйКодСтроки = 9;
    Док.ДатаСогласования = Док.Дата;
    Док.ФормаОплаты = Соглашение.ФормаОплаты;

    Если Не ЗначениеЗаполнено(Док.БанковскийСчет) Тогда
        Запрос = Новый Запрос;
        Запрос.Текст =
"ВЫБРАТЬ ПЕРВЫЕ 1
|   ЗаказКлиента.БанковскийСчет КАК БанковскийСчет
|ИЗ
|   Документ.ЗаказКлиента КАК ЗаказКлиента
|ГДЕ
|   ЗаказКлиента.БанковскийСчет.Владелец = &Организация
| И ЗаказКлиента.Проведен = ИСТИНА
|
|УПОРЯДОЧИТЬ ПО
|   ЗаказКлиента.Дата УБЫВ";

        Запрос.УстановитьПараметр("Владелец", Организация);
        РезультатЗапроса = Запрос.Выполнить();
        Выборка = РезультатЗапроса.Выбрать();
        Пока Выборка.Следующий() Цикл
            Док.БанковскийСчет = Выборка.БанковскийСчет;
        КонецЦикла;
    КонецЕсли;
КонецЕсли;

Если РазрешенДокумент Тогда

```

```

Для Каждого СтрТЗ Из Товары Цикл
    СтрДок = Док.Товары.Добавить();
    ЗаполнитьЗначенияСвойств(СтрДок, СтрТЗ);
    СтрДок.ДатаОтгрузки = Док.ДатаОтгрузки;
    //СтрДок.Номенклатура = "";
    //СтрДок.Характеристика = "";
    //СтрДок.Упаковка = "";
    СтрДок.КоличествоУпаковок = СтрДок.Количество;
    //СтрДок.Количество = ;
    //СтрДок.ВидЦены = Справочники.ВидыЦен.ПолучитьСсылку(Новый УникальныйИдентификатор("73867e74-73c8-11df-b338-0011955cba6b"));
//Оптовая;
    //СтрДок.Цена = "";
    //СтрДок.Сумма = "";
    //СтрДок.УдалитьСтавкаНДС = Перечисления.СтавкиНДС.ПустаяСсылка();
    //СтрДок.СтавкаНДС = СтрДок.Номенклатура.ВидНоменклатуры.СтавкаНДС;
    //СтрДок.СуммаНДС = 0;
    //СтрДок.СуммаСНДС = 0;
    //СтрДок.ПроцентРучнойСкидки = 0;
    //СтрДок.СуммаРучнойСкидки = 0;
    //СтрДок.ПроцентАвтоматическойСкидки = 0;
    //СтрДок.СуммаАвтоматическойСкидки = 0;
    //СтрДок.ПричинаОтмены = "";
    //СтрДок.КодСтроки = 0;
    //СтрДок.Отменено = Ложь;
    //СтрДок.КлючСвязи = 0;
    //СтрДок.Склад = Справочники.Склады.ПолучитьСсылку(Новый УникальныйИдентификатор("6f87e83f-722c-11df-b336-0011955cba6b"));
//Центральный склад;
    //СтрДок.СрокПоставки = 0;
    //СтрДок.Содержание = "";
    //СтрДок.СтатусУказанияСерий = 0;
    СтрДок.ВариантОбеспечения = Перечисления.ВариантыОбеспечения.РезервироватьПоМереПоступления;
    //СтрДок.Серия = "";
    //СтрДок.НоменклатураНабора = "";
    //СтрДок.Обособленно = Ложь;
    //СтрДок.ХарактеристикаНабора = "";
    //СтрДок.ИдентификаторСтроки = "";
    //СтрДок.ВариантОбеспеченияДоИзмененияОбновлениемИБ = Перечисления.ВариантыОбеспечения.ПустаяСсылка();
    //СтрДок.Подразделение = "";
    //СтрДок.НоменклатураПартнера = "";
    //СтрДок.СуммаБонусныхБалловКСписанию = 0;
    //СтрДок.СуммаБонусныхБалловКСписаниюВВалюте = 0;
    //СтрДок.СуммаНачисленныхБонусныхБалловВВалюте = 0;

СтруктураПересчетаСуммы = ОбработкаТабличнойЧастиКлиентСервер.ПараметрыПересчетаСуммыНДСВСтрокеТЧ(Док);

СтруктураДействий = Новый Структура;
//СтруктураДействий.Вставить("ПроверитьХарактеристикуПовладельцу", СтрДок.Характеристика);
СтруктураДействий.Вставить("ПроверитьЗаполнитьУпаковкуПовладельцу", СтрДок.Упаковка);
СтруктураДействий.Вставить("ПересчитатьКоличествоЕдиниц");
СтруктураДействий.Вставить("ЗаполнитьСтавкуНДС", Новый Структура("НалогообложениеНДС, Дата, Организация", Док.НалогообложениеНДС,
Док.Дата, Организация));

СтруктураДействий.Вставить("ПересчитатьСуммуНДС", СтруктураПересчетаСуммы);
СтруктураДействий.Вставить("ПересчитатьСуммуСНДС", СтруктураПересчетаСуммы);
СтруктураДействий.Вставить("ПересчитатьСумму");
СтруктураДействий.Вставить("ПересчитатьСуммуСУчетомРучнойСкидки", Новый Структура("Очищать", Истина));
СтруктураДействий.Вставить("ПересчитатьСуммуСУчетомАвтоматическойСкидки", Новый Структура("Очищать", Истина));
СтруктураДействий.Вставить("ПроверитьЗаполнитьНазначение");

ДополнительнаяСтруктура = ОбработкаТабличнойЧастиКлиентСервер.ПолучитьСтруктуруЗаполненияУсловийПродажВСтрокеТЧ(Док);
ОбщегоНазначенияКлиентСервер.ДополнитьСтруктуру(СтруктураДействий, ДополнительнаяСтруктура, Истина);

ИспользоватьСоглашенияСКлиентами = Константы.ИспользоватьСоглашенияСКлиентами.Получить();
Если ИспользоватьСоглашенияСКлиентами И ЗначениеЗаполнено(Док.Соглашение) Тогда
    СтруктураДействий.Вставить("ЗаполнитьУсловияПродаж",
ОбработкаТабличнойЧастиКлиентСервер.ПолучитьСтруктуруЗаполненияУсловийПродажВСтрокеТЧ(Док));
Иначе
    СтруктураДействий.Вставить("ЗаполнитьЦенуПродажи",
ОбработкаТабличнойЧастиКлиентСервер.ПолучитьСтруктуруЗаполненияЦеныВСтрокеТЧ(Док));
КонецЕсли;

КэшированныеЗначения = Неопределено;
ТекущаяСтрокаСтруктура = ПолучитьТекущуюСтрокуСтруктурой(СтрДок, СтруктураДействий);

ОбработкаТабличнойЧастиСервер.ОбработатьСтрокуТЧ(ТекущаяСтрокаСтруктура, СтруктураДействий, КэшированныеЗначения);
ЗаполнитьЗначенияСвойств(СтрДок, ТекущаяСтрокаСтруктура);

КонецЦикла;

//СтрДок = Док.ЭтапыГрафикаОплаты.Добавить();
//СтрДок.ВариантОплаты =
//СтрДок.ДатаПлатежа =
//СтрДок.ПроцентПлатежа =
//СтрДок.СуммаПлатежа =
//СтрДок.ПроцентЗалогаЗаТару =
//СтрДок.СуммаЗалогаЗаТару =
//СтрДок.СуммаОтклоненияМерныхТоваров =
//СтрДок.Сдвиг =
//СтрДок.ВариантОтсчета =

//СтрДок = Док.СкидкиНаценки.Добавить();
//СтрДок.КлючСвязи =
//СтрДок.СкидкаНаценка =
//СтрДок.Сумма =
//СтрДок.НапомнитьПозже =

//СтрДок = Док.НачислениеБонусныхБаллов.Добавить();
//СтрДок.КлючСвязи =
//СтрДок.БонуснаяПрограммаЛояльности =
//СтрДок.ДатаНачисления =
//СтрДок.ДатаСписания =
//СтрДок.СуммаБонусныхБаллов =

//СтрДок = Док.ДополнительныеРеквизиты.Добавить();
//СтрДок.Свойство =

```

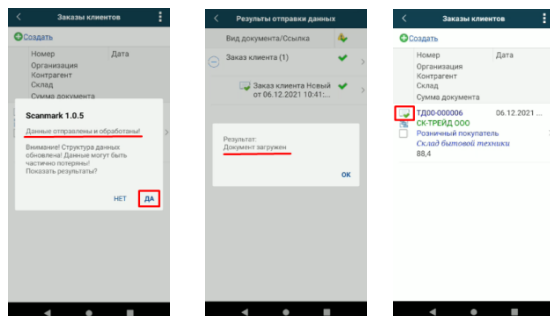


```
//СтрДок.Значение =
//СтрДок.ТекстоваяСтрока =

//Док.Записать ();
Док.Записать (РежимЗаписиДокумента.Проведение);
```

КонецЕсли;

Отправим документ в «1С:Предприятие» повторно. Получим результат: документ загружен и проведен.



Проверим документ «Заказ клиента» в «1С:Предприятие»:

The first screenshot shows the 'Заказы клиентов' (Customer orders) list in the 1C desktop application. The table below represents the data visible in the screenshot:

Номер	Дата	Сумма	Клиент	Приоритет	Дата отгрузки	Валюта	Операция	Сделка
TD00-000026	15.05.2017	11 107,40	Розничный покуп...	Средний	18.05.2017	RUB	Реализация	
TD00-000027	20.05.2017	117 625,00	Альфа	Средний	16.07.2017	RUB	Реализация	
TD00-000028	13.07.2017	117 625,00	Альфа	Средний	16.07.2017	RUB	Реализация	
TD00-000033	01.11.2021	12 234,00		Средний		RUB	Реализация	
TD00-000002	01.11.2021	7 305,00	Бытовая техника...	Средний	04.10.2021	RUB	Реализация	
TD00-000004	08.27	8 880,60	Розничный покуп...	Средний	06.12.2021	RUB	Реализация	
TD00-000005	08.29	9 490,50	Обувь-стиль (Та...	Средний	06.12.2021	RUB	Реализация	
TD00-000006	11.05	88,40	Розничный покуп...	Средний	06.12.2021	EUR	Реализация	

The second screenshot shows the detailed view of the selected order 'Заказ клиента TD00-000006 от 06.12.2021 11:05:45'. The 'Основное' (Main) tab is active, showing the following details:

- Номер: TD00-000006
- Дата: 06.12.2021 11:05:45
- Операция: Реализация
- Клиент: Розничный покупатель
- Организация: СК-ТРЕЙД ООО
- Контрагент: Розничный покупатель
- Склад: Склад бытовой техники
- Соглашение: Оптовые продажи (экспорт)
- Статус: К выполнению
- Приоритет: Средний
- Оплата: К оплате 06.12.2021 (100%)
- Оплачено по заказу: 0.00 EUR 0%

The third screenshot shows the 'Товары (2)' (Goods (2)) tab, displaying a list of goods:

N	Номенклатура	Характеристика	Действ.	Серия	Содержимое	Количество	Ед. изм.	Вид цены	Цена	% авт.
1	Вентилятор BINATONE ALPINE 160...	<характеристика>	Отг...	0...	<для раб...>	1,000	шт	Оптовая	20,33	
2	Кондиционер ELEKTA	Без управления	Отг...	0...	<для раб...>	1,000	шт	Оптовая	68,07	

Заказ корректно заполнен и проведен. В коде следует отметить две переменные, которые инициализируются до выполнения кода:

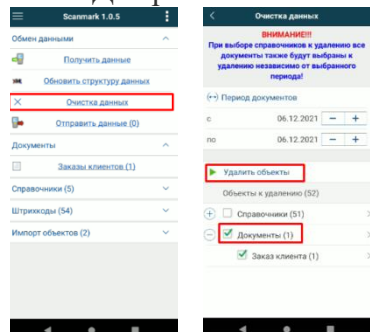
ОписаниеОшибки — используйте эту переменную для передачи описания ошибки или предупреждения пользователю на ТСД;

ДатаДок - использование имени переменной Дата нежелательно, поэтому переданное значение документа Дата инициализируется переменной ДатаДок.

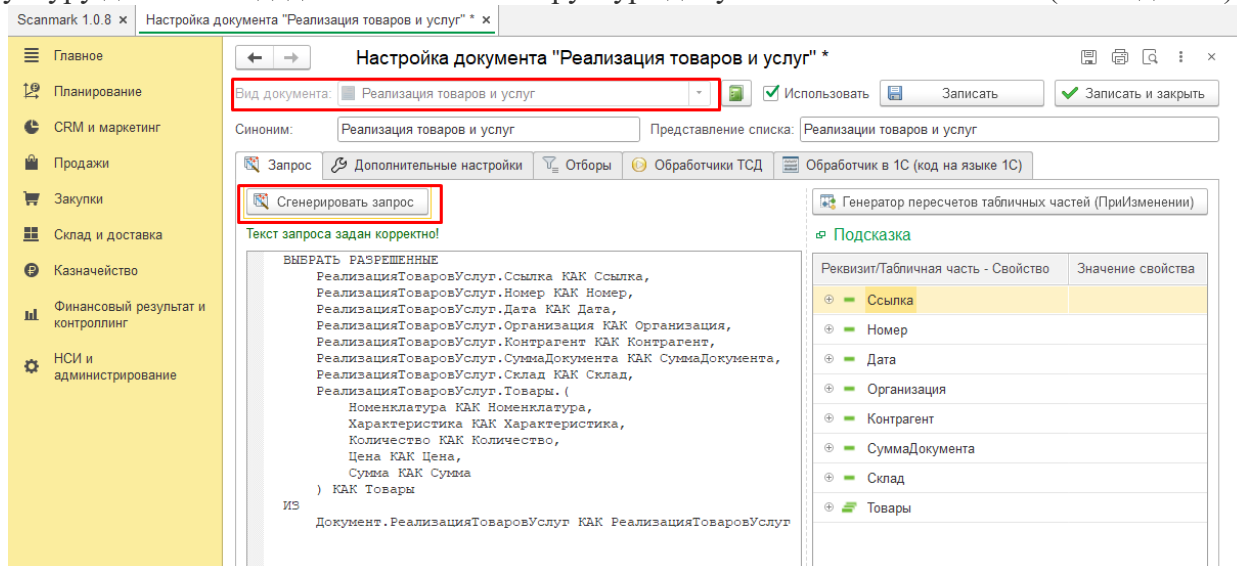
## Задача 8: На ТСД на основании документа «Заказ клиента» оформить, подобрать коды маркировки DataMatrix GS1 и отправить документ «Реализация товаров и услуг» в «1С:Предприятие»

В этой задаче рассмотрим новую типовую ситуацию: В учетной системе имеется заказ клиента. Пользователю (кладовщику) необходимо загрузить заказ клиента на ТСД, на основании этого заказа создать «Реализацию товаров и услуг», подобрать товар в соответствии с заказом клиента и отправить документ «Реализация товаров и услуг» в учетную систему «1С:Предприятие».

На ТСД через меню «Очистить данные» удалим документы:



В обработке «Scanmark» добавим новый документ «Реализация товаров и услуг». Структуру данных создадим по аналогии структуре документа «Заказ клиента» (см. Задача 2).



Изменим запрос – добавим три вычисляемых поля: «СуммаДокументаФакт», «КоличествоФакт» и «СуммаФакт». Текст запроса станет таким:

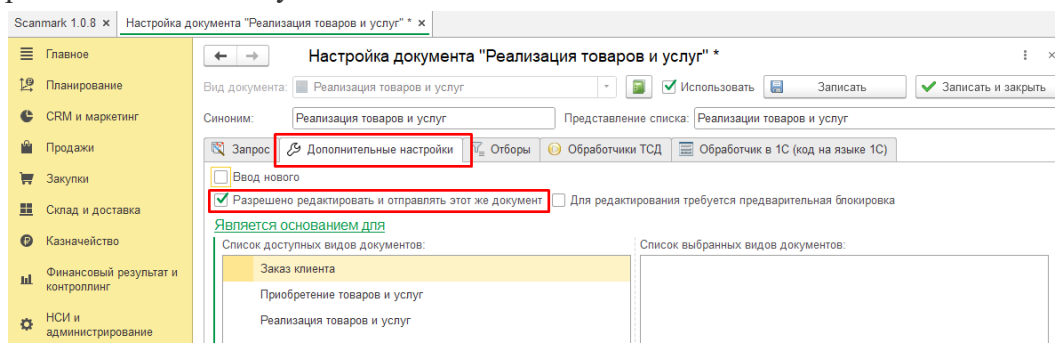
```
ВЫБРАТЬ РАЗРЕШЕННЫЕ
РеализацияТоваровУслуг.Ссылка КАК Ссылка,
РеализацияТоваровУслуг.Номер КАК Номер,
РеализацияТоваровУслуг.Дата КАК Дата,
РеализацияТоваровУслуг.Организация КАК Организация,
РеализацияТоваровУслуг.Контрагент КАК Контрагент,
РеализацияТоваровУслуг.Склад КАК Склад,
РеализацияТоваровУслуг.СуммаДокумента КАК СуммаДокумента,
РеализацияТоваровУслуг.СуммаДокумента КАК СуммаДокументаФакт,
РеализацияТоваровУслуг.Товары. (
    Номенклатура КАК Номенклатура,
    Характеристика КАК Характеристика,
    Цена КАК Цена,
    Количество КАК Количество,
    Сумма КАК Сумма,
    Количество КАК КоличествоФакт,
    Сумма КАК СуммаФакт
) КАК Товары
```

В правой части настройки, раскроем дерево и изменим значения свойств следующих реквизитов:

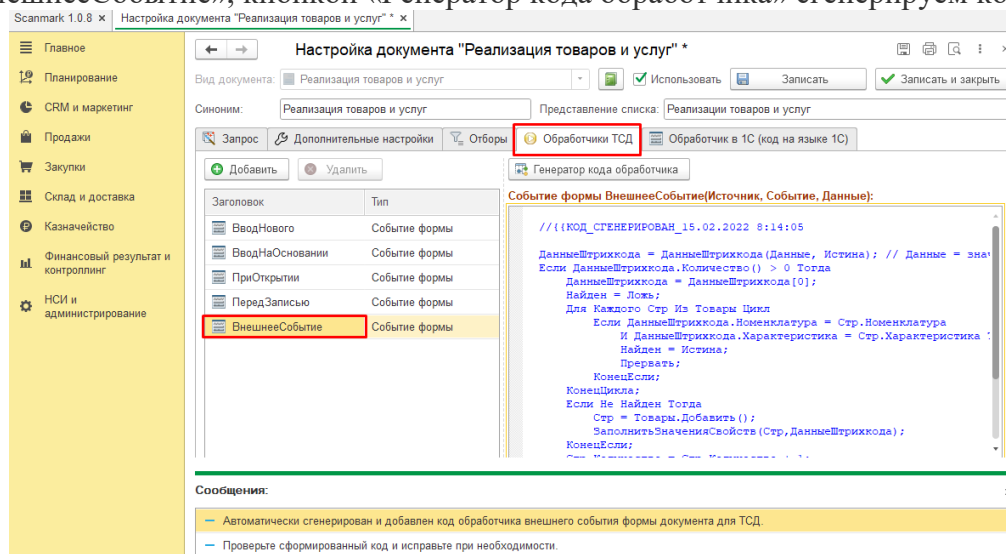
- «Ссылка»: Видимость = Нет;
- «Номер»: ТолькоПросмотр = Истина; Ширина = 10; Шрифт, Цвет = 0,0,128;
- «Дата»: Ширина = 15; Положение = В той же колонке; Шрифт, Цвет = 0,0,128;
- «Организация»: ТолькоПросмотр = Истина; Шрифт, Цвет = 0,128, 0;
- «Контрагент»: ТолькоПросмотр = Истина; Шрифт, Цвет = 29,47, 190;
- «Склад»: ТолькоПросмотр = Истина; Шрифт, Курсив = Истина, Цвет = 29,47, 190;
- «СуммаДокумента»: ТолькоПросмотр = Истина;
- «СуммаДокументаФакт»: Синоним = Сумма документа факт; ТолькоПросмотр = Истина;
- «Товары, Номенклатура»: Шрифт, Цвет = 128,0,0;
- «Товары, Характеристика»: Шрифт, Курсив = Истина; Шрифт, Цвет = 128,0,0;
- «Товары, Цена»: ТолькоПросмотр = Истина; Шрифт, Размер = 10, Жирность = Истина, Цвет = 0,128, 0;
- «Товары, Количество»: ТолькоПросмотр = Истина; Синоним = Кол-во Заказ; Шрифт, Размер = 12, Жирность = Истина;
- «Товары, Сумма»: ТолькоПросмотр = Истина; Положение = В той же колонке; Шрифт, Размер = 10, Жирность = Истина, Цвет = 0,128, 0;
- «Товары, КоличествоФакт»: Синоним = Кол-во факт; Шрифт, Размер = 12, Жирность = Истина;
- «Товары, СуммаФакт»: Синоним = Сумма факт; ТолькоПросмотр = Истина; Положение = В той же колонке, Шрифт, Размер = 12, Жирность = Истина, Цвет = 0,128, 0;

С помощью кнопки «Генератор пересчетов табличных частей (ПриИзменении)» сгенерируем код пересчета. Сохраним настройки.

На закладке «Дополнительные настройки» установим флажок «Разрешено редактировать и отправлять этот же документ».



На закладке «Обработчик на ТСД» в таблице слева установим текущее событие «ВнешнееСобытие», кнопкой «Генератор кода обработчика» сгенерируем код:



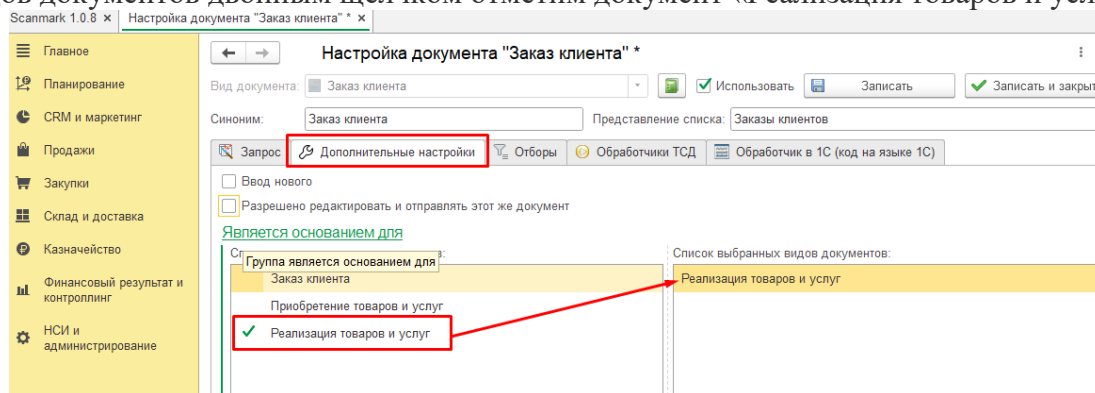
Сгенерированный код является шаблоном, и нас не устраивает такой код «по умолчанию».

Внесем изменения в текст кода:

```
Если Событие = "KeyBoard" Тогда
//ваш код обработки события нажатия клавиши
ИначеЕсли Событие = "RFID" Тогда
//ваш код обработки события считывания радиометок
ИначеЕсли Событие = "Barcode" Тогда
ДанныеШтрихкода = ДанныеШтрихкода(Данные); // Данные = значение внешнего события
Если ДанныеШтрихкода.Количество() > 0 Тогда
    ДанныеШтрихкода = ДанныеШтрихкода[0];
    Найден = Ложь;
    Для Каждого Стр Из Товары Цикл
        Если ДанныеШтрихкода.Номенклатура = Стр.Номенклатура
        И ДанныеШтрихкода.Характеристика = Стр.Характеристика Тогда
            Найден = Истина;
            Прервать;
        КонецЕсли;
    КонецЦикла;
Если Найден Тогда
    Стр.КоличествоФакт = Стр.КоличествоФакт + 1;
    Стр.СуммаФакт = Окр(Стр.Цена * Стр.КоличествоФакт, 2);
    СуммаДокументаФакт = Товары.Итог("СуммаФакт");
Иначе
    СтрТовар = СокрЛП(ДанныеШтрихкода.Номенклатура);
    Если ЗначениеЗаполнено(ДанныеШтрихкода.Характеристика) Тогда
        СтрТовар = СтрТовар + ", " + СокрЛП(ДанныеШтрихкода.Характеристика);
    КонецЕсли;
    Сообщить("Товар "" + СтрТовар + "" отсутствует в заказе и не может быть обработан!");
КонецЕсли;
Иначе
    Сообщить("Ничего не найдено: " + Символы.ПС + Данные);
КонецЕсли;
КонецЕсли;
```

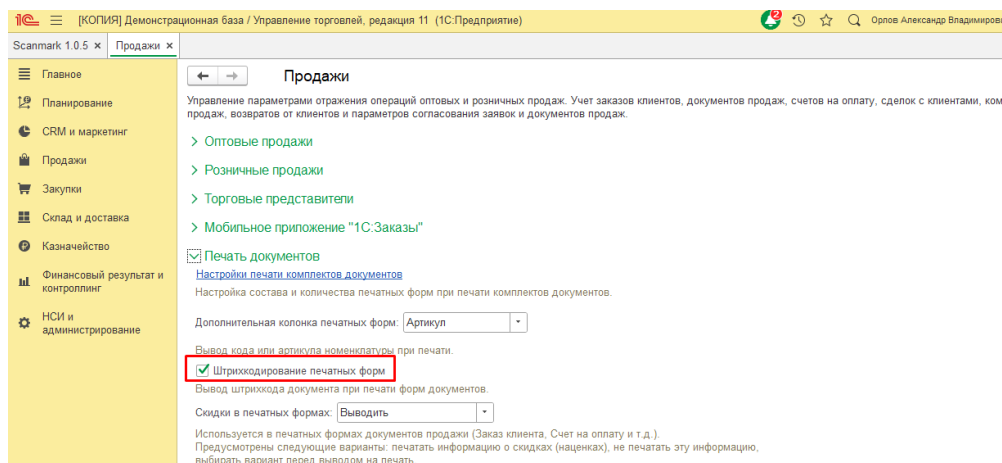
Сохраним настройку документа «Реализация товаров и услуг».

Сделаем дополнительные изменения в настройке документа «Заказ клиента». Откроем соответствующую настройку. На закладке «Дополнительные настройки» снимем флажки «Ввод нового» и «Разрешено редактировать и отправлять этот же документ». В списке доступных видов документов двойным щелчком отметим документ «Реализация товаров и услуг».

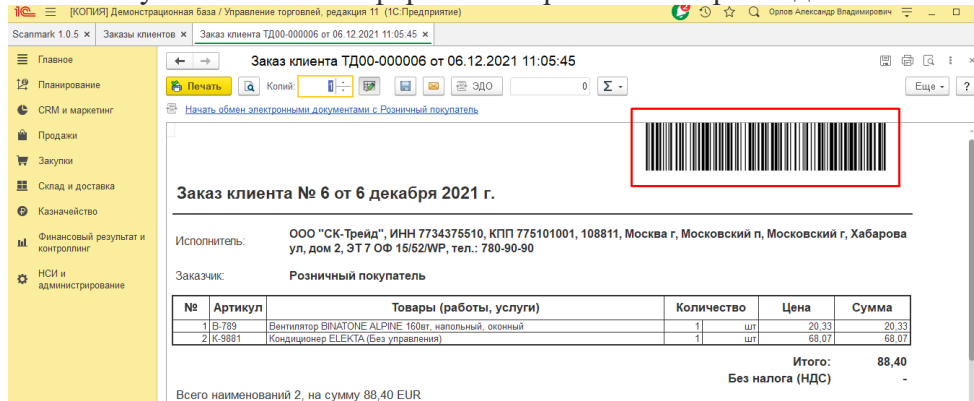


Сохраним настройку документа «Заказ клиента».

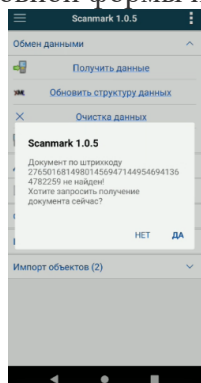
Рассмотрим возможные способы получения «Заказа клиента» на ТСД. Первый способ был описан ранее: через команду на ТСД «Получить данные». Второй способ применим, если в учетной системе настроено штрихкодирование печатных форм. Для типовой конфигурации «Управление торговлей 11.х» нужно перейти: НСИ и администрирование – Настройка НСИ и разделов – Продажи – Печать документов и убедиться, что установлен флажок «Штрихкодирование печатных форм».



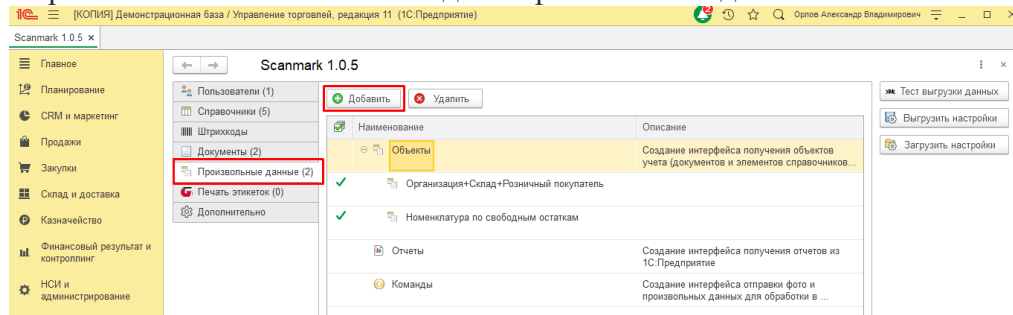
В этом случае в печатных формах отображается штрихкод:



Документ «Заказ клиента» на ТСД можно получить сканируя штрихкод документа из основной формы или из журнала документов «Заказы клиентов»:



Мы реализуем еще один способ через настройку произвольных данных «Объекты». В Обработке «Scanmark» на закладке «Произвольные данные» нажмем кнопку «Добавить».



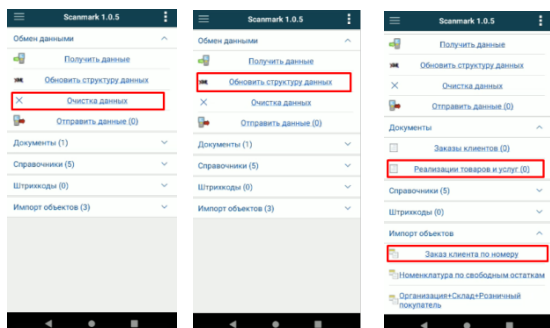
Зададим наименование «Заказ клиента по номеру», через «Конструктор запросов» сформируем текст запроса, а затем сохраним настройку:

ВЫБРАТЬ РАЗРЕШЕННЫЕ ПЕРВЫЕ 1  
ЗаказКлиента.Ссылка КАК Ссылка  
ИЗ  
Документ.ЗаказКлиента КАК ЗаказКлиента

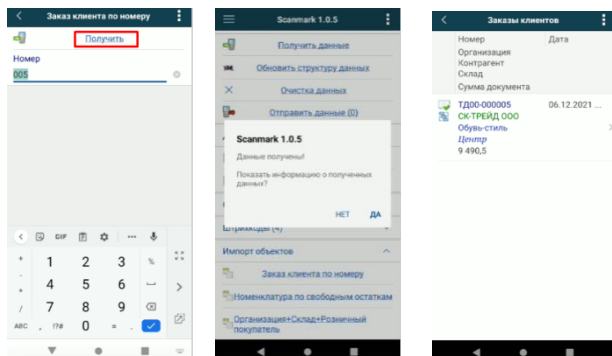
ГДЕ  
ЗаказКлиента.Номер ПОДОВНО "%" + &Номер

УПОРЯДОЧИТЬ ПО  
ЗаказКлиента.Дата УБЫВ

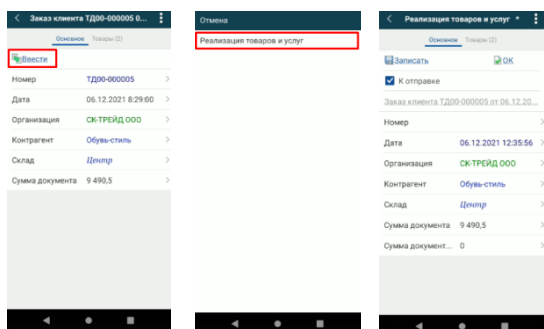
На ТСД последовательно выполним команды «Очистка данных» (отметим все объекты учета) и «Обновить структуру данных». Убедимся, что на ТСД появились новые виды объектов учета:



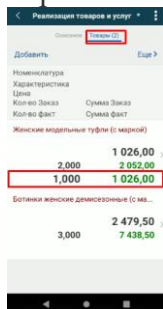
Откроем «Импорт объектов. Заказ клиента по номеру» и получим документ «Заказ клиента»:



Откроем документ «Заказ клиента». В нем появилась кнопка «Ввести». Нажмем ее, выберем создание документа «Реализация товаров и услуг». Будет открыта форма нового документа:



Перейдем на закладку «Товары» и начнем сканировать штрихкоды товаров. Документ обрабатывает данные корректно, что нам и было нужно на данном этапе решения задачи.



Мы видим, что у нас в документе присутствует обувная продукция. Такие товары подлежат обязательной маркировке в системе «Честный знак». Реализуем работу с учетом «Честного знака» в документе «Реализация товаров и услуг».

В первую очередь, добавим дополнительный реквизит «ОсобенностьУчета» в настройке справочника «Номенклатура». Исправим запрос:

```

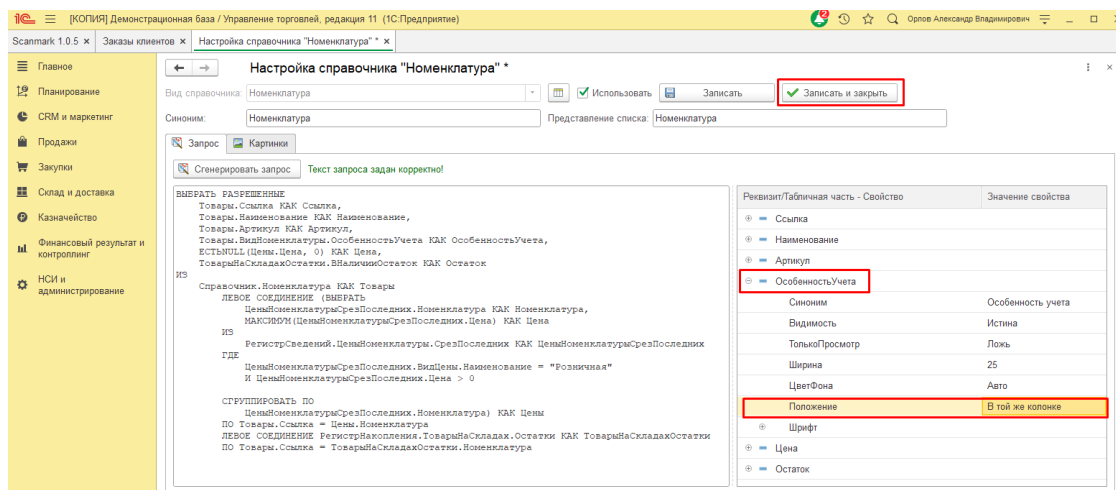
ВЫБРАТЬ РАЗРЕШЕННЫЕ
Товары.Ссылка КАК Ссылка,
Товары.Наименование КАК Наименование,
Товары.Артикул КАК Артикул,
Товары.ВидНоменклатуры.ОсобенностьУчета КАК ОсобенностьУчета,
ЕСТЬNULL(Цены.Цена, 0) КАК Цена,
ТоварыНаСкладахОстатки.ВНаличииОстаток КАК Остаток
ИЗ
Справочник.Номенклатура КАК Товары
ЛЕВОЕ СОЕДИНЕНИЕ (ВЫБРАТЬ
    ЦеныНоменклатурыСрезПоследних.Номенклатура КАК Номенклатура,
    МАКСИМУМ(ЦеныНоменклатурыСрезПоследних.Цена) КАК Цена
ИЗ
    РегистрСведений.ЦеныНоменклатуры.СрезПоследних КАК ЦеныНоменклатурыСрезПоследних
ГДЕ
    ЦеныНоменклатурыСрезПоследних.ВидЦены.Наименование = "Розничная"
И ЦеныНоменклатурыСрезПоследних.Цена > 0

СГРУППИРОВАТЬ ПО
    ЦеныНоменклатурыСрезПоследних.Номенклатура) КАК Цены
ПО Товары.Ссылка = Цены.Номенклатура
ЛЕВОЕ СОЕДИНЕНИЕ РегистрНакопления.ТоварыНаСкладах.Остатки КАК ТоварыНаСкладахОстатки
ПО Товары.Ссылка = ТоварыНаСкладахОстатки.Номенклатура
    
```

Изменим свойство нового реквизита и сохраним настройку:

- «ОсобенностьУчета»: Положение = В той же колонке



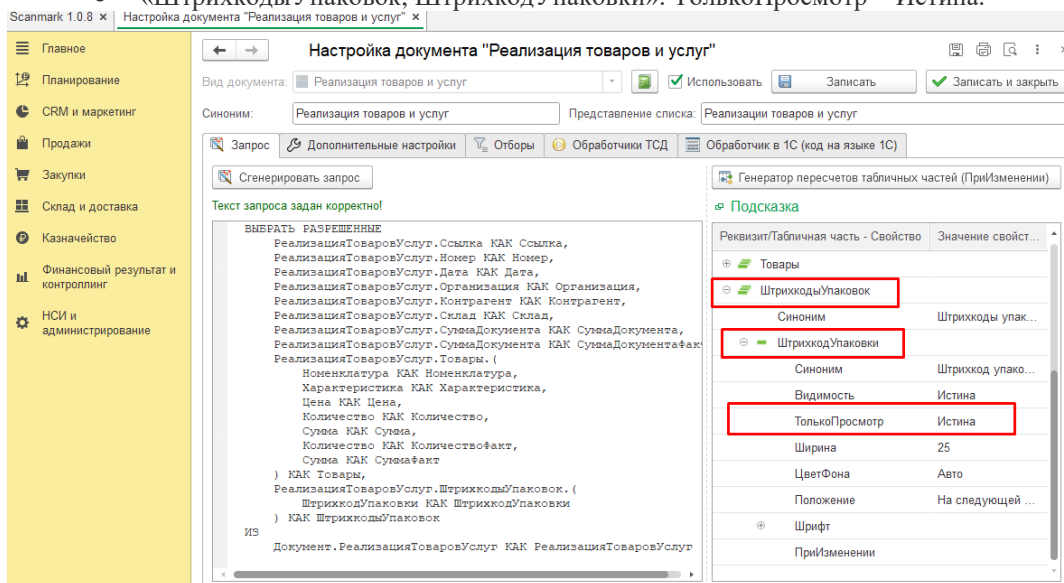


В настройке документа «Реализация товаров и услуг» на закладке «Запрос» в тексте запроса добавим табличную часть «ШтрихкодыУпаковок»:

ВЫБРАТЬ РАЗРЕШЕННЫЕ  
 РеализацияТоваровУслуг.Ссылка КАК Ссылка,  
 РеализацияТоваровУслуг.Номер КАК Номер,  
 РеализацияТоваровУслуг.Дата КАК Дата,  
 РеализацияТоваровУслуг.Организация КАК Организация,  
 РеализацияТоваровУслуг.Контрагент КАК Контрагент,  
 РеализацияТоваровУслуг.Склад КАК Склад,  
 РеализацияТоваровУслуг.СуммаДокумента КАК СуммаДокумента,  
 РеализацияТоваровУслуг.СуммаДокумента КАК СуммаДокументаФакт,  
 РеализацияТоваровУслуг.Товары. (  
   Номенклатура КАК Номенклатура,  
   Характеристика КАК Характеристика,  
   Цена КАК Цена,  
   Количество КАК Количество,  
   Сумма КАК Сумма,  
   Количество КАК КоличествоФакт,  
   Сумма КАК СуммаФакт  
 ) КАК Товары,  
 РеализацияТоваровУслуг.ШтрихкодыУпаковок. (  
   ШтрихкодУпаковки КАК ШтрихкодУпаковки  
 ) КАК ШтрихкодыУпаковок  
 ИЗ  
 Документ.РеализацияТоваровУслуг КАК РеализацияТоваровУслуг

Изменим свойство реквизита в новой табличной части:

- «ШтрихкодыУпаковок, ШтрихкодУпаковки»: ТолькоПросмотр = Истина.



Откроем закладку «Обработчики ТСД» и внесем изменения в обработчик события «ВнешнееСобытие», которые для подлежащих учету в системе «Честный знак» товаров разрешат работать только с типом штрихкода DataMatrix GS1:



```

Если Событие = "Barcode" Тогда
    Если Событие = "KeyBoard" Тогда
        //ваш код обработки события нажатия клавиши
    ИначеЕсли Событие = "RFID" Тогда
        //ваш код обработки события считывания радиометок
    ИначеЕсли Событие = "Barcode" Тогда

        ШтрихкодEAN13          = Данные;
        ШтрихкодДМ              = "";
        РазрешеноОбрабатывать   = Истина;
        ДанныеТовара            = Неопределено;
        ОсобенностьУчета        = "";

        Если СтрДлина(Данные) > 32 Тогда
            Если КодСимвола(Данные, 32) = 29 Тогда
                ШтрихкодДМ      = "(" + Сред(Данные, 1, 2) + ")" + Сред(Данные, 3, 14) + "(" + Сред(Данные, 17, 2) + ")" +
                Сред(Данные, 19, 13);
                ШтрихкодEAN13    = Сред(Данные, 4, 13);

                Строки = ШтрихкодыУпаковок.НайтиСтроки(Новый Структура("ШтрихкодУпаковки", ШтрихкодДМ));
                Если Строки.Количество() > 0 Тогда
                    РазрешеноОбрабатывать = Ложь;
                    Сообщить("Такой код маркировки уже добавлен!");
                КонецЕсли;
            КонецЕсли;
        КонецЕсли;

        Если РазрешеноОбрабатывать Тогда
            ДанныеШтрихкода = ДанныеШтрихкода(ШтрихкодEAN13);

            Если ДанныеШтрихкода.Количество() > 0 Тогда
                ДанныеТовара = ДанныеШтрихкода[0];
                оТовар        = ОбъектСканМарк(ДанныеТовара.Номенклатура);
                Если ЗначениеЗаполнено(оТовар) Тогда
                    ОсобенностьУчета = оТовар.ОсобенностьУчета;
                КонецЕсли;

                Если ЗначениеЗаполнено(ОсобенностьУчета) Тогда
                    Если ПустаяСтрока(ШтрихкодДМ) Тогда
                        Сообщить("Тип товара "" + ОсобенностьУчета + "" следует подбирать только через штрихкод
                        DataMatrix GS1!");
                        РазрешеноОбрабатывать = Ложь;
                    КонецЕсли;
                КонецЕсли;
            Иначе
                Сообщить("Ничего не найдено: " + Символы.ПС + Данные);
                РазрешеноОбрабатывать = Ложь;
            КонецЕсли;
        КонецЕсли;

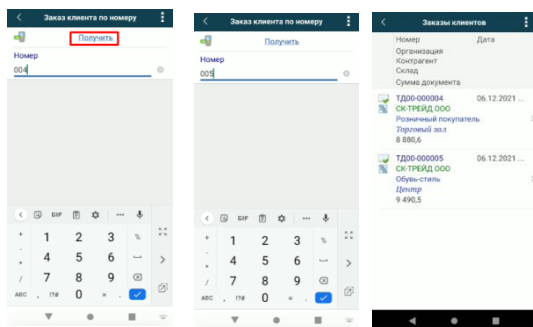
        Если РазрешеноОбрабатывать Тогда
            Строки =
                Структура("Номенклатура, Характеристика", ДанныеТовара.Номенклатура, ДанныеТовара.Характеристика);
            Если Строки.Количество() > 0 Тогда
                Стр = Строки[0];
                Если Стр.КоличествоФакт < Стр.Количество Тогда
                    Стр.КоличествоФакт = Стр.КоличествоФакт + 1;
                    Стр.СуммаФакт      = Окр(Стр.Цена * Стр.КоличествоФакт, 2);
                    СуммаДокументаФакт = Товары.Итог("СуммаФакт");
                    Если Не ПустаяСтрока(ШтрихкодДМ) и ЗначениеЗаполнено(ОсобенностьУчета) Тогда
                        СтрШК      = ШтрихкодыУпаковок.Добавить();
                        СтрШК.ШтрихкодУпаковки = ШтрихкодДМ;
                    КонецЕсли;
                Иначе
                    СтрТовар = СокрЛП(ДанныеТовара.Номенклатура);
                    Если ЗначениеЗаполнено(ДанныеТовара.Характеристика) Тогда
                        СтрТовар = СтрТовар + ", " + СокрЛП(ДанныеТовара.Характеристика);
                    КонецЕсли;
                    Сообщить("Количество товара "" + СтрТовар + "" не должно превышать количества в заказе!");
                КонецЕсли;

            Иначе
                СтрТовар = СокрЛП(ДанныеТовара.Номенклатура);
                Если ЗначениеЗаполнено(ДанныеТовара.Характеристика) Тогда
                    СтрТовар = СтрТовар + ", " + СокрЛП(ДанныеТовара.Характеристика);
                КонецЕсли;
                Сообщить("Товар по штрихкоду " + ШтрихкодEAN13 + " "" + СтрТовар + "" отсутствует в заказе и не
                может быть обработан!");

            КонецЕсли;
        КонецЕсли;
    КонецЕсли;

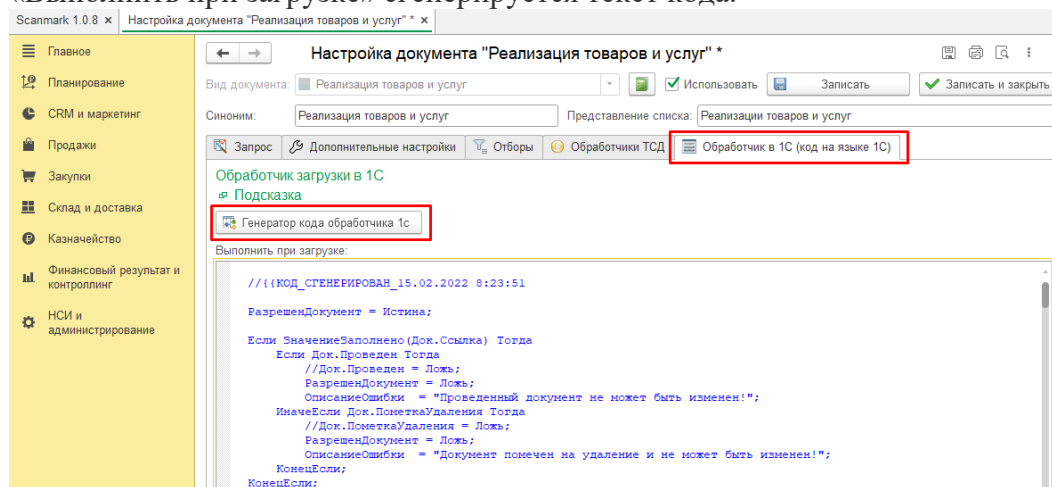
```

Сохраним настройку. На терминале сбора данных откроем «Импорт объектов, Заказ клиента по номеру» и поочередно получим заказа клиента:



Создадим на основании полученных заказов документы «Реализация товаров и услуг» и попробуем сканировать штрихкоды (EAN13 и DataMatrix GS1), и посмотреть корректность обработки штрихкодов.

Теперь осталось в настройке документа «Реализация товаров и услуг» настроить обработку получения данных по документу с ТСД. Откроем настройку, перейдем на закладку «Обработчик в 1С (код на языке 1С)», нажмем кнопку «Генератор кода обработчика 1с». В поле «Выполнить при загрузке» сгенерируется текст кода.



Внесем изменения в текст кода и сохраним настройку.

```
РазрешенДокумент = Истина;
Если ЗначениеЗаполнено (Док.Ссылка) Тогда
    Если Док.Проведен Тогда
        //Док.Проведен = Ложь;
        РазрешенДокумент = Ложь;
        ОписаниеОшибки = "Проведенный документ не может быть изменен!";
    ИначеЕсли Док.ПометкаУдаления Тогда
        //Док.ПометкаУдаления = Ложь;
        РазрешенДокумент = Ложь;
        ОписаниеОшибки = "Документ помечен на удаление и не может быть изменен!";
    КонецЕсли;
КонецЕсли;

Если РазрешенДокумент и ЗначениеЗаполнено (Док.Ссылка) Тогда
    //Док.Номер =
    Если Не ЗначениеЗаполнено (Док.Дата) Тогда
        Док.Дата = ТекущаяДата ();
    КонецЕсли;

    Док.Товары.Очистить ();
    Док.СкидкиНаценки.Очистить ();
    Док.ВидыЗапасов.Очистить ();
    Док.ДополнительныеРеквизиты.Очистить ();
    Док.РасшифровкаПлатежа.Очистить ();
    Док.Серии.Очистить ();
    Док.ЭтапыГрафикаОплаты.Очистить ();
    Док.ШтрихкодыУпаковок.Очистить ();
КонецЕсли;

Если РазрешенДокумент Тогда
    Если ЗначениеЗаполнено (ДокументОснование) Тогда
        Док.Заполнить (ДокументОснование);
    КонецЕсли;
    Док.СкидкиРассчитаны = Истина;

    НомСтр = Док.Товары.Количество ();
    Пока НомСтр > 0 Цикл
        НомСтр = НомСтр - 1;
        СтрДок = Док.Товары[НомСтр];
```

```

Если ЗначениеЗаполнено(СтрДок.Характеристика) Тогда
    ТЗ Строки = Товары.Скопировать(Новый Структура("Номенклатура, Характеристика",
СтрДок.Номенклатура, СтрДок.Характеристика));
    Иначе
        ТЗ_Строки = Товары.Скопировать(Новый Структура("Номенклатура", СтрДок.Номенклатура));
    КонецЕсли;

Количество = СтрДок.Количество;
СтрДок.Количество = ТЗ Строки.Итог("КоличествоФакт");

Если СтрДок.Количество = Количество Тогда
    ИначеЕсли СтрДок.Количество > 0 Тогда
        СтрДок.КоличествоУпаковок = СтрДок.Количество;
        СтруктураПересчетаСуммы =
ОбработкаТабличнойЧастиКлиентСервер.ПараметрыПересчетаСуммыНДСВСтрокеТЧ(Док);

        СтруктураДействий = Новый Структура;
        //СтруктураДействий.Вставить("ПроверитьХарактеристикуПоВладельцу", СтрДок.Характеристика);
        СтруктураДействий.Вставить("ПроверитьЗаполнитьУпаковкуПоВладельцу", СтрДок.Упаковка);
        СтруктураДействий.Вставить("ПересчитатьКоличествоЕдиниц");
        СтруктураДействий.Вставить("ЗаполнитьСтавкуНДС", Новый Структура("НалогообложениеНДС, Дата",
Док.НалогообложениеНДС, Док.Дата));
        СтруктураДействий.Вставить("ПересчитатьСуммуНДС", СтруктураПересчетаСуммы);
        СтруктураДействий.Вставить("ПересчитатьСуммуСНДС", СтруктураПересчетаСуммы);
        СтруктураДействий.Вставить("ПересчитатьСумму");
        СтруктураДействий.Вставить("ПересчитатьСуммуСУчетомРучнойСкидки", Новый Структура("Очищать",
Истина));
        СтруктураДействий.Вставить("ПересчитатьСуммуСУчетомАвтоматическойСкидки", Новый
Структура("Очищать", Истина));
        СтруктураДействий.Вставить("ПроверитьЗаполнитьНазначение");

        ДополнительнаяСтруктура =
ОбработкаТабличнойЧастиКлиентСервер.ПолучитьСтруктуруЗаполненияУсловийПродажВСтрокеТЧ(Док);
        ОбщегоНазначенияКлиентСервер.ДополнитьСтруктуру(СтруктураДействий, ДополнительнаяСтруктура,
Истина);

        ИспользоватьСоглашенияСКлиентами = Константы.ИспользоватьСоглашенияСКлиентами.Получить();
        Если ИспользоватьСоглашенияСКлиентами И ЗначениеЗаполнено(Док.Соглашение) Тогда
            СтруктураДействий.Вставить("ЗаполнитьУсловияПродаж",
ОбработкаТабличнойЧастиКлиентСервер.ПолучитьСтруктуруЗаполненияУсловийПродажВСтрокеТЧ(Док));
        Иначе
            СтруктураДействий.Вставить("ЗаполнитьЦенуПродажи",
ОбработкаТабличнойЧастиКлиентСервер.ПолучитьСтруктуруЗаполненияЦеныВСтрокеТЧ(Док));
        КонецЕсли;

        КэшированныеЗначения = Неопределено;
        ТекущаяСтрокаСтруктура = ПолучитьТекущуюСтрокуСтруктурой(СтрДок, СтруктураДействий);

        ОбработкаТабличнойЧастиСервер.ОбработатьСтрокуТЧ(ТекущаяСтрокаСтруктура, СтруктураДействий,
КэшированныеЗначения);
        ЗаполнитьЗначенияСвойств(СтрДок, ТекущаяСтрокаСтруктура);

        Иначе
            Док.Товары.Удалить(СтрДок);
        КонецЕсли;

КонецЦикла;

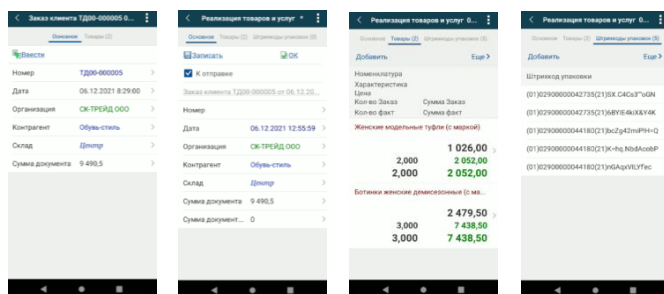
Для Каждого СтрТЗ Из ШтрихкодыУпаковок Цикл
    СтрДок = Док.ШтрихкодыУпаковок.Добавить();
    ЗаполнитьЗначенияСвойств(СтрДок, СтрТЗ);
    //СтрДок.ШтрихкодУпаковки = "";
КонецЦикла;

//Док.Записать();
Док.Записать(РежимЗаписиДокумента.Проведение);

КонецЕсли;

```

На ТСД на основании заказа клиента создаем документ «Реализация товаров и услуг», по кодам маркировки «Честный знак» подбираем товар, сохраняем и отправляем документ в 1с:



Получаем результат в 1С:

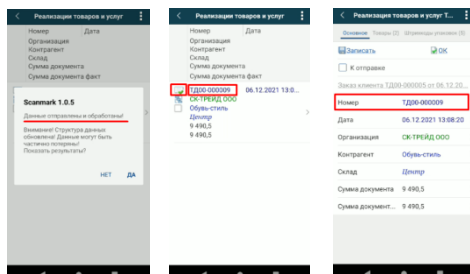
Scitmark 1.0.5 x Заказы клиентов x Документы реализации x Настройка документа "Реализация товаров и услуг" x Реализация товаров и услуг TDD0-000009 от 06.12.2021 13:08:20

Основное Товары (2) Дополнительно

Добавить

N	Номенклатура	Характеристика	Серия	Количество	Ед. изм.	Вид цены	Цена	% авт.	Сумма авт.	% руч.	Сумма руч.	Сумма	Ставка НДС	НДС
1	Женские модаль...	<характеристика>	<серия на указыв...	2,000	пар	Дистрибьютор...	1 026.00					2 052.00	20%	
2	Ботинки женские...	<характеристика>	<серия на указыв...	3,000	пар	Дистрибьютор...	2 479.50					7 438.50	20%	

И результат на ТСД:

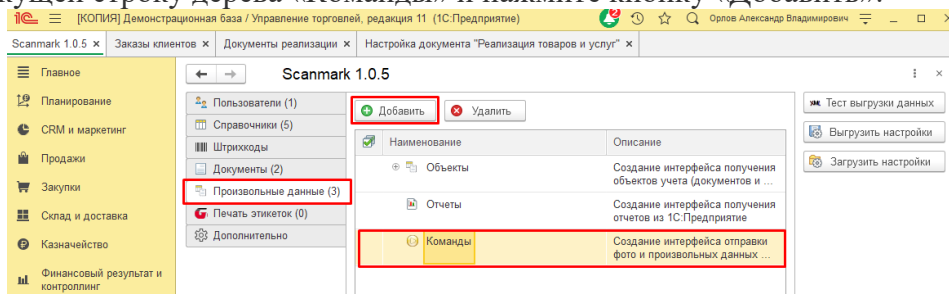


Задача успешно выполнена.

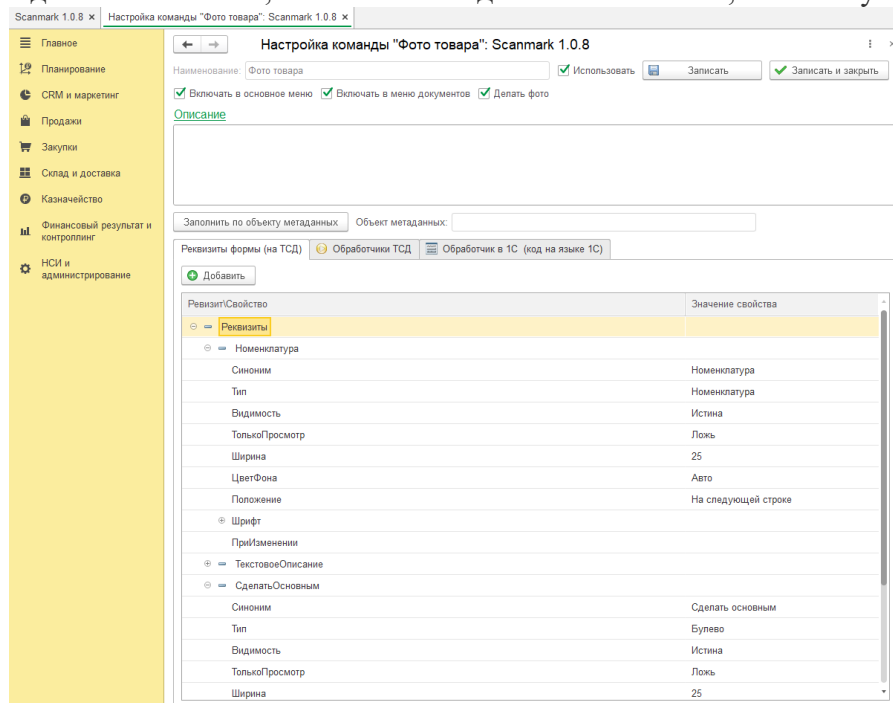
## Задача 9: На ТСД создать команду для фотографирования и присоединения фотографии к номенклатуре в «1С:Предприятие»

Реализация этой настройки позволит делать фото на ТСД, отправлять файл фотографии в «1С:Предприятие» и связывать этой файл с номенклатурой.

В обработке «Scanmark» перейдите на страницу «Произвольные данные», сделайте текущей строку дерева «Команды» и нажмите кнопку «Добавить»:

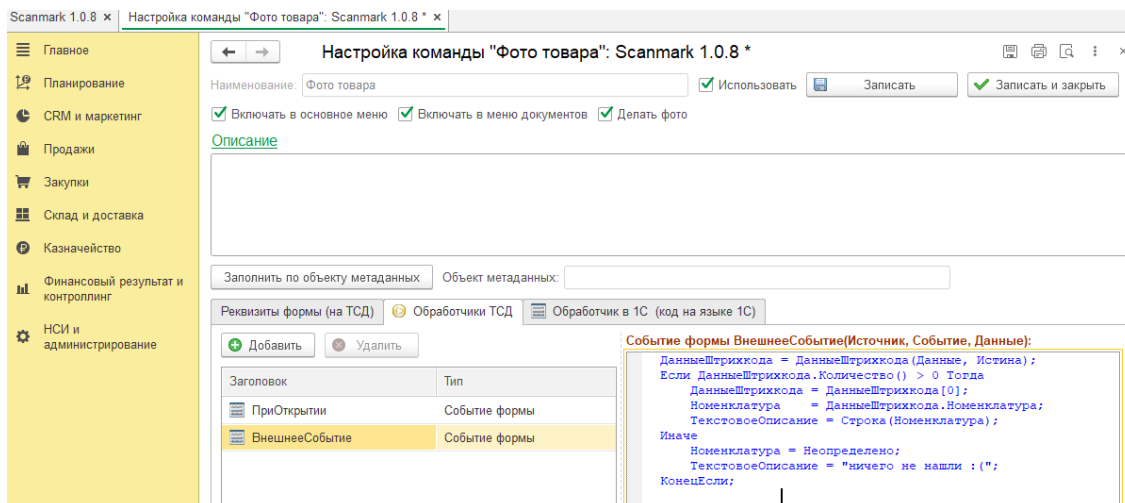


В форме команды Наименование = «Фото товара», устанавливаем флажок «Делать фото», внизу формы добавляем реквизиты формы (на ТСД): Номенклатура, тип = «Номенклатура», СделайОсновным, Синоним = «Сделать основным», тип = «Булево»:



На закладке «Обработчики ТСД» для события «ВнешнееСобытие» добавляем текст кода:

```
Если Событие = "KeyBoard" Тогда
    //ваш код обработки события нажатия клавиши
ИначеЕсли Событие = "RFID" Тогда
    //ваш код обработки события считывания радиометок
ИначеЕсли Событие = "Barcode" Тогда
    ДанныеШтрихкода = ДанныеШтрихкода(Данные, Истина);
    Если ДанныеШтрихкода.Количество() > 0 Тогда
        ДанныеШтрихкода = ДанныеШтрихкода[0];
        Номенклатура = ДанныеШтрихкода.Номенклатура;
        ТекстовоеОписание = Строка(Номенклатура);
    Иначе
        Номенклатура = Неопределено;
        ТекстовоеОписание = "ничего не нашли :(";
        //сообщить("Ничего не найдено: " + Символы.ПС + Данные);
    КонецЕсли;
КонецЕсли;
```



## На закладке «Обработчик в 1С (код на языке 1С)» добавляем текст кода:

```

Результат = "Команда не изменила данные!";
Если ЗначениеЗаполнено (Номенклатура) Тогда
    Если Фотографии.Количество() > 0 Тогда

        Для Каждого СтрФото Из Фотографии Цикл
            Картинка = СтрФото.Картинка;
            ДвоичныеДанные = Картинка.ПолучитьДвоичныеДанные();
            ХранилищеКартинки = Новый ХранилищеЗначения (Картинка);

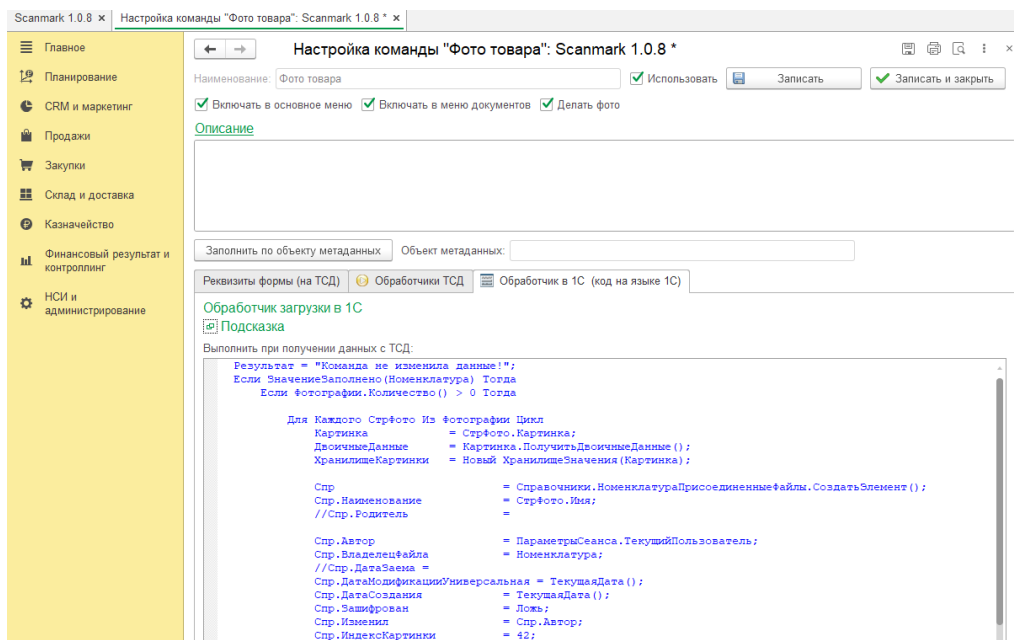
            Спр = Справочники.НоменклатураПрисоединенныеФайлы.СоздатьЭлемент();
            Спр.Наименование = СтрФото.Имя;
            //Спр.Родитель =

            Спр.Автор = ПараметрыСеанса.ТекущийПользователь;
            Спр.ВладелецФайла = Номенклатура;
            //Спр.ДатаЗаема =
            Спр.ДатаМодификацииУниверсальная = ТекущаяДата();
            Спр.ДатаСоздания = ТекущаяДата();
            Спр.Зашифрован = Ложь;
            Спр.Изменил = Спр.Автор;
            Спр.ИндексКартинки = 42;
            //Спр.Описание = ТекстовоеОписание;
            Спр.ПодписанЭП = Ложь;
            //Спр.ПутьКФайлу =
            Спр.Размер = ДвоичныеДанные.Размер();
            Спр.Расширение = СтрФото.Расширение;
            //Спр.Редактирует =
            Спр.СтатусИзвлеченияТекста = Перечисления.СтатусыИзвлеченияТекстаФайлов.НеИзвлечен;
            //Спр.ТекстХранилище =
            Спр.ТипХраненияФайла = Перечисления.ТипыХраненияФайлов.ВИнформационнойБазе;
            //Спр.Том =
            //Спр.ФайлХранилище =
            Спр.ХранитьВерсии = Ложь;
            Спр.Записать();

            МенеджерЗаписи = РегистрыСведений.ДвоичныеДанныеФайлов.СоздатьМенеджерЗаписи();
            МенеджерЗаписи.Файл = Спр.Ссылка;
            МенеджерЗаписи.ДвоичныеДанныеФайла = ХранилищеКартинки;
            МенеджерЗаписи.Записать (Истина);

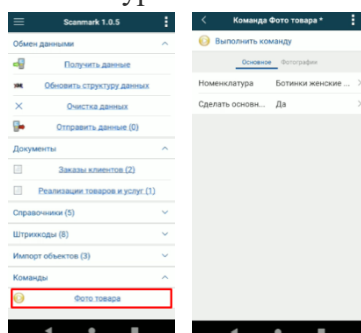
            Если Не ЗначениеЗаполнено (Номенклатура.ФайлКартинки) или СделатьОсновным Тогда
                СпрНом = Номенклатура.ПолучитьОбъект();
                СпрНом.ФайлКартинки = Спр.Ссылка;
                СпрНом.Записать();
            КонецЕсли;
        КонецЦикла;
        Результат = "Фото (" + Фотографии.Количество() + ") прикреплены к " + Номенклатура;
    КонецЕсли;
КонецЕсли;

```

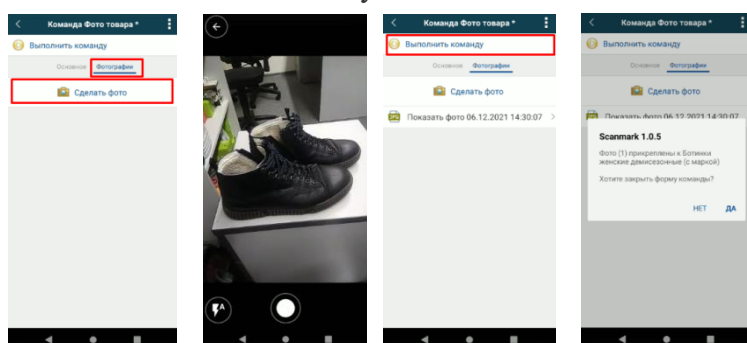


Сохраним настройку.

На ТСД обновим структуру данных. Откроем команду «Фото товара». Зададим значения: номенклатура = Ботинки женские демисезонные (с маркой), Сделать основным = Да.



Перейдем на закладку «Фотографии», нажмем «Сделать фото», сделаем снимок, далее нажмем «Выполнить команду»:



Проверим результат в «1С:Предприятие». Найдем товар «Ботинки женские демисезонные (с маркой)» и убедимся, что фото загружено и сделано основным у элемента номенклатуры.

Scalpayk 1.0.5 x Заказы клиентов x Документы реализации x Настройка документ... x Заказ к... TJ00-000005 x Настройка команды... x Ботинки женские демисезонные x

Ботинки женские демисезонные (с маркой) (Номенклатура)


Записать и закрыть | Печать | Отчеты | Перейти | Еще ?

Карточка | Реализации

Рабочее наименование: **Ботинки женские демисезонные (с маркой)**  
 Наименование для печати: **Ботинки женские демисезонные (с маркой)**  
 Артикул: <не указан>  
 Код: 90-00000290

Номенклатура с аналогичными свойствами  
 Штрихкоды (2)

Описание



Основы параметров учета

Вид номенклатуры:	Обувь с нарисовкой
Тип номенклатуры:	Товар
Характеристики:	Не используются
Серия:	Не используется
Оформление продаж:	Реализация товаров и услуг
Группа списка:	<не указана>
Группа доступа:	Обувь
Качество:	Новый
Товары другого качества (0)	

Единицы измерения и условия хранения

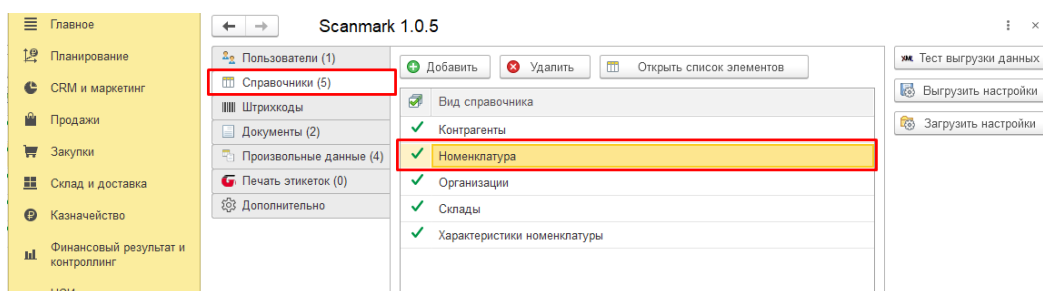
Упаковки:	Упаковки обуви
Единица хранения:	Список (3)
Единица для отчетов:	пар
Вес:	<не измеряется>
Объем:	<не измеряется>

Текстовое описание: <не указано>  
 Фото (6)

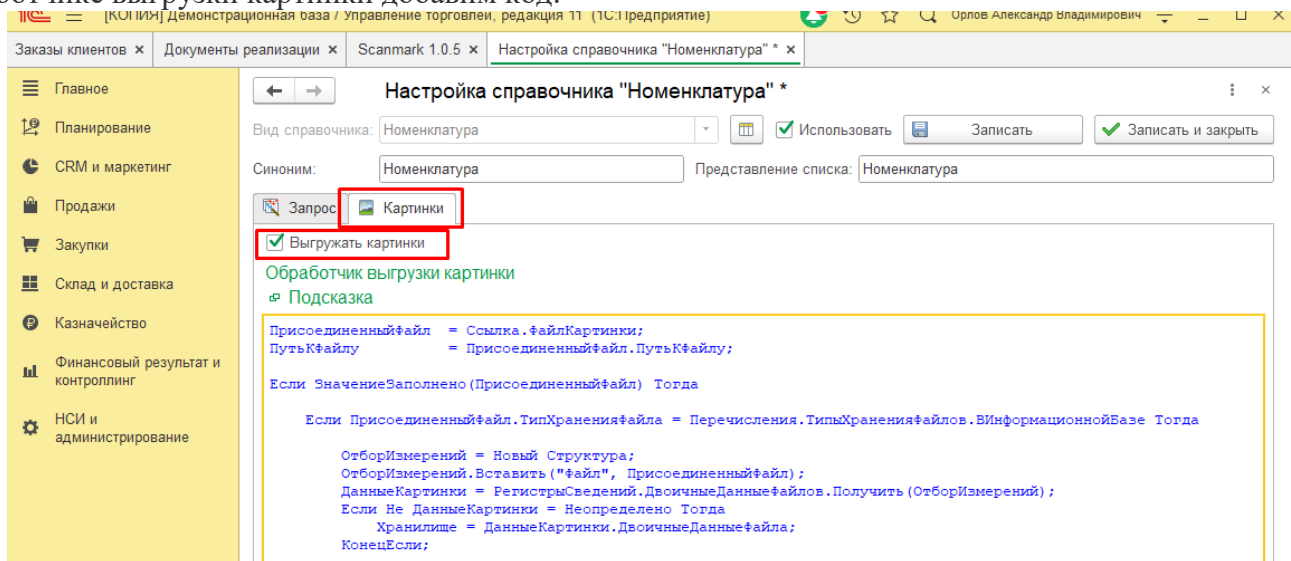


## Задача 10: Получить фото номенклатуры из «1С:Предприятие» на ТСД

В «1С:Предприятие» откроем настройку справочника «Номенклатура»:



Перейдем на закладку «Картинки», установим флажок «Выгружать картинки». В обработчике выгрузки картинки добавим код:



```
ПрисоединенныйФайл = Ссылка.ФайлКартинки;
ПутьКФайлу = ПрисоединенныйФайл.ПутьКФайлу;
```

```
Если ЗначениеЗаполнено (ПрисоединенныйФайл) Тогда
```

```
    Если ПрисоединенныйФайл.ТипХраненияФайла = Перечисления.ТипыХраненияФайлов.ВинформационнойБазе Тогда
```

```
        ОтборИзмерений = Новый Структура;
        ОтборИзмерений.Вставить ("Файл", ПрисоединенныйФайл);
        ДанныеКартинки = РегистрыСведений.ДвоичныеДанныеФайлов.Получить (ОтборИзмерений);
        Если Не ДанныеКартинки = Неопределено Тогда
            Хранилище = ДанныеКартинки.ДвоичныеДанныеФайла;
        КонецЕсли;
```

```
    ИначеЕсли ПрисоединенныйФайл.ТипХраненияФайла = Перечисления.ТипыХраненияФайлов.ВТомНаДиске Тогда
        ПутьКФайлу = ПрисоединенныйФайл.ПутьКФайлу;
        Если Не ПустаяСтрока (ПутьКФайлу) Тогда
```

```
            ПолныйПутьWindows = ПрисоединенныйФайл.Том.ПолныйПутьWindows;
            ПолныйПуть = ПолныйПутьWindows + ПутьКФайлу;
            ФайлФото = Новый Файл (ПолныйПуть);
            Если ФайлФото.Существует () Тогда
                ДвоичныеДанные = Новый ДвоичныеДанные (ПолныйПуть);
                Хранилище = Новый ХранилищеЗначения (ДвоичныеДанные);
            КонецЕсли;
        КонецЕсли;
```

```
    КонецЕсли;
```

```
КонецЕсли;
```

```
КонецЕсли;
```

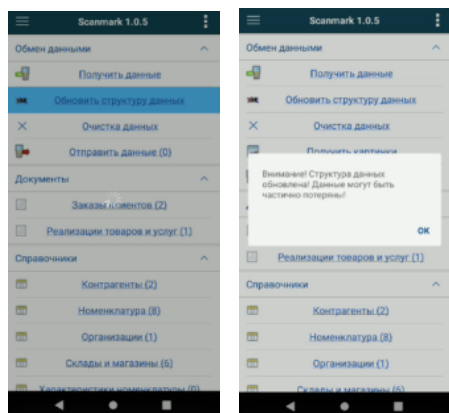
В обработчике выгрузки картинки следует указать код, который будет выполнен при получении данных при запросе картинок. В качестве входных параметров инициализируются переменные:

<Ссылка> - тип "СправочникСсылка.XXX", где XXX - вид текущего справочника).

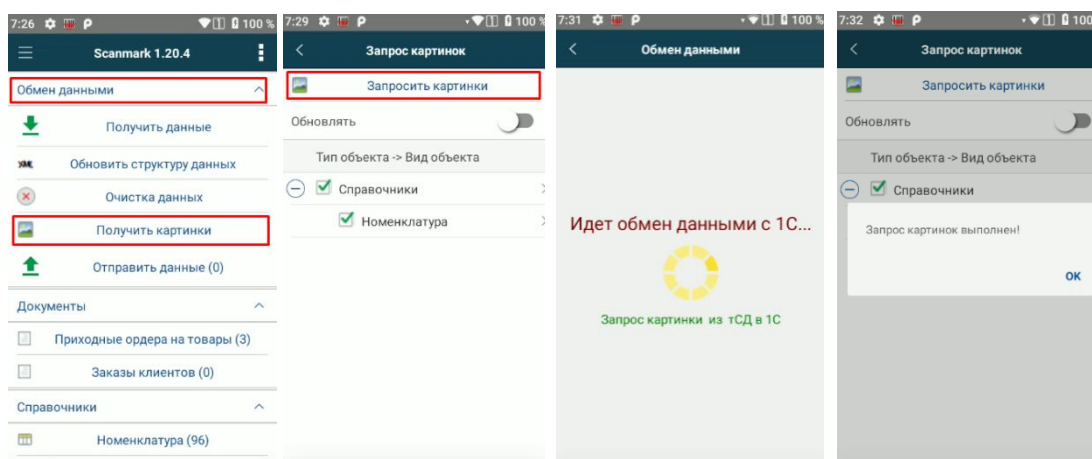
<Хранилище> - Неопределено.

В переменную Хранилище нужно поместить данные картинки, тип значения "ХранилищеЗначения".

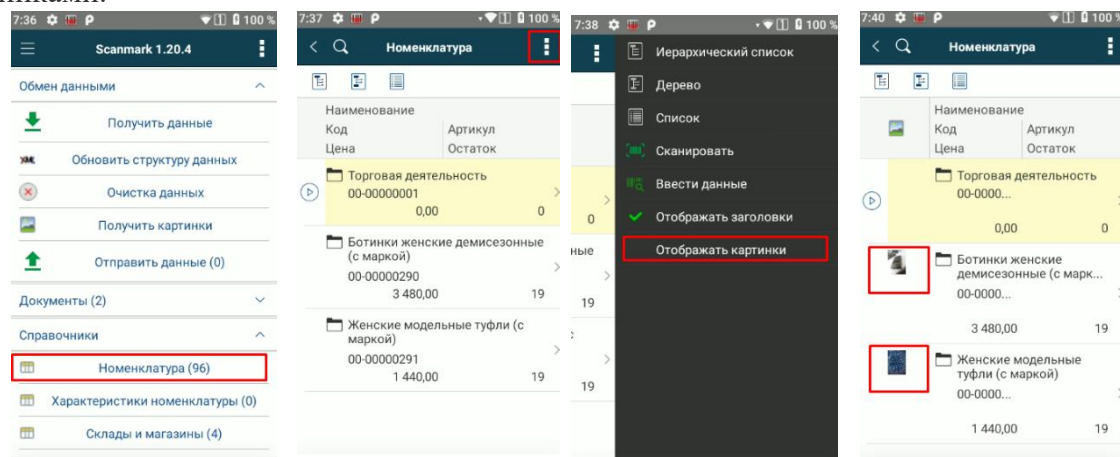
Запишем настройку. На ТСД выполним обновление структуры данных.



На главном экране в основном меню «Обмен данными» появится команда «Получить картинки», с помощью которой можно получить/обновить картинки по выбранным видам справочников:



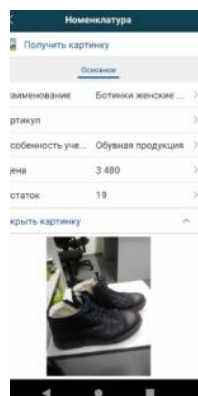
Откроем список номенклатуры. Чтобы отобразить картинки в списке нажмите команду «Отобразить картинки» в меню формы списка. В левой части списка появится колонка с картинками.



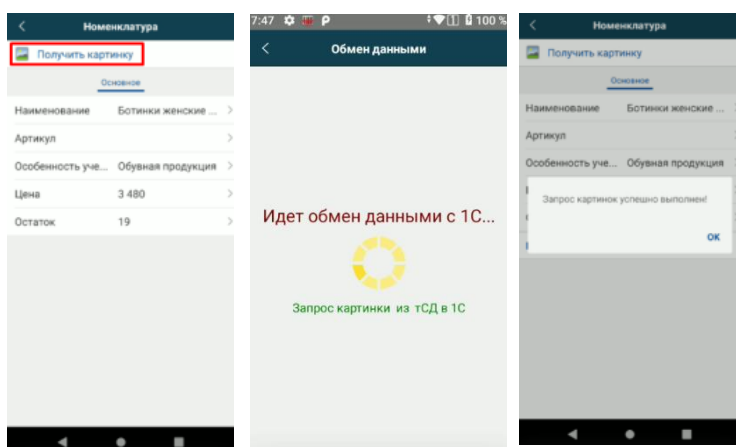
Нажимая на иконки картинок можно открыть картинку в отдельной форме:



В форме элемента справочника при наличии картинка, эта картинка будет расположена в нижней части формы на странице «Основное»:



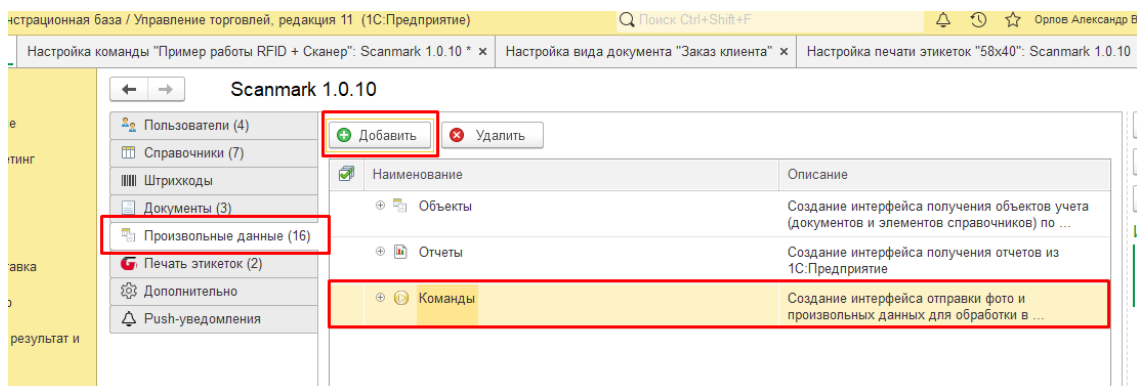
В форме элемента справочника на ТСД картинку можно получить/обновить из «1С:Предприятие». Нажмем кнопку «Получить картинку».



Как в форме списка элементов справочника открыть картинку в отдельной форме можно кликнув по картинке.

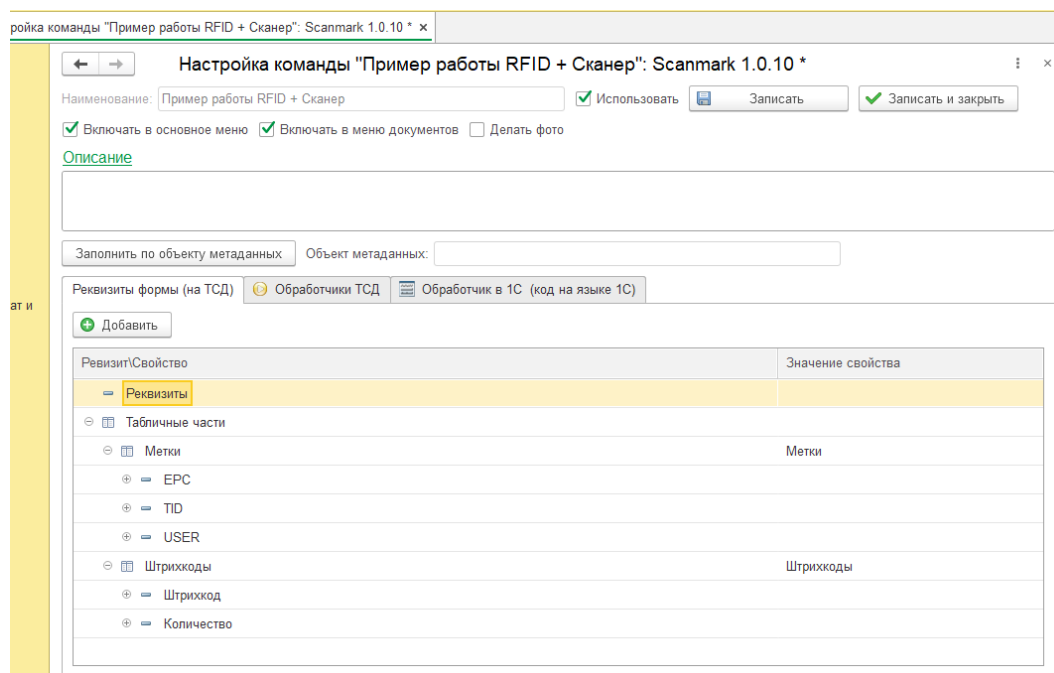
## Задача 11: Создать пример команды для работы со считывателем RFID

В форме настройки в «1С:Предприятие» на закладке «Произвольные данные» создадим новую команду.



Зададим наименование команды «Пример работы RFID + Сканер», добавим табличную часть «Метки» с реквизитами «EPC», «TID», «USER»:

Добавим табличную часть «Штрихкоды» с реквизитами «Штрихкод» и «Количество». Тип реквизита «Количество» установим «Число».



На закладке «Обработчики ТСД» для события «ВнешнееСобытие» добавим код:

```
Если Событие = "Keyboard" Тогда
    //ваш код обработки события нажатия клавиши
ИначеЕсли Событие = "RFID" Тогда
    ТекущаяСтраница ("Метки");
    Добавлено = 0;
    Обновлено = 0;
    Если ТипЗнч(Данные) = Тип("Массив") Тогда
        Для К = 0 По Данные.ВГраница() Цикл
            Строки = Метки.НайтиСтроки(Новый Структура("EPC,TID", Данные[К][0], Данные[К][1]));
            Если Строки.Количество() = 0 Тогда
                Стр = Метки.Добавить();
                Стр.EPC = Данные[К][0];
                Стр.TID = Данные[К][1];

                Стр.стрEPC = "EPC";
                Стр.стрTID = "TID";
```

```

Стр.стрUSER = "USER";

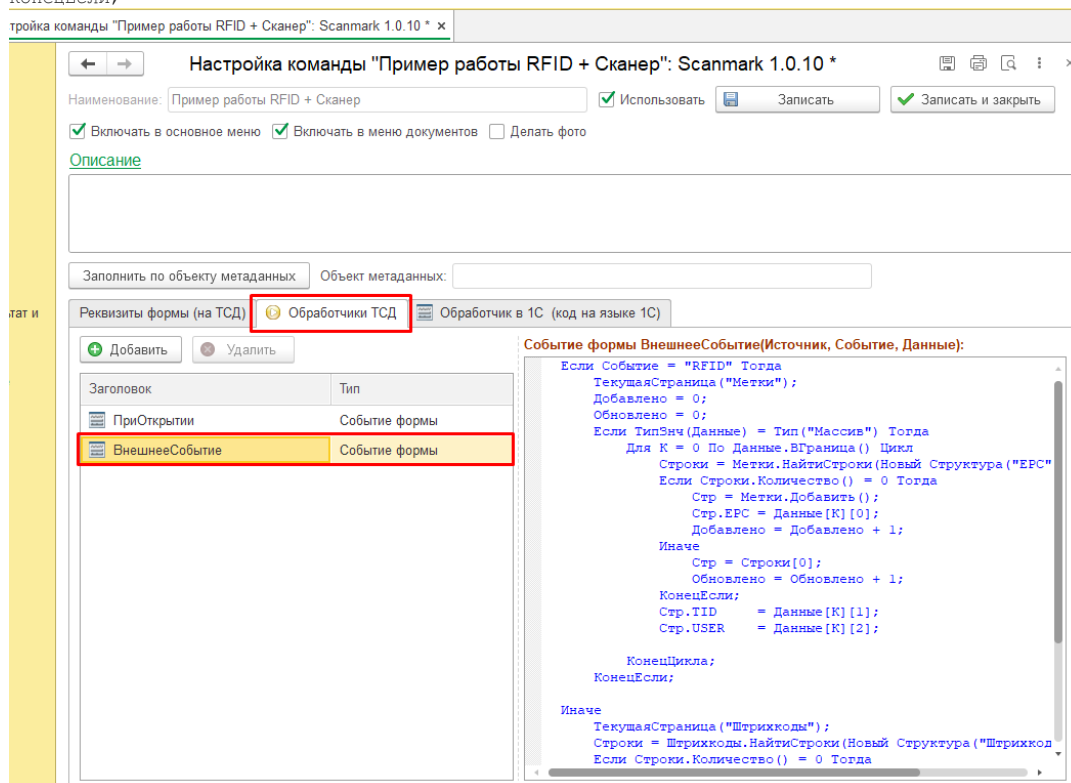
Добавлено = Добавлено + 1;
Иначе
Стр = Строки[0];
Обновлено = Обновлено + 1;
КонецЕсли;
Стр.USER = Данные[K][2];

КонецЦикла;
КонецЕсли;

//Сообщить ("Добавлено меток - " + Добавлено + Символы.ПС + "Обновлено меток - " + Обновлено);

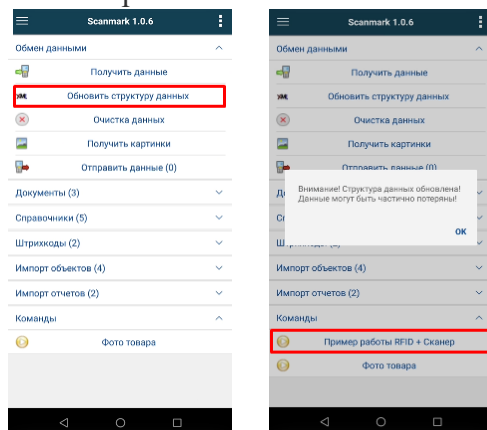
ИначеЕсли Событие = "Barcode" Тогда
ТекущаяСтраница ("Штрихкоды");
Строки = Штрихкоды.НайтиСтроки (Новый Структура ("Штрихкод", Данные));
Если Строки.Количество() = 0 Тогда
Стр = Штрихкоды.Добавить();
Стр.Штрихкод = Данные;
Иначе
Стр = Строки[0];
КонецЕсли;
Стр.Количество = Стр.Количество + 1;
КонецЕсли;

```

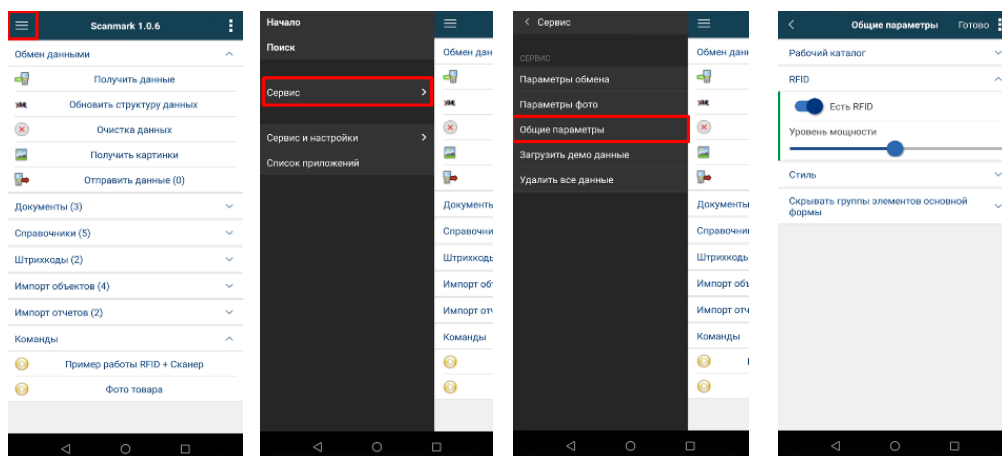


Следует обратить внимание, что при событии «RFID» тип данных уже не «Строка», а «Массив(п, 3)». Запишем команду.

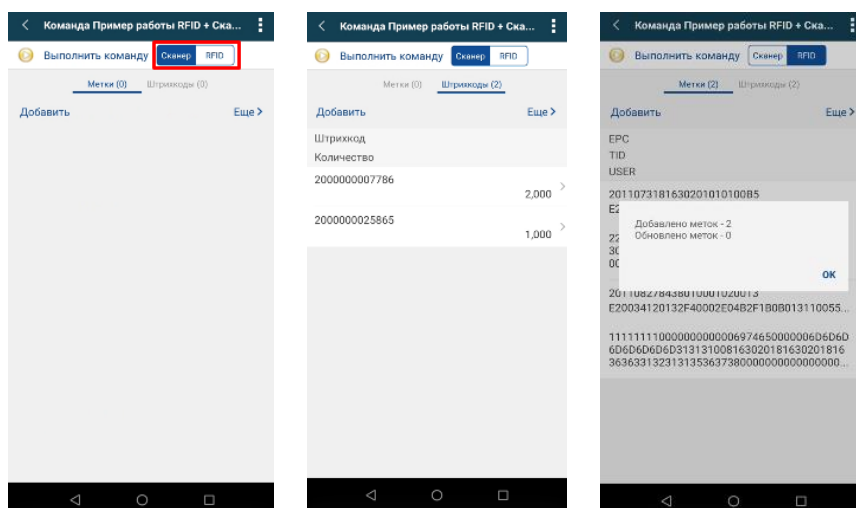
На ТСД выполним обновление структуры данных. Появится команда «Пример работы RFID + Сканер».



Далее откроем общие настройки, раскроем настройку «RFID», установим переключатель «Есть RFID», уровень мощности выставим примерно посередине.



Откроем команду «Пример работы RFID + Сканер» и, переключая тип устройства Сканер/RFID, поочередно сканировать сканером штрихкодов и считывателем RFID. В результате табличные части заполняются соответствующими данными:



Создание этикетки следует начать в редакторе [GoLabel](#). Вид примера этикетки размером 58x40:



^Q40,3  
^W58  
^H5  
^P1  
^S2  
^AD  
^C1  
^R0  
~Q+0  
^O0  
^D0  
^E10  
~R255  
^L  
Dy2-me-dd  
Th:m:s

```

V00,36,name1,j136
V01,36,name2,j136
V02,12,articule1,j112
V03,12,articule2,j112
V04,3,size1,j13
V05,13,color1,j113
V06,13,color2,j113
V07,16,brand1,j116
V08,16,brand2,j116
V09,16,product1,j116
V10,16,product2,j116
V11,16,manufacturer1,j116
V12,16,country1,j116
V13,13,markingcode1,j113
V14,13,markingcode2,j113
V15,13,markingcode3,j113
V16,13,barcode1,j113
Lo,0,64,463,65
Lo,308,65,309,318
Lo,0,127,307,128
Lo,0,187,307,188
AB,4,3,1,1,0,0E,^V00
AB,4,33,1,1,0,0E,^V01
AA,2,68,1,1,0,0E,АРТИКУЛ
AA,115,68,1,1,0,0E,РАЗМЕР
AA,189,68,1,1,0,0E,ЦВЕТ
AA,1,88,1,1,0,0E,^V02
AA,1,106,1,1,0,0E,^V03
AD,120,85,1,1,0,0E,^V04
AA,189,88,1,1,0,0E,^V05
AA,189,106,1,1,0,0E,^V06
AA,1,129,1,1,0,0E,МАРКА (БРЕНД)
AA,1,149,1,1,0,0E,^V07
AA,1,166,1,1,0,0E,^V08
AA,156,129,1,1,0,0E,ПРОДУКЦИЯ
AA,156,149,1,1,0,0E,^V09
AA,156,166,1,1,0,0E,^V10
AA,1,189,1,1,0,0E,ПРОИЗВОДИТЕЛЬ
AA,1,209,1,1,0,0E,^V11
AA,156,189,1,1,0,0E,СТРАНА
AA,156,209,1,1,0,0E,^V12
BE,38,238,2,5,55,0,0,^V16
AA,319,204,1,1,0,0E,^V13
AA,319,220,1,1,0,0E,^V14
AA,319,236,1,1,0,0E,^V15
XRB314,71,3,0,131
~lthis_is_gsl_data_matrix0123456789this_is_gsl_data_matrix0123456789this_is_gsl_data_matrix0123456789th
is_is_gsl_data_matrix0123456
Y323,259,RightSign
E

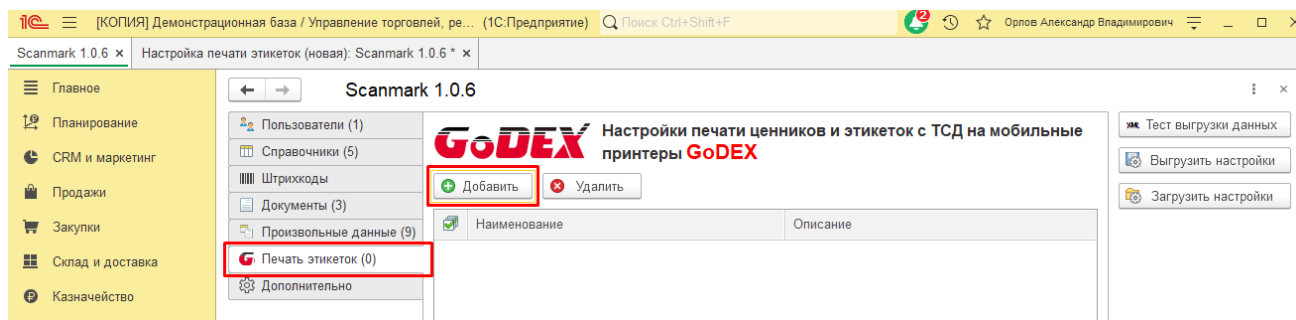
```

Кроме этого в этикетке есть ресурс (картинка) – логотип «Честный знак». Сохраните картинку в файл как RightSign.bmp (808x301):

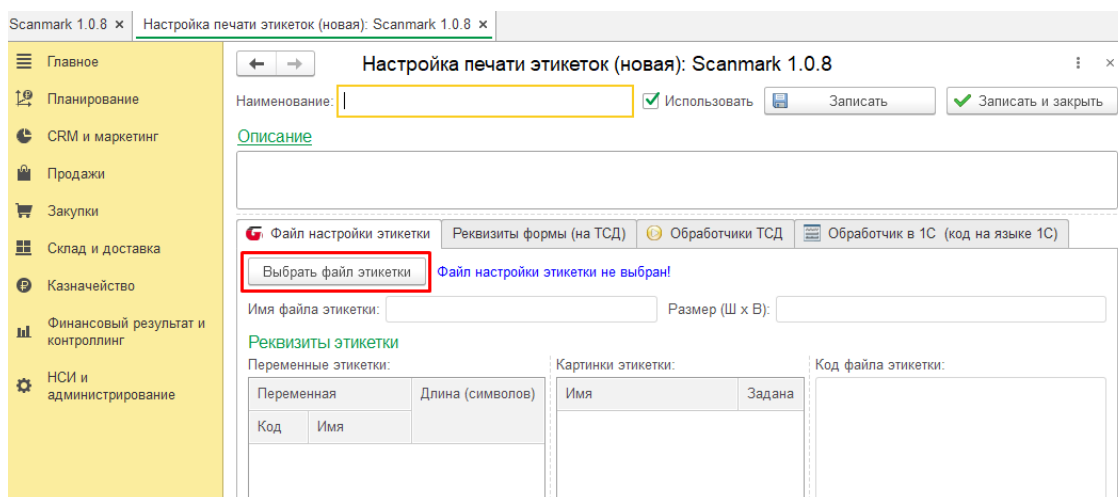


В форме настройки в «1С:Предприятие» на закладке «Печать этикеток» добавим настройку этикетки:

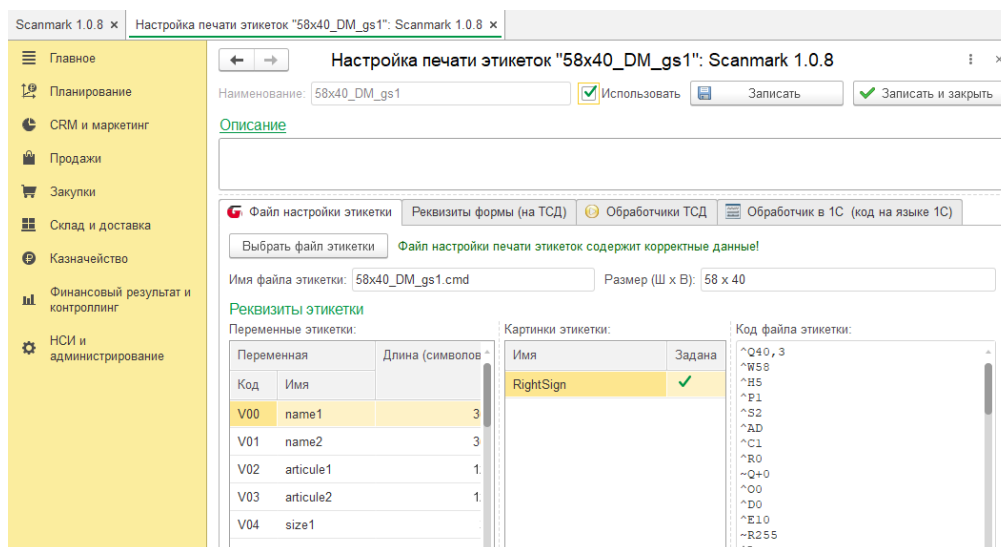




В форме настройки этикетки выберем файл 58x40\_DM\_gs1.cmd:



В результате будут загружен и разобран файл этикетки:



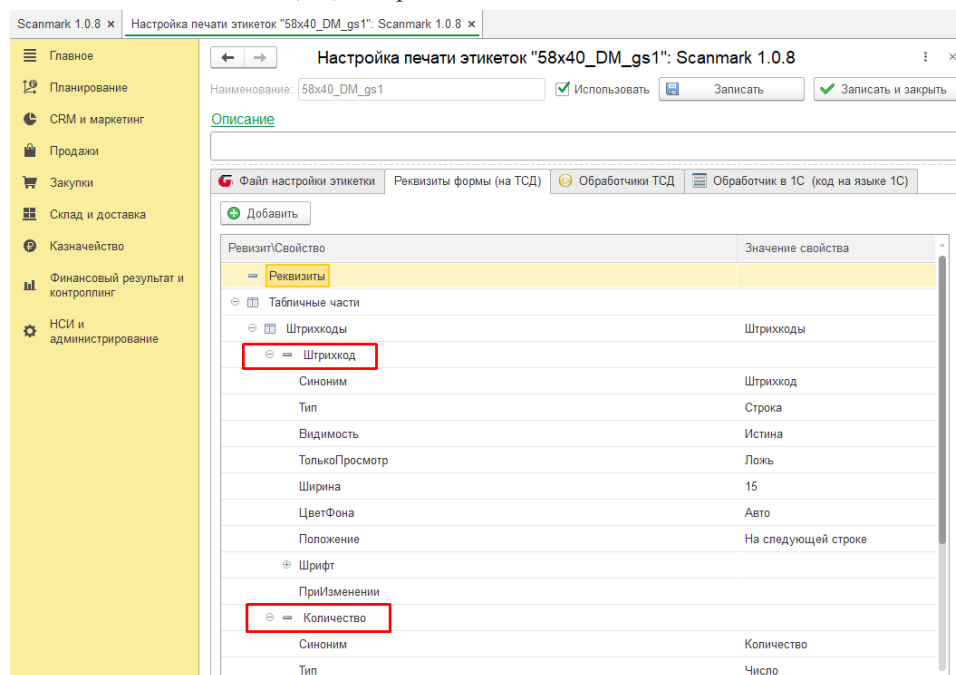
Файл ресурса RightSign.bmp был загружен в настройку автоматически, т.к. находился в одном каталоге с 58x40\_DM\_gs1.cmd.

**ВАЖНО!!!** В правой части в поле «Код файла этикетки» есть строка: "`~1this_is_gs1_data_matrix0123456789this_is_gs1_data_matrix0123456789this_is_gs1_data_matrix0123456789this_is_gs1_data_matrix0123456`". Эта строка является шаблоном для замены реальным кодом DataMatrix GS1 в приложении Scanmark. Используйте всегда этот шаблон при настройке печати маркировки «Честный знак».

На закладке «Штрихкоды (на ТСД)» добавим табличную часть «Штрихкод». В

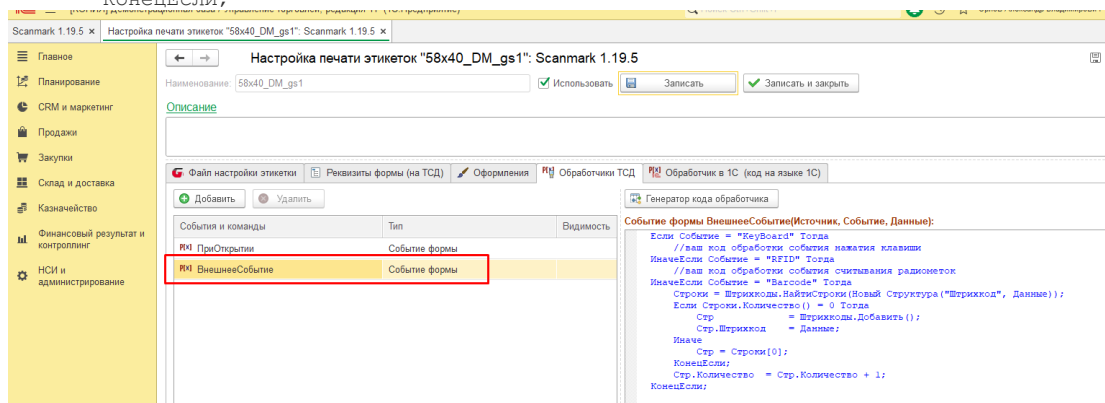
табличной части добавим реквизиты «Штрихкод» и «Количество», изменим свойства реквизитов на:

- «Штрихкод»: Ширина = 15, ПриИзменении =  
Если ТекущиеДанные.Количество = 0 Тогда  
ТекущиеДанные.Количество = 1;  
КонецЕсли;
- «Количество»: Тип = Число(3,0), Ширина = 8, Положение = Новая колонка;



На закладке «Обработчики ТСД» для события «ВнешнееСобытие» добавим код:

```
Если Событие = "KeyBoard" Тогда
    //ваш код обработки события нажатия клавиши
ИначеЕсли Событие = "RFID" Тогда
    //ваш код обработки события считывания радиометок
ИначеЕсли Событие = "Barcode" Тогда
    Строки = Штрихкоды.НайтиСтроки(Новый Структура("Штрихкод", Данные));
    Если Строки.Количество() = 0 Тогда
        Стр = Штрихкоды.Добавить();
        Стр.Штрихкод = Данные;
    Иначе
        Стр = Строки[0];
    КонецЕсли;
    Стр.Количество = Стр.Количество + 1;
КонецЕсли;
```



На закладке «Обработчик в 1С (код на языке 1С)» добавим код:

```
НужноМарок = Штрихкоды.Итог("Количество");

Запрос = Новый Запрос;
Запрос.Текст =
    "ВЫБРАТЬ ПЕРВЫЕ " + НужноМарок + "
    | ПулКодовМаркировкиСУЗ.ДокументОснование КАК ДокументОснование,
```

```

| ПулКодовМаркировкиСУЗ.ЗаказНаЭмиссию КАК ЗаказНаЭмиссию,
| ПулКодовМаркировкиСУЗ.КодМаркировки КАК КодМаркировки,
| ПулКодовМаркировкиСУЗ.ХешСуммаКодаМаркировки КАК ХешСуммаКодаМаркировки,
| ПулКодовМаркировкиСУЗ.Статус КАК Статус,
| ПулКодовМаркировкиСУЗ.Организация КАК Организация,
| ПулКодовМаркировкиСУЗ.СтанцияУправленияЗаказами КАК СтанцияУправленияЗаказами,
| ПулКодовМаркировкиСУЗ.Номенклатура КАК Номенклатура,
| ПулКодовМаркировкиСУЗ.Характеристика КАК Характеристика,
| ПулКодовМаркировкиСУЗ.ПолныйКодМаркировки КАК ПолныйКодМаркировки,
| ПулКодовМаркировкиСУЗ.ДатаЭмиссииУниверсальная КАК ДатаЭмиссииУниверсальная,
| ПулКодовМаркировкиСУЗ.ДатаПечатиУниверсальная КАК ДатаПечатиУниверсальная,
| ПулКодовМаркировкиСУЗ.ШтрихкодУпаковки КАК ШтрихкодУпаковки,
| ПулКодовМаркировкиСУЗ.GTIN КАК GTIN,
| ПулКодовМаркировкиСУЗ.ВидПродукции КАК ВидПродукции,
| ПулКодовМаркировкиСУЗ.МаксимальнаяРозничнаяЦена КАК МаксимальнаяРозничнаяЦена,
| ПулКодовМаркировкиСУЗ.СпособВводаВОборот КАК СпособВводаВОборот,
| ПулКодовМаркировкиСУЗ.Шаблон КАК Шаблон,
| ПулКодовМаркировкиСУЗ.ХешСуммаУпаковки КАК ХешСуммаУпаковки,
| ПулКодовМаркировкиСУЗ.ТипШтрихкода КАК ТипШтрихкода,
| ПулКодовМаркировкиСУЗ.МаркировкаОстатков КАК МаркировкаОстатков
| ИЗ
| РегистрСведений.ПулКодовМаркировкиСУЗ КАК ПулКодовМаркировкиСУЗ
| ГДЕ
| ПулКодовМаркировкиСУЗ.ДатаПечатиУниверсальная = ДАТАВРЕМЯ(1, 1, 1)
//| И ПулКодовМаркировкиСУЗ.ВидПродукции = ЗНАЧЕНИЕ(Перечисление.ВидыПродукцииИС.Обувь)
| И ПулКодовМаркировкиСУЗ.МаркировкаОстатков = ИСТИНА";

```

```

ТЗ Коды = Запрос.Выполнить().Выгрузить();
Хватает = ТЗ Коды.Количество() = НужноМарок;

```

```

Если Хватает Тогда
    Запрос = Новый Запрос;
    Запрос.Текст =

```

```

        "ВЫБРАТЬ
        | ШтрихкодыНоменклатуры.Штрихкод КАК Штрихкод,
        | ШтрихкодыНоменклатуры.Номенклатура КАК Номенклатура,
        | ШтрихкодыНоменклатуры.Характеристика КАК Характеристика,
        | ШтрихкодыНоменклатуры.Упаковка КАК Упаковка,
        | ШтрихкодыНоменклатуры.Номенклатура.ЕдиницаИзмерения КАК ЕдиницаИзмерения,
        | ШтрихкодыНоменклатуры.Номенклатура.Артикул КАК Артикул,
        | ШтрихкодыНоменклатуры.Номенклатура.Марка КАК Бренд,
        | ШтрихкодыНоменклатуры.Номенклатура.Производитель КАК Производитель
        | ИЗ
        | РегистрСведений.ШтрихкодыНоменклатуры КАК ШтрихкодыНоменклатуры
        | ГДЕ
        | ШтрихкодыНоменклатуры.Штрихкод В (&Штрихкоды) ";

```

```

Запрос.УстановитьПараметр("Штрихкоды", Штрихкоды.ВыгрузитьКолонку("Штрихкод"));

```

```

Данные = Запрос.Выполнить().Выгрузить();

```

```

НачатьТранзакцию();

```

```

пп = 0;

```

```

Для Каждого СтрокаДанные Из Данные Цикл

```

```

    СтрокиШтрихкоды = Штрихкоды.НайтиСтроки(Новый Структура("Штрихкод", СтрокаДанные.Штрихкод));
    Если СтрокиШтрихкоды.Количество() = 0 Тогда
        Продолжить;

```

```

    Иначе

```

```

        Кодов = СтрокиШтрихкоды[0].Количество;

```

```

    КонецЕсли;

```

```

    Имя = СокрЛП(СокрЛП(СтрокаДанные.Номенклатура) + " " + СокрЛП(СтрокаДанные.Характеристика));

```

```

    Для К = 1 по Кодов Цикл

```

```

        пп = пп + 1;

```

```

        СтрокаПула = ТЗ_Коды[пп - 1];

```

```

        // инициализация данных таблицы "ДанныеЭтикетки"

```

```

        СтрТСД = ДанныеЭтикетки.Добавить();
        СтрТСД.name1 = Сред(Имя, 1, 36);
        СтрТСД.name2 = Сред(Имя, 37, 36);
        СтрТСД.articule1 = Сред(СтрокаДанные.Артикул, 1, 12);
        СтрТСД.articule2 = Сред(СтрокаДанные.Артикул, 13, 12);
        //СтрТСД.size1 = СтрокаДанные.;
        //СтрТСД.color1 = СтрокаДанные.;
        //СтрТСД.color2 = СтрокаДанные.;
        СтрТСД.brand1 = Сред(СтрокаДанные.Бренд, 1, 16);
        СтрТСД.brand2 = Сред(СтрокаДанные.Бренд, 17, 16);
        //СтрТСД.product1 = СтрокаДанные.;
        //СтрТСД.product2 = СтрокаДанные.;

```

```

СтрТсД.manufacturer1 = СтрокаДанные.Производитель;
//СтрТсД.country1 = СтрокаДанные.;
СтрТсД.markingcode1 = Сред (СтрокаПула.КодМаркировки, 5, 14);
СтрТсД.markingcode2 = Сред (СтрокаПула.КодМаркировки, 23, 13);
//СтрТсД.markingcode3 =
СтрТсД.barcode1 = СтрокаДанные.Штрихкод;
СтрТсД.КодДатаМатрикс = СтрокаПула.ПолныйКодМаркировки;

```

```

МенеджерЗаписи = РегистрыСведений.ПулКодовМаркировкиСУЗ.СоздатьМенеджерЗаписи();
ЗаполнитьЗначенияСвойств (МенеджерЗаписи, СтрокаПула);
МенеджерЗаписи.ДатаПечатиУниверсальная = ТекущаяДата();
МенеджерЗаписи.Записать (Истина);

```

```

КонецЦикла;
КонецЦикла;

```

```

ЗафиксироватьТранзакцию();

```

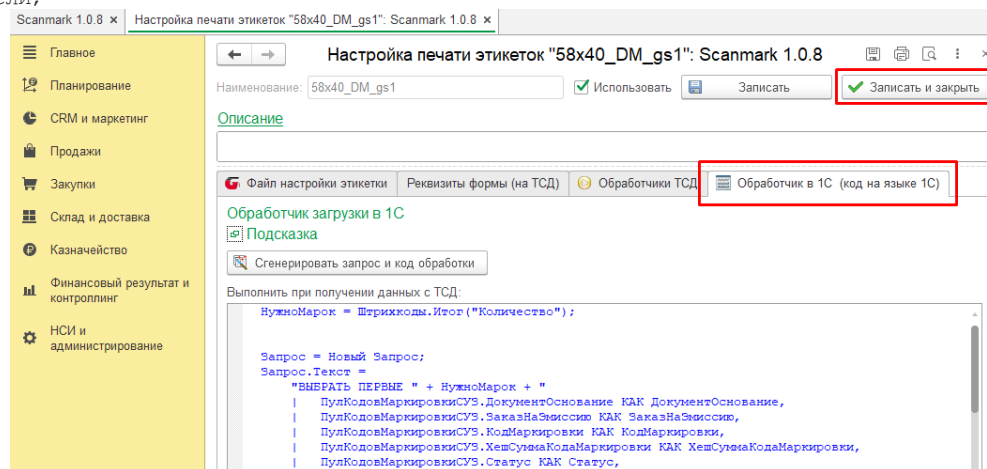
Иначе

```

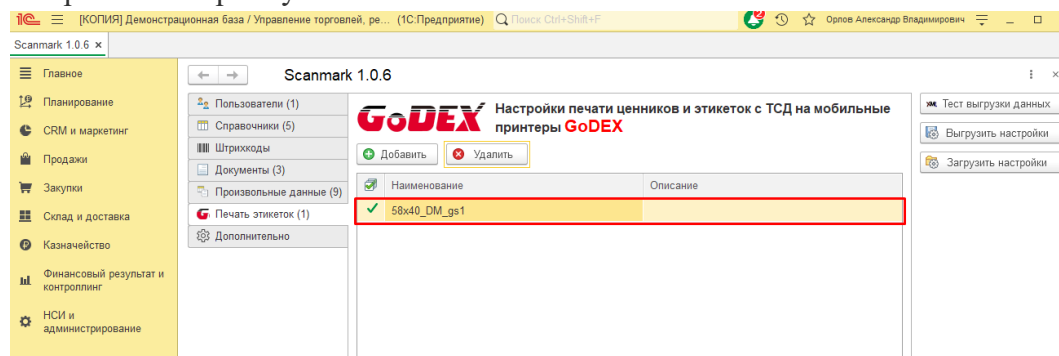
Результат = "Пул кодов маркировки не содержит необходимое количество марок: " + Символы.ПС +
ТЗ Коды.Количество() + " < " + НужноМарок + "!";

```

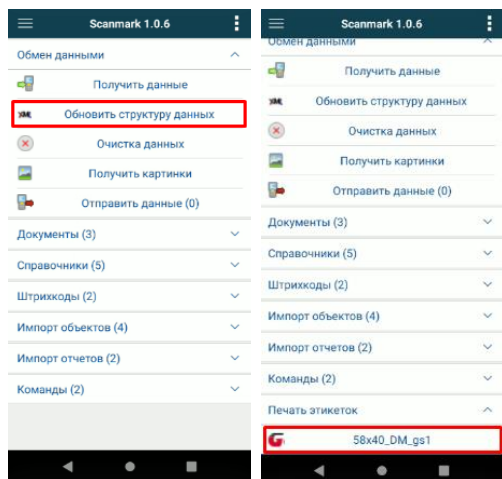
КонецЕсли;



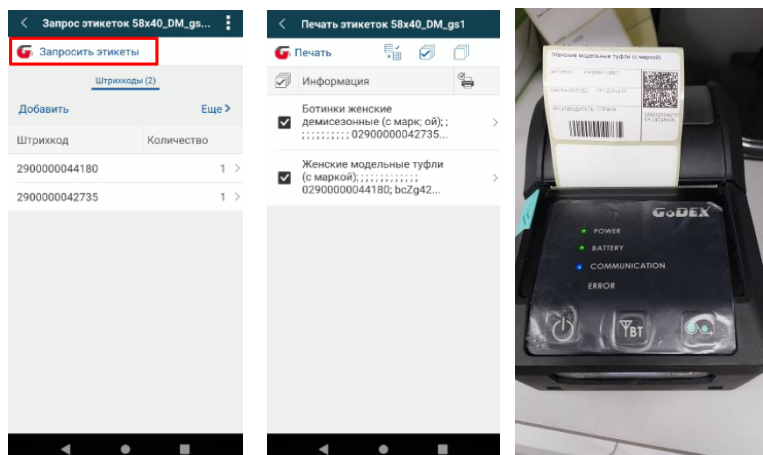
Сохраним настройку.



На ТсД выполним «Обновить структуру данных». В результате появится раздел «Печать этикеток» с пунктом «58x40\_DM\_gs1»:

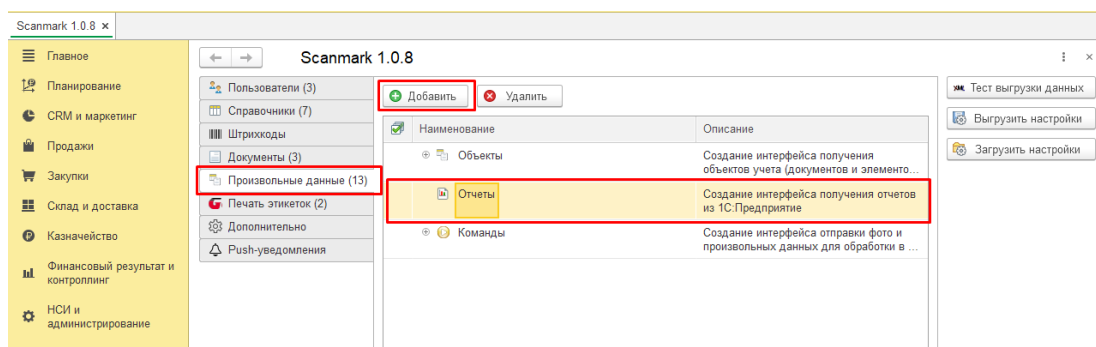


Откроем этикетку «58x40\_DM\_gs1», сканируем штрихкоды товара (EAN-13) и выполним «Запросить этикетки». Следует убедиться, что в «1С:Предприятие» в регистре сведений «Пул маркировки СУЗ» имеются записи с кодами DataMatrix G1. Если такие записи есть, то получим этикетки. Убедимся, что настроено подключение к мобильному принтеру этикеток, принтер включен и заряжен лентой с этикетками размером 58x40. На ТСД нажмем печать.

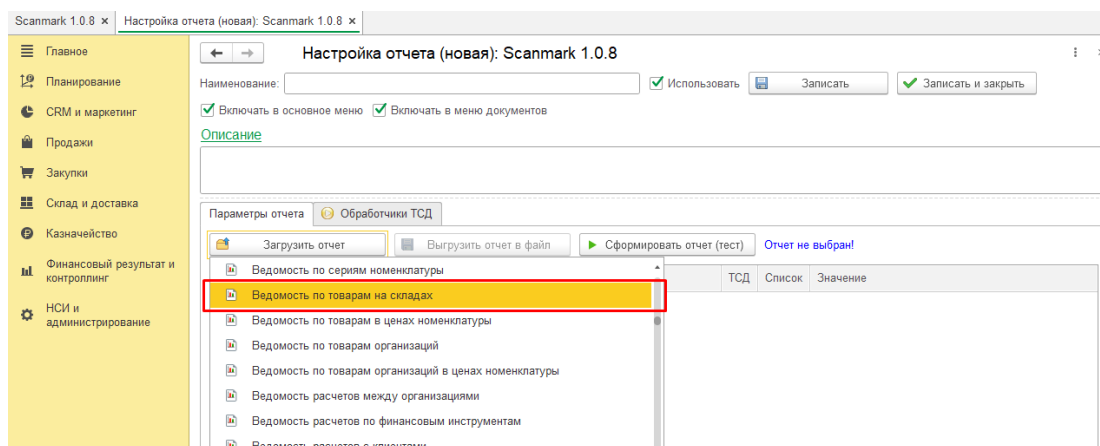
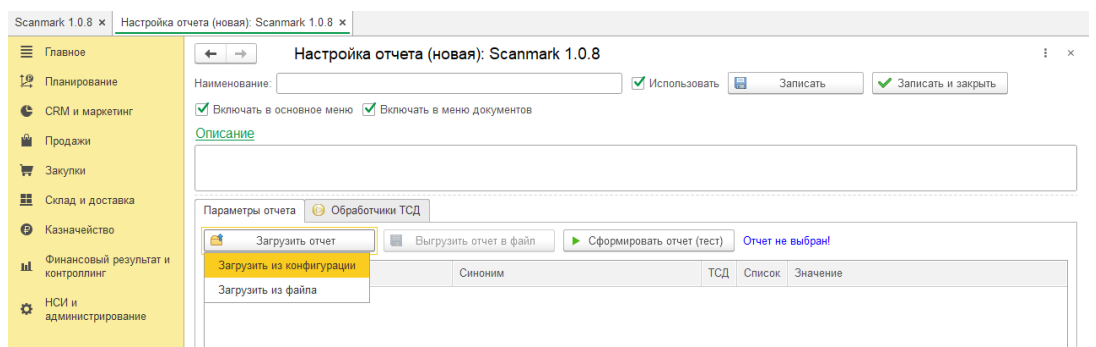


## Задача 13: Создать отчет

В «1С:Предприятие» в обработке «Scanmark» на закладке «Произвольные данные» создадим новую настройку отчета:



В форме настройки отчета загрузим отчет «Ведомость по товарам на складах»:



Отчет возможно загрузить из конфигурации или из файла. Загружаемый отчет должен быть сделан на основе Системы компоновки данных (СКД).

Установим значения в колонках «ТСД» и «Значение» как на картинке ниже:

Демонстрационная база / Управление торговлей, редакция 11 (1С:Предприятие) | Орлов Александр Владимирович

Scanmark 1.0.6 x | Настройка отчета (новая): Scanmark 1.0.6 \* | Тестовый отчет

Наименование: Ведомость по товарам на складах | ☒ Использовать |  | ☒ Записать и закрыть

☒ Включать в основное меню | ☒ Включать в меню документов

[Описание](#)

Параметры отчета | ☒ Обработчик внешнего события (в форме на ТСД)

|  |  | 

Параметр	Синоним	ТСД	Список	Значение
Период	Период	<input type="checkbox"/>		
НачалоПериода	Начало периода	<input checked="" type="checkbox"/>		
КонецПериода	Конец периода	<input checked="" type="checkbox"/>		
ТекущаяДата	Текущая дата	<input type="checkbox"/>		
ИспользуетсяОтборПоСегментуНоменклатуры	Используется отбор по сегменту номенклатуры	<input type="checkbox"/>		
ИспользуетсяОтборПоПересчетуТоваров	Используется отбор по пересчету товаров	<input type="checkbox"/>		
ЕдиницыКоличества	Количество товаров	<input type="checkbox"/>		1
КоличественныйИтогПоЕдИзм	Количественные итоги	<input type="checkbox"/>		Нет
ТекстЗапросаВесНоменклатуры	Текст запроса вес номенклатуры	<input type="checkbox"/>		
ТекстЗапросаОбъемНоменклатуры	Текст запроса объем номенклатуры	<input type="checkbox"/>		

Сохраним настройку отчета, выберем параметры напротив установленных флажков из колонки «ТСД» и сформируем отчет:

Scanmark 1.0.8 x | Настройка отчета (новая): Scanmark 1.0.8 \* | Тестовый отчет

Наименование: Ведомость по товарам на складах | ☒ Использовать |  | ☒ Записать и закрыть

☒ Включать в основное меню | ☒ Включать в меню документов

[Описание](#)

Параметры отчета | ☒ Обработчик ТСД

|  |  | 

Параметр	Синоним	ТСД	Список	Значение
Период	Период	<input type="checkbox"/>		
НачалоПериода	Начало периода	<input checked="" type="checkbox"/>		01.02.2022 0:00:00
КонецПериода	Конец периода	<input checked="" type="checkbox"/>		16.02.2022 0:00:00
ТекущаяДата	Текущая дата	<input type="checkbox"/>		
ИспользуетсяОтборПоСегментуНоменклатуры	Используется отбор по сегменту номенклатуры	<input type="checkbox"/>		
ИспользуетсяОтборПоПересчетуТоваров	Используется отбор по пересчету товаров	<input type="checkbox"/>		
ЕдиницыКоличества	Количество товаров	<input type="checkbox"/>		1
КоличественныйИтогПоЕдИзм	Количественные итоги	<input type="checkbox"/>		Нет
ТекстЗапросаВесНоменклатуры	Текст запроса вес номенклатуры	<input type="checkbox"/>		
ТекстЗапросаОбъемНоменклатуры	Текст запроса объем номенклатуры	<input type="checkbox"/>		

Scanmark 1.0.8 x | Настройка отчета "Ведомость по товарам на складах": Scanmark 1.0.8 x | Ведомость по товарам на складах x

Параметры: Период: Количество товаров: В единицах для отчетов

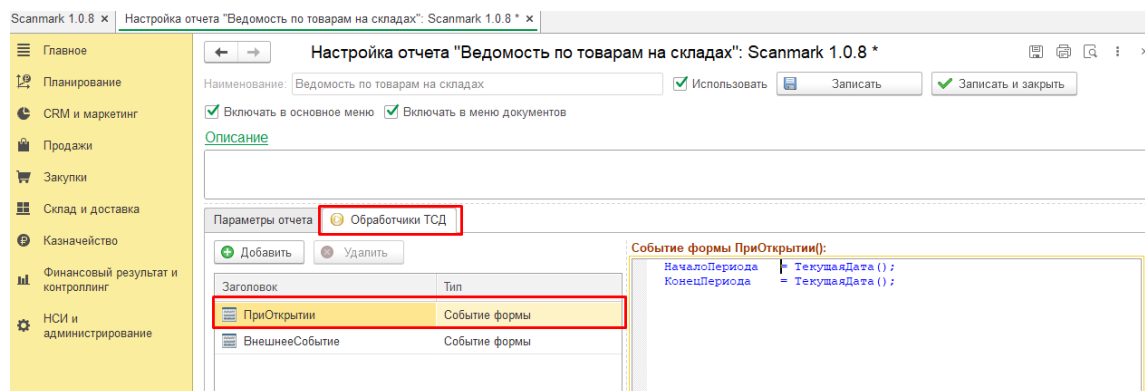
Склад	Артикул	Номенклатура, Характеристика, Серия	Ед. изм.	Количество	Начальный остаток	Приход	Расход
Бытовая техника	B-789	Вентилятор BINATONE ALPINE 160вт, напольный, оконный, ,	шт	170,000			
	B-987	Вентилятор JIPONIC (Таил.), напольный, ,	шт	5,000			
	ВН-900	Вентилятор настольный, ,	шт	2,000			
	ВН-901	Вентилятор настольный, Модель 901, ,	шт	10,000			
	M-77	Комбайн MOULINEX A77 4C, ,	шт	6,000			
	KB-900	Комбайн кухонный BINATONE FP 67, ,	шт	10,000			
	K-9881	Кондиционер ELEKTA. Без управления, ,	шт	10,000			
	K-9881	Кондиционер ELEKTA. С дистанционным управлением, ,	шт	9,000			
	K-9881	Кондиционер ELEKTA. С ручным управлением, ,	шт	6,000			
	K-7778	Кофе-машина "Универсал", ,	шт	7,000			
	KA-879	Кофеварка BRAUN KF22R, ,	шт	10,000			
	M-678	Кофеварка JACOBS (Австрия), ,	шт	7,000			
	M-545	Миксер BINATONE HM 212.6 скор. 150вт, ,	шт	10,000			
	M-545	Миксер SOLAC мод. 545, ,	шт	10,000			

Отчет сформирован, ошибок нет. Закроем печатную форму.

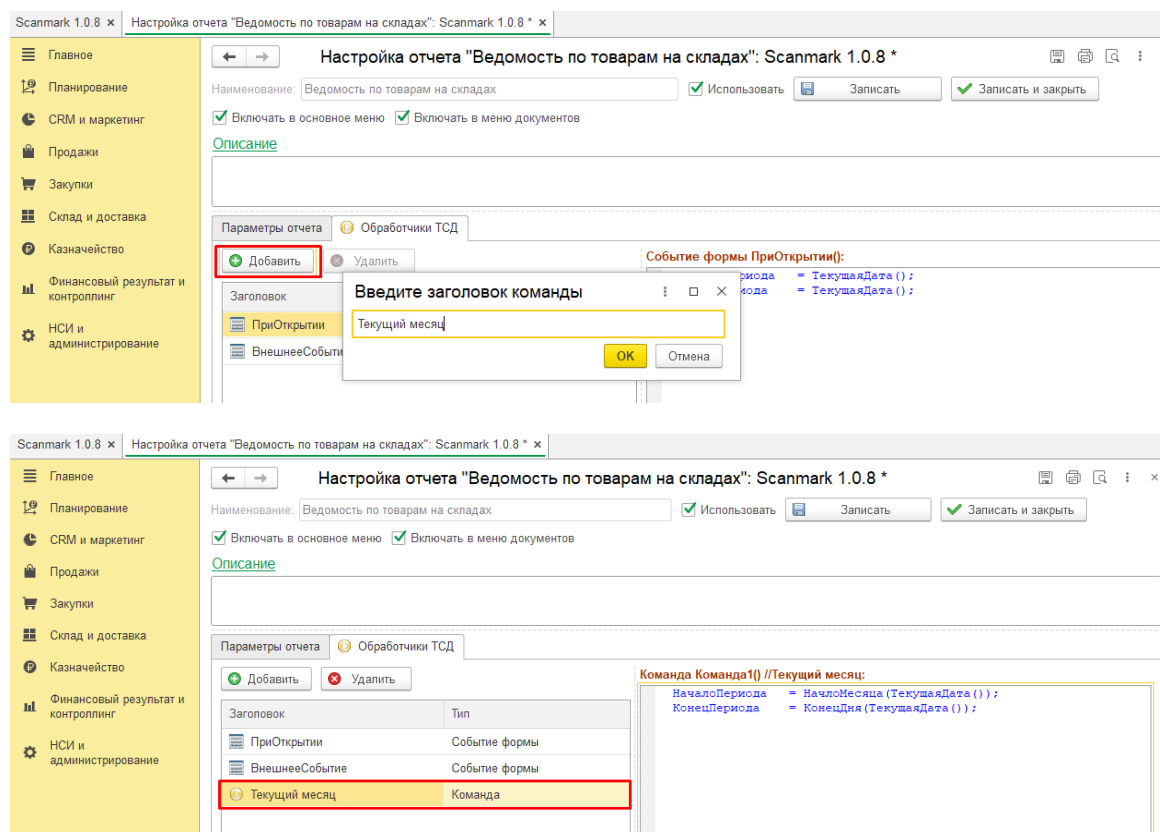
Добавим немного автоматизации. На закладке «Обработчики ТСД» в таблице событий в левой части формы сделаем текущим «ПриОткрытии», в правой части формы в текстовом поле

добавим код:

```
НачалоПериода = ТекущаяДата();  
КонецПериода = ТекущаяДата();
```



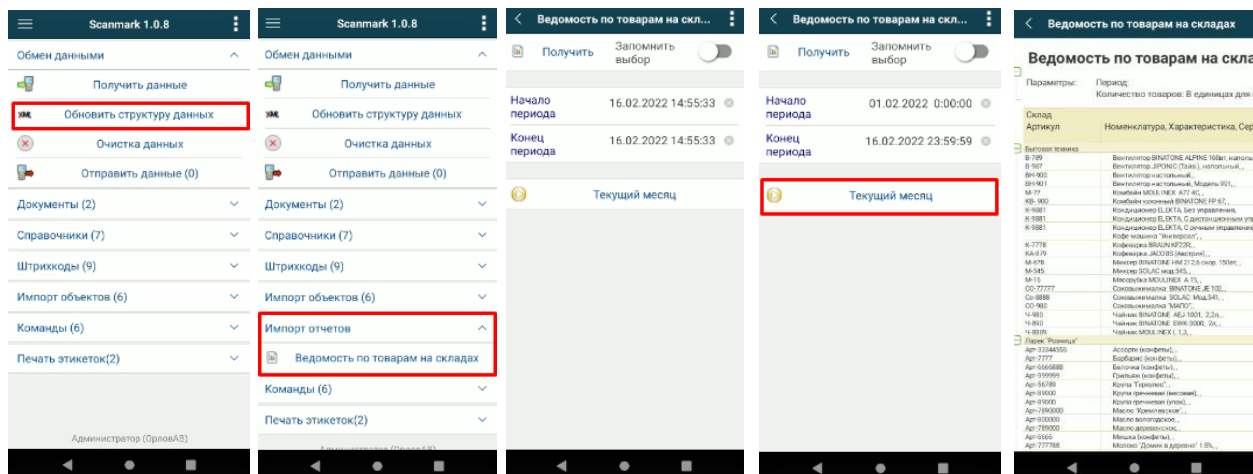
Добавим команду «Текущий месяц». Для созданной команды добавим код обработчика:



```
НачалоПериода = НачалоМесяца(ТекущаяДата());  
КонецПериода = КонецДня(ТекущаяДата());
```

Сохраним настройку отчета. На ТСД обновим структуры данных. Появятся новый раздел «Импорт отчетов» и отчет «Ведомость по товарам на складах» в этом разделе. При открытии формы параметры «Начало периода» и «Конец периода» установились значениями текущей даты и временем. Нажмем на команду (кнопку) «Текущий месяц», в результате период установится с начала текущего месяца по конец текущего дня. Нажмем «Получить». Отчет сформирован и получен, ошибок нет:

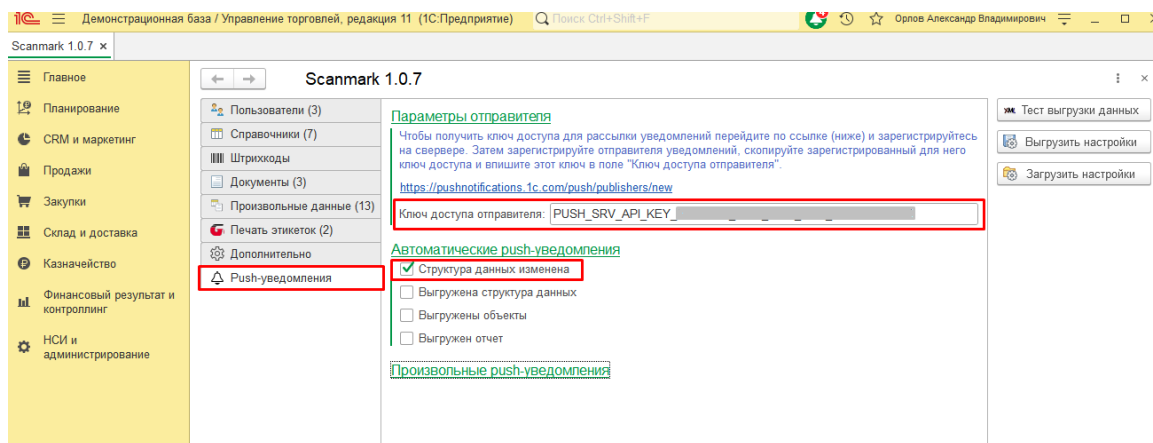




## Задача 14: Пример работы push-уведомлениями

В «1С:Предприятие» в обработке «Scanmark» на закладке «Push-уведомления» произведем настройку.

Перейдем по ссылке <https://pushnotifications.1c.com/push/publishers/new> группе «Параметры отправителя» и выполним регистрацию нового отправителя. Полученный ключ доступа впишем в поле «Ключ доступа отправителя»:



Доставляемые уведомления > Отправители > Новый отправитель

Доставляемые уведомления

Отправители уведомлений

Мобильные приложения

Пользовательское соглашение

### Регистрация нового отправителя уведомлений

Код отправителя \*  
xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxx

Наименование отправителя \*  
Моя база в УТ

Разрешенные IP  
IP адреса (через запятую), с которых разрешена отправка уведомлений

Сохранить Назад

Доставляемые уведомления > Отправители > Редактирование отправителя

Доставляемые уведомления

Отправители уведомлений

Мобильные приложения

Пользовательское соглашение

### Редактирование отправителя уведомлений

Код отправителя \*  
[Redacted]

Наименование отправителя \*  
[Redacted]

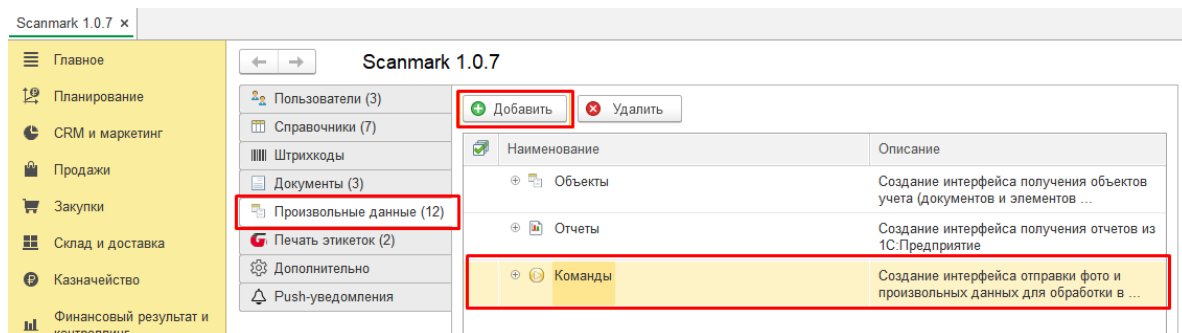
Ключ доступа отправителя  
PUSH\_SRV\_API\_KEY\_ [Redacted] [Перегенерировать](#)

Разрешенные IP  
IP адреса (через запятую), с которых разрешена отправка уведомлений

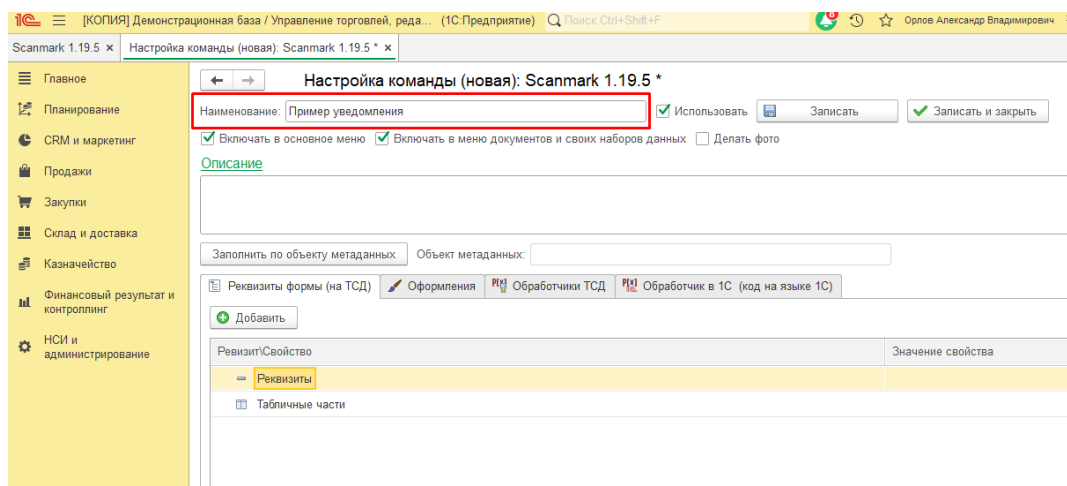
Сохранить Назад

В группе «Автоматические push-уведомления» установим флажок «Структура данных изменена».

На закладке «Произвольные данные» добавим новую команду, в которой сделаем получение системной информации из 1С в виде уведомления:



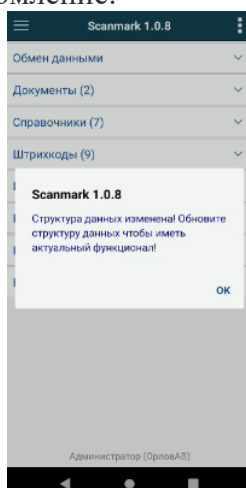
Зададим наименование «Пример уведомления».



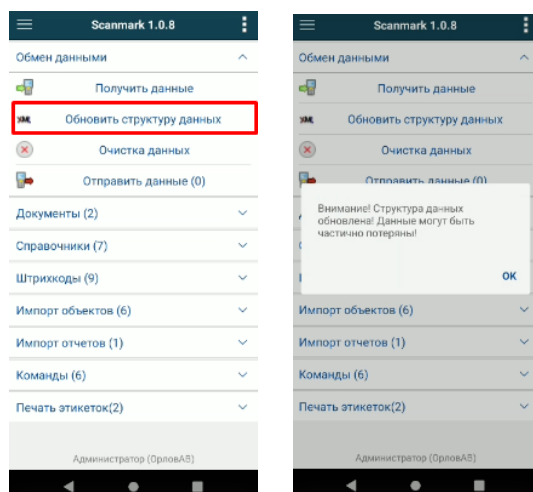
На закладке «Обработчик в 1С (код на языке 1С)» добавим код:

```
Текст = "Имя пользователя = " + ИмяПользователя();
СисИнфо = Новый СистемнаяИнформация;
Текст = Текст + "; " + "Версия ОС = " + СисИнфо.ВерсияОС;
Текст = Текст + "; " + "Версия приложения = " + СисИнфо.ВерсияПриложения;
Текст = Текст + "; " + "Тип платформы = " + СисИнфо.ТипПлатформы;
PushУведомление(Текст, "Системная информация");
```

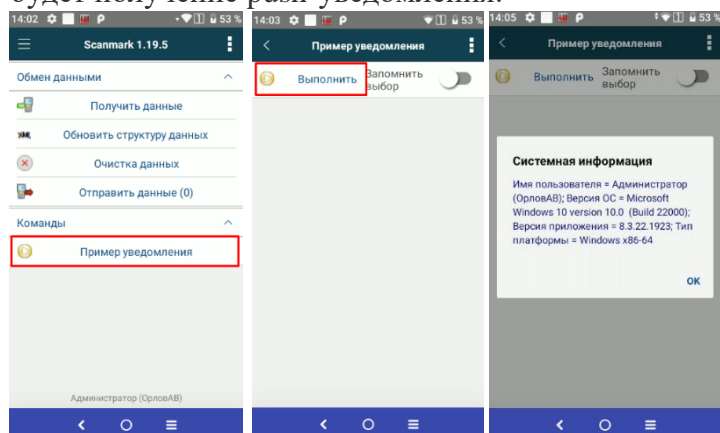
Сохраним настройку команды. Сразу после сохранения настройки на ТСД получим уведомление:



Обновим структуру данных на ТСД.



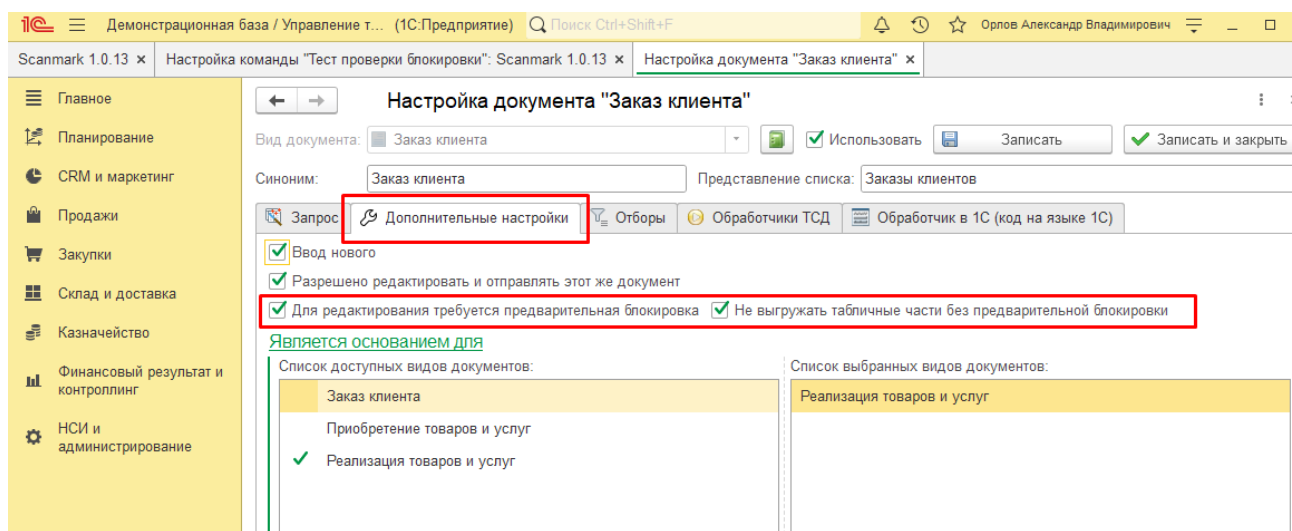
Откроем новую команду, нажмем «Выполнить». Результатом выполнения команды будет получение push-уведомления.



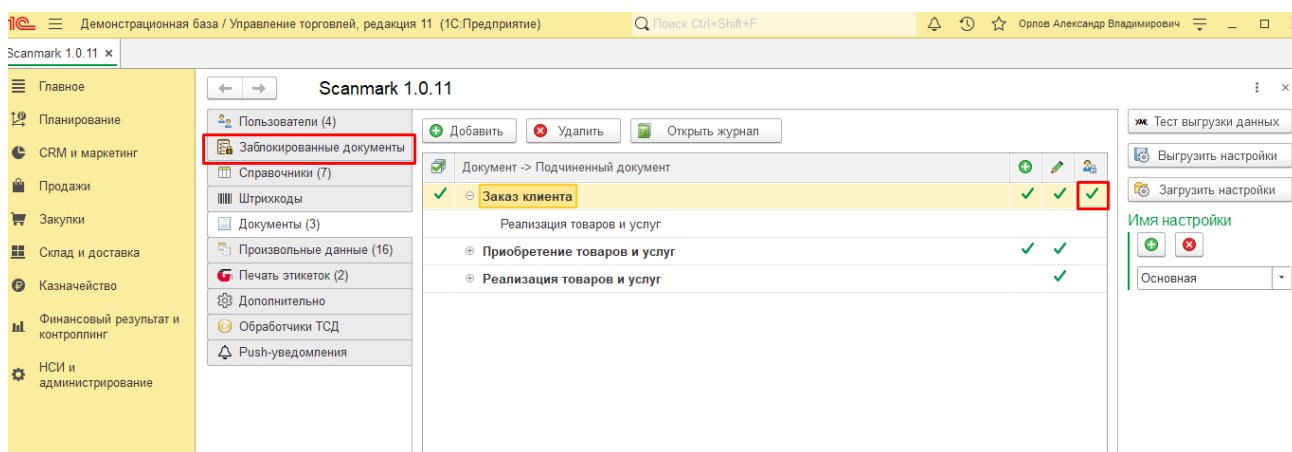
## Задача 15: Работа с блокировками документов

В процессе работы с данными может возникнуть ситуация, когда операторы мобильных устройств могут независимо получать и обрабатывать одни и те же документы. Например, на складе работают одновременно несколько кладовщиков, которые загружают на мобильные устройства документы «Заказ клиента», а затем совершают набор номенклатуры в документы «Реализация товаров и услуг». Чтобы исключить вероятность одновременной обработки кладовщиками одних и тех же заказов предусмотрен режим блокировки документов.

Работа с блокировками настраивается в форме настройки документа. Режим блокировки доступен только при установленном флажке «Разрешено редактировать и отправлять этот же документ».

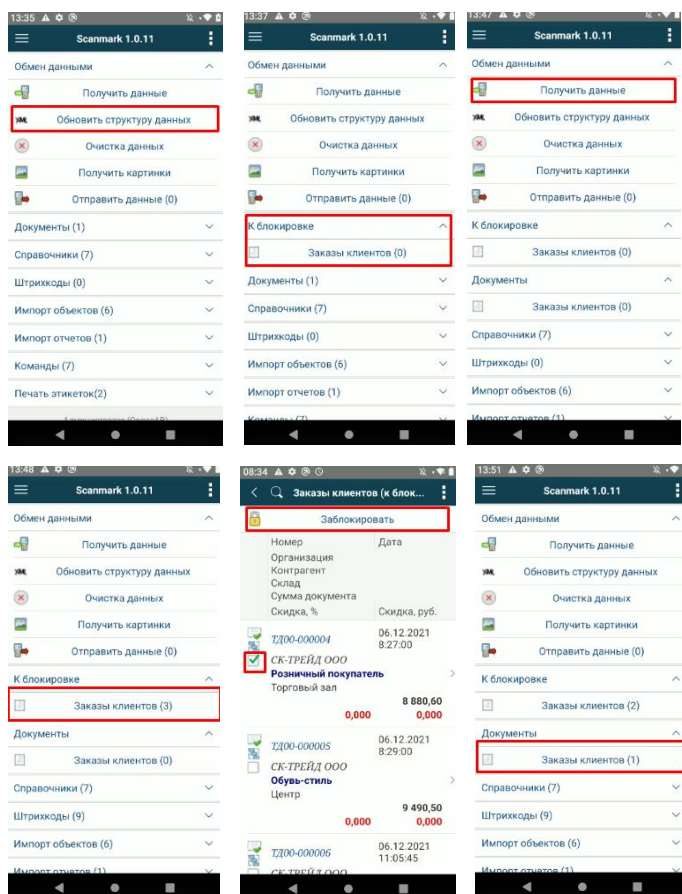


Если хотя бы один вид документа настроен для работы с блокировками, то в основной форме появляется закладка «Заблокированные документы».

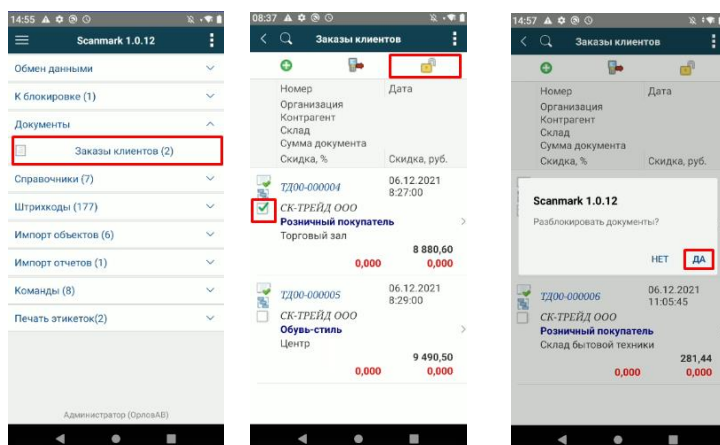


После такой настройки на мобильных устройствах необходимо обновить структуру данных. При этом в основной форме приложения Scanmark появится дополнительное меню «К блокировке». При получении данных в мобильном устройстве документы «Заказ покупателя» будут помещены в меню основной формы «К блокировке». Пользователь может выбрать документы, которые необходимо заблокировать, и выполнить обмен. Выбранные документы будут заблокированы для работы только этого пользователя + мобильное устройство и будут

доступны для дальнейшей работы в меню основной формы «Документы».



Если пользователь решил отказаться от блокировки ранее заблокированных им документов, то это возможно сделать в списке документов. Выберите документы, которые требуется разблокировать, и нажмите кнопку разблокировки. При успешном обмене документы будут разблокированы.



В форме настройки «1С:Предприятие» также возможно управление блокировками. Данную возможность следует использовать только в качестве получения информации или для «наведения порядка» в списке заблокированных документов. Например, разблокировать «зависшие» блокировки документов. Для управления списком заблокированных документов перейдите на страницу «Заблокированные документы» и произведите необходимые действия.

Демонстрационная база / Управление торговлей, редакция 11 (1С-Предприятие) Поиск Ctrl+Shift+F Орлов Александр Владимирович

Scanmark 1.0.11 x Заказы клиентов x

Главное  
Планирование  
CRM и маркетинг  
Продажи  
Закупки  
Склад и доставка  
Казначейство  
Финансовый результат и контроль  
НСИ и администрирование

Scanmark 1.0.11

Пользователи (4)  
**Заблокированные документы**  
Справочники (7)  
Штрихкоды  
Документы (3)  
Произвольные данные (16)  
Печать этикеток (2)  
Дополнительно  
Обработчики ТСД  
Push-уведомления

Порядок группировки:  
Вид документа  
Имя пользователя

Добавить Разблокировать Очистить хранилище заблокированных документов Обновить

Значение	Дата блокировки
ЗаказКлиента	
Администратор (ОрловАВ)	
Заказ клиента ТД00-000004 от 06.12.2021 8:27:00	09.08.2022 13:50:54

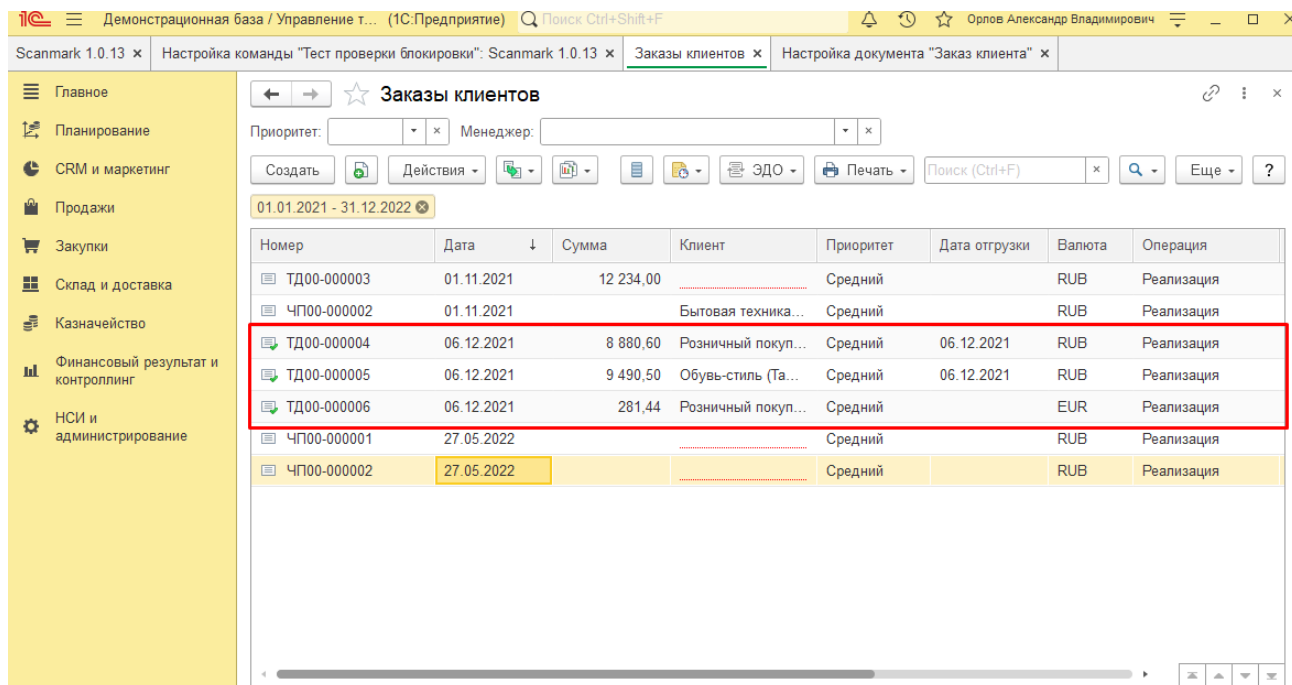
Тест выгрузки данных  
Выгрузить настройки  
Загрузить настройки

Имя настройки  
Основная

## Задача 16: Удалить заблокированные документы другими пользователями.

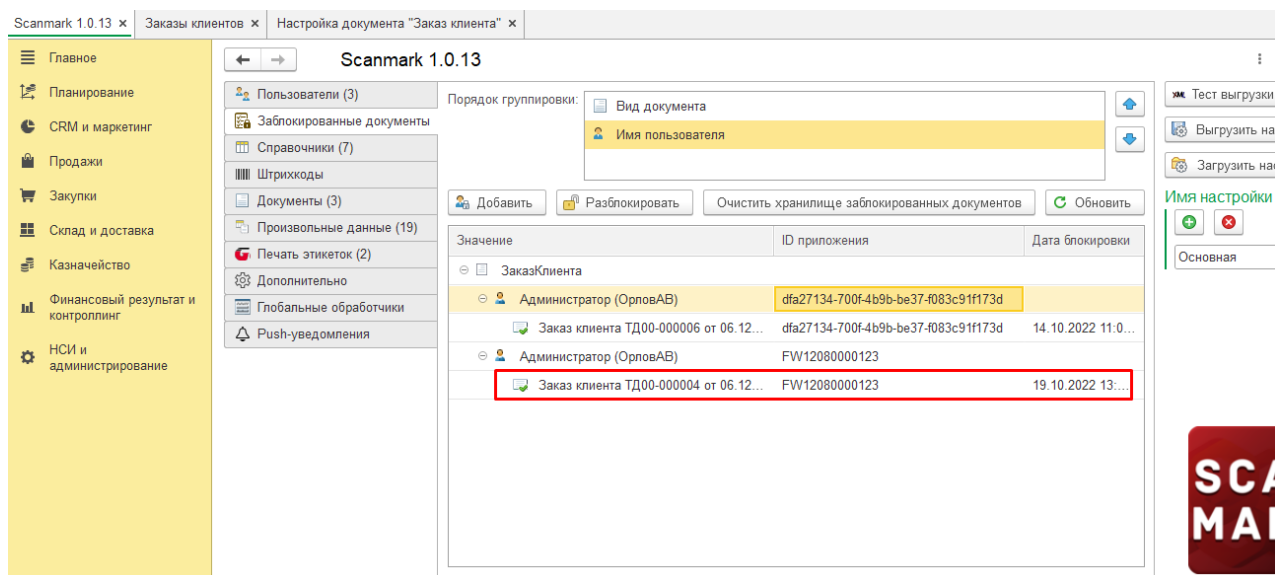
В ПО Scanmark реализованы два типа глобальных обработчиков. Рассмотрим назначение и использование таких обработчиков на примерах в этой и следующей задачах.

Наша задача загрузить проведенные Заказы клиентов и оставить только те документы, которые не заблокированы другими пользователями. В нашем примере это три заказа.



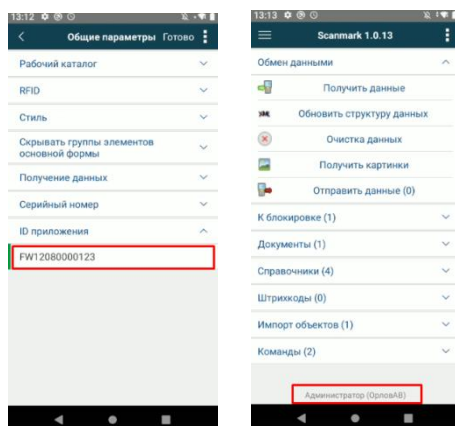
Номер	Дата	Сумма	Клиент	Приоритет	Дата отгрузки	Валюта	Операция
TD00-000003	01.11.2021	12 234,00		Средний		RUB	Реализация
ЧП00-000002	01.11.2021		Бытовая техника...	Средний		RUB	Реализация
TD00-000004	06.12.2021	8 880,60	Розничный покуп...	Средний	06.12.2021	RUB	Реализация
TD00-000005	06.12.2021	9 490,50	Обувь-стиль (Та...	Средний	06.12.2021	RUB	Реализация
TD00-000006	06.12.2021	281,44	Розничный покуп...	Средний		EUR	Реализация
ЧП00-000001	27.05.2022			Средний		RUB	Реализация
ЧП00-000002	27.05.2022			Средний		RUB	Реализация

Проверим какие блокировки уже установлены на такие документы.



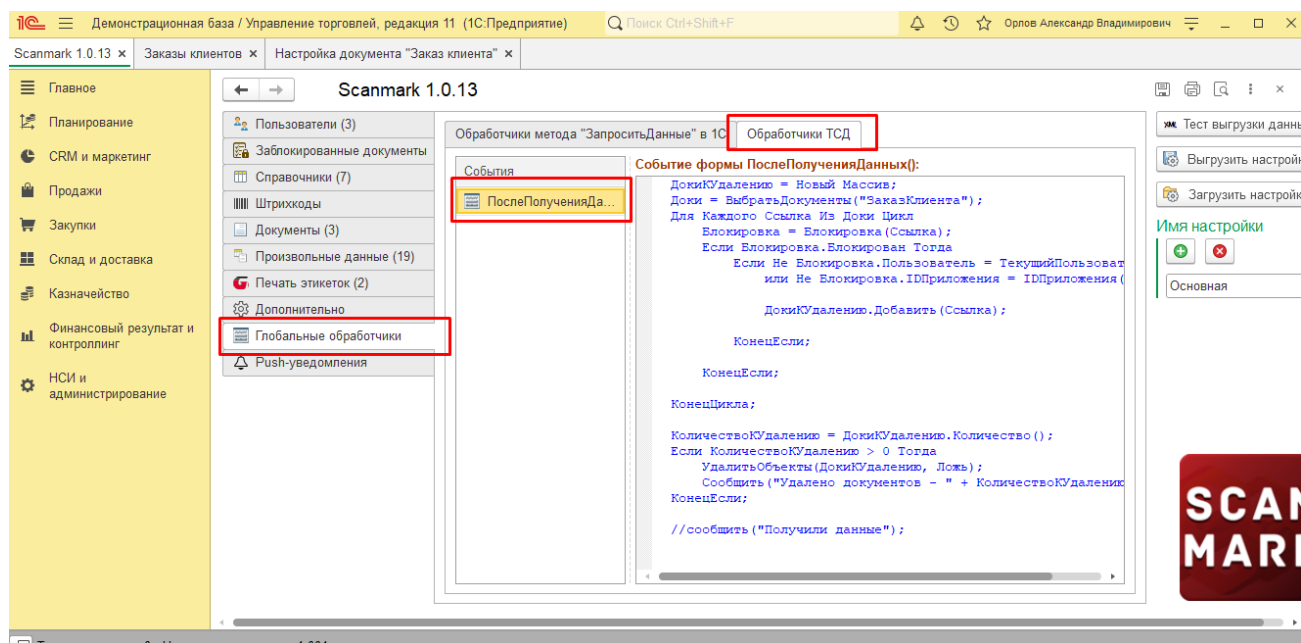
Значение	ID приложения	Дата блокировки
ЗаказКлиента		
Администратор (ОрловАВ)	dfa27134-700f-4b9b-be37-f083c91f173d	
Заказ клиента TD00-000006 от 06.12...	dfa27134-700f-4b9b-be37-f083c91f173d	14.10.2022 11:0...
Администратор (ОрловАВ)	FW12080000123	
Заказ клиента TD00-000004 от 06.12...	FW12080000123	19.10.2022 13:...





Очевидно, что после получения данных из 1С на ТСД должны остаться два документа: один незаблокированный и один уже ранее заблокированный нашим устройством.

В настройках Scanmark 1С:Предприятие перейдем на закладку «Глобальные обработчики», выберем «Обработчики ТСД», где уже присутствует единственный обработчик (события) «ПослеПолученияДанных».



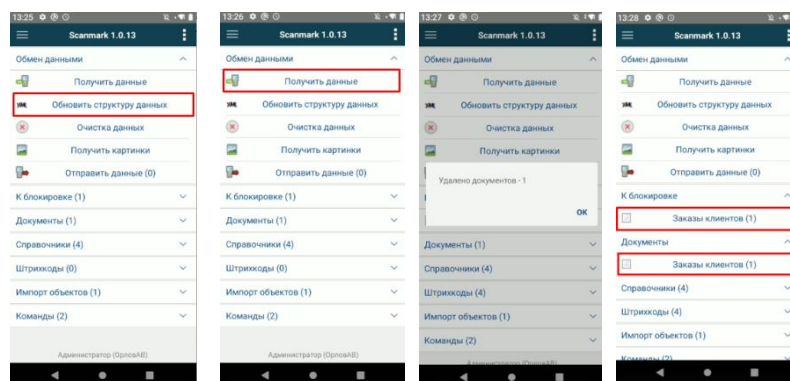
В редакторе Событие формы ПослеПолученияДанных() напишем код на языке 1С, используя специальные функции Scanmark:

```

ДокиКУдалению = Новый Массив;
Доки = ВыбратьДокументы ("ЗаказКлиента");
Для Каждого Ссылка Из Доки Цикл
    Блокировка = Блокировка (Ссылка);
    Если Блокировка.Блокирован Тогда
        Если Не Блокировка.Пользователь = ТекущийПользователь ()
            или Не Блокировка.IDПриложения = IDПриложения () Тогда
            ДокиКУдалению.Добавить (Ссылка);
        КонецЕсли;
    КонецЕсли;
КонецЦикла;
КоличествоКУдалению = ДокиКУдалению.Количество ();
Если КоличествоКУдалению > 0 Тогда
    УдалитьОбъекты (ДокиКУдалению, Ложь);
    Сообщить ("Удалено документов - " + КоличествоКУдалению);
КонецЕсли;

```

Обновим структуру данных на ТСД. После чего получим данные.

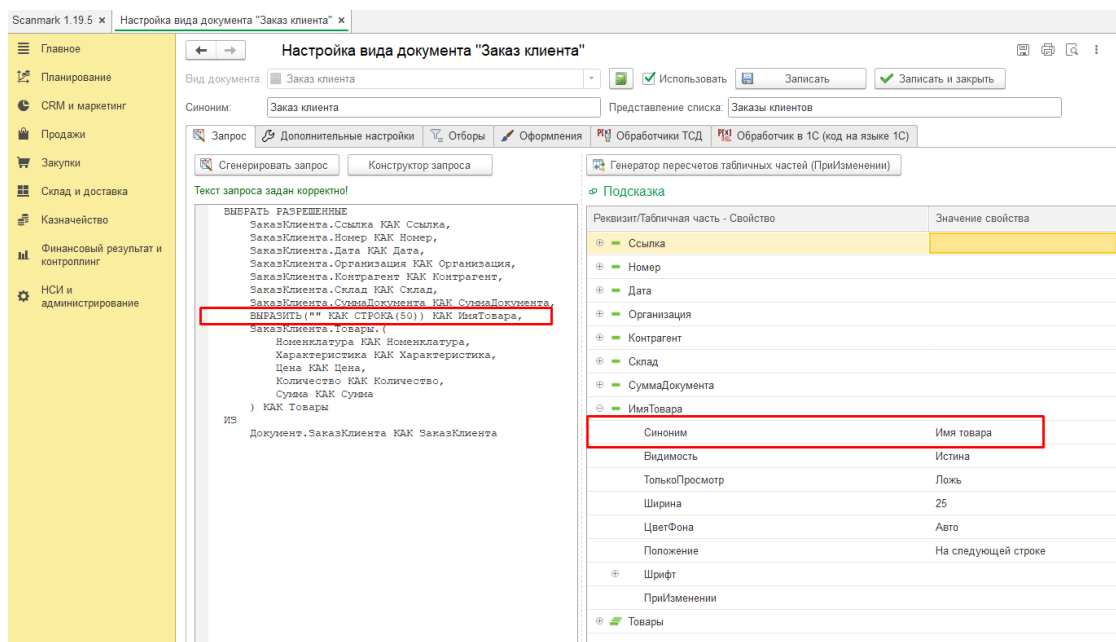


В результате мы видим, что после получения данных осталось два Заказа клиента: один в списке «К блокировке» и один уже в списке «Документы» (заблокирован нашим устройством).

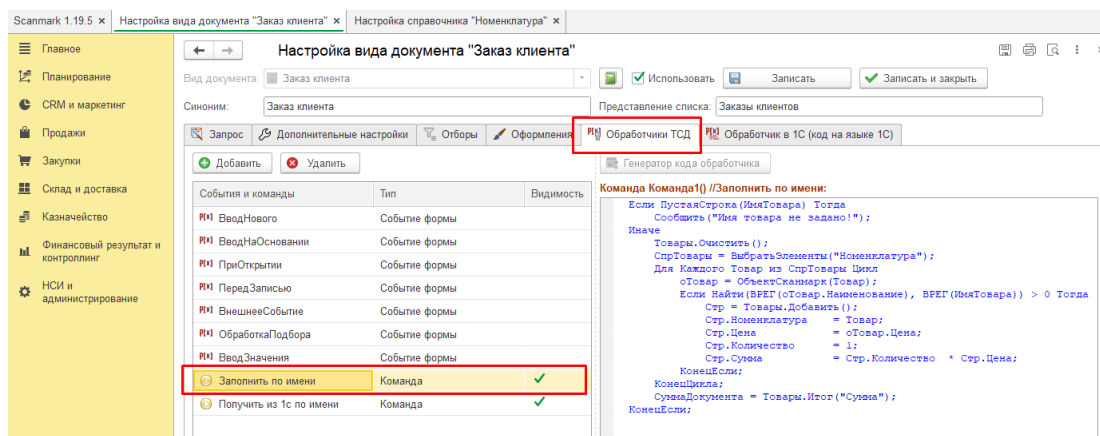
## Задача 17: Пример использования метода «ПолучитьДанныеИз1С».

В качестве примера поставим задачу: на ТСД в форме документа «Заказ клиента» получить номенклатуру из 1С, в наименовании которых есть заданный текст, и заполнить табличную часть актуальными данными.

В настройке документа «Заказ клиента» в запросе добавим виртуальное поле «ИмяТовара»:



На закладке «Обработчики ТСД» добавим команду «Заполнить по имени»



Пропишем код:

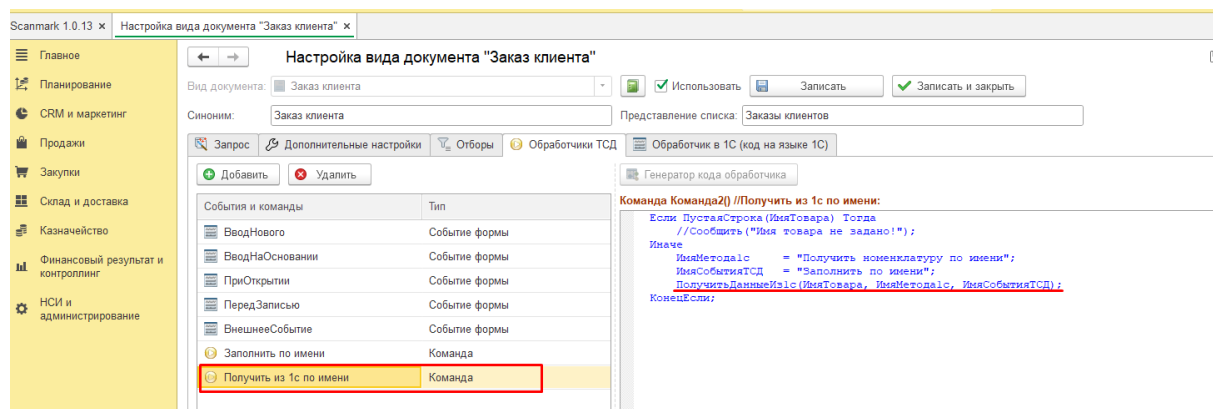
```
Если ПустаяСтрока(ИмяТовара) Тогда
    Сообщить("Имя товара не задано!");
Иначе
    Товары.Очистить();
    СпрТовары = ВыбратьЭлементы("Номенклатура");
    Для Каждого Товар из СпрТовары Цикл
        oТовар = ОбъектСканмарк(Товар);
        Если Найти(ВРЕГ(oТовар.Наименование), ВРЕГ(ИмяТовара)) > 0 Тогда
            Стр = Товары.Добавить();
            Стр.Номенклатура = Товар;
            Стр.Цена = oТовар.Цена;
            Стр.Количество = 1;
            Стр.Сумма = Стр.Количество * Стр.Цена;
```

```

Стр.Сумма = Стр.Количество * Стр.Цена;
КонецЕсли;
КонецЦикла;
СуммаДокумента = Товары.Итог("Сумма");
КонецЕсли;

```

Добавим команду «Получить из 1с по имени» с кодом:

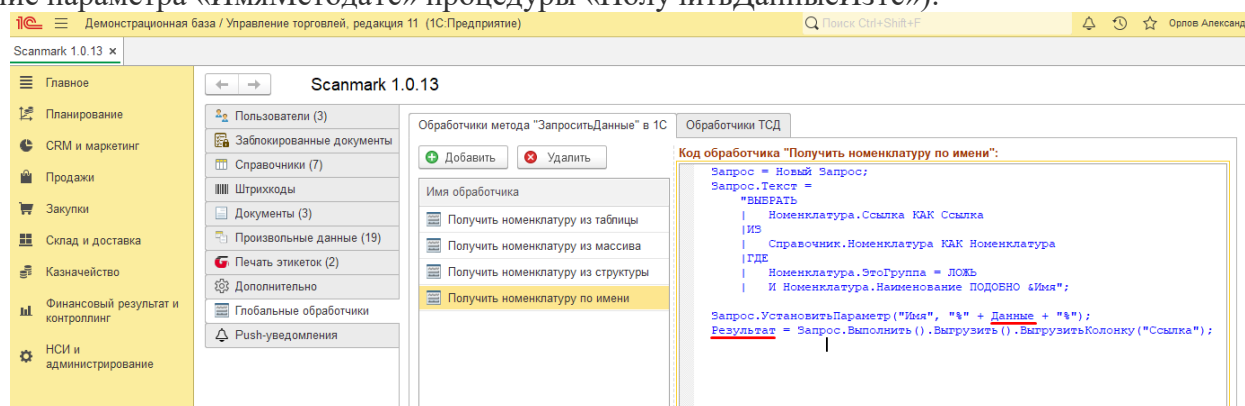


```

Если ПустаяСтрока(ИмяТовара) Тогда
    //Сообщить("Имя товара не задано!");
Иначе
    ИмяМетода1с = "Получить номенклатуру по имени";
    ИмяСобытияТсд = "Заполнить по имени";
    ПолучитьДанныеИз1с(ИмяТовара, ИмяМетода1с, ИмяСобытияТсд);
КонецЕсли;

```

Сохраним настройку документа «Заказ клиента». Перейдем на страницу «Глобальные обработчики» и добавим новый метод «Получить номенклатуру по имени» (см. выше код: значение параметра «ИмяМетода1с» процедуры «ПолучитьДанныеИз1с»):



Добавим код обработчика:

```

Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
| Номенклатура.Ссылка КАК Ссылка
| ИЗ
| Справочник.Номенклатура КАК Номенклатура
| ГДЕ
| Номенклатура.ЭтоГруппа = ЛОЖЬ
| И Номенклатура.Наименование ПОДОВОНО &Имя";

```

```

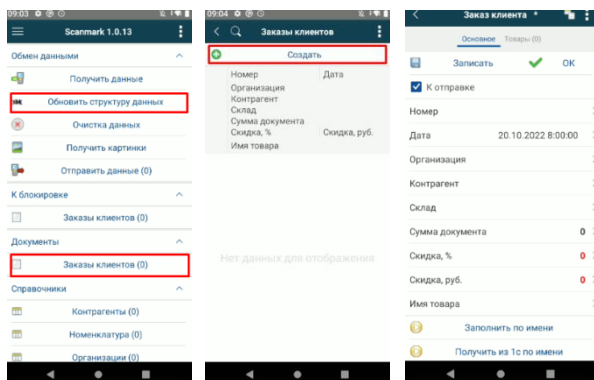
Запрос.УстановитьПараметр("Имя", "%" + Данные + "%");
Результат = Запрос.Выполнить().Выгрузить().ВыгрузитьКолонку("Ссылка");

```

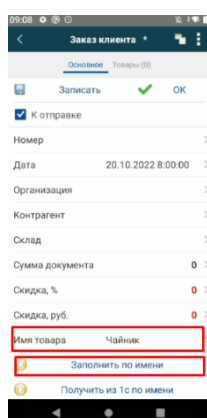
В обработчике всегда присутствует входящий параметр «Данные», который инициализируется значением параметра <Данные> процедуры «ПолучитьДанныеИз1с». Результатом выполнения кода является переменная <Результат>, в который передаются данные объектов для передачи на Тсд. Допустимые типы значений переменной <Результат>: ДокументСсылка, СправочникСсылка, ТаблицаЗначений, СписокЗначений,

Массив, Структура.

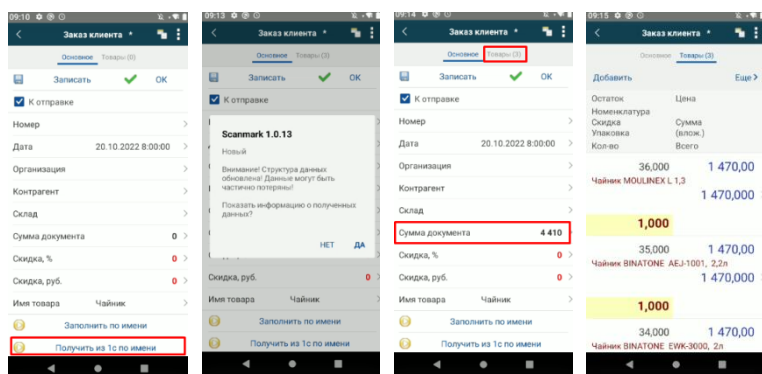
На ТСД обновим структуру данных, а затем создадим новый документ «Заказ клиента»:



Будем искать «чайники» ☺. Сначала попробуем найти в базе данных на ТСД – жмем «Заполнить по имени». Ожидаемо ничего не происходит, т.к. справочник номенклатура пустой:



Теперь попробуем получить товары из 1с. Жмем «Получить из 1с по имени». После выполнения команды видим результат – документ заполнился «чайниками» и произошел пересчет документа:



## Задача 18: Пример использования метода «ВыполнитьСканмарк».

Обработчики ТСД могут содержать подобные части кода. Например, при изменении пользователем количества или цены пересчитывать сумму и НДС:

Сумма = Цена \* Количество;

НДС = Сумма / 120 \* 20;

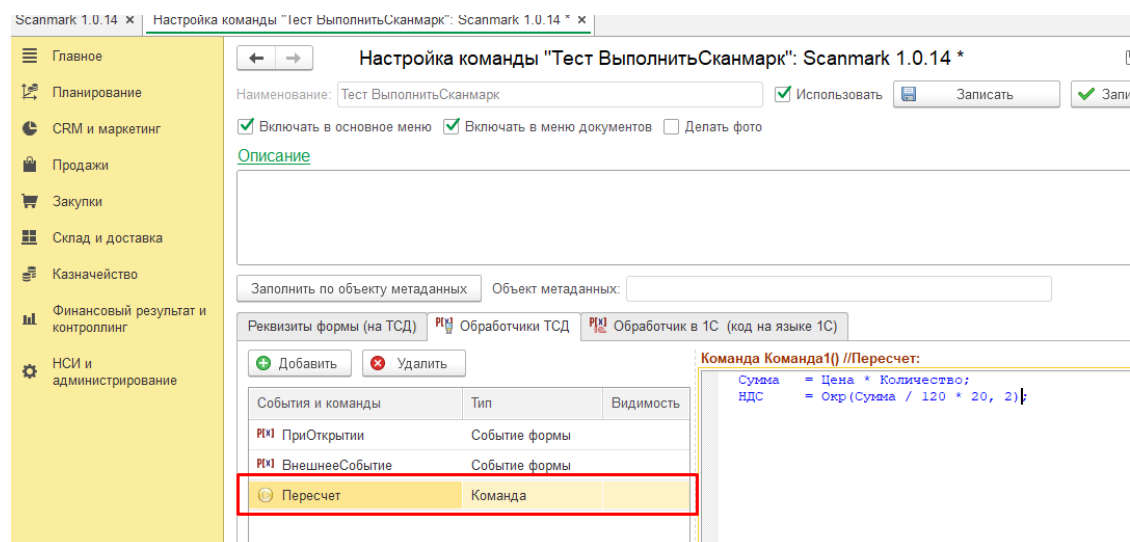
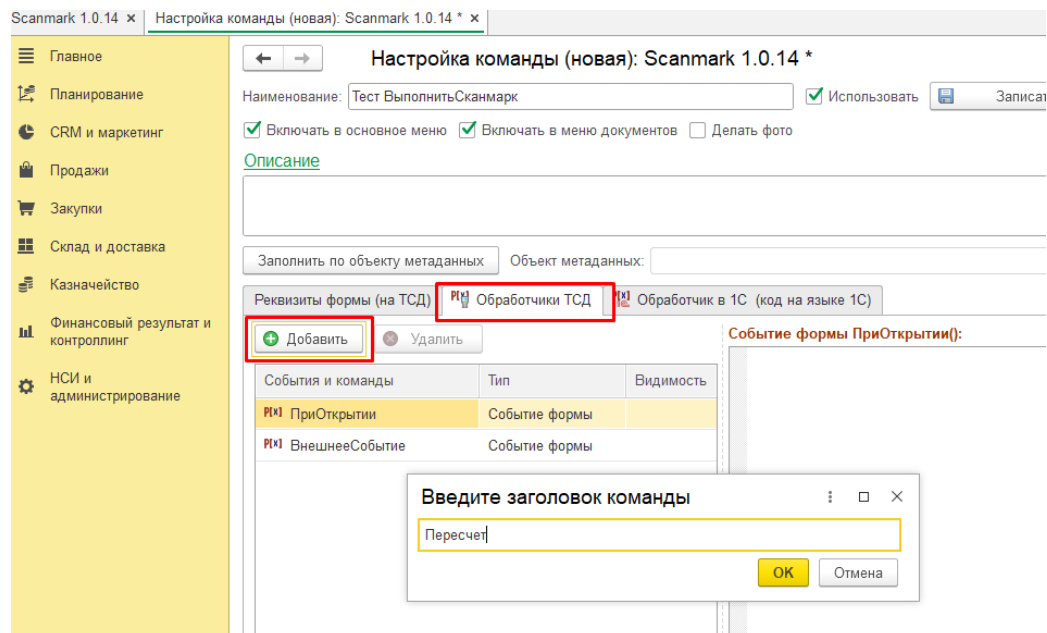
Чтобы не прописывать идентичный код в обработчиках событий при изменении цены или количество, можно использовать процедуру «ВыполнитьСканмарк».

Создадим новую команду и добавим реквизиты Цена, Количество, Сумма, НДС (все тип «Число»), как на картинках ниже:

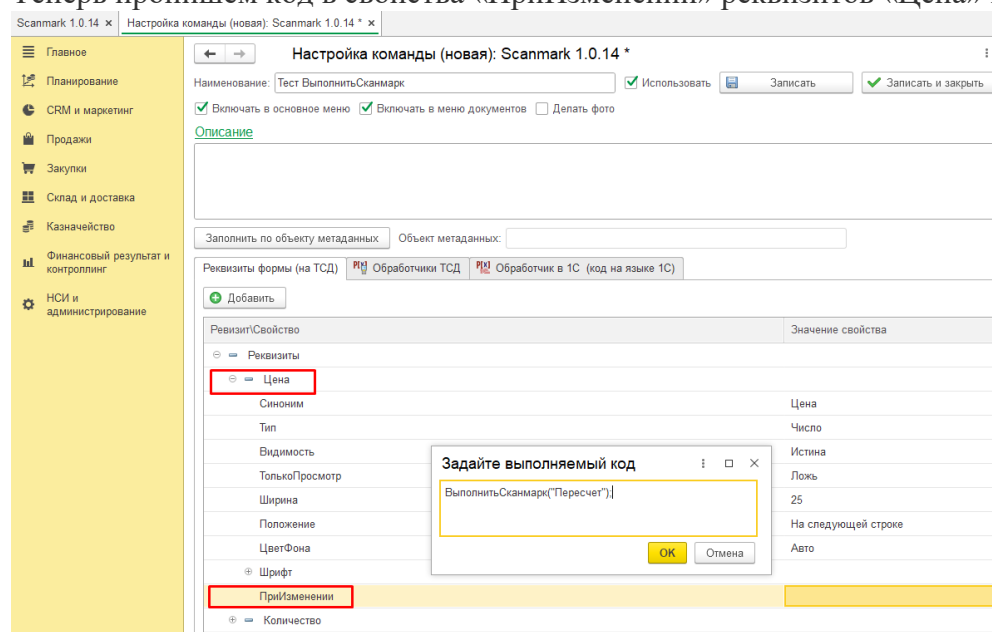
Реквизит\Свойство	Значение свойства
Реквизиты	
Цена	Цена
Тип	Число
Видимость	Истина
ТолькоПросмотр	Ложь
Ширина	25
Положение	На следующей строке
ЦветФона	Авто

Реквизит\Свойство	Значение свойства
Реквизиты	
Цена	Цена
Количество	Количество
Сумма	Сумма
НДС	НДС

Добавим команду «Пересчет» и пропишем код для этой команды:



Теперь пропишем код в свойства «ПриИзменении» реквизитов «Цена» и «Количество»:



Сохраним изменения. На ТСД обновим структуру данных и проверим как это работает:

Scanmark 1.0.14 x Настройка команды (новая): Scanmark 1.0.14 \* x

Настройка команды (новая): Scanmark 1.0.14 \*

Наименование: Тест ВыполнитьСканмарк ☒ Использовать

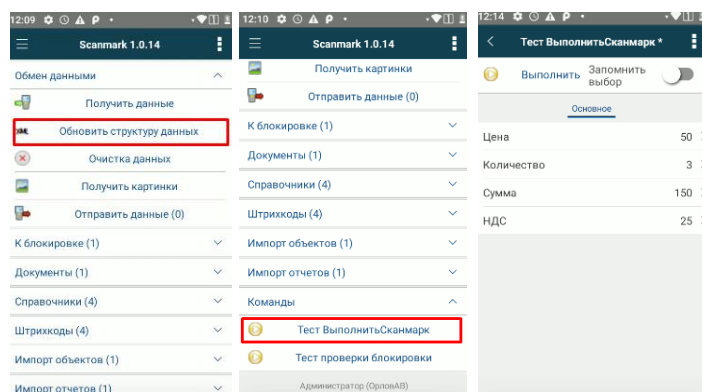
☒ Включать в основное меню ☒ Включать в меню документов ☐ Делать фото

Описание

Заполнить по объекту метаданных:  Объект метаданных:

Реквизиты формы (на ТСД) ☒ Обработчики ТСД ☒ Обработчик в 1С (код на языке 1С)

Реквизит\Свойство	Значение свойства
⊖ Реквизиты	
⊕ Цена	
⊖ Количество	
Синоним	Количество
Тип	Число
Видимость	Истина
ТолькоПросмотр	Ложь
Ширина	25
Положение	На следующей строке
ЦветФона	Авто
⊕ Шрифт	
ПриИзменении	ВыполнитьСканмарк("Пересчет");
⊕ Сумма	
⊕ НДС	

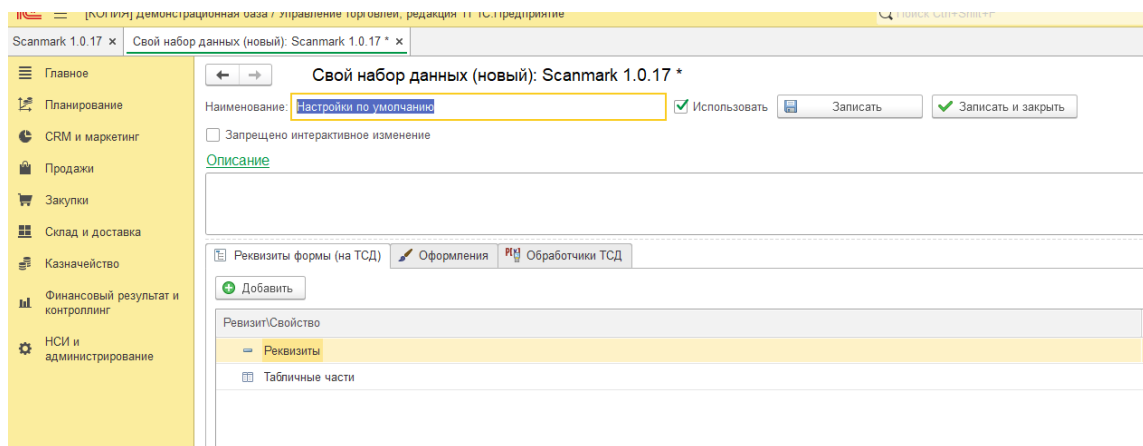
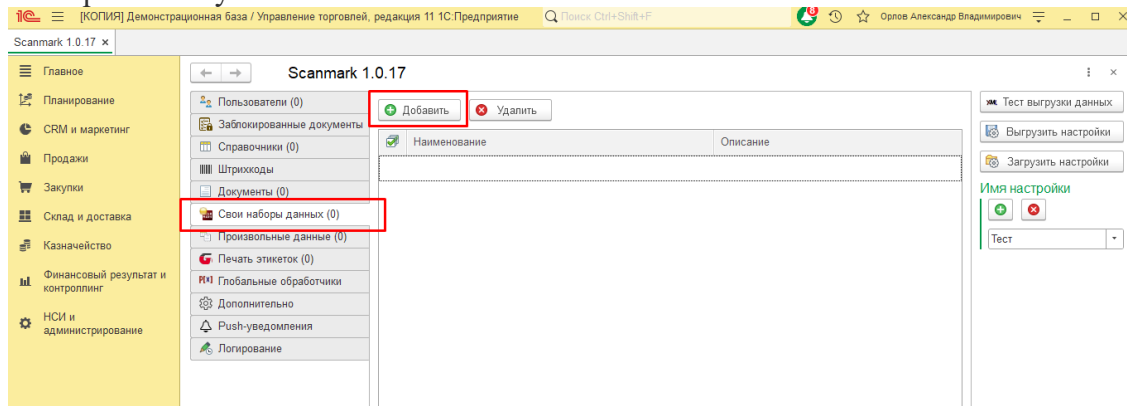




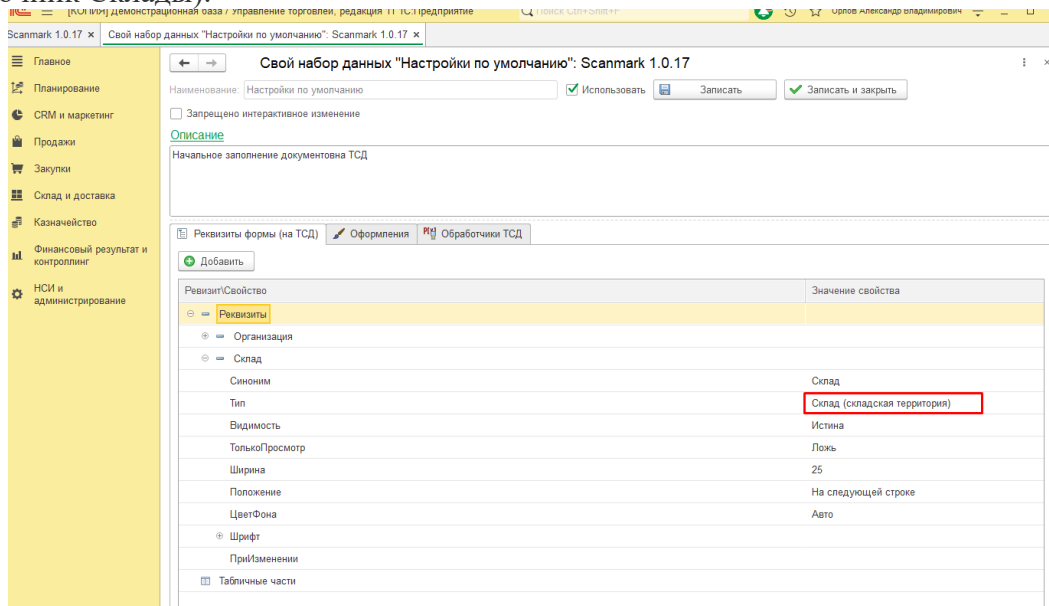
## Задача 19: Пример заполнения «по умолчанию».

При работе на ТСД часто возникает потребность установить значения реквизитов «по умолчанию». Например, при вводе нового документа «Заказ клиента» пользователь заполняет реквизиты «Организация» и «Склад» одними и теми же значениями. Этот процесс можно автоматизировать.

В обработке «Scanmark» откроем закладку «Свои наборы данных» и создадим новый набор «Настройки по умолчанию»:

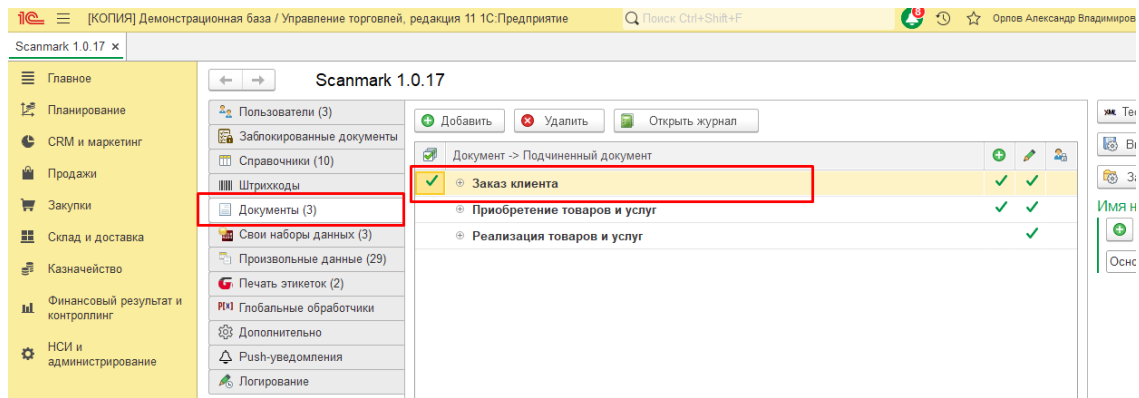


Добавим реквизиты «Организация» (тип Справочник Организации) и «Склад» » (тип Справочник Склады):



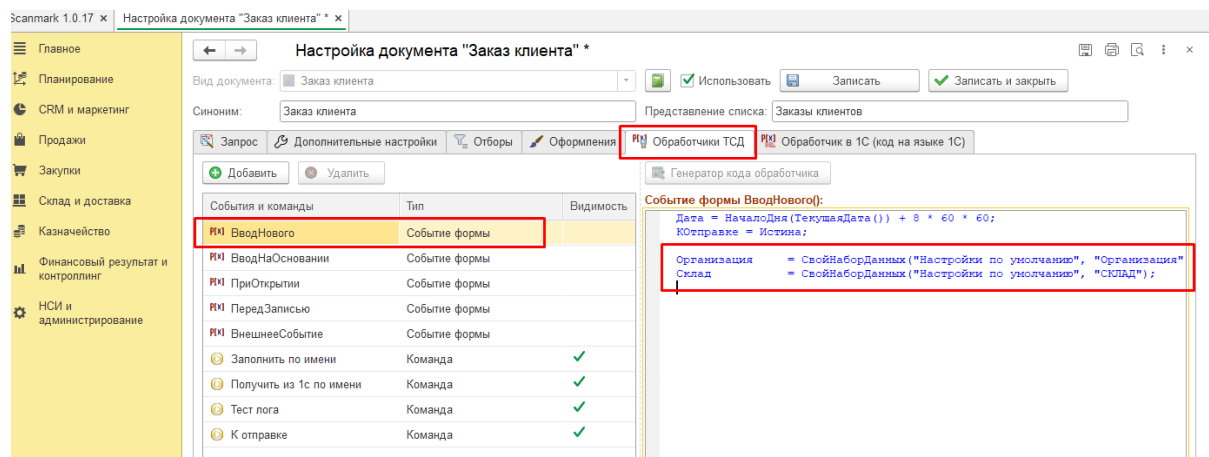
Запишем «Настройки по умолчанию».

Откроем настройку документа «Заказ клиента»:



На закладке «Обработчики ТСД» в событии «Ввод нового» добавим код:

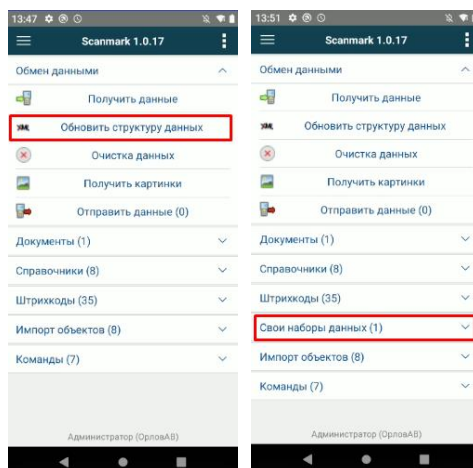
```
Организация = СвойНаборДанных("Настройки по умолчанию", "Организация");
Склад = СвойНаборДанных("Настройки по умолчанию", "СКЛАД");
```



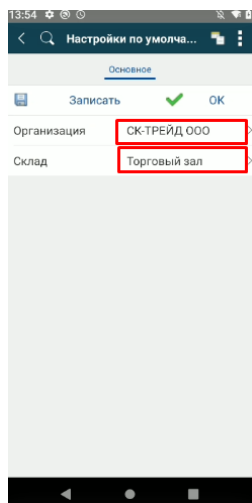
Внимание! В данном примере используется функция «СвойНаборДанных()», описание см. в Разделе «Процедуры и функции обработчиков «Scanmark» данного руководства.

Запишем настройку документа.

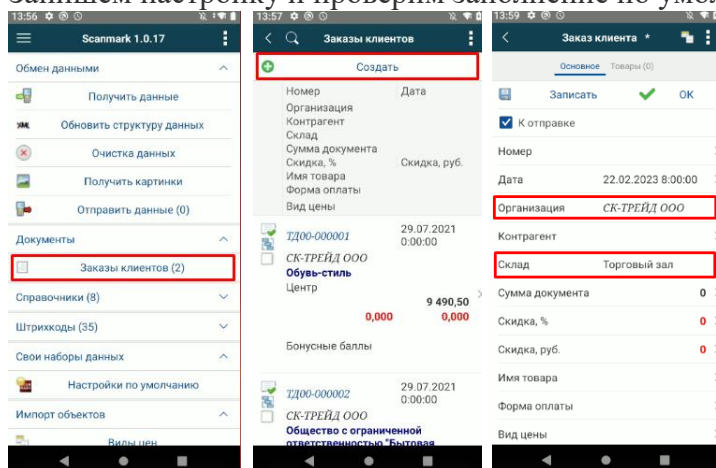
На ТСД в приложении «Scanmark» выполним обновление структуры данных, после которого появится группа объектов «Свои наборы данных»:



Зайдем в новый «набор» и зададим настройки реквизитов по умолчанию:



Запишем настройку и проверим заполнение по умолчанию документа «Заказ клиента»:

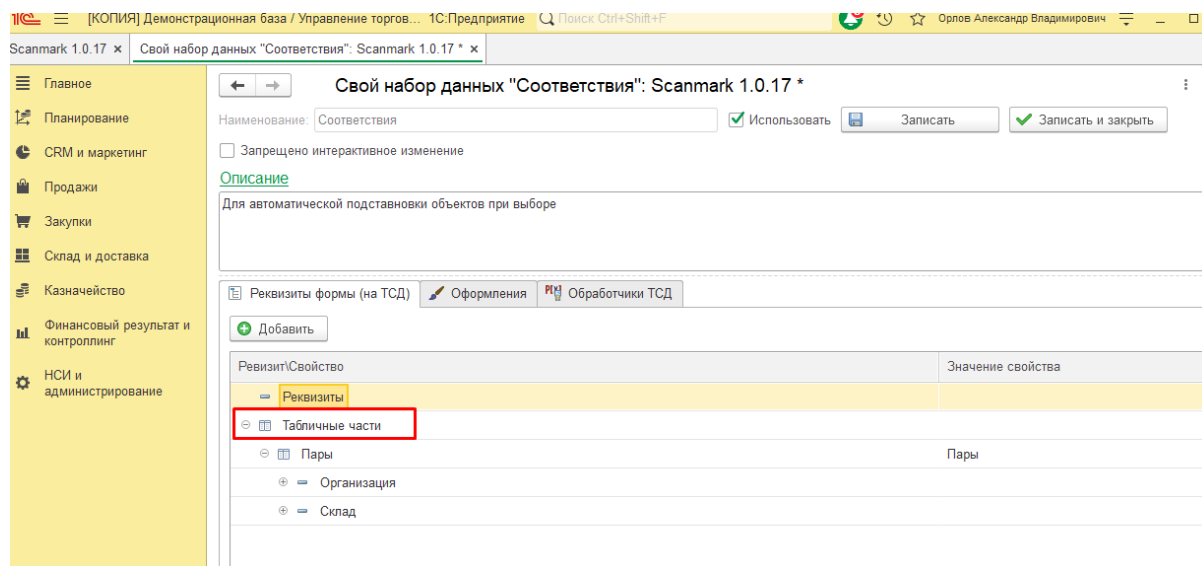


Работает!

## Задача 20: Пример заполнения «по соответствию».

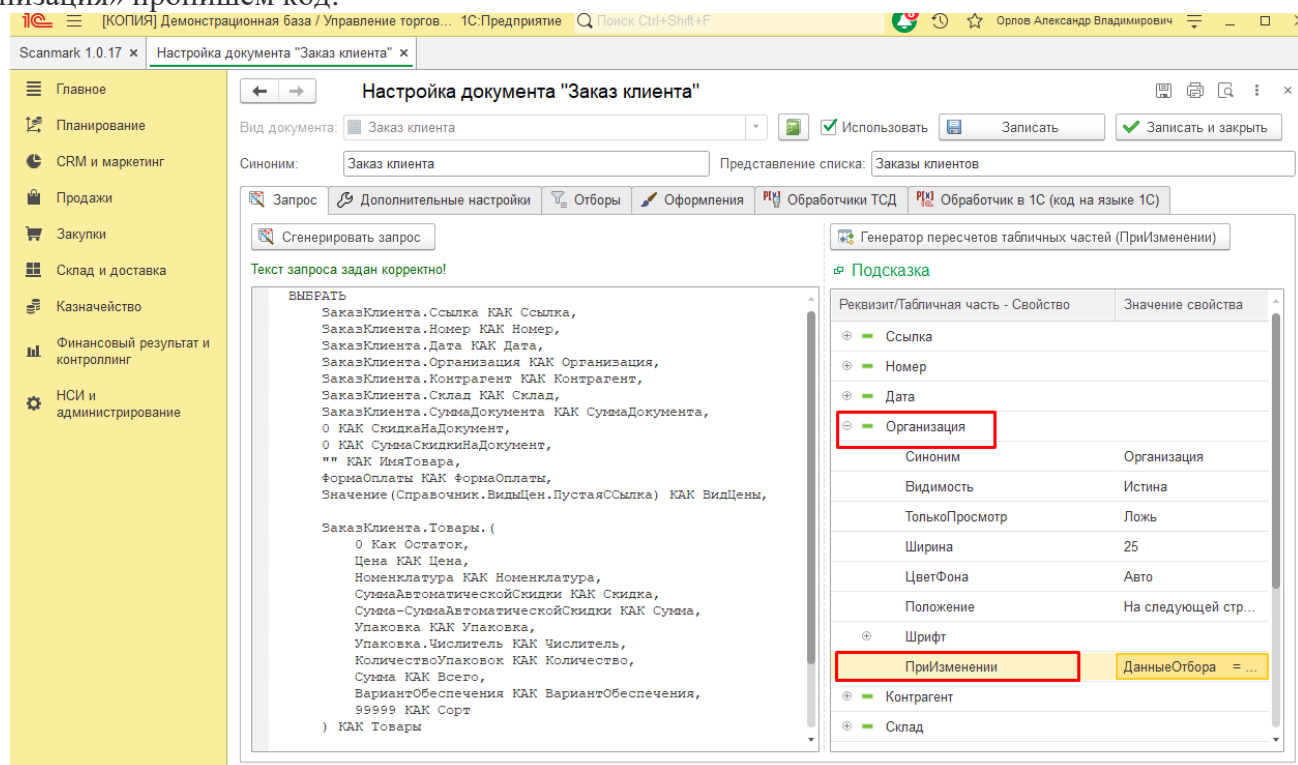
При работе на ТСД часто возникает желание установить значения реквизитов «по соответствию». Например, в документе «Заказ клиента» при заполнении реквизита «Организация» нужно установить определенное (соответствующее) значение реквизита «Склад».

В обработке «Scanmark» откроем закладку «Свои наборы данных» и создадим новый набор «Соответствия», в форме настройки набора добавим табличную часть «Пары», которой добавим реквизиты «Организация» (тип Справочник Организации) и «Склад» (тип Справочник Склады):



Запишем настройку «Соответствия».

Откроем настройку документа «Заказ клиента», в свойстве «ПриИзменении» реквизита «Организация» пропишем код:



```
ДанныеОтбора = Новый Структура();  
ДанныеОтбора.Вставить("Организация", Организация);
```

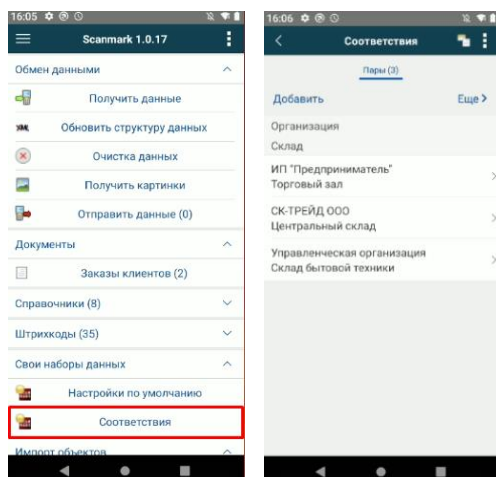
```

ТЗ = СвойНаборДанных("Соответствия", "Пары", ДанныеОтбора);
Если ТЗ.Количество() > 0 Тогда
    Склад = ТЗ[0].Склад;
КонецЕсли;

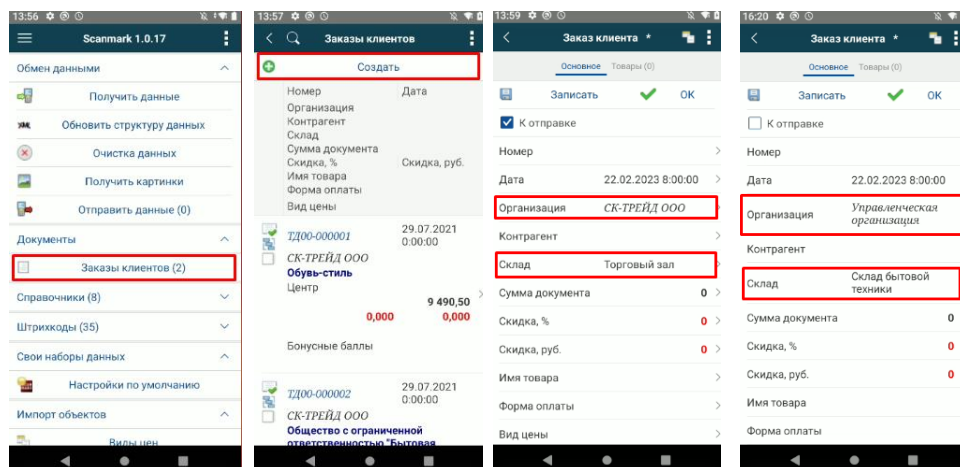
```

Запишем настройку документа.

На ТСД в приложении «Scanmark» выполним обновление структуры данных. В появившемся в приложении «Scanmark» набора данных «Соответствия» зададим несколько пар соответствий Организация – Склад:



После завершения настройки проверим «работу» настройки в документе «Заказ клиента». Создадим новый документ и пробуем выбирать различные организации:

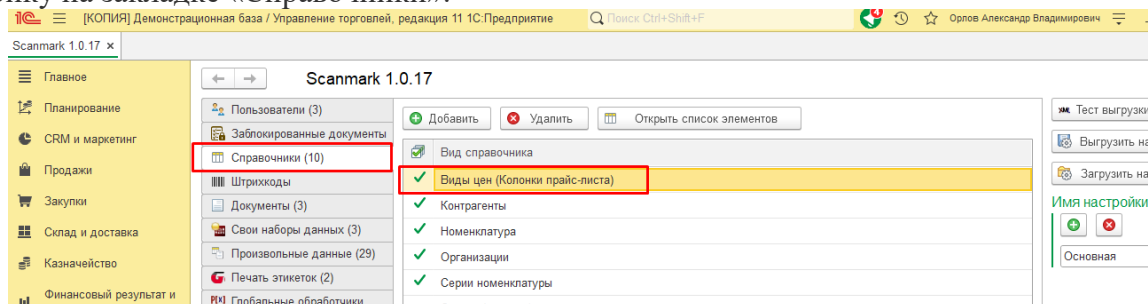


Работает!

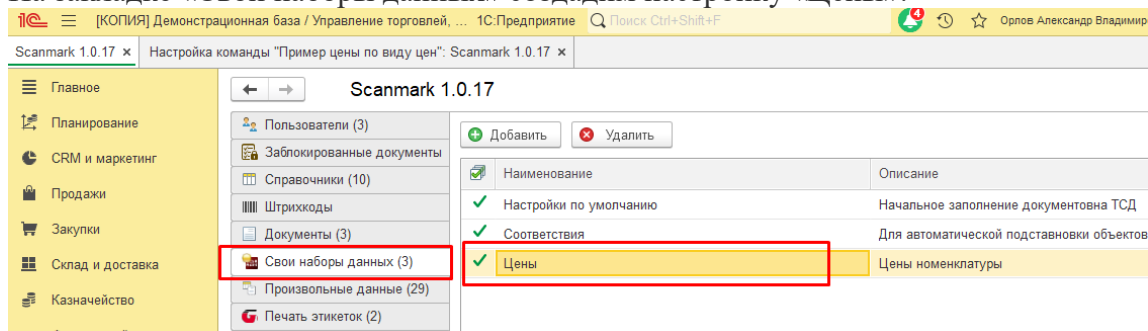
## Задача 21: Пересчет табличной части документа «Заказ клиента» после выбора типа цен.

Сначала составим схему решения задачи. Нам нужно получить и хранить таблицу товаров с ценами по различным видам цен. Хранение и взаимодействие с таким видом данных организуем с помощью объекта учета «Свои наборы данных». Приступим к реализации задачи.

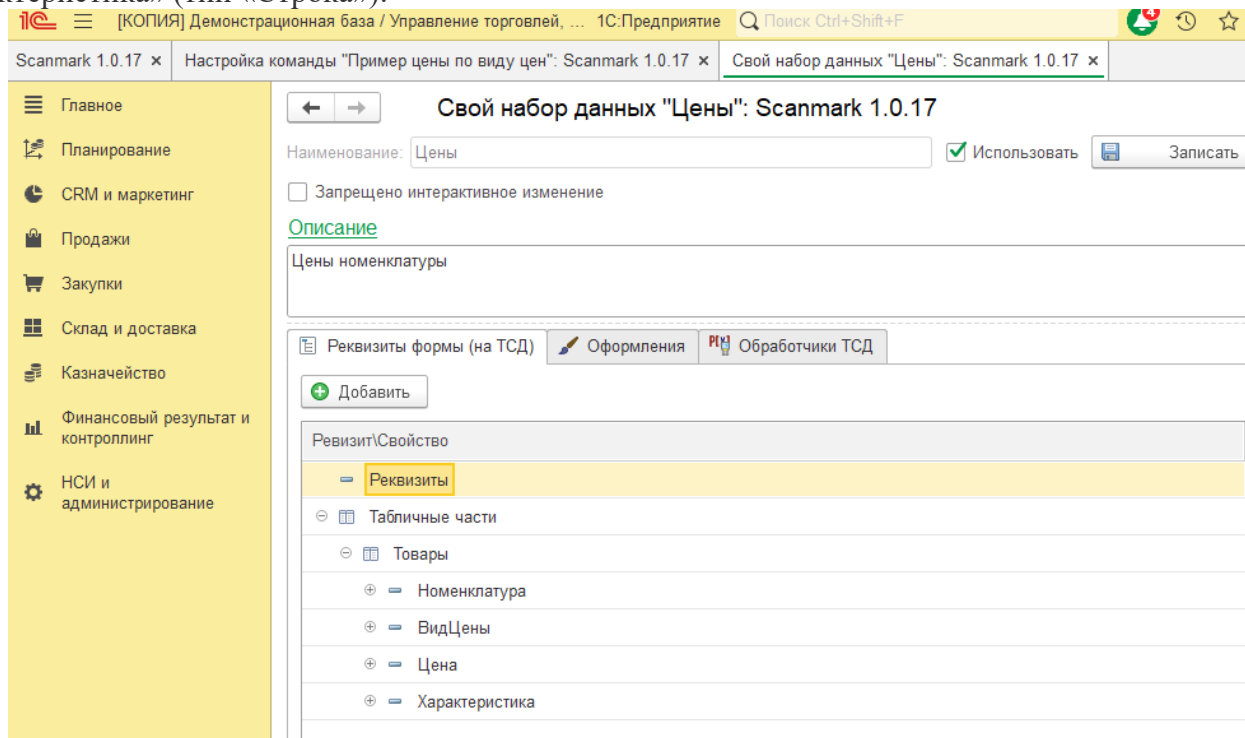
Проверим наличие настройки справочника «Виды цен», при отсутствии добавляем настройку на закладке «Справочники»:



На закладке «Свои наборы данных» создадим настройку «Цены»:

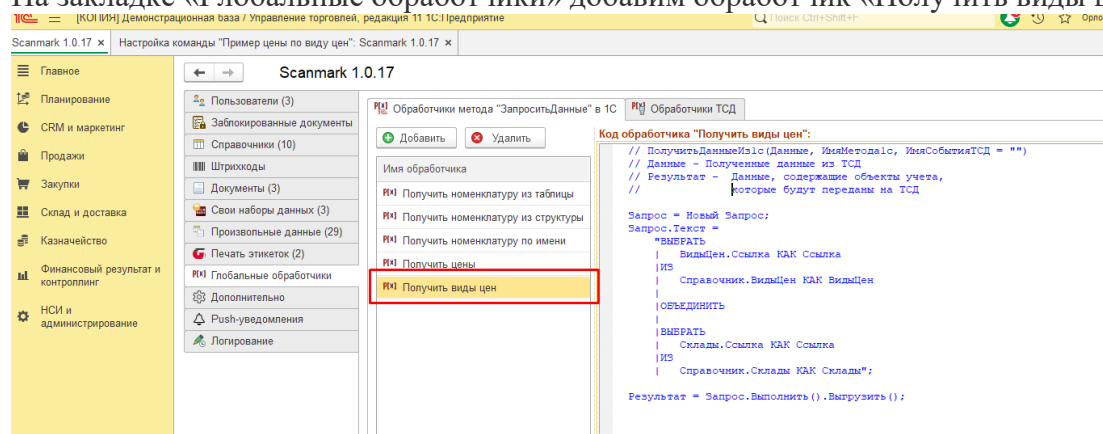


В настройке «Цены» создадим табличную часть с реквизитами «Номенклатура» (Справочник «Номенклатура»), «ВидЦены» (Справочник «Виды цен»), «Цена» (тип «Число») и «Характеристика» (тип «Строка»):



Сохраним и пока закроем настройку «Цены».

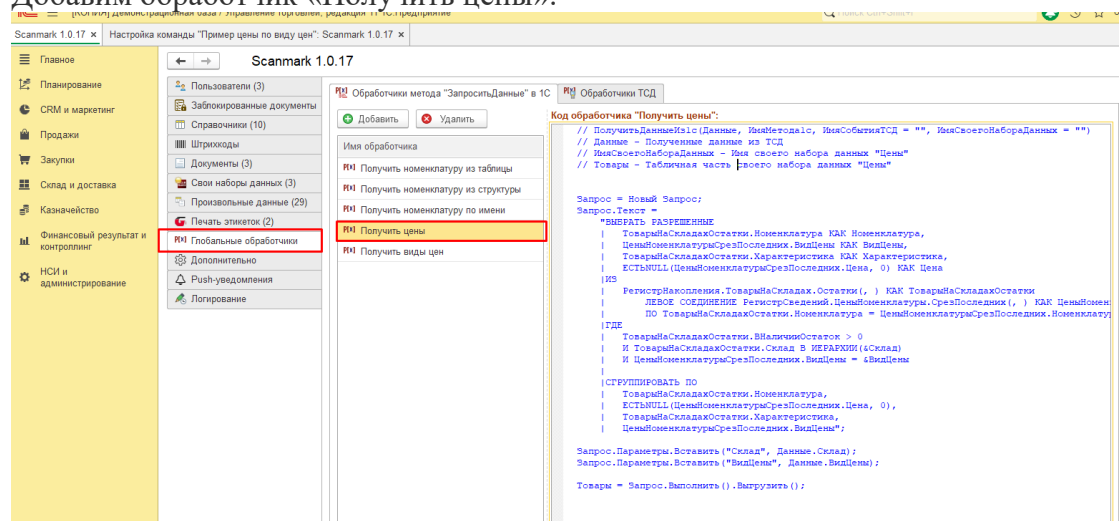
На закладке «Глобальные обработчики» добавим обработчик «Получить виды цен»:



```
Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
| ВидыЦен.Ссылка КАК Ссылка
| ИЗ
| Справочник.ВидыЦен КАК ВидыЦен
|
| ОБЪЕДИНИТЬ
|
| ВЫБРАТЬ
| Склады.Ссылка КАК Ссылка
| ИЗ
| Справочник.Склады КАК Склады";
```

```
Результат = Запрос.Выполнить().Выгрузить();
```

Добавим обработчик «Получить цены»:



```
Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ РАЗРЕШЕННЫЕ
| ТоварыНаСкладахОстатки.Номенклатура КАК Номенклатура,
| ЦеныНоменклатурыСрезПоследних.ВидЦены КАК ВидЦены,
| ТоварыНаСкладахОстатки.Характеристика КАК Характеристика,
| ЕСТЬNULL(ЦеныНоменклатурыСрезПоследних.Цена, 0) КАК Цена
| ИЗ
| РегистрНакопления.ТоварыНаСкладах.Остатки(, ) КАК ТоварыНаСкладахОстатки
| ЛЕВОЕ СОЕДИНЕНИЕ РегистрСведений.ЦеныНоменклатуры.СрезПоследних(, ) КАК
ЦеныНоменклатурыСрезПоследних
| ПО
ТоварыНаСкладахОстатки.Номенклатура
ЦеныНоменклатурыСрезПоследних.Номенклатура
| ГДЕ
```

```

| ТоварыНаСкладахОстатки.ВНаличииОстаток > 0
| И ТоварыНаСкладахОстатки.Склад В ИЕРАРХИИ(&Склад)
| И ЦеныНоменклатурыСрезПоследних.ВидЦены = &ВидЦены
|
| СГРУППИРОВАТЬ ПО
| ТоварыНаСкладахОстатки.Номенклатура,
| ЕСТЬNULL(ЦеныНоменклатурыСрезПоследних.Цена, 0),
| ТоварыНаСкладахОстатки.Характеристика,
| ЦеныНоменклатурыСрезПоследних.ВидЦены";

```

```

Запрос.Параметры.Вставить("Склад", Данные.Склад);
Запрос.Параметры.Вставить("ВидЦены", Данные.ВидЦены);

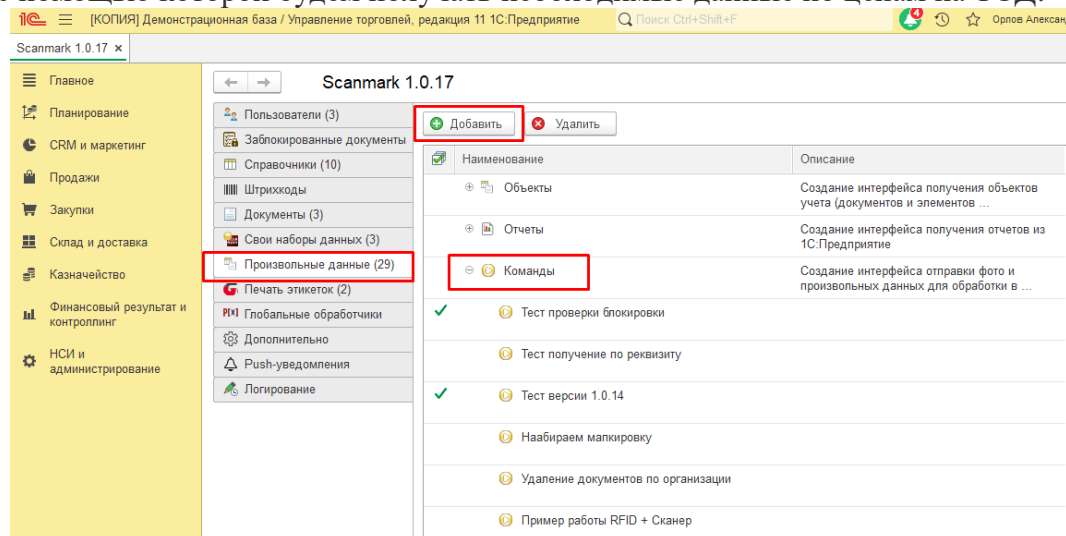
```

```

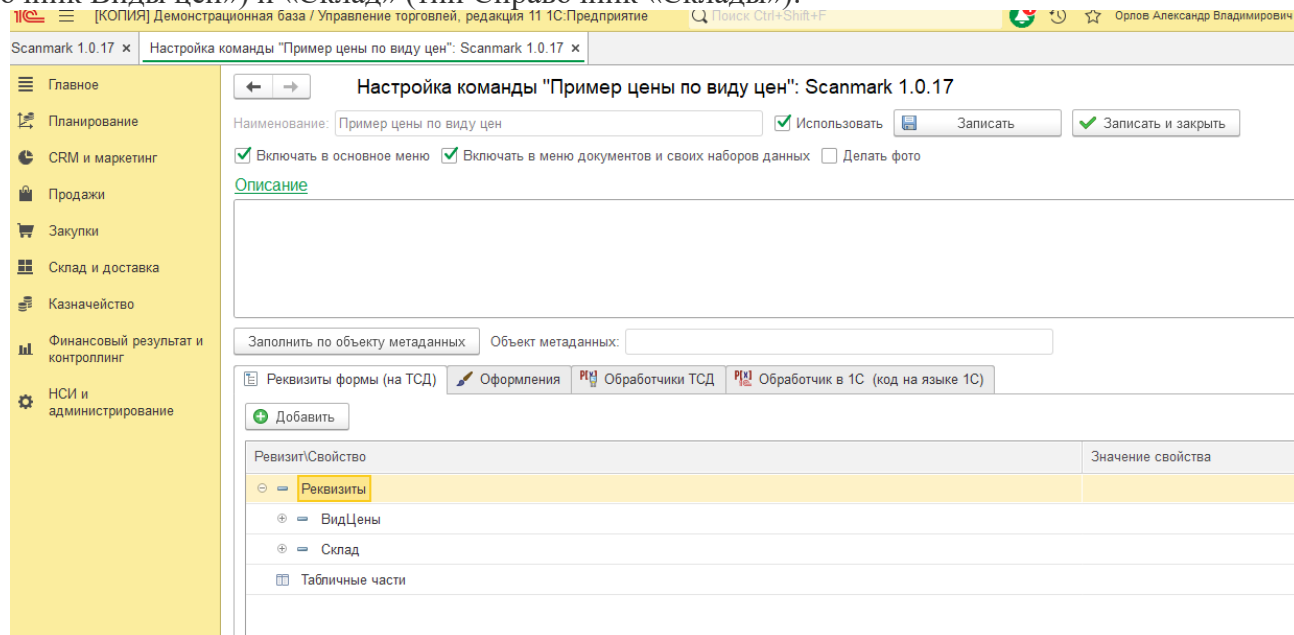
Товары = Запрос.Выполнить().Выгрузить();

```

На закладке «Произвольные данные» создадим новую команду «Пример цены по виду цен», с помощью которой будем получать необходимые данные по ценам на ТСД:



На закладке «Реквизиты формы (на ТСД)» добавим реквизиты «ВидЦены» (тип Справочник Виды цен») и «Склад» (тип Справочник «Склады»):



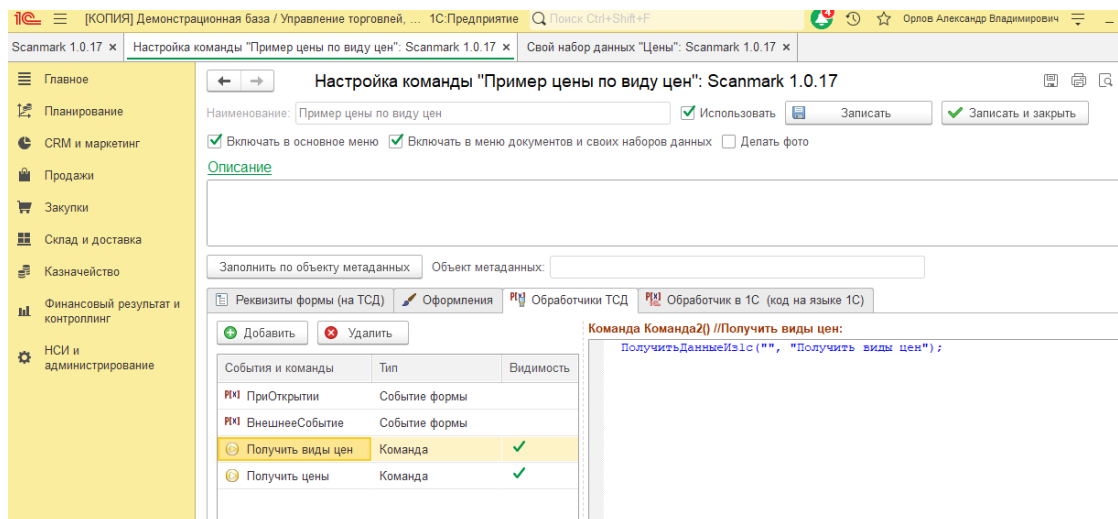
На закладке «Обработчики ТСД» добавим две команды: «Получить виды цен» с кодом обработчика

```

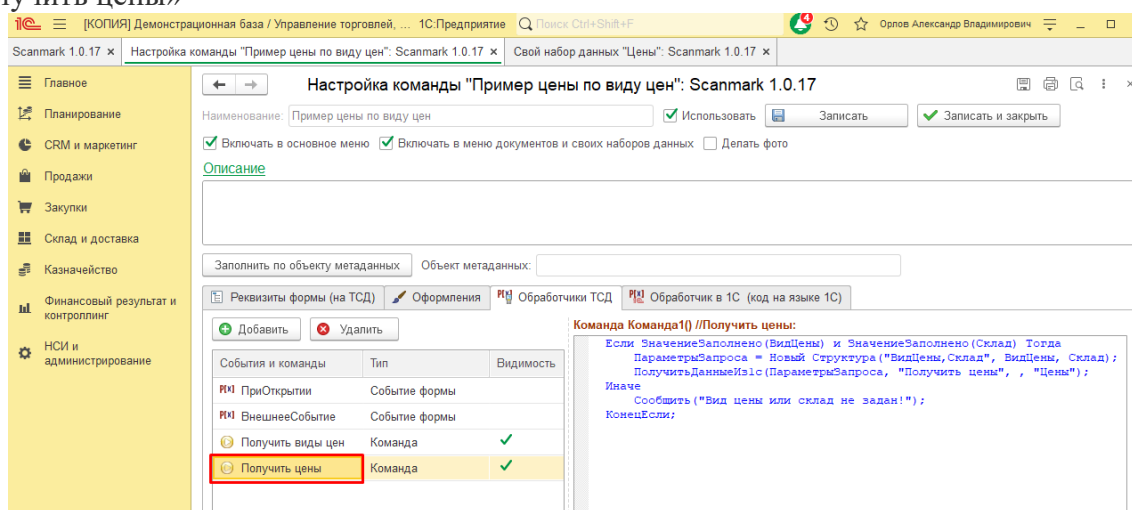
ПолучитьДанныеИз1с("", "Получить виды цен");

```





и «Получить цены»

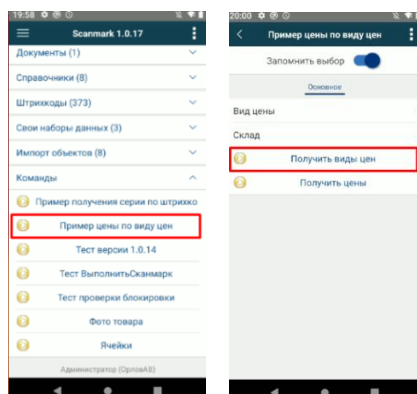


с кодом

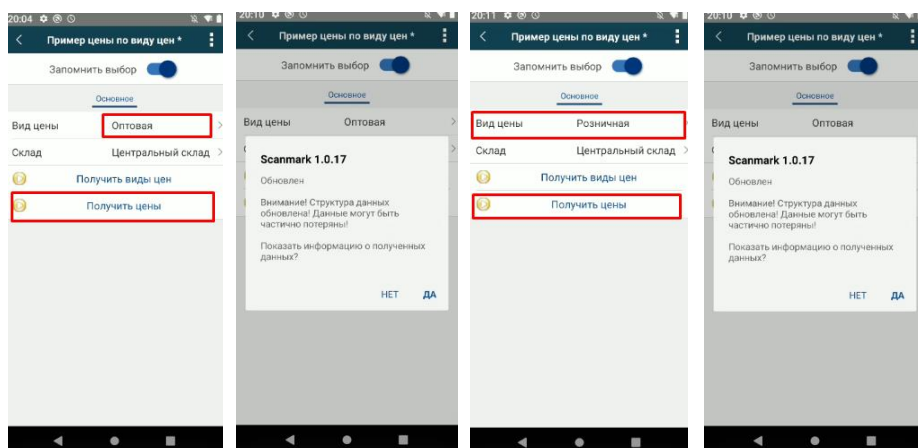
Если ЗначениеЗаполнено(ВидЦены) и ЗначениеЗаполнено(Склад) Тогда  
ПараметрыЗапроса = Новый Структура("ВидЦены,Склад", ВидЦены, Склад);  
ПолучитьДанныеИз1с(ПараметрыЗапроса, "Получить цены", , "Цены");  
Иначе  
Сообщить("Вид цены или склад не задан!");  
КонецЕсли;

С помощью данной настройки мы сможем получить элементы справочников «Виды цен» и «Склады» (Получить виды цен), а также данные по ценам номенклатуры (Свой набор данных «Цены»). Запишем и закроем команду.

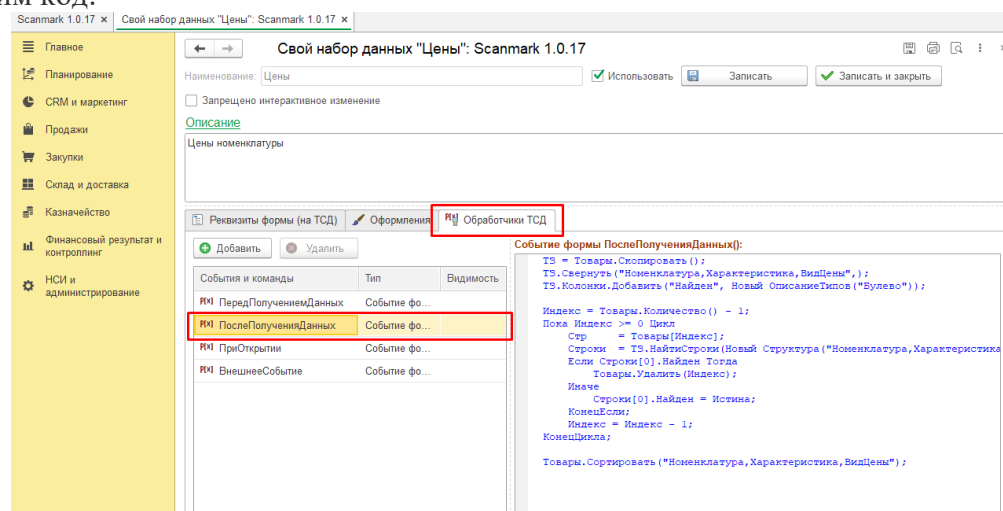
Теперь немного пояснения того, какой функционал уже написан, и как это работает. Из команды произвольных данных «Пример цены по виду цен» на ТСД мы можем получить элементы справочников «Виды цен» и «Склады» (кнопка «Получить виды цен»):



Далее, нужно выбрать вид цены, склад и получить данные своего набора «Цены» (кнопка «Получить цены»). В нашем случае поочередно выберем и получим данные по видам цены Оптовая и Розничная:



При получении данных на ТСД по типу «Свой набор данных» приложение обрабатывает полученные данные следующим образом: Данные из реквизитов заменяют текущие данные заданного набора, данные же табличных частей не заменяют уже имеющиеся данные, а добавляют новые записи в соответствующие табличные части. При настройке пользователь может обработать данные объекта «Свой набор данных» в методах «ПередПолучениемДанных» и «ПослеПолученияДанных» настройки такого набора. В нашем случае при повторном получении данных цен номенклатуры нам нужно исключить повтор строк с совпадением номенклатуры, характеристики и вида цен. Для этого в методе «ПослеПолученияДанных» добавим код:

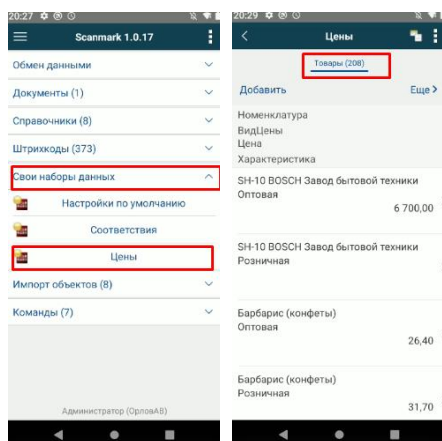


```
T3 = Товары.Скопировать();
T3.Свернуть ("Номенклатура, Характеристика, ВидЦены", );
T3.Колонки.Добавить ("Найден", Новый ОписаниеТипов ("Булево"));

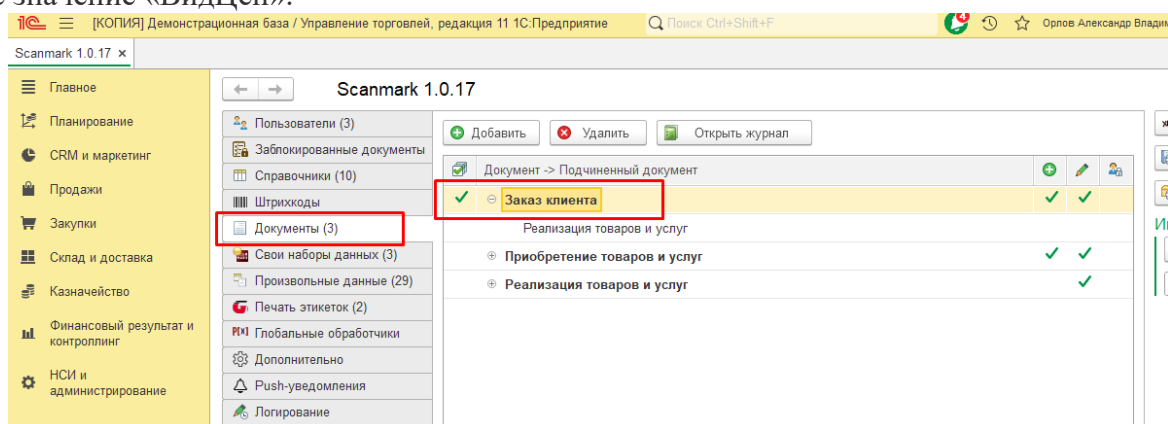
Индекс = Товары.Количество() - 1;
Пока Индекс >= 0 Цикл
    Стр = Товары[Индекс];
    Строки = T3.НайтиСтроки (Новый Структура ("Номенклатура, Характеристика, ВидЦены",
Стр.Номенклатура, Стр.Характеристика, Стр.ВидЦены));
    Если Строки[0].Найден Тогда
        Товары.Удалить (Индекс);
    Иначе
        Строки[0].Найден = Истина;
    КонецЕсли;
    Индекс = Индекс - 1;
КонецЦикла;

Товары.Сортировать ("Номенклатура, Характеристика, ВидЦены");
```

Проверим наличие данных «Цены» в «Свои наборы данных»:



Теперь осталось доработать настройку документа «Заказ клиента». Добавим в запрос пустое значение «ВидЦен»:



Для этого в текст запроса добавим строку

Значение (Справочник.ВидыЦен.ПустаяССылка) КАК ВидЦены,  
а в свойство реквизита «ВидЦены» «ПриИзменении» добавим код:

Если ЗначениеЗаполнено (ВидЦены) Тогда

ДанныеОтбора = Новый Структура ();

ДанныеОтбора.Вставить ("ВидЦены", ВидЦены);

ТЗ\_Цены = СвойНаборДанных ("Цены", "Товары", ДанныеОтбора);

Для Каждого Стр Из Товары Цикл

Строки = ТЗ\_Цены.НайтиСтроки (Новый Структура ("ВидЦены,Номенклатура",  
ВидЦены, Стр.Номенклатура));

Если Строки.Количество() = 0 Тогда

Продолжить;

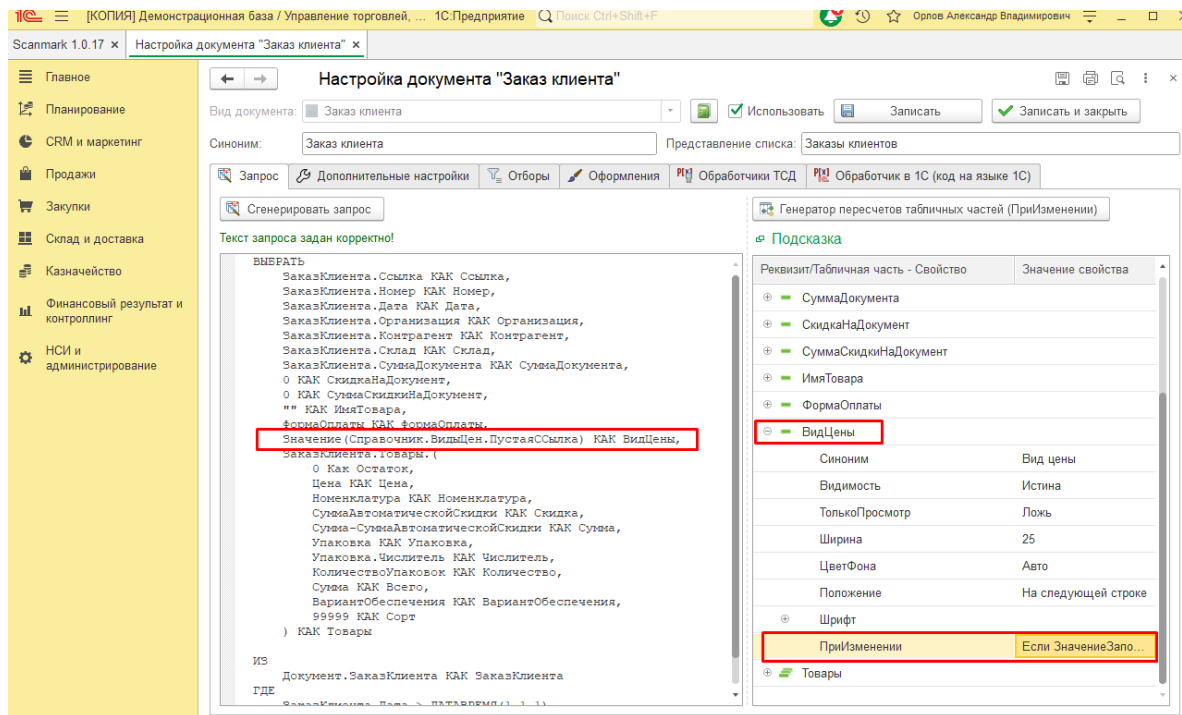
КонецЕсли;

Стр.Цена = Строки[0].Цена;

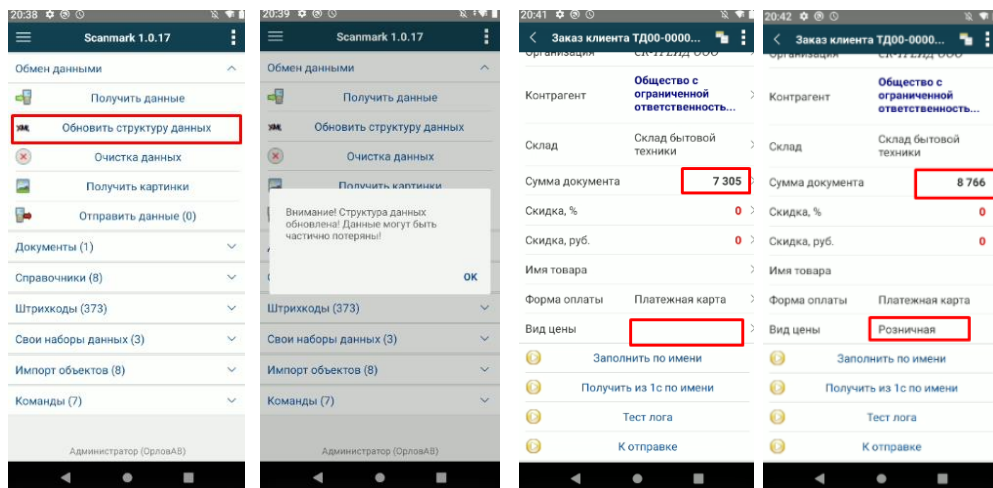
Стр.Сумма = Стр.Цена \* Стр.Количество;

КонецЦикла;

СуммаДокумента = Товары.Итог ("Сумма");  
КонецЕсли;

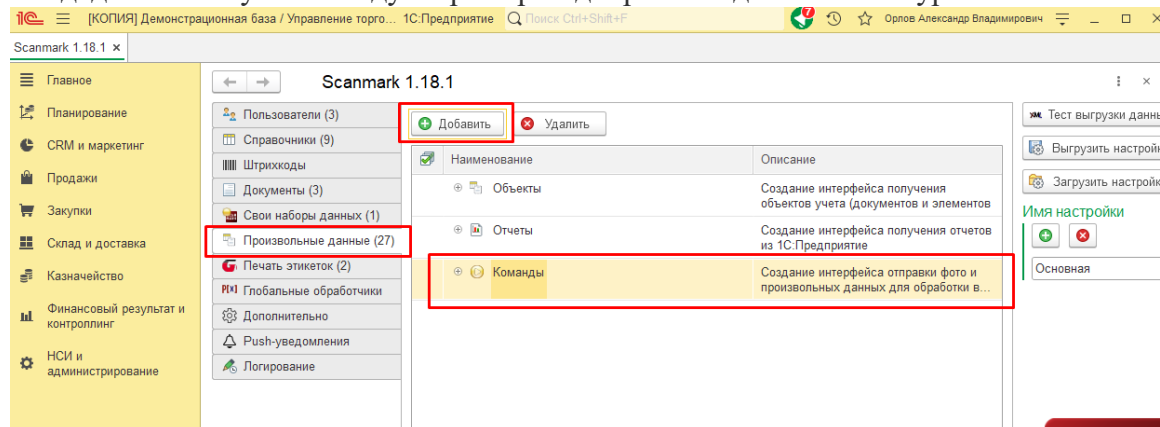


Сохраним и закроем настройку документа. Обновим структуру данных и проверим работу пересчета в приложении на ТСД выбирая в документе «Заказ клиента» вид цены:



## Задача 22: Пример подбора номенклатуры по штрихкоду с вводом количества.

Создадим тестовую команду «Пример подбора и ввода номенклатуры».

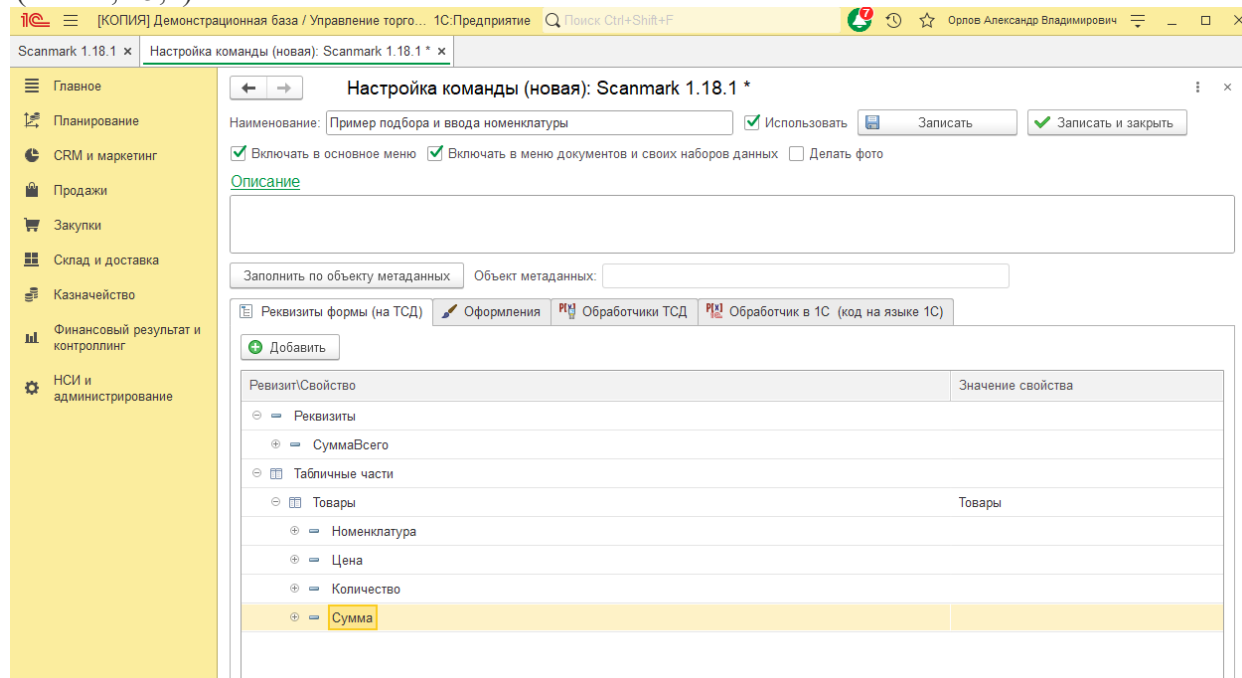


Добавим реквизит «СуммаВсего» (Число, 15, 2).

Добавим Табличную часть «Товары». В Табличной части создадим реквизиты:

Номенклатура (Справочник.Номенклатура), Цена (Число, 15,2), Количество (Число,10,0),

Сумма (Число, 15,2).



На закладке «Обработчики ТСД» для события «ВнешнееСобытие» пропишем код:

```
Если Событие = "Barcode" Тогда
    ДанныеШтрихкода = ДанныеШтрихкода(Данные, Истина);
    Если ДанныеШтрихкода.Количество() > 0 Тогда
        ДанныеШтрихкода = ДанныеШтрихкода[0];
        Строки = Товары.НайтиСтроки(Новый Структура("Номенклатура", ДанныеШтрихкода.Номенклатура));

        Если Строки.Количество() = 0 Тогда
            Количество = 0;
        Иначе
            Количество = Строки[0].Количество;
        КонецЕсли;

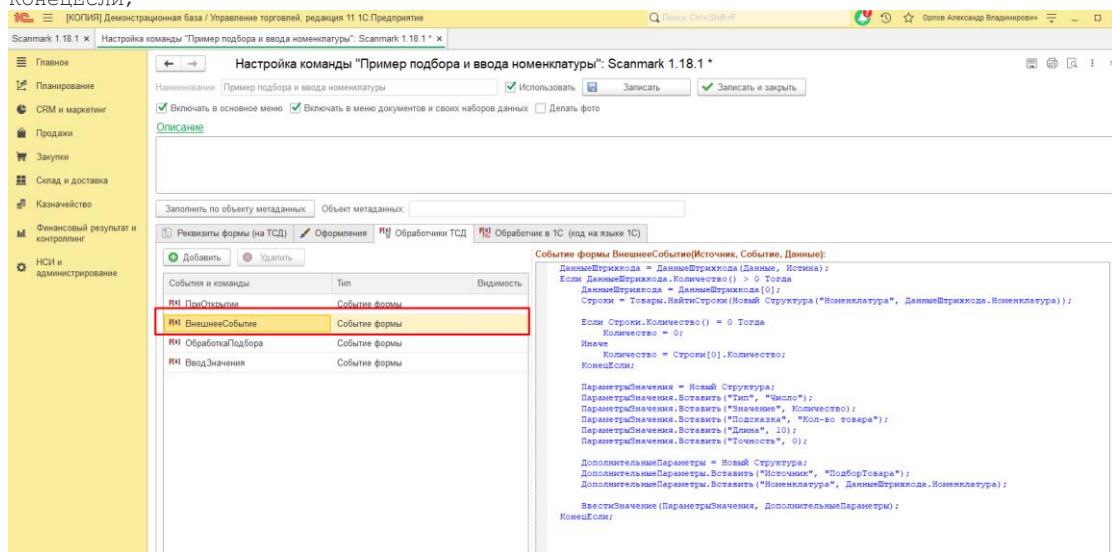
        ПараметрыЗначения = Новый Структура;
        ПараметрыЗначения.Вставить("Тип", "Число");
        ПараметрыЗначения.Вставить("Значение", Количество);
        ПараметрыЗначения.Вставить("Подсказка", "Кол-во товара");
        ПараметрыЗначения.Вставить("Длина", 10);
        ПараметрыЗначения.Вставить("Точность", 0);

        ДополнительныеПараметры = Новый Структура;
```

ДополнительныеПараметры.Вставить ("Источник", "ПодборТовара");  
ДополнительныеПараметры.Вставить ("Номенклатура", ДанныеШтрихкода.Номенклатура);

ВвестиЗначение (ПараметрыЗначения, ДополнительныеПараметры);  
КонецЕсли;

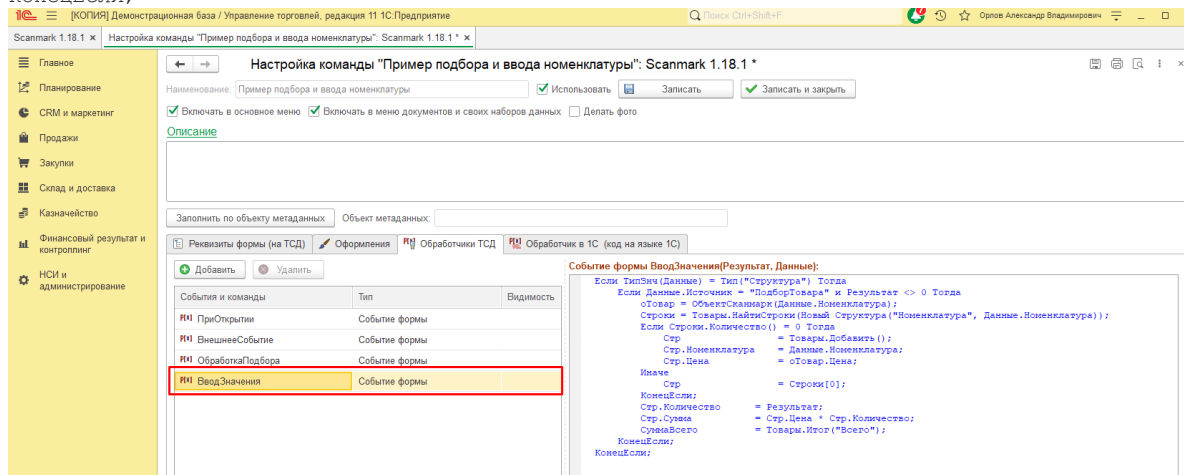
КонецЕсли;



В событии «ВводЗначения» пропишем код:

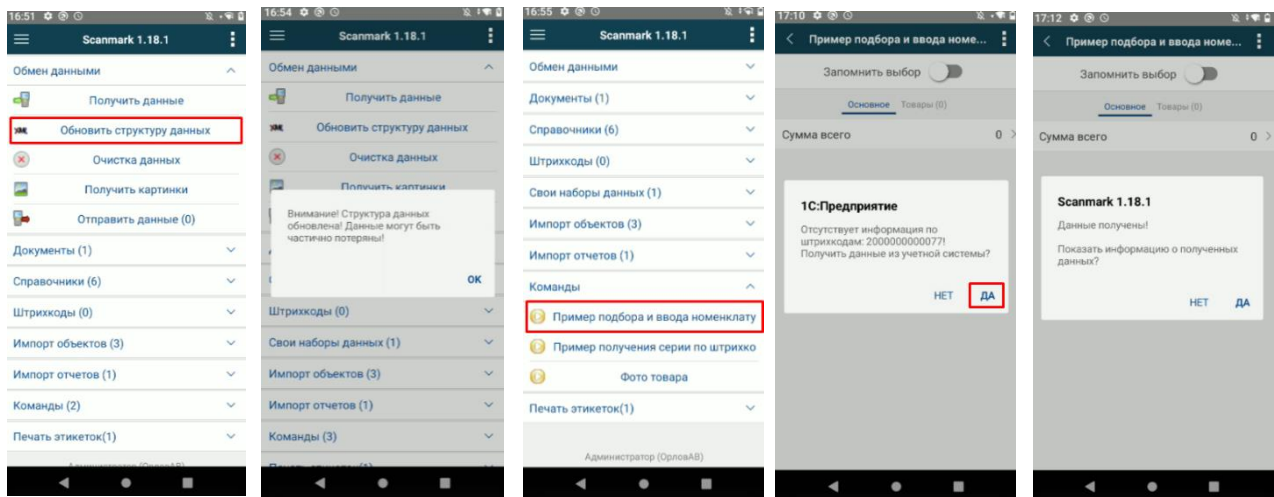
```
Если ТипЗнч(Данные) = Тип("Структура") Тогда  
    Если Данные.Источник = "ПодборТовара" и Результат <> 0 Тогда  
        оТовар = ОбъектСканмарк(Данные.Номенклатура);  
        Строки = Товары.НайтиСтроки(Новый Структура("Номенклатура", Данные.Номенклатура));  
        Если Строки.Количество() = 0 Тогда  
            Стр = Товары.Добавить();  
            Стр.Номенклатура = Данные.Номенклатура;  
            Стр.Цена = оТовар.Цена;  
        Иначе  
            Стр = Строки[0];  
        КонецЕсли;  
        Стр.Количество = Результат;  
        Стр.Сумма = Стр.Цена * Стр.Количество;  
        СуммаВсего = Товары.Итог("Сумма");  
    КонецЕсли;
```

КонецЕсли;

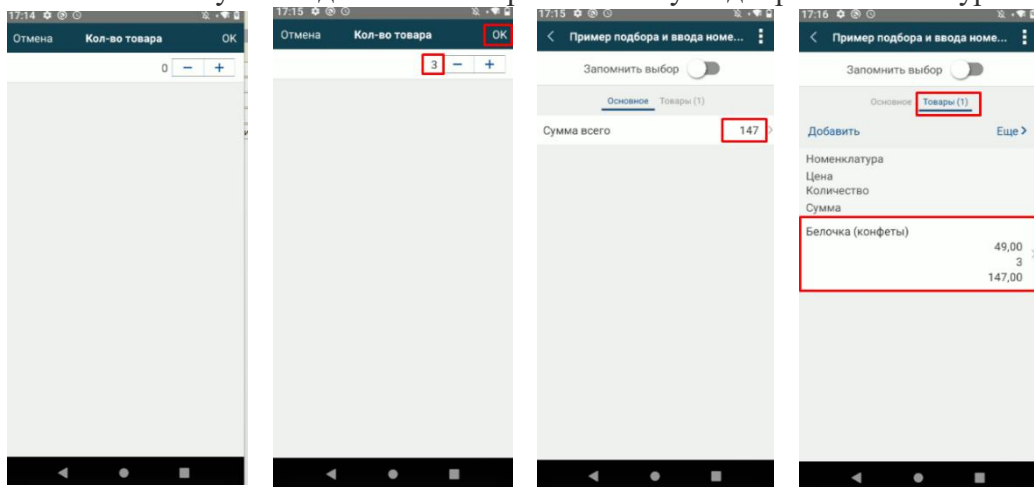


Сохраним настройку.

На ТСД обновим структуру и протестируем функционал:



После получения данных повторим попытку подбора номенклатуры по штрихкоду:



Результат достигнут.



## Задача 23: Пример подбора номенклатуры из списка справочника с вводом количества.

В команду «Пример подбора и ввода номенклатуры» (см. предыдущую задачу «Пример подбора номенклатуры по штрихкоду с вводом количества») добавим команду «Подбор» и пропишем код в обработчике:

События и команды	Тип	Видимость
ПриОткрытии	Событие формы	
ВнешнееСобытие	Событие формы	
ОбработкаПодбора	Событие формы	
ВводЗначения	Событие формы	
Подбор	Команда	✓

```
Команда Команда1() /Подбор:
ПараметрыПодбора = Новый Структура;
ПараметрыПодбора.Вставить("Объект", "Справочник.Номенклатура");
ПараметрыПодбора.Вставить("ВыбиратьЭлементы", Истина);
ПараметрыПодбора.Вставить("ВыбиратьГруппы", Ложь);
ПараметрыПодбора.Вставить("МножественныйВыбор", Истина);
ПараметрыПодбора.Вставить("Источник", "ПодборТовара");
ОткрытьПодбор(ПараметрыПодбора);
```

```
ПараметрыПодбора = Новый Структура;
ПараметрыПодбора.Вставить("Объект", "Справочник.Номенклатура");
ПараметрыПодбора.Вставить("ВыбиратьЭлементы", Истина);
ПараметрыПодбора.Вставить("ВыбиратьГруппы", Ложь);
ПараметрыПодбора.Вставить("МножественныйВыбор", Истина);
ПараметрыПодбора.Вставить("Источник", "ПодборТовара");
ОткрытьПодбор(ПараметрыПодбора);
```

В событии «ОбработкаПодбора» пропишем код:

События и команды	Тип	Видимость
ПриОткрытии	Событие формы	
ВнешнееСобытие	Событие формы	
ОбработкаПодбора	Событие формы	
ВводЗначения	Событие формы	
Подбор	Команда	✓

```
Событие формы ОбработкаПодбора(Источник.Данные):
Если Источник = "ПодборТовара" Тогда
    Строки = Товары.НайтиСтроки(Новый Структура("Номенклатура", Данные));
    Если Строки.Количество() = 0 Тогда
        Количество = 0;
    Иначе
        Количество = Строки[0].Количество;
    КонецЕсли;
    ПараметрыЗначения = Новый Структура;
    ПараметрыЗначения.Вставить("Тип", "Число");
    ПараметрыЗначения.Вставить("Значение", Количество);
    ПараметрыЗначения.Вставить("Подсказка", "Кол-во товара");
    ПараметрыЗначения.Вставить("Длина", 10);
    ПараметрыЗначения.Вставить("Точность", 0);
    ДополнительныеПараметры = Новый Структура;
    ДополнительныеПараметры.Вставить("Источник", "ПодборТовара");
    ДополнительныеПараметры.Вставить("Номенклатура", Данные);
    ВвестиЗначение(ПараметрыЗначения, ДополнительныеПараметры);
КонецЕсли;
```

```
Если Источник = "ПодборТовара" Тогда
    Строки = Товары.НайтиСтроки(Новый Структура("Номенклатура", Данные));
    Если Строки.Количество() = 0 Тогда
        Количество = 0;
    Иначе
        Количество = Строки[0].Количество;
    КонецЕсли;
    ПараметрыЗначения = Новый Структура;
    ПараметрыЗначения.Вставить("Тип", "Число");
    ПараметрыЗначения.Вставить("Значение", Количество);
    ПараметрыЗначения.Вставить("Подсказка", "Кол-во товара");
    ПараметрыЗначения.Вставить("Длина", 10);
    ПараметрыЗначения.Вставить("Точность", 0);
    ДополнительныеПараметры = Новый Структура;
```

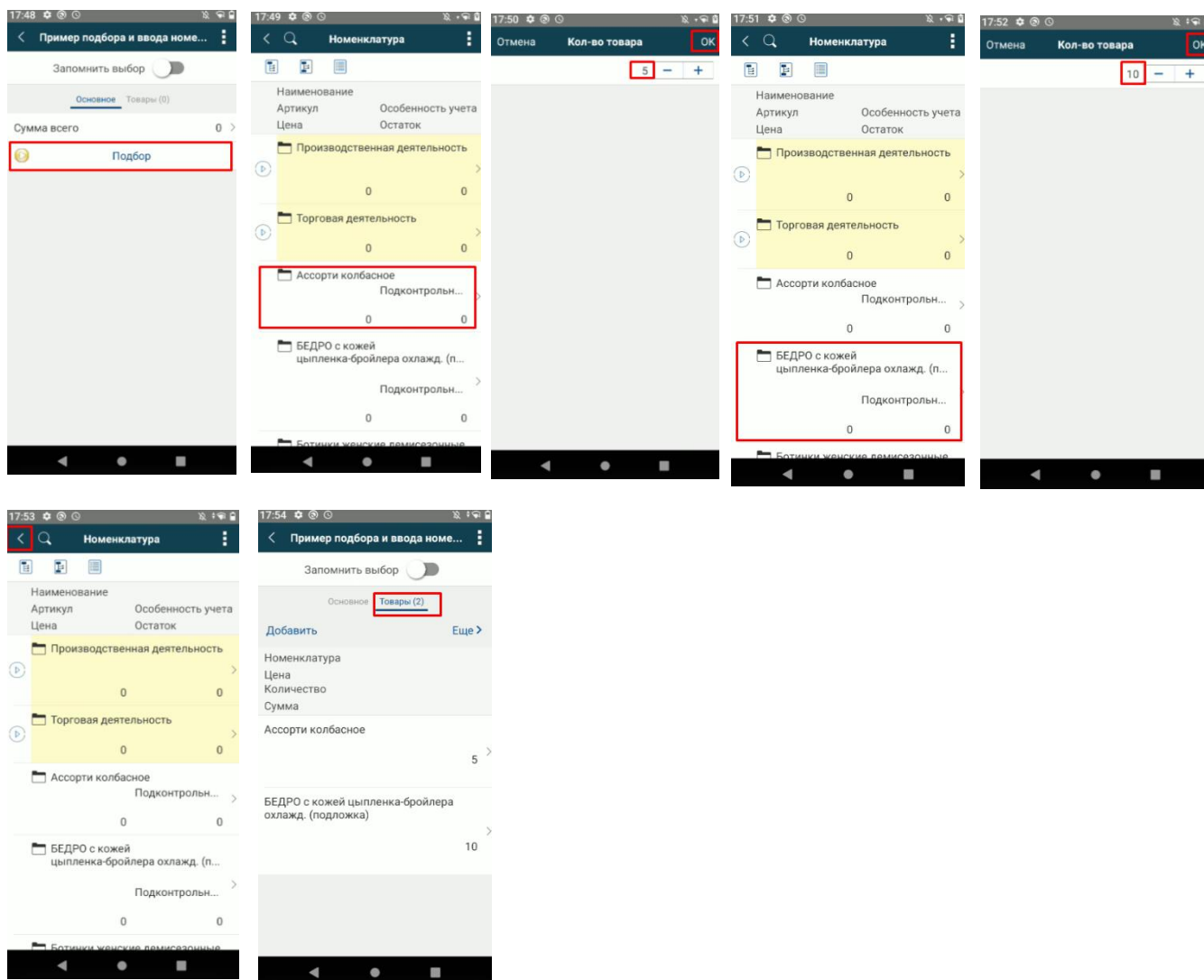
```
ДополнительныеПараметры = Новый Структура;
```



ДополнительныеПараметры.Вставить ("Источник", "ПодборТовара");  
ДополнительныеПараметры.Вставить ("Номенклатура", Данные);

ВвестиЗначение (ПараметрыЗначения, ДополнительныеПараметры);  
КонецЕсли;

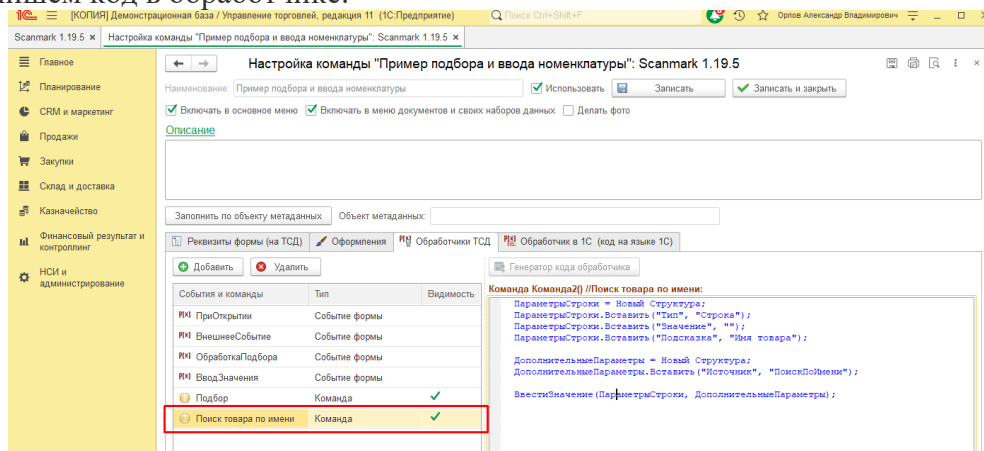
Сохраним настройку. На ТСД обновим структуру данных и протестируем новый функционал:



Результат достигнут.

## Задача 24: Пример поиска номенклатуры по наименованию с вводом количества и возможностью запроса номенклатуры с сервера.

В команду «Пример подбора и ввода номенклатуры» (см. задачу «Пример подбора номенклатуры по штрихкоду с вводом количества») добавим команду «Поиск товара по имени» и пропишем код в обработчике:



```
ПараметрыСтроки = Новый Структура;  
ПараметрыСтроки.Вставить ("Тип", "Строка");  
ПараметрыСтроки.Вставить ("Значение", "");  
ПараметрыСтроки.Вставить ("Подсказка", "Имя товара");
```

```
ДополнительныеПараметры = Новый Структура;  
ДополнительныеПараметры.Вставить ("Источник", "ПоискПоИмени");
```

```
ВвестиЗначение (ПараметрыСтроки, ДополнительныеПараметры);
```

### В событии «ВводЗначения» изменим код на:

```
Если ТипЗнч (Данные) = Тип ("Структура") Тогда  
Если Данные.Источник = "ПодборТовара" и Результат <> 0 Тогда  
оТовар = ОбъектСканмарк (Данные.Номенклатура);  
Строки = Товары.НайтиСтроки (Новый Структура ("Номенклатура", Данные.Номенклатура));  
Если Строки.Количество () = 0 Тогда  
Стр = Товары.Добавить ();  
Стр.Номенклатура = Данные.Номенклатура;  
Стр.Цена = оТовар.Цена;  
Иначе  
Стр = Строки [0];  
КонецЕсли;  
Стр.Количество = Результат;  
Стр.Сумма = Стр.Цена * Стр.Количество;  
СуммаВсего = Товары.Итог ("Сумма");  
  
ИначеЕсли Данные.Источник = "ПоискПоИмени" Тогда  
ИмяТовара = Результат;  
Если СтрДлина (ИмяТовара) >= 3 Тогда  
Список = Новый СписокЗначений;  
Список.Добавить ("< НАЙТИ НА СЕРВЕРЕ >", "< НАЙТИ НА СЕРВЕРЕ >");  
СпрТовары = НайтиПоНаименованию ("Номенклатура", "%" + ИмяТовара + "%");  
Для Каждого Товар из СпрТовары Цикл  
оТовар = ОбъектСканмарк (Товар);  
Список.Добавить (Товар, оТовар.Наименование + " (" + оТовар.Остаток + ")");  
КонецЦикла;  
  
Если Список.Количество () = 1 Тогда  
ПараметрыВопроса = Новый Структура;  
ПараметрыВопроса.Вставить ("Тип", "Булево");  
ПараметрыВопроса.Вставить ("Подсказка", "Ничего не найдено!" + Символы.ПС + "Найти на сервере?");  
  
ДополнительныеПараметры = Новый Структура;  
ДополнительныеПараметры.Вставить ("Источник", "Найти на сервере");  
ДополнительныеПараметры.Вставить ("ИмяТовара", ИмяТовара);  
  
ВвестиЗначение (ПараметрыВопроса, ДополнительныеПараметры);  
Иначе
```

```

ПараметрыСписок = Новый Структура;
ПараметрыСписок.Вставить ("Тип", "СписокЗначений");
ПараметрыСписок.Вставить ("Значение", Список);

ДополнительныеПараметры = Новый Структура;
ДополнительныеПараметры.Вставить ("Источник", "ВыбратьТоварИзСписка");
ДополнительныеПараметры.Вставить ("ИмяТовара", ИмяТовара);

ВвестиЗначение (ПараметрыСписок, ДополнительныеПараметры);
КонецЕсли;
КонецЕсли;

ИначеЕсли Данные.Источник = "ВыбратьТоварИзСписка" Тогда
    Если Результат = "< НАЙТИ НА СЕРВЕРЕ >" Тогда
        ИмяМетода1с = "Получить номенклатуру по имени";
        ИмяСобытияТСД = "";
        ПолучитьДанныеИз1с (Данные.ИмяТовара, ИмяМетода1с, ИмяСобытияТСД);
    Иначе
        Строки = Товары.НайтиСтроки (Новый Структура ("Номенклатура", Результат));
        Если Строки.Количество() = 0 Тогда
            Количество = 0;
        Иначе
            Количество = Строки[0].Количество;
        КонецЕсли;
        ПараметрыЗначения = Новый Структура;
        ПараметрыЗначения.Вставить ("Тип", "Число");
        ПараметрыЗначения.Вставить ("Значение", Количество);
        ПараметрыЗначения.Вставить ("Подсказка", "Кол-во товара");
        ПараметрыЗначения.Вставить ("Длина", 10);
        ПараметрыЗначения.Вставить ("Точность", 0);

        ДополнительныеПараметры = Новый Структура;
        ДополнительныеПараметры.Вставить ("Источник", "ПодборТовара");
        ДополнительныеПараметры.Вставить ("Номенклатура", Результат);

        ВвестиЗначение (ПараметрыЗначения, ДополнительныеПараметры);
        КонецЕсли;

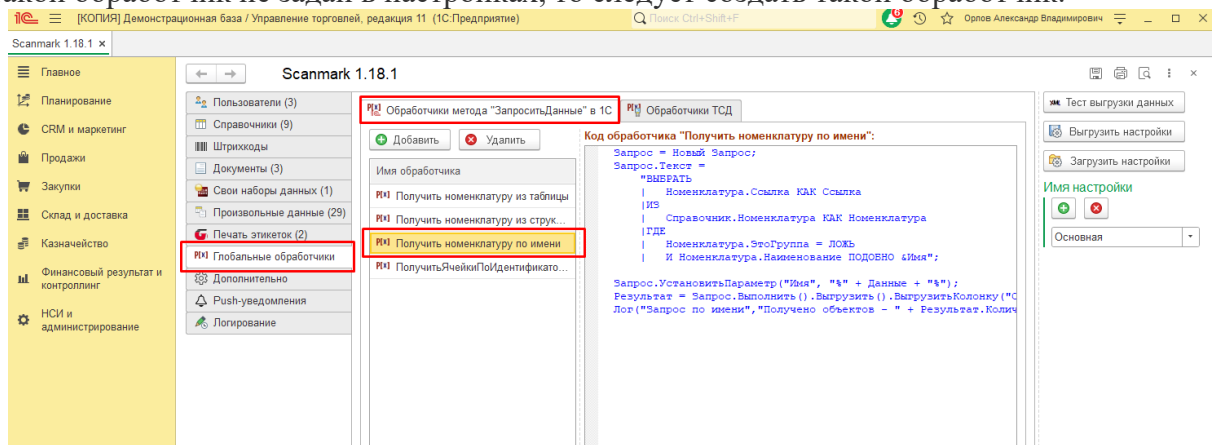
ИначеЕсли Данные.Источник = "Найти на сервере" Тогда
    Если Результат Тогда
        ИмяМетода1с = "Получить номенклатуру по имени";
        ИмяСобытияТСД = "";
        ПолучитьДанныеИз1с (Данные.ИмяТовара, ИмяМетода1с, ИмяСобытияТСД);
    КонецЕсли;

КонецЕсли;

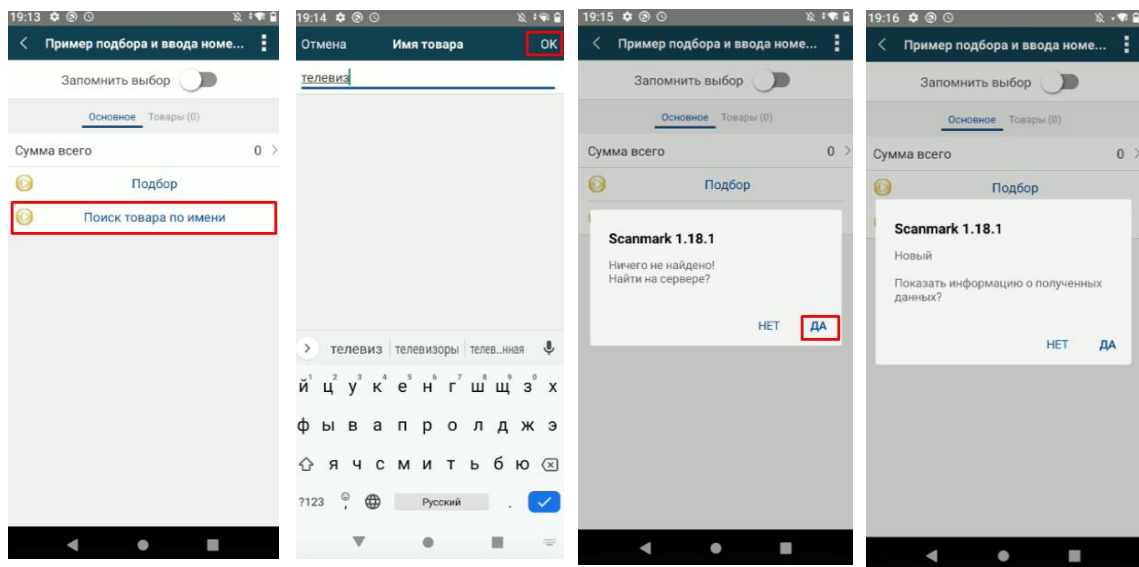
КонецЕсли;

```

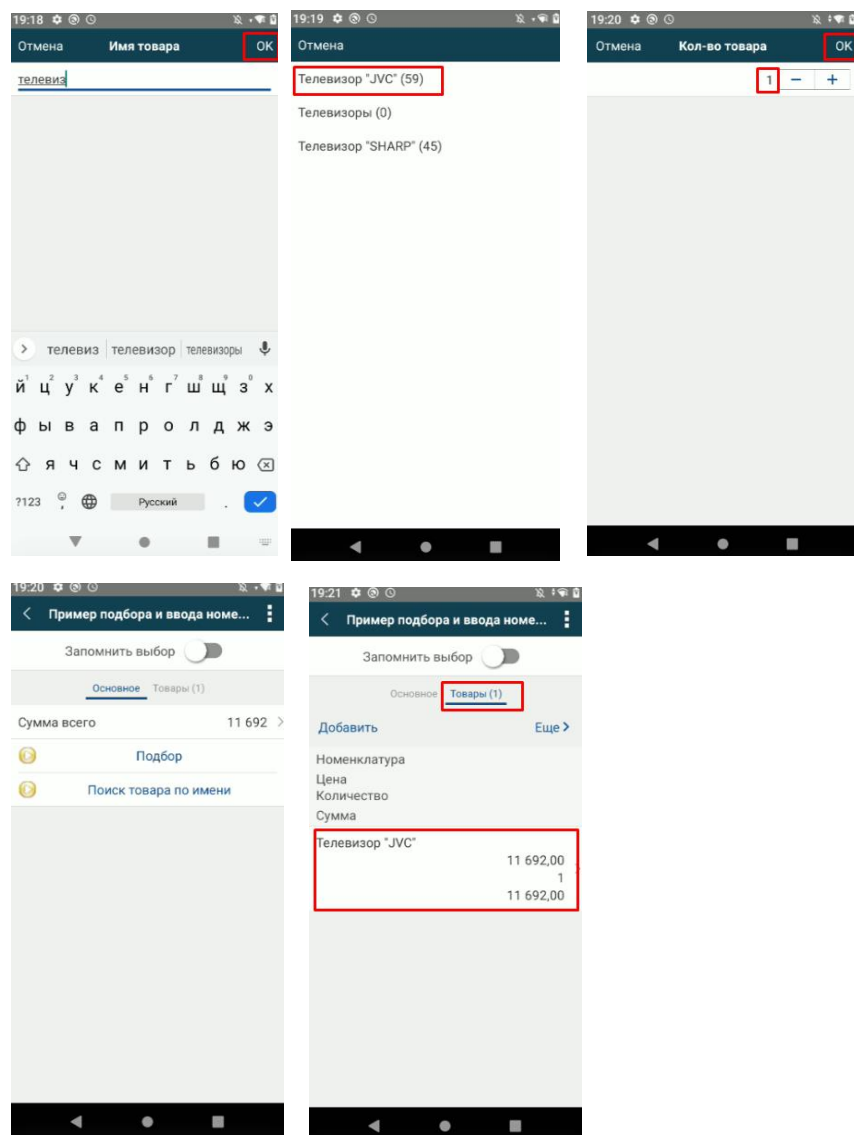
В коде есть обращение к глобальному обработчику "Получить номенклатуру по имени", который был рассмотрен в задаче «Пример использования метода «ПолучитьДанныеИз1с». Если такой обработчик не задан в настройках, то следует создать такой обработчик.



Сохраним настройку. На ТСД обновим структуру данных и проверим новый функционал:



Выполним поиск по строке еще раз:



Результат достигнут.

## Задача 25: Пример настройки нажатия клавиш.

### **ВНИМАНИЕ!!!**

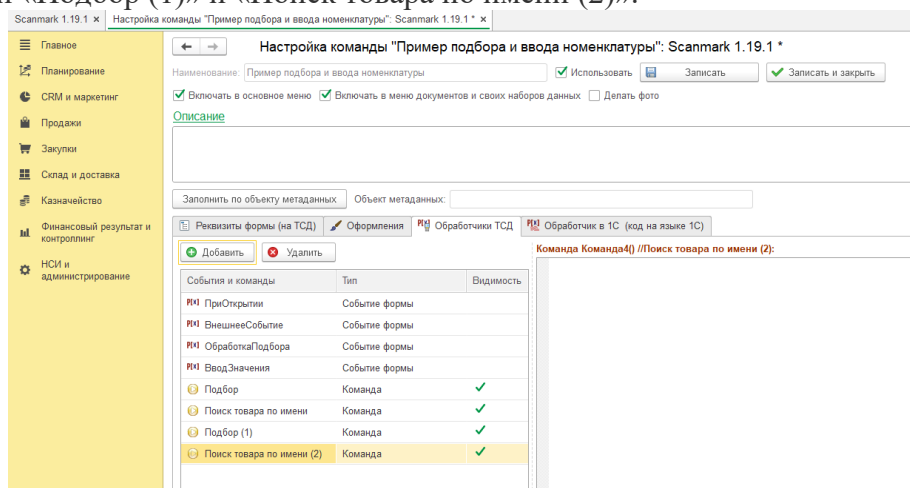
**В версии 1.19.1 появился обработчик перехвата нажатия клавиш. Поэтому, во всех обработчиках «ВнешнееСобытие» необходимо проверить код на наличие проверки параметра «Событие».**

Пример:

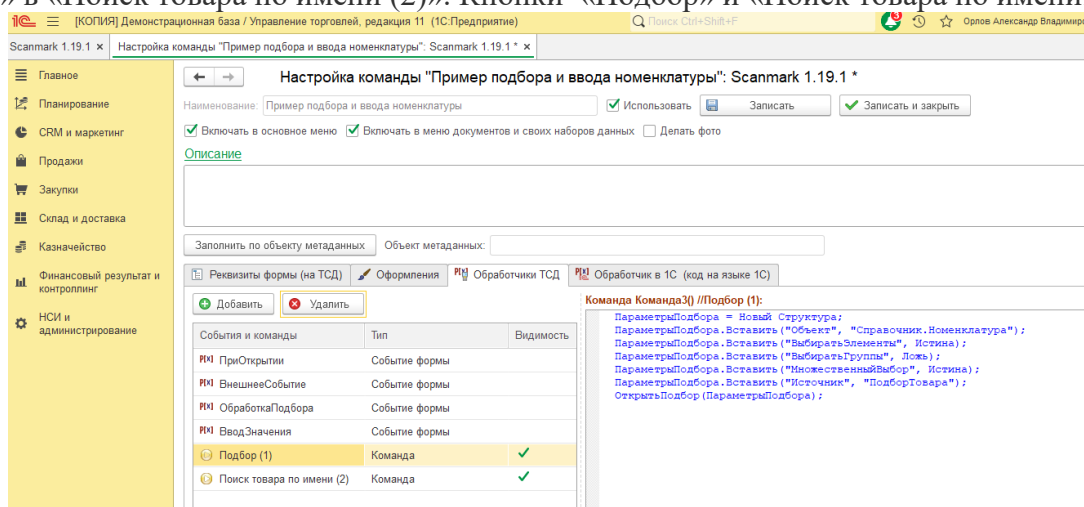
```
Если Событие = "KeyBoard" Тогда  
  < ваш код обработки события нажатия клавиши >  
ИначеЕсли Событие = "Barcode" Тогда  
  < ваш код обработки события сканирования штрихкода >  
ИначеЕсли Событие = "RFID" Тогда  
  < ваш код обработки события считывания радиометок >  
КонецЕсли;
```

В моделях ТСД с клавиатурой часто удобно к нажатию клавиш привязать какие-либо обработчики. Например, выполнение обработчиков или смена закладки страницы.

Для примера реализации настройки нажатия клавиш возьмем команду «Пример подбора и ввода номенклатуры» (см. задачи «Пример подбора номенклатуры по штрихкоду с вводом количества», «Пример поиска номенклатуры по наименованию с вводом количества и возможностью запроса номенклатуры с сервера»). В данной команде создадим две новые кнопки «Подбор (1)» и «Поиск товара по имени (2)»:



Скопируем коды обработчиков: из «Подбор» в «Подбор (1)», из «Поиск товара по имени» в «Поиск товара по имени (2)». Кнопки «Подбор» и «Поиск товара по имени» удалим:



## В обработчике «ВнешнееСобытие» изменим код на:

```
Если Событие = "KeyBoard" Тогда
```

```
    Если Данные = "7" Тогда // клавиша 0
    ИначеЕсли Данные = "8" Тогда // клавиша 1
        ВыполнитьСканмарк("Подбор (1)");
    ИначеЕсли Данные = "9" Тогда // клавиша 2
        ВыполнитьСканмарк("Поиск товара по имени (2)");
    ИначеЕсли Данные = "10" Тогда // клавиша 3
    ИначеЕсли Данные = "11" Тогда // клавиша 4
    ИначеЕсли Данные = "12" Тогда // клавиша 5
    ИначеЕсли Данные = "13" Тогда // клавиша 6
    ИначеЕсли Данные = "14" Тогда // клавиша 7
    ИначеЕсли Данные = "15" Тогда // клавиша 8
    ИначеЕсли Данные = "16" Тогда // клавиша 9
    ИначеЕсли Данные = "19" Тогда // клавиша вверх
    ИначеЕсли Данные = "20" Тогда // клавиша вниз
    ИначеЕсли Данные = "21" Тогда // клавиша влево
        ТекущаяСтраница("Основное");
    ИначеЕсли Данные = "22" Тогда // клавиша вправо
        ТекущаяСтраница("Товары");
    Иначе
        //Сообщить (Данные);
    КонецЕсли;
```

```
ИначеЕсли Событие = "Barcode" Тогда
```

```
    ДанныеШтрихкода = ДанныеШтрихкода(Данные, Истина);
    Если ДанныеШтрихкода.Количество() = 0 Тогда
    Иначе
        ДанныеШтрихкода = ДанныеШтрихкода[0];
        Строки = Товары.НайтиСтроки(Новый Структура("Номенклатура", ДанныеШтрихкода.Номенклатура));

        Если Строки.Количество() = 0 Тогда
            Количество = 0;
        Иначе
            Количество = Строки[0].Количество;
        КонецЕсли;

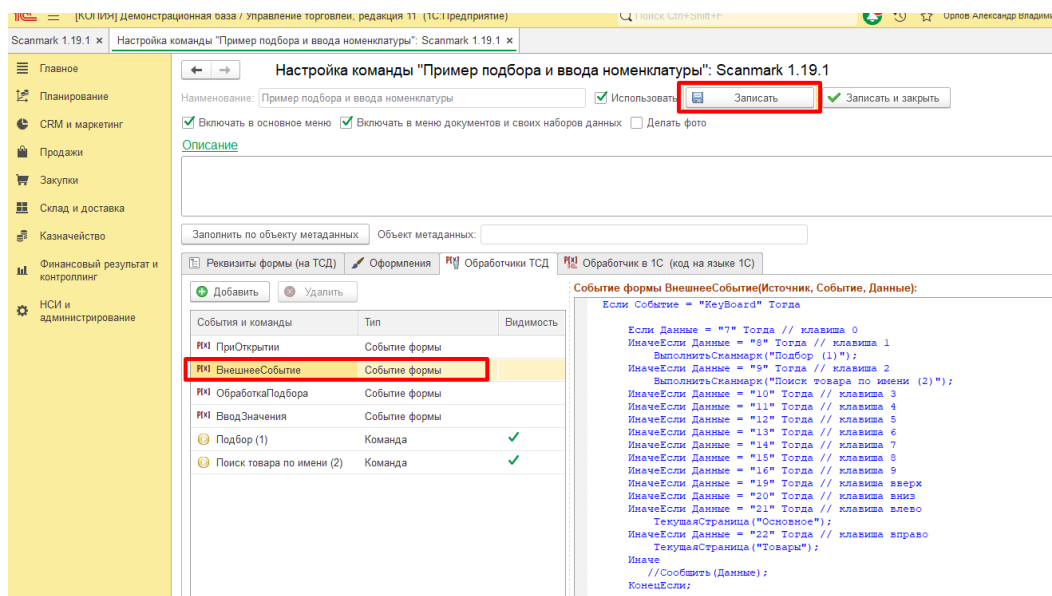
        ПараметрыЗначения = Новый Структура;
        ПараметрыЗначения.Вставить("Тип", "Число");
        ПараметрыЗначения.Вставить("Значение", Количество);
        ПараметрыЗначения.Вставить("Подсказка", "Кол-во товара");
        ПараметрыЗначения.Вставить("Длина", 10);
        ПараметрыЗначения.Вставить("Точность", 0);

        ДополнительныеПараметры = Новый Структура;
        ДополнительныеПараметры.Вставить("Источник", "ПодборТовара");
        ДополнительныеПараметры.Вставить("Номенклатура", ДанныеШтрихкода.Номенклатура);

        ВвестиЗначение(ПараметрыЗначения, ДополнительныеПараметры);
    КонецЕсли;

КонецЕсли;
```

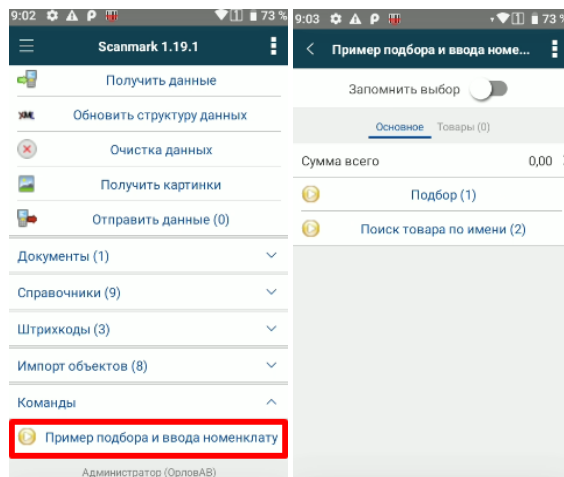
В данном методе мы добавили код обработки для события "KeyBoard", где назначили некоторым клавишам выполнение predetermined команд. Сохраним настройку:



**Внимание!** Данный пример был реализован для ТСД GlobalPOS C5000. Набор клавиш клавиатуры для различных моделей ТСД, а также возвращаемые коды (параметр «Данные») могут отличаться. Поэтому предварительно необходимо определить получаемые внешним событием коды клавиатуры экспериментально.

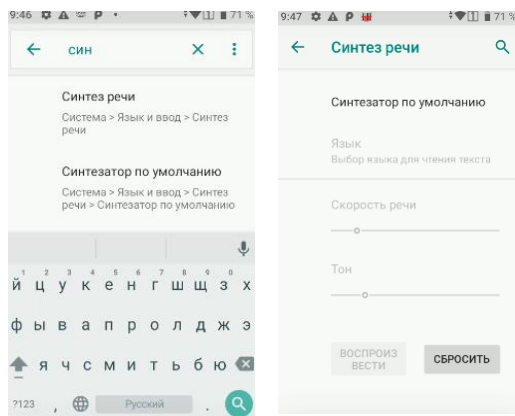
На ТСД выполним обновление структуры данных и проверим работу нового функционала:

- клавиша «1» = Команда «Подбор (1);
- клавиша «2» = Команда «Поиск товара по имени (2)»;
- клавиша «>» (стрелка вправо) = переключение на страницу «Товары»;
- клавиша «<» (стрелка влево) = переключение на страницу «Основное».

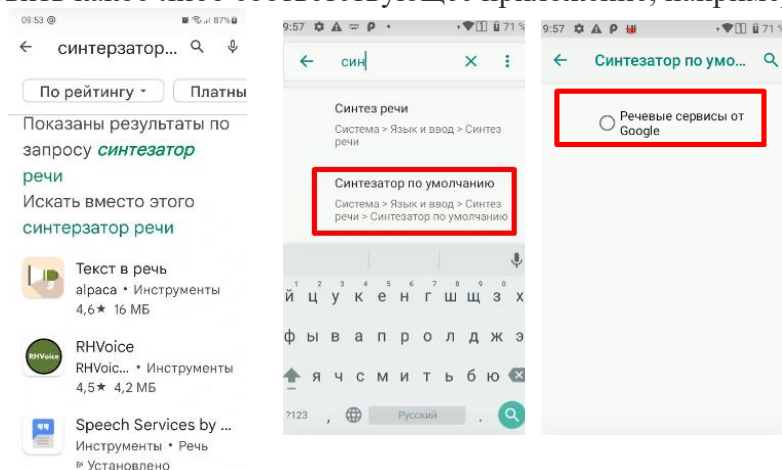


## Задача 26: Пример работы с синтезатором речи.

Для возможности воспроизводить речь на ТСД должен быть установлен синтезатор речи. Проверить наличие установленного синтезатора речи можно в настройках Android:



Если в системе отсутствует установленный синтезатор речи, то можно скачать и установить какое-либо соответствующее приложение, например, с Google Play:



Добавим использование синтезатора речи в созданную в предыдущих задачах команду «Пример подбора и ввода номенклатуры».

Для этого добавить оператор СказатьФразу() в обработчики «ВнешнееСобытие»:  
Если Событие = "KeyBoard" Тогда

```
Если Данные = "7" Тогда // клавиша 0
ИначеЕсли Данные = "8" Тогда // клавиша 1
    ВыполнитьСканмарк("Подбор (1)");
ИначеЕсли Данные = "9" Тогда // клавиша 2
    ВыполнитьСканмарк("Поиск товара по имени (2)");
ИначеЕсли Данные = "10" Тогда // клавиша 3
ИначеЕсли Данные = "11" Тогда // клавиша 4
ИначеЕсли Данные = "12" Тогда // клавиша 5
ИначеЕсли Данные = "13" Тогда // клавиша 6
ИначеЕсли Данные = "14" Тогда // клавиша 7
ИначеЕсли Данные = "15" Тогда // клавиша 8
ИначеЕсли Данные = "16" Тогда // клавиша 9
ИначеЕсли Данные = "19" Тогда // клавиша вверх
ИначеЕсли Данные = "20" Тогда // клавиша вниз
ИначеЕсли Данные = "21" Тогда // клавиша влево
    ТекущаяСтраница("Основное");
ИначеЕсли Данные = "22" Тогда // клавиша вправо
    ТекущаяСтраница("Товары");
Иначе
    //Сообщить(Данные);
КонецЕсли;
```

ИначеЕсли Событие = "Barcode" Тогда

```
ДанныеШтрихкода = ДанныеШтрихкода(Данные, Истина);
```



```

Если ДанныеШтрихкода.Количество() = 0 Тогда
    СказатьФразу("Отсутствует информация по штрихкоду " + Данные + "!");
    СказатьФразу("Получить данные из учетной системы?");
Иначе
    ДанныеШтрихкода = ДанныеШтрихкода[0];
    Строки = Товары.НайтиСтроки(Новый Структура("Номенклатура", ДанныеШтрихкода.Номенклатура));

    Если Строки.Количество() = 0 Тогда
        Количество = 0;
    Иначе
        Количество = Строки[0].Количество;
    КонецЕсли;

    ПараметрыЗначения = Новый Структура;
    ПараметрыЗначения.Вставить("Тип", "Число");
    ПараметрыЗначения.Вставить("Значение", Количество);
    ПараметрыЗначения.Вставить("Подсказка", "Кол-во товара");
    ПараметрыЗначения.Вставить("Длина", 10);
    ПараметрыЗначения.Вставить("Точность", 0);

    ДополнительныеПараметры = Новый Структура;
    ДополнительныеПараметры.Вставить("Источник", "ПодборТовара");
    ДополнительныеПараметры.Вставить("Номенклатура", ДанныеШтрихкода.Номенклатура);

    ВвестиЗначение(ПараметрыЗначения, ДополнительныеПараметры);
    КонецЕсли;

КонецЕсли;

```

## и «ВводЗначения»:

```

Если ТипЗнч(Данные) = Тип("Структура") Тогда
    Если Данные.Источник = "ПодборТовара" и Результат <> 0 Тогда
        оТовар = ОбъектСканмарк(Данные.Номенклатура);
        Строки = Товары.НайтиСтроки(Новый Структура("Номенклатура", Данные.Номенклатура));
        Если Строки.Количество() = 0 Тогда
            Стр = Товары.Добавить();
            Стр.Номенклатура = Данные.Номенклатура;
            Стр.Цена = оТовар.Цена;
        Иначе
            Стр = Строки[0];
        КонецЕсли;
        Стр.Количество = Результат;
        Стр.Сумма = Стр.Цена * Стр.Количество;
        СуммаВсего = Товары.Итог("Сумма");

        СказатьФразу("Цена на товар " + оТовар.Наименование + " = " + оТовар.Цена + " руб");

    ИначеЕсли Данные.Источник = "ПоискПоИмени" Тогда
        ИмяТовара = Результат;
        Если СтрДлина(ИмяТовара) >= 3 Тогда
            Список = Новый СписокЗначений;
            Список.Добавить("< НАЙТИ НА СЕРВЕРЕ >", "< НАЙТИ НА СЕРВЕРЕ >");
            СпрТовары = НайтиПоНаименованию("Номенклатура", "%" + ИмяТовара + "%");
            Для Каждого Товар из СпрТовары Цикл
                оТовар = ОбъектСканмарк(Товар);
                Список.Добавить(Товар, оТовар.Наименование + " (" + оТовар.Остаток + ")");
            КонецЦикла;

            Если Список.Количество() = 1 Тогда
                ПараметрыВопроса = Новый Структура;
                ПараметрыВопроса.Вставить("Тип", "Булево");
                ПараметрыВопроса.Вставить("Подсказка", "Ничего не найдено!" + Символы.ПС + "Найти на сервере?");

                ДополнительныеПараметры = Новый Структура;
                ДополнительныеПараметры.Вставить("Источник", "Найти на сервере");
                ДополнительныеПараметры.Вставить("ИмяТовара", ИмяТовара);

                ВвестиЗначение(ПараметрыВопроса, ДополнительныеПараметры);
                Иначе

                ПараметрыСписок = Новый Структура;
                ПараметрыСписок.Вставить("Тип", "СписокЗначений");
                ПараметрыСписок.Вставить("Значение", Список);

                ДополнительныеПараметры = Новый Структура;
                ДополнительныеПараметры.Вставить("Источник", "ВыбратьТоварИзСписка");

```

```

        ДополнительныеПараметры.Вставить ("ИмяТовара", ИмяТовара);

        ВвестиЗначение (ПараметрыСписок, ДополнительныеПараметры);
        КонецЕсли;
        КонецЕсли;

ИначеЕсли Данные.Источник = "ВыбратьТоварИзСписка" Тогда
    Если Результат = "< НАЙТИ НА СЕРВЕРЕ >" Тогда
        ИмяМетода1с = "Получить номенклатуру по имени";
        ИмяСобытияТСД = "";
        ПолучитьДанныеИз1с (Данные.ИмяТовара, ИмяМетода1с, ИмяСобытияТСД);
    Иначе
        Строки = Товары.НайтиСтроки (Новый Структура ("Номенклатура", Результат));
        Если Строки.Количество() = 0 Тогда
            Количество = 0;
        Иначе
            Количество = Строки[0].Количество;
        КонецЕсли;
        ПараметрыЗначения = Новый Структура;
        ПараметрыЗначения.Вставить ("Тип", "Число");
        ПараметрыЗначения.Вставить ("Значение", Количество);
        ПараметрыЗначения.Вставить ("Подсказка", "Кол-во товара");
        ПараметрыЗначения.Вставить ("Длина", 10);
        ПараметрыЗначения.Вставить ("Точность", 0);

        ДополнительныеПараметры = Новый Структура;
        ДополнительныеПараметры.Вставить ("Источник", "ПодборТовара");
        ДополнительныеПараметры.Вставить ("Номенклатура", Результат);

        ВвестиЗначение (ПараметрыЗначения, ДополнительныеПараметры);
        КонецЕсли;

ИначеЕсли Данные.Источник = "Найти на сервере" Тогда
    Если Результат Тогда
        ИмяМетода1с = "Получить номенклатуру по имени";
        ИмяСобытияТСД = "";
        ПолучитьДанныеИз1с (Данные.ИмяТовара, ИмяМетода1с, ИмяСобытияТСД);
        КонецЕсли;

    КонецЕсли;

КонецЕсли;

```

Сохраним настройку, на ТСД обновим структуру данных и попробуем осуществить подбор номенклатуры с помощью сканирования штрихкодов номенклатуры или кнопкой «Подбор (1)». В результате при подборе ТСД будет произносить текст.

## Задача 27: Пример подбора номенклатуры по штрихкоду по данным на ТСД + из 1С с возможностью установить новый штрихкод номенклатуры в 1С.

В данной задаче реализуем алгоритм:

1. сканируем штрихкод;
2. если на ТСД находим номенклатуру по штрихкоду, то добавляем номенклатуру и штрихкод в табличную часть;
3. если п.2 не выполняется (по штрихкоду не находим номенклатуру на ТСД), то пробуем получить данные из 1С, и при успешном поиске, добавляем номенклатуру и штрихкод в табличную часть;
4. если п.3 не выполняется (информация по штрихкоду отсутствует и на ТСД и в 1С), то пробуем подобрать соответствующую номенклатуру по имени по данным на ТСД;
5. если подобрать номенклатуру в п.4 не удалось, то пробуем подобрать соответствующую номенклатуру по данным из 1С;
6. если удалось подобрать соответствующую штрихкоду номенклатуру в п.4 или в п.5, то отдельной командой передадим информацию в 1С и добавим новый штрихкод в учетную систему, после чего обновим данные на ТСД.

В Обработке Scanmark на закладке «Произвольные данные» добавим команду «Номенклатура по штрихкоду»:

The image shows two screenshots of the Scanmark 1.19.4 application interface. The top screenshot shows the main menu on the left with 'Произвольные данные (20)' highlighted. The right pane shows a list of commands, with 'Команды' (Commands) highlighted. The bottom screenshot shows the 'Настройка команды (новая): Scanmark 1.19.4 \*' dialog. The 'Наименование' (Name) field is set to 'Номенклатура по штрихкоду'. The 'Использовать' (Use) checkbox is checked. The 'Описание' (Description) field is empty. The 'Заполнить по объекту метаданных' (Fill by object metadata) checkbox is checked. The 'Объект метаданных' (Object metadata) field is empty. The 'Реквизиты формы (на ТСД)' (Form requisites (on ТСД)) tab is selected, showing a list of requisites with 'Реквизиты' (Requisites) selected.

Добавим реквизиты и изменим свойства как на картинках:

Scanmark 1.19.4 x Настройка команды "Номенклатура по штрихкоду": Scanmark 1.19.4 x

Главное

Планирование

CRM и маркетинг

Продажи

Закупки

Склад и доставка

Казначейство

Финансовый результат и контроллинг

НСИ и администрирование

Настройка команды "Номенклатура по штрихкоду": Scanmark 1.19.4

Включать в основное меню Включать в меню документов и своих наборов данных Делать фото

Описание

Заполнить по объекту метаданных: Объект метаданных:

Реквизиты формы (на ТСД) Оформление Обработки ТСД Обработка в 1С (код на языке 1С)

Добавить

Реквизит/Свойство	Значение свойства
Реквизиты	
Штрихкод	
Синоним	Штрихкод
Тип	Строка
Видимость	Ложь
ТолькоПросмотр	Ложь
Ширина	25
Положение	На следующей строке

mark 1.19.4 x Настройка команды "Номенклатура по штрихкоду": Scanmark 1.19.4 x

Главное

Планирование

CRM и маркетинг

Продажи

Закупки

Склад и доставка

Казначейство

Финансовый результат и контроллинг

НСИ и администрирование

Настройка команды "Номенклатура по штрихкоду": Scanmark 1.19.4

Включать в основное меню Включать в меню документов и своих наборов данных Делать фото

Описание

Заполнить по объекту метаданных: Объект метаданных:

Реквизиты формы (на ТСД) Оформление Обработки ТСД Обработка в 1С (код на языке 1С)

Добавить

Реквизит/Свойство	Значение свойства
Реквизиты	
Штрихкод	
ИмяТовара	
Синоним	ИмяТовара
Тип	Строка
Видимость	Ложь
ТолькоПросмотр	Ложь
Ширина	25
Положение	На следующей строке

ic [КОПИЯ] Демонстрационная база / Управление торговлей, редакция 11 (1С:Предприятие)

Scanmark 1.19.4 x Настройка команды "Номенклатура по штрихкоду": Scanmark 1.19.4 x

Главное

Планирование

CRM и маркетинг

Продажи

Закупки

Склад и доставка

Казначейство

Финансовый результат и контроллинг

НСИ и администрирование

Настройка команды "Номенклатура по штрихкоду": Scanmark 1.19.4

Наименование: Номенклатура по штрихкоду Использовать Записать Записать и закрыть

Включать в основное меню Включать в меню документов и своих наборов данных Делать фото

Описание

Заполнить по объекту метаданных: Объект метаданных:

Реквизиты формы (на ТСД) Оформление Обработки ТСД Обработка в 1С (код на языке 1С)

Добавить

Реквизит/Свойство	Значение свойства
Реквизиты	
Штрихкод	
ИмяТовара	
Табличные части	
Товары	Товары
Номенклатура	
Синоним	Номенклатура
Тип	Номенклатура
Видимость	Истина
ТолькоПросмотр	Ложь
Ширина	25

Scanmark 1.19.4 x Настройка команды "Номенклатура по штрихкоду": Scanmark 1.19.4 x

Наименование: Номенклатура по штрихкоду ☒ Использовать ☐ Записать ☒ Записать и закрыть

☒ Включать в основное меню ☒ Включать в меню документов и своих наборов данных ☐ Делать фото

Описание

Заполнить по объекту метаданных: Объект метаданных:

☒ Ревизиты формы (на ТСД) ☒ Оформление ☒ Обработчики ТСД ☒ Обработчик в 1С (код на языке 1С)

☒ Добавить

Ревизит/Свойство	Значение свойства
⇒ Ревизиты	
⇒ Штрихкод	
⇒ ИмяТовара	
⇒ Табличные части	
⇒ Товары	Товары
⇒ Номенклатура	
⇒ Штрихкод	
Синоним	Штрихкод
Тип	Строка
Видимость	Истина
ТолькоПросмотр	Ложь
Ширина	25

Scanmark 1.19.4 x Настройка команды "Номенклатура по штрихкоду": Scanmark 1.19.4 x

Наименование: Номенклатура по штрихкоду ☒ Использовать ☐ Записать ☒ Записать и закрыть

☒ Включать в основное меню ☒ Включать в меню документов и своих наборов данных ☐ Делать фото

Описание

Заполнить по объекту метаданных: Объект метаданных:

☒ Ревизиты формы (на ТСД) ☒ Оформление ☒ Обработчики ТСД ☒ Обработчик в 1С (код на языке 1С)

☒ Добавить

Ревизит/Свойство	Значение свойства
⇒ Ревизиты	
⇒ Штрихкод	
⇒ ИмяТовара	
⇒ Табличные части	
⇒ Товары	Товары
⇒ Номенклатура	
⇒ Штрихкод	
⇒ Создать	
Синоним	Создать
Тип	Булево
Видимость	Истина
ТолькоПросмотр	Ложь
Ширина	10
Положение	В той же колонке
ЦветФона	Авто

На закладке «Обработчики ТСД» создадим команды «ДобавитьТовар», «ПовторныйВыборТовара» и «Передать новые штрихкоды в 1с» как на картинке:

Scanmark 1.19.4 x Настройка команды "Номенклатура по штрихкоду": Scanmark 1.19.4 x

Наименование: Номенклатура по штрихкоду ☒ Использовать ☐ Записать ☒ Записать и закрыть

☒ Включать в основное меню ☒ Включать в меню документов и своих наборов данных ☐ Делать фото

Описание

Заполнить по объекту метаданных: Объект метаданных:

☒ Ревизиты формы (на ТСД) ☒ Оформление ☒ Обработчики ТСД ☒ Обработчик в 1С (код на языке 1С)

☒ Добавить ☐ Удалить

Генератор кода обработчика

События и команды	Тип	Видимость
ПриОткрытии	Событие формы	
ВнешнееСобытие	Событие формы	
ОбработкаПодбора	Событие формы	
ВводЗначения	Событие формы	
ДобавитьТовар	Команда	
ПовторныйВыборТовара	Команда	
Передать новые штрихкоды в 1с	Команда	<input checked="" type="checkbox"/>

Событие формы ПриОткрытии(): ТекущаяСтраница ("Товары"):

Добавим код для событий и команд:

**1. ПриОткрытии:**

```
ТекущаяСтраница ("Товары");
```

**2. ВнешнееСобытие:**

```
Если Событие = "KeyBoard" Тогда
//ваш код обработки события нажатия клавиши
ИначеЕсли Событие = "RFID" Тогда
//ваш код обработки события считывания радиометок
ИначеЕсли Событие = "Barcode" Тогда
Строки = Товары.НайтиСтроки(Новый Структура("Штрихкод", Данные));
Если Строки.Количество() = 0 Тогда
ДанныеШтрихкода = ДанныеШтрихкода(Данные);
Если ДанныеШтрихкода.Количество() > 0 Тогда
ДанныеШтрихкода = ДанныеШтрихкода[0];
Стр = Товары.Добавить();
Стр.Номенклатура = ДанныеШтрихкода.Номенклатура;
Стр.Штрихкод = Данные;
Стр.Создать = Ложь;
Иначе
Штрихкод = Данные;
ИмяМетода1с = "Получить номенклатуру по штрихкоду";
ИмяСобытияТСД = "ДобавитьТовар";
ПолучитьДанныеИз1с(Данные, ИмяМетода1с, ИмяСобытияТСД);
КонецЕсли;
КонецЕсли;
КонецЕсли;
```

**3. ВводЗначения:**

```
Если ТипЗнч(Данные) = Тип("Структура") Тогда
Если Данные.Источник = "ПоискПоИмени" Тогда
ИмяТовара = Результат;
Если СтрДлина(ИмяТовара) >= 3 Тогда
Список = Новый СписокЗначений;
Список.Добавить("< НАЙТИ НА СЕРВЕРЕ >", "< НАЙТИ НА СЕРВЕРЕ >");
СпрТовары = НайтиПоНаименованию("Номенклатура", "%" + ИмяТовара +
"%");
Для Каждого Товар из СпрТовары Цикл
оТовар = ОбъектСканмарк(Товар);
Если оТовар.ЭтоГруппа = Ложь Тогда
Список.Добавить(Товар, оТовар.Наименование);
КонецЕсли;
КонецЦикла;

ПараметрыСписок = Новый Структура;
ПараметрыСписок.Вставить("Тип", "СписокЗначений");
ПараметрыСписок.Вставить("Значение", Список);

ДополнительныеПараметры = Новый Структура;
ДополнительныеПараметры.Вставить("Источник",
"ВыбратьТоварИзСписка");
ДополнительныеПараметры.Вставить("ИмяТовара", ИмяТовара);

ВвестиЗначение(ПараметрыСписок, ДополнительныеПараметры);

КонецЕсли;

ИначеЕсли Данные.Источник = "ВыбратьТоварИзСписка" Тогда
Если Результат = "< НАЙТИ НА СЕРВЕРЕ >" Тогда
ИмяТовара = Данные.ИмяТовара;
ИмяМетода1с = "Получить номенклатуру по имени";
ИмяСобытияТСД = "ПовторныйВыборТовара";
ПолучитьДанныеИз1с(ИмяТовара, ИмяМетода1с, ИмяСобытияТСД);
```

```

        Иначе
            Стр          = Товары.Добавить ();
            Стр.Номенклатура = Результат;
            Стр.Штрихкод   = Штрихкод;
            Стр.Создать    = Истина;
        КонецЕсли;
    КонецЕсли;
КонецЕсли;

```

#### 4. ДобавитьТовар:

```

ДанныеШтрихкода = ДанныеШтрихкода (Штрихкод);
Если ДанныеШтрихкода.Количество() > 0 Тогда
    ДанныеШтрихкода = ДанныеШтрихкода[0];
    Стр          = Товары.Добавить ();
    Стр.Номенклатура = ДанныеШтрихкода.Номенклатура;
    Стр.Штрихкод   = Штрихкод;
    Стр.Создать    = Ложь;

```

Иначе

```

Подсказка = "Штрихкод " + СокрЛП(Штрихкод) + " в информационной базе не
зарегистрирован!";
Подсказка = Подсказка + Символы.ПС + "Подобрать номенклатуру по имени?";
ПараметрыСтроки = Новый Структура;
ПараметрыСтроки.Вставить("Тип", "Строка");
ПараметрыСтроки.Вставить("Значение", "");
ПараметрыСтроки.Вставить("Подсказка", Подсказка);

```

```

ДополнительныеПараметры = Новый Структура;
ДополнительныеПараметры.Вставить("Источник", "ПоискПоИмени");

```

```

ВвестиЗначение(ПараметрыСтроки, ДополнительныеПараметры);

```

КонецЕсли;

#### 5. ПовторныйВыборТовара:

```

Если СтрДлина(ИмяТовара) >= 3 Тогда
    Список = Новый СписокЗначений;
    Список.Добавить("< НАЙТИ НА СЕРВЕРЕ >", "< НАЙТИ НА СЕРВЕРЕ >");
    СпрТовары = НайтиПоНаименованию("Номенклатура", "%" + ИмяТовара + "%");
    Для Каждого Товар из СпрТовары Цикл
        оТовар = ОбъектСканмарк(Товар);
        Если оТовар.ЭтоГруппа = Ложь Тогда
            Список.Добавить(Товар, оТовар.Наименование);
        КонецЕсли;
    КонецЦикла;

```

```

ПараметрыСписок = Новый Структура;
ПараметрыСписок.Вставить("Тип", "СписокЗначений");
ПараметрыСписок.Вставить("Значение", Список);

```

```

ДополнительныеПараметры = Новый Структура;
ДополнительныеПараметры.Вставить("Источник", "ВыбратьТоварИзСписка");
ДополнительныеПараметры.Вставить("ИмяТовара", ИмяТовара);

```

```

ВвестиЗначение(ПараметрыСписок, ДополнительныеПараметры);

```

КонецЕсли;

#### 6. Передать новые штрихкоды в 1с:

```

ТЗ = Товары.Скопировать(Новый Структура("Создать", Истина));
Если ТЗ.Количество() > 0 Тогда
    ИмяМетода1с = "Передать новые штрихкоды";

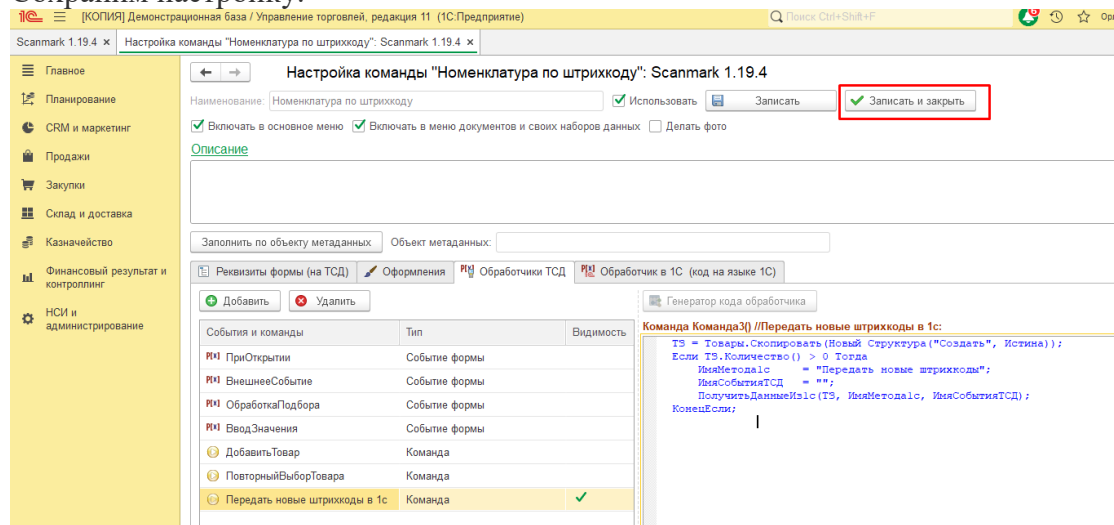
```

```

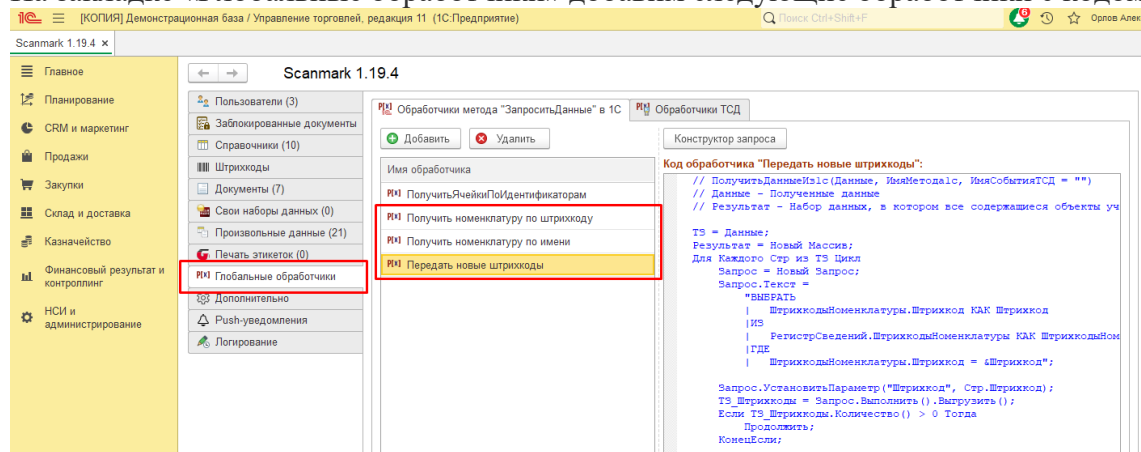
ИмяСобытияТСД = "";
ПолучитьДанныеИз1с(ТЗ, ИмяМетода1с, ИмяСобытияТСД);
КонецЕсли;

```

## Сохраним настройки:



## На закладке «Глобальные обработчики» добавим следующие обработчики с кодом:



### 1. Получить номенклатуру по штрихкоду:

```

Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
| ШтрихкодыНоменклатуры.Номенклатура КАК Номенклатура,
| ШтрихкодыНоменклатуры.Упаковка КАК Упаковка
| ИЗ
| РегистрСведений.ШтрихкодыНоменклатуры КАК ШтрихкодыНоменклатуры
| ГДЕ
| ШтрихкодыНоменклатуры.Штрихкод = &Штрихкод";

```

```

Запрос.УстановитьПараметр("Штрихкод", Данные);
Результат = Запрос.Выполнить().Выгрузить();

```

### 2. Получить номенклатуру по имени:

```

Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
| Номенклатура.Ссылка КАК Ссылка
| ИЗ
| Справочник.Номенклатура КАК Номенклатура
| ГДЕ
| Номенклатура.ЭтоГруппа = ЛОЖЬ

```



```
| И Номенклатура.Наименование ПОДОБНО &Имя";
```

```
Запрос.УстановитьПараметр("Имя", "%" + Данные + "%");  
Результат = Запрос.Выполнить().Выгрузить();
```

### 3. Передать новые штрихкоды:

```
// ПолучитьДанныеИз1с(Данные, ИмяМетода1с, ИмяСобытияТСД = "")  
// Данные - Полученные данные  
// Результат - Набор данных, в котором все содержащиеся объекты учета  
будут переданы на ТСД
```

```
ТЗ = Данные;  
Результат = Новый Массив;  
Для Каждого Стр из ТЗ Цикл  
Запрос = Новый Запрос;  
Запрос.Текст =  
    "ВЫБРАТЬ  
    | ШтрихкодыНоменклатуры.Штрихкод КАК Штрихкод  
    | ИЗ  
    | РегистрСведений.ШтрихкодыНоменклатуры КАК ШтрихкодыНоменклатуры  
    | ГДЕ  
    | ШтрихкодыНоменклатуры.Штрихкод = &Штрихкод";
```

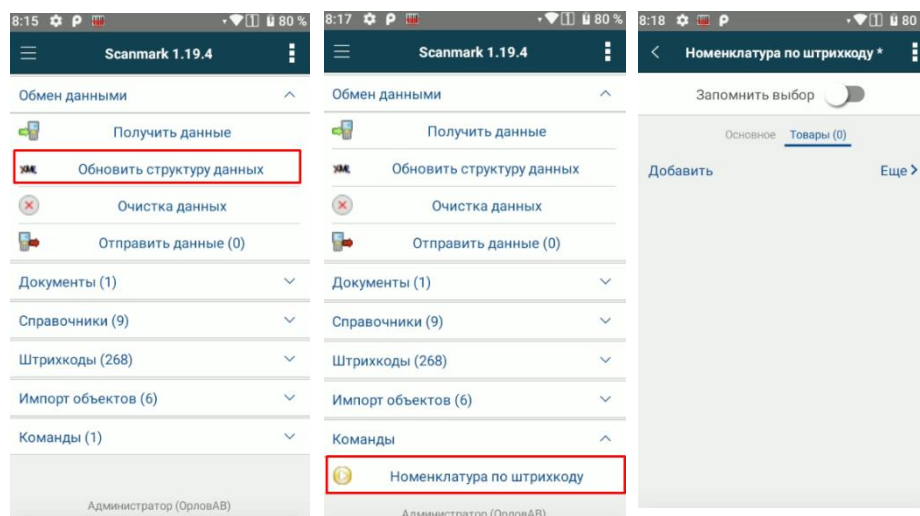
```
Запрос.УстановитьПараметр("Штрихкод", Стр.Штрихкод);  
ТЗ_Штрихкоды = Запрос.Выполнить().Выгрузить();  
Если ТЗ_Штрихкоды.Количество() > 0 Тогда  
    Продолжить;  
КонецЕсли;
```

```
МенеджерЗаписи =  
РегистрыСведений.ШтрихкодыНоменклатуры.СоздатьМенеджерЗаписи();  
МенеджерЗаписи.Штрихкод = Стр.Штрихкод;  
МенеджерЗаписи.Номенклатура = Стр.Номенклатура;  
//МенеджерЗаписи.Характеристика =  
//МенеджерЗаписи.Упаковка =  
МенеджерЗаписи.Записать(Истина);
```

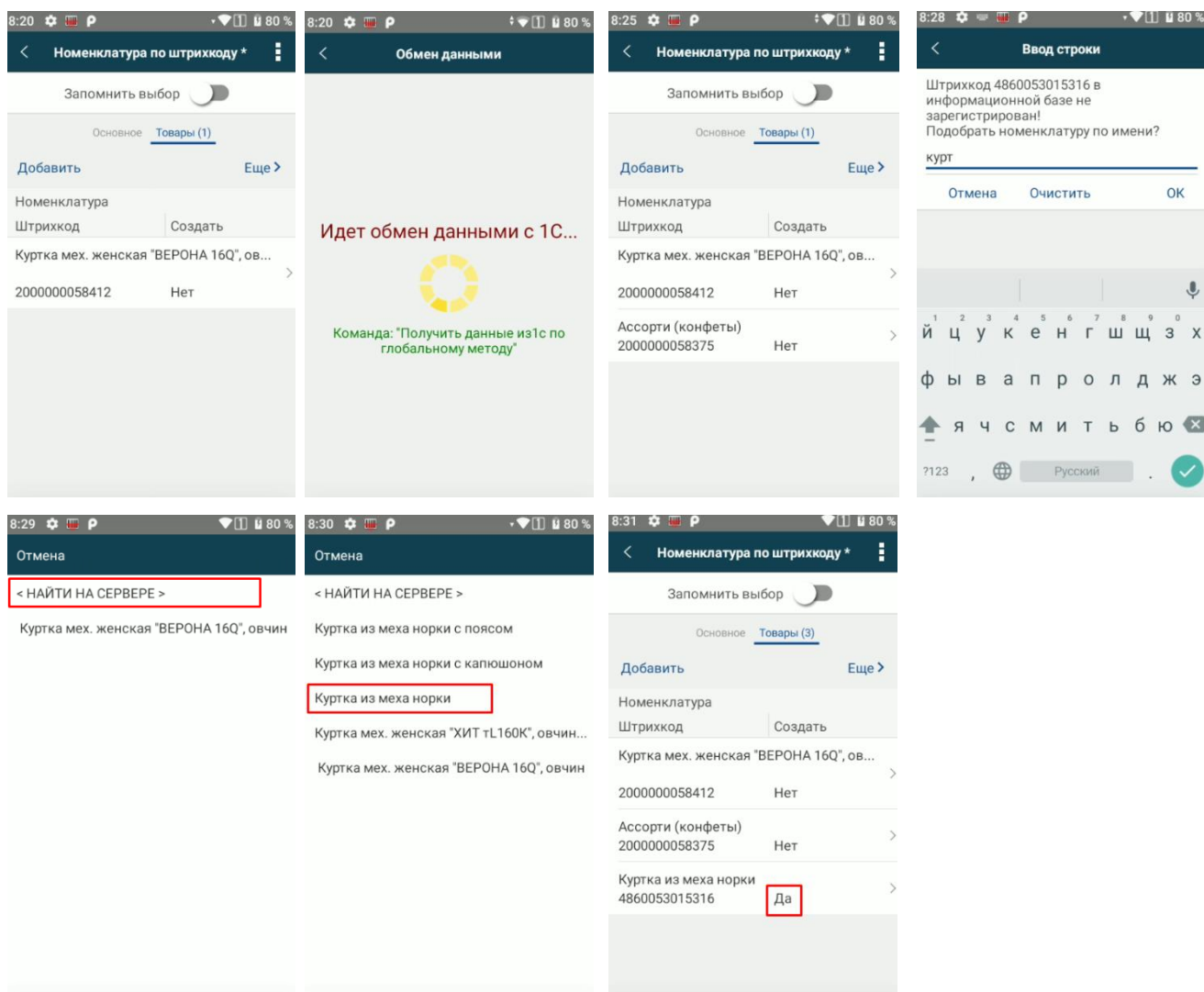
```
Результат.Добавить(Стр.Номенклатура);  
КонецЦикла;
```

Настройка сделана. Осталось проверить.

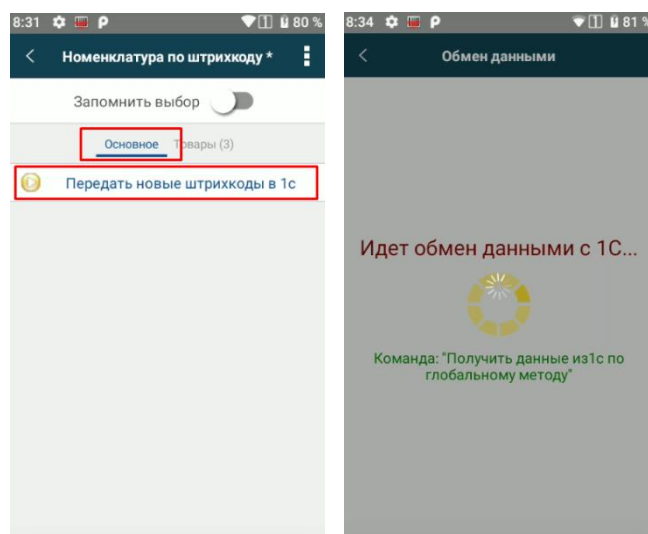
Обновим структуру данных на ТСД. Откроем команду «Номенклатура по штрихкоду».



Для примера сначала пробуем сканировать штрихкоды, которые уже есть в учетной системе, а потом новые штрихкоды.



Командой «Передать новые штрихкоды в 1с» создаем в 1С новый штрихкод для номенклатуры «Куртка из меха норки»:



☰

Сcanmark 1.19.4 x

Настройка справочника "Номенклатура" x

Номенклатура x

Куртка из меха норки (Номенклатура) x

Штрихкоды номенклатуры x

Поиск Ctrl+Shift+F

🔄 ⭐

Ори

Главное

Планирование

CRM и маркетинг

Продажи

Закупки

Склад и доставка

...

← → ☆ Штрихкоды номенклатуры

Создать 🔒

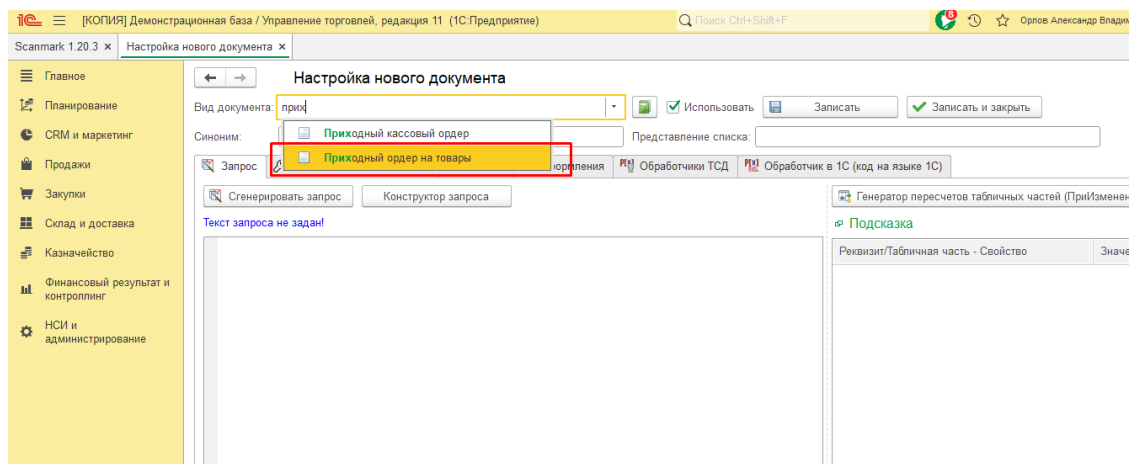
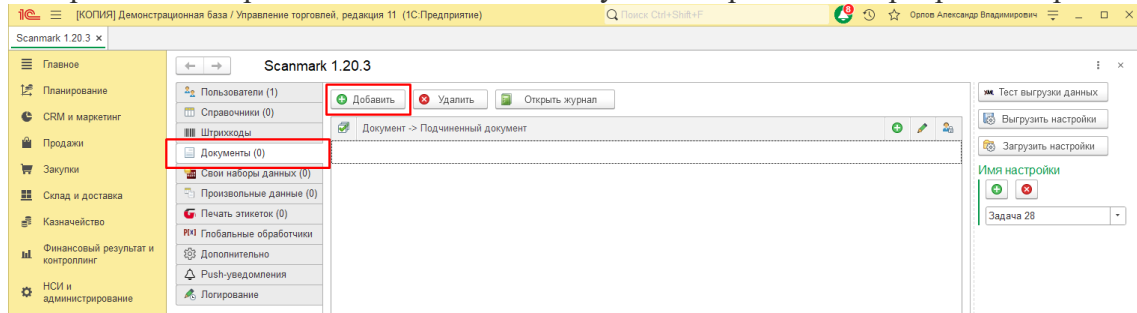
Поиск (Ctrl+F)

Штрихкод	Номенклатура	Характеристика
4640019487285	Куртка из меха норки	52 Черный
4670020041671	Куртка из меха норки	42 Белый
4860053015316	Куртка из меха норки	<характеристики не используются>

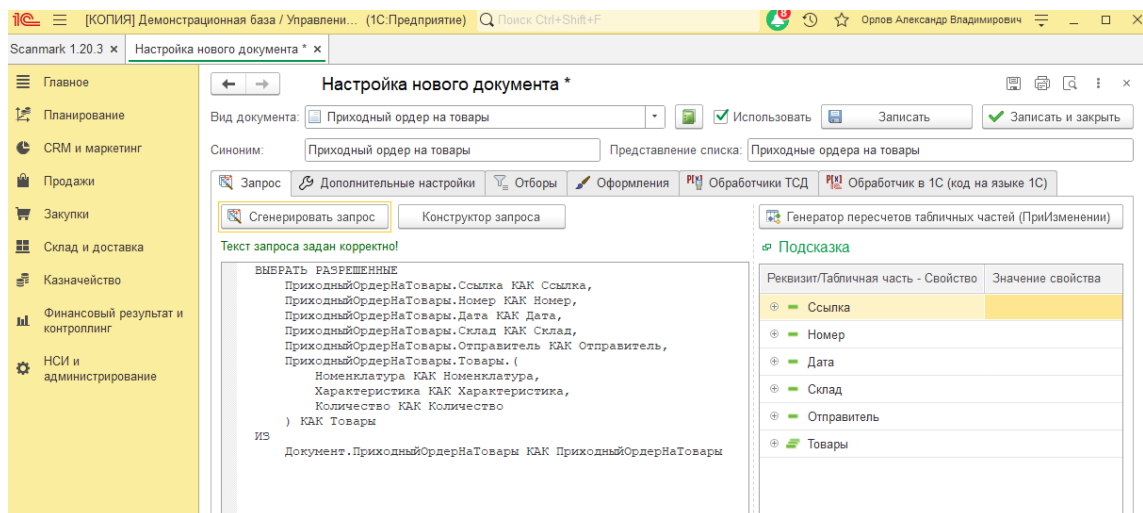
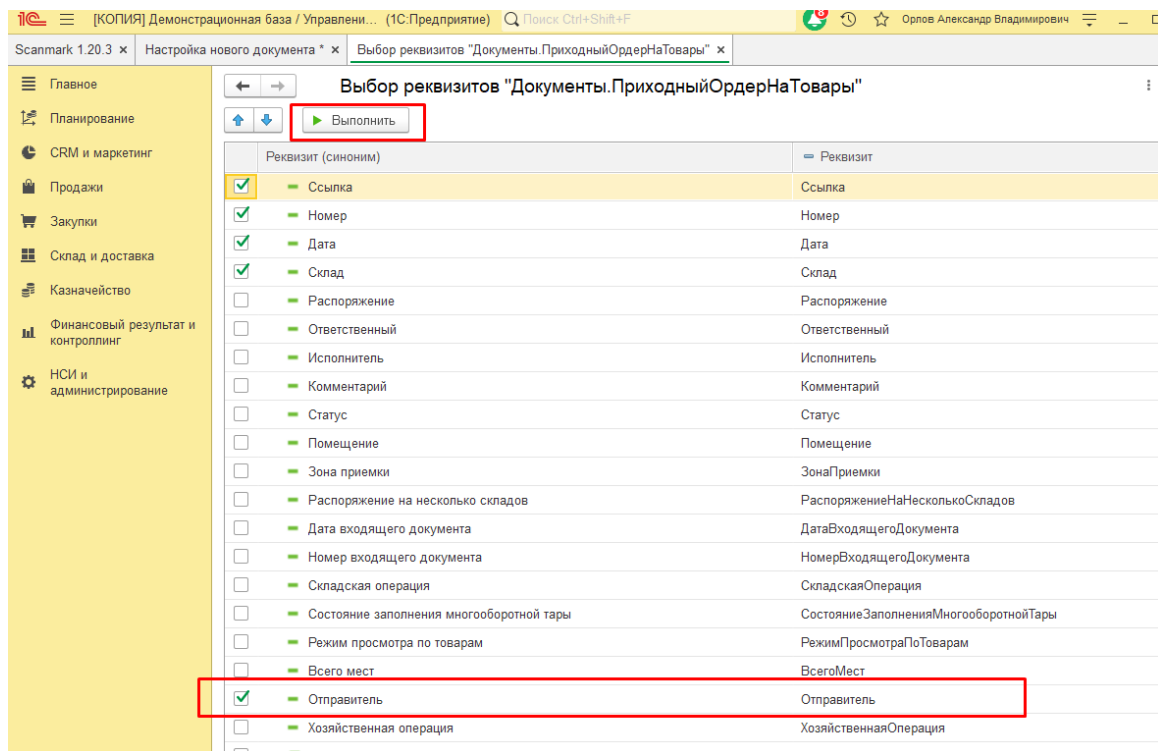
## Задача 28: Пример работы с составным типом данных, преобразование типа данных.

В 1С:Предприятие описание типов значений некоторых реквизитов объектов метаданных является составным типом данных. Мобильное приложение «Scanmark» не поддерживает составной тип данных. В данной задаче реализуем настройку, в которой реализуем работу с таким типом данных, а также используем методы управления отображением элементов формы (спойлер: видимость элементов, расположение кнопок (команд) на форме ТСД).

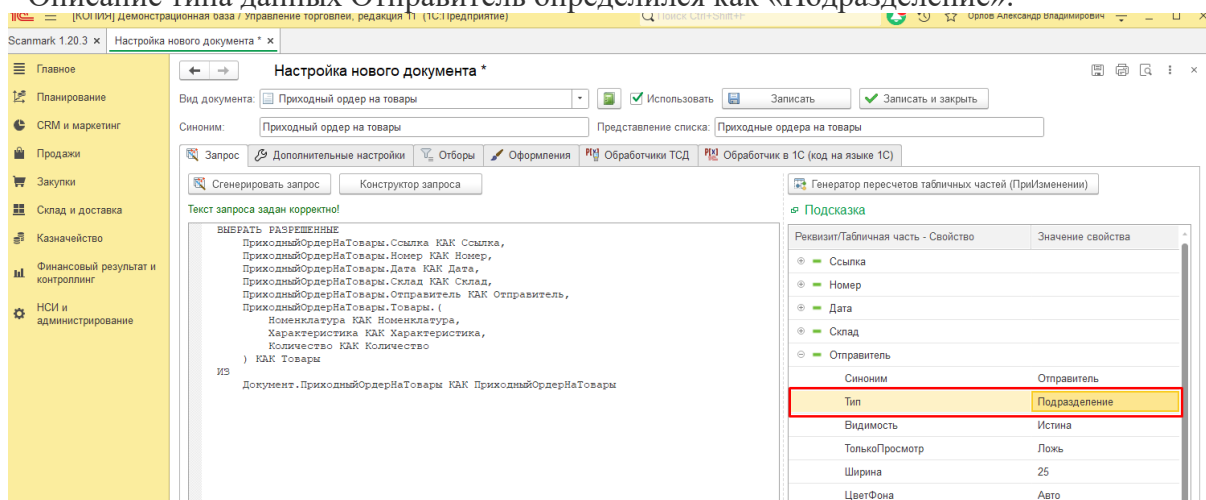
В обработке настройки добавим новый документ «ПриходныйОрдерНаТовары»:



Некоторые реквизиты уже выбран по умолчанию. Отметим дополнительно реквизит «Отправитель»:

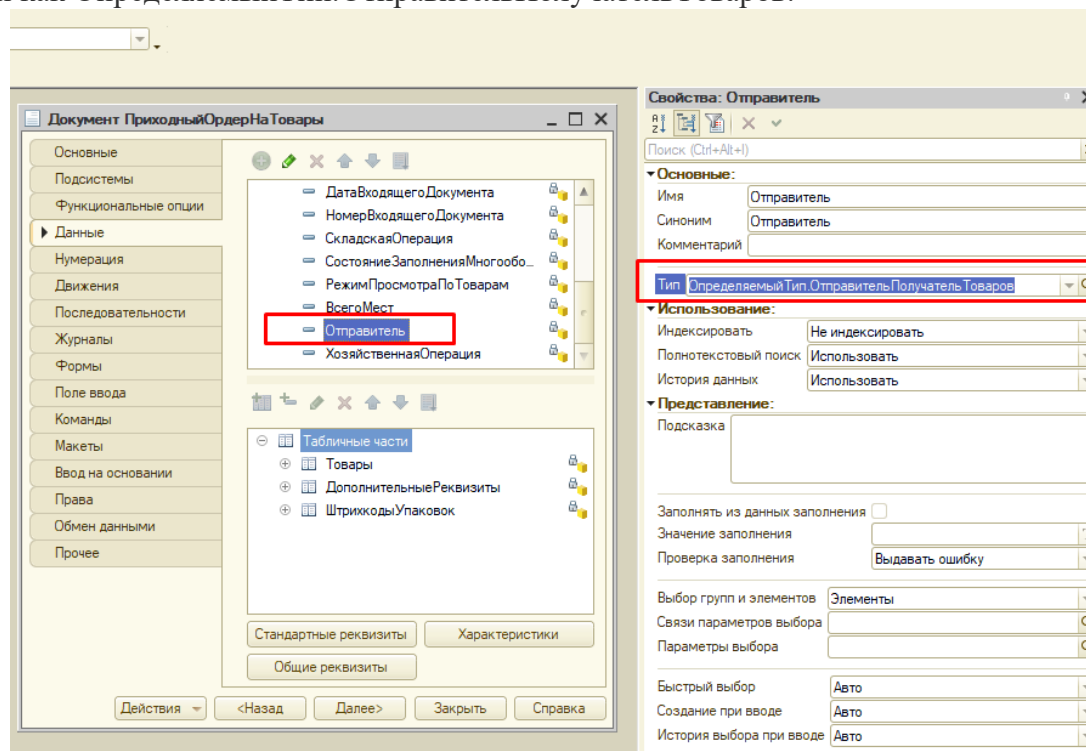


Описание типа данных Отправитель определится как «Подразделение»:

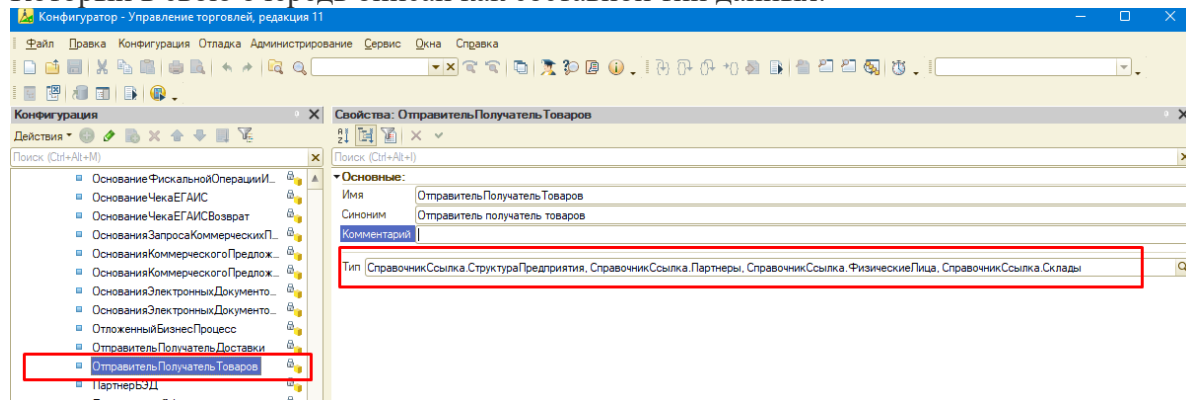


В конфигурации в документе «ПриходныйОрдерНаТовары» реквизит отправитель

описан как ОпределяемыйТип.ОтправительПолучательТоваров:



Который в свою очередь описан как составной тип данных:



С помощью конструктора запросов изменим текст запроса и добавим несколько вычисляемых реквизитов: СтрОтправитель, ОтправительСклад, ОтправительПартнер, ОтправительПодразделение, ОтправительФизЛицо, ВидОтправителя:

ВЫБРАТЬ

ПриходныйОрдерНаТовары.Ссылка КАК Ссылка,  
ПриходныйОрдерНаТовары.Номер КАК Номер,  
ПриходныйОрдерНаТовары.Дата КАК Дата,  
ПриходныйОрдерНаТовары.Склад КАК Склад,  
ПриходныйОрдерНаТовары.Отправитель.Представление КАК СтрОтправитель,

ВЫБОР

КОГДА ПриходныйОрдерНаТовары.Отправитель ССЫЛКА Справочник.Склады

ТОГДА ПриходныйОрдерНаТовары.Отправитель

ИНАЧЕ ЗНАЧЕНИЕ (Справочник.Склады.ПустаяСсылка)

КОНЕЦ КАК ОтправительСклад,

ВЫБОР

КОГДА ПриходныйОрдерНаТовары.Отправитель ССЫЛКА Справочник.Партнеры

ТОГДА ПриходныйОрдерНаТовары.Отправитель

ИНАЧЕ ЗНАЧЕНИЕ (Справочник.Партнеры.ПустаяСсылка)

КОНЕЦ КАК ОтправительПартнер,

ВЫБОР

КОГДА ПриходныйОрдерНаТовары.Отправитель ССЫЛКА Справочник.СтруктураПредприятия

ТОГДА ПриходныйОрдерНаТовары.Отправитель

ИНАЧЕ ЗНАЧЕНИЕ (Справочник.СтруктураПредприятия.ПустаяСсылка)

КОНЕЦ КАК ОтправительПодразделение,  
ВЫБОР

КОГДА ПриходныйОрдерНаТовары.Отправитель ССЫЛКА Справочник.ФизическиеЛица

ТОГДА ПриходныйОрдерНаТовары.Отправитель

ИНАЧЕ ЗНАЧЕНИЕ (Справочник.ФизическиеЛица.ПустаяСсылка)

КОНЕЦ КАК ОтправительФизЛицо,  
ВЫБОР

КОГДА ПриходныйОрдерНаТовары.Отправитель ССЫЛКА Справочник.Склады

ТОГДА "Склады"

ИНАЧЕ ВЫБОР

КОГДА ПриходныйОрдерНаТовары.Отправитель ССЫЛКА Справочник.Партнеры

ТОГДА "Партнеры"

ИНАЧЕ ВЫБОР

КОГДА

ПриходныйОрдерНаТовары.Отправитель

ССЫЛКА

Справочник.СтруктураПредприятия

ТОГДА "СтруктураПредприятия"

ИНАЧЕ ВЫБОР

КОГДА

ПриходныйОрдерНаТовары.Отправитель

ССЫЛКА

Справочник.ФизическиеЛица

ТОГДА "ФизическиеЛица"

ИНАЧЕ ""

КОНЕЦ

КОНЕЦ

КОНЕЦ

КОНЕЦ КАК ВидОтправителя,

ПриходныйОрдерНаТовары.Товары. (

Номенклатура КАК Номенклатура,

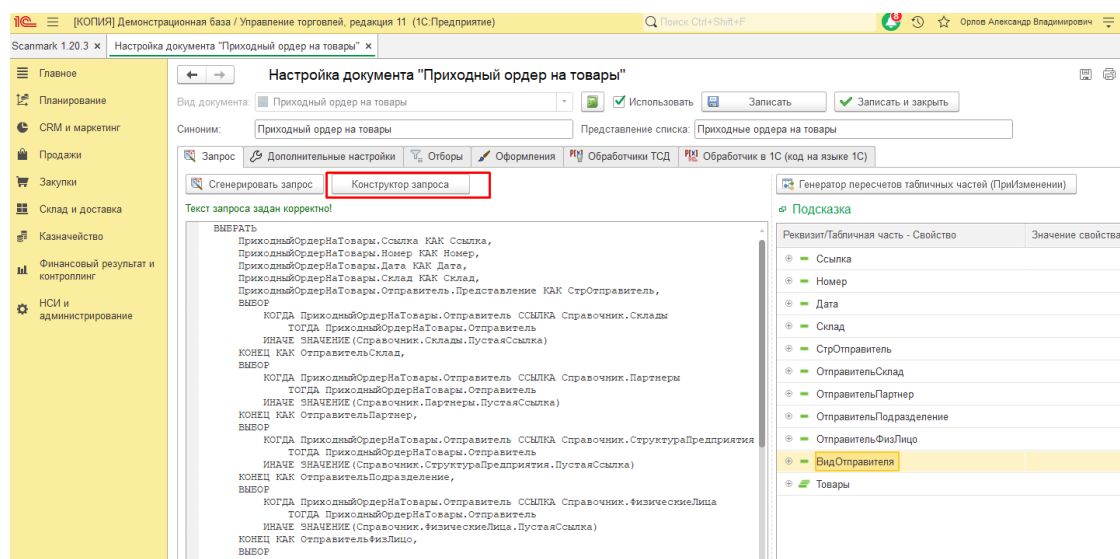
Характеристика КАК Характеристика,

КоличествоУпаковок КАК Количество

) КАК Товары

ИЗ

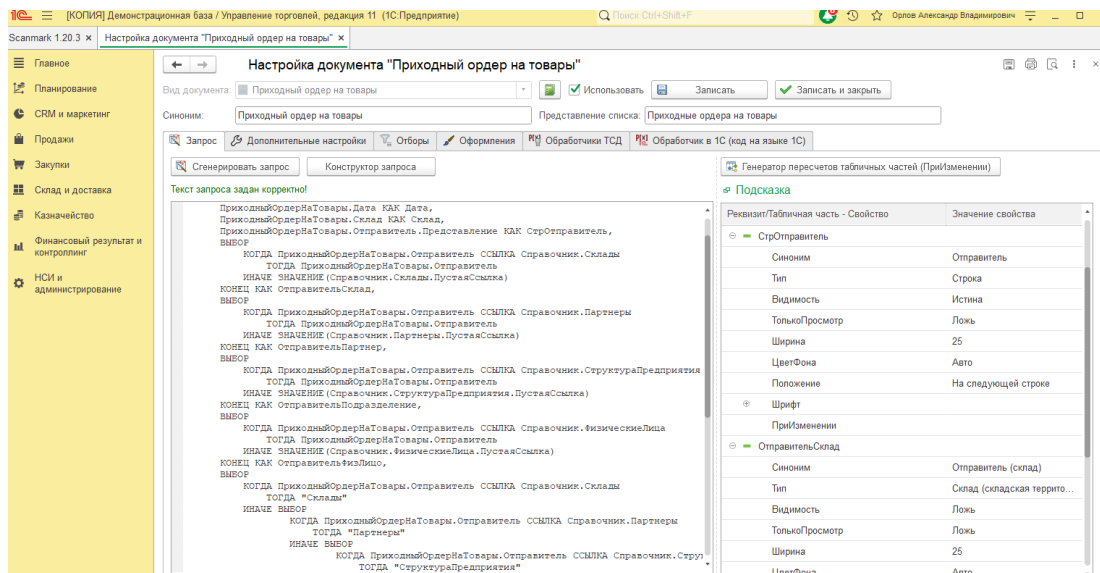
Документ.ПриходныйОрдерНаТовары КАК ПриходныйОрдерНаТовары



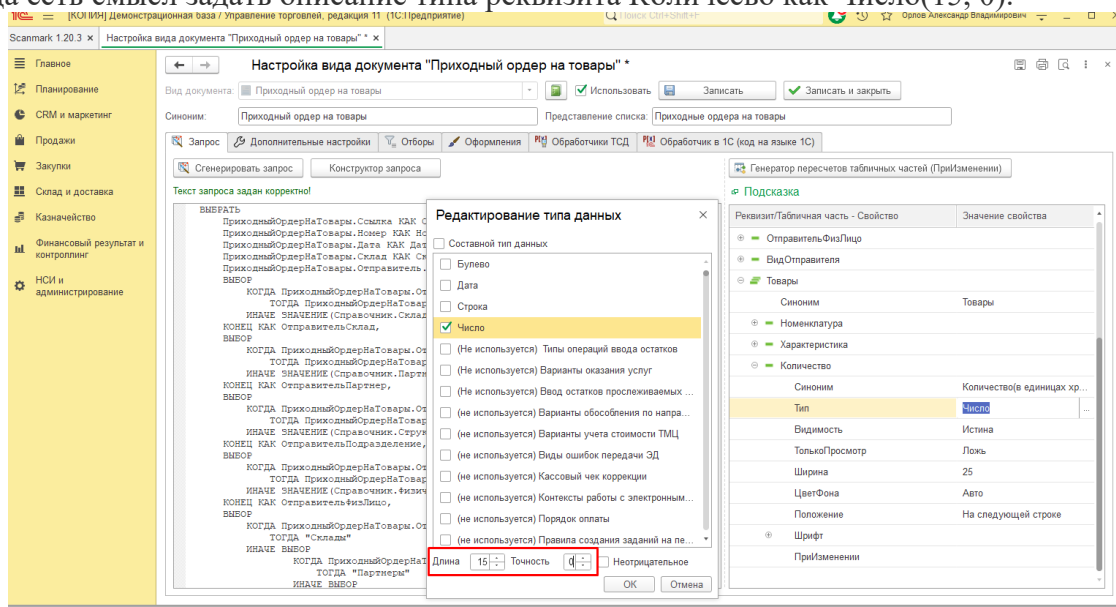
Изменим свойства реквизитов:

- СтрОтправитель: Синоним = «Отправитель»;
- ОтправительСклад: Синоним = «Отправитель (склад)», Тип = Склад (складская территория), Видимость = Ложь;
- ОтправительПартнер: Синоним = Отправитель (партнер), Тип = Партнер, Видимость = Ложь;
- ОтправительПодразделение: Синоним = «Отправитель (подразделение)», Тип = Подразделение, Видимость = Ложь;
- ОтправительФизЛицо: Синоним = «Отправитель (физ.лицо)», Тип = Физическое лицо, Видимость = Ложь;
- ВидОтправителя: Видимость = Ложь.

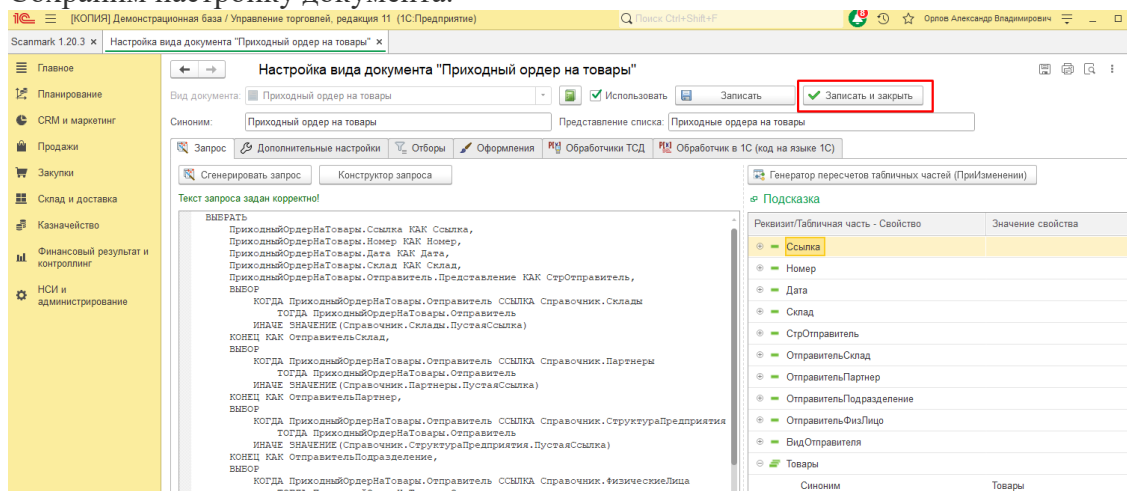




В конфигурации в табличной части товара реквизит Количество описан как Число(15, 3). Если учет ведется только штучного товара, то количество товара – это почти всегда целое число. И тогда есть смысл задать описание типа реквизита Количество как Число(15, 0):

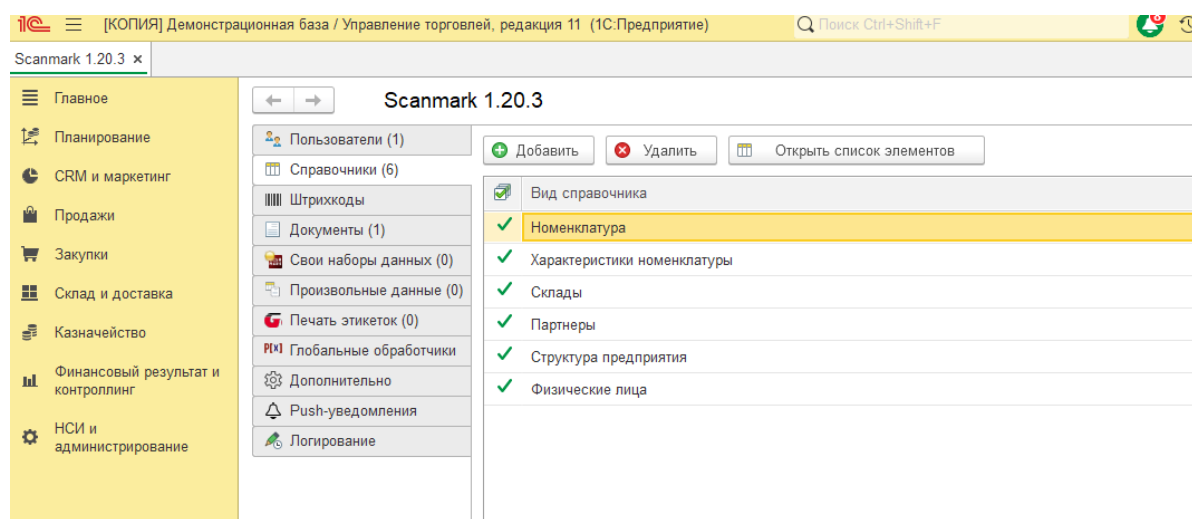
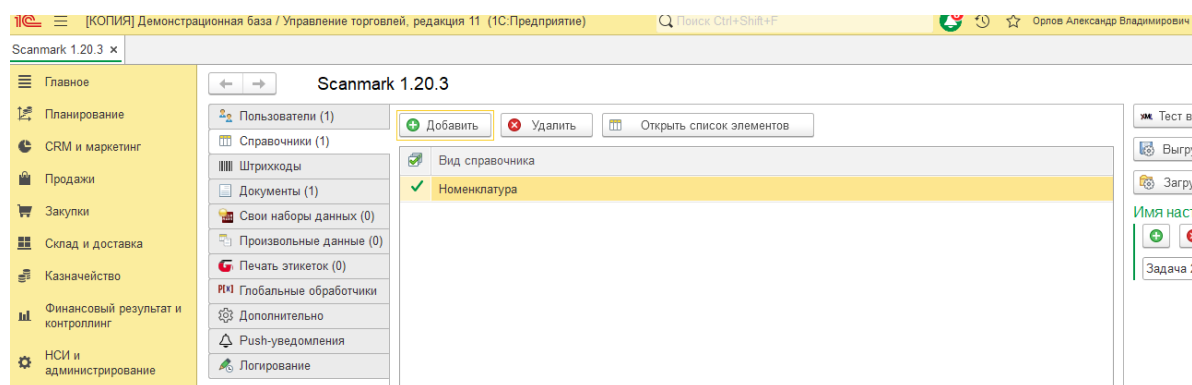
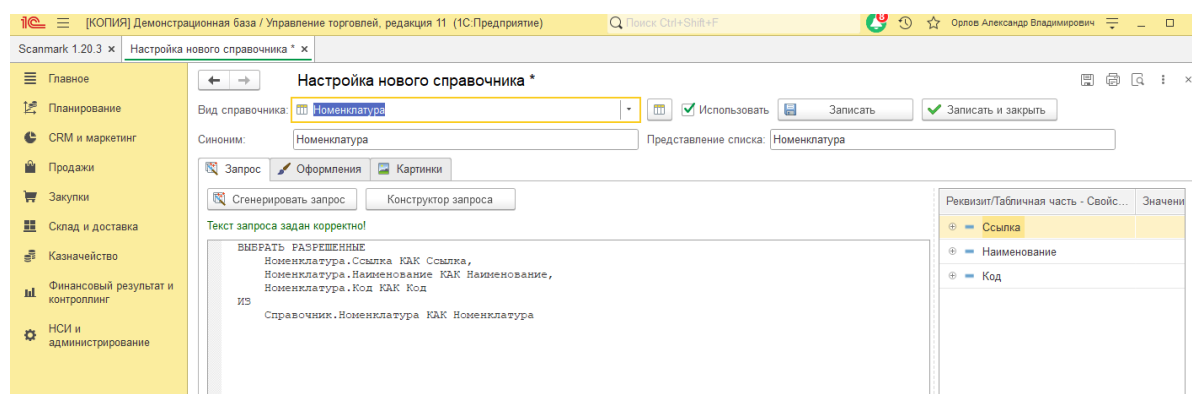
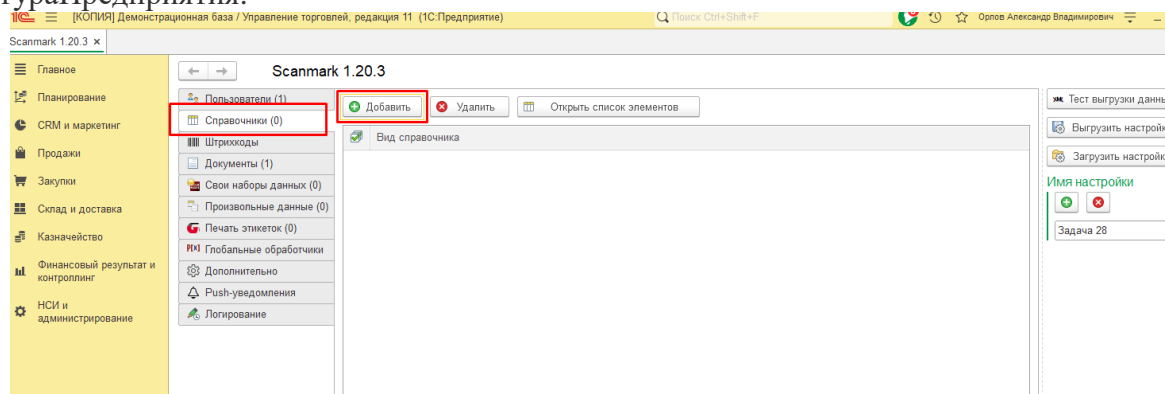


## Сохраним настройку документа.

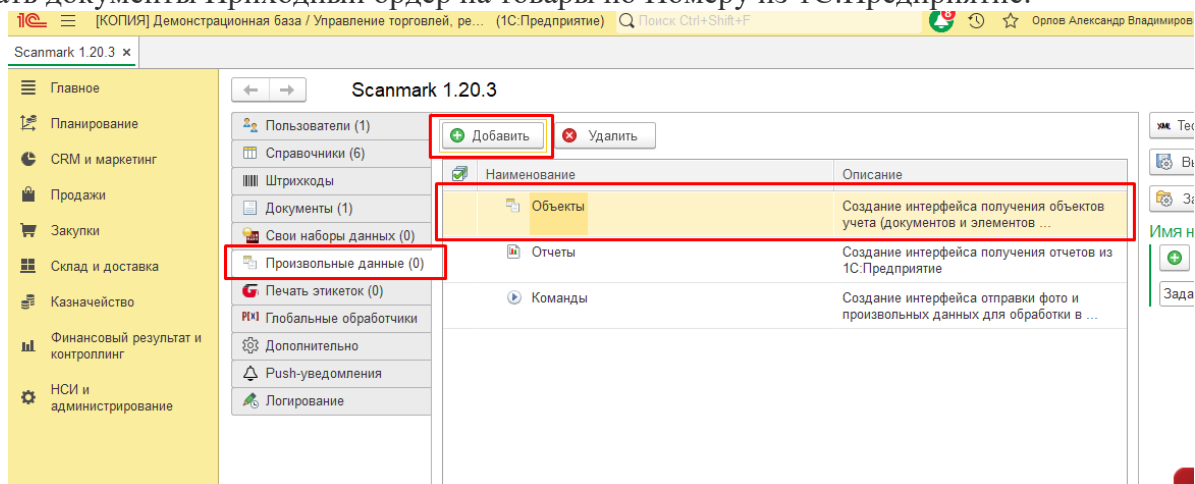




Перейдем на закладку «Справочники» и добавим следующие виды справочников: Номенклатура, ХарактеристикиНоменклатуры, Склады, Партнеры, ФизическиеЛица, СтруктураПредприятия.



На закладке «Произвольные данные» добавим Объект, с помощью которого сможем получать документы Приходный ордер на товары по Номеру из 1С:Предприятие:



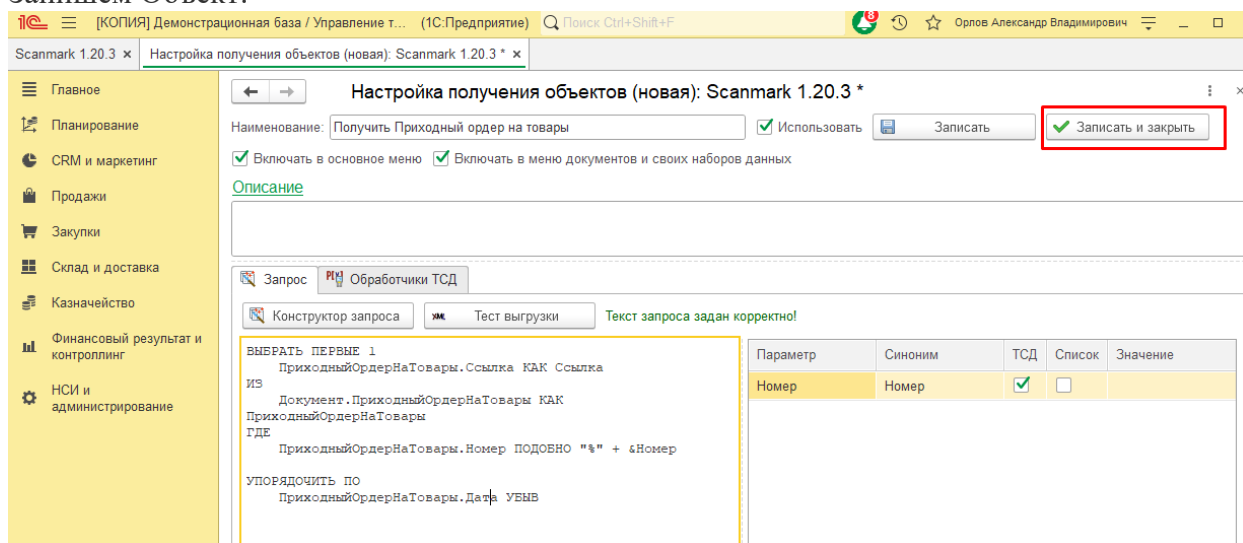
Зададим наименование «Получить Приходный ордер на товары». Добавим текст запроса:

```

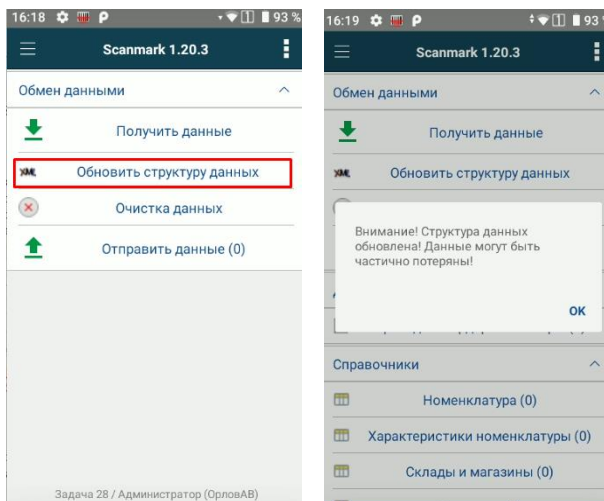
ВЫБРАТЬ ПЕРВЫЕ 1
ПриходныйОрдерНаТовары.Ссылка КАК Ссылка
ИЗ
Документ.ПриходныйОрдерНаТовары КАК ПриходныйОрдерНаТовары
ГДЕ
ПриходныйОрдерНаТовары.Номер ПОДОВОНО "%" + &Номер

УПОРЯДОЧИТЬ ПО
ПриходныйОрдерНаТовары.Дата УБЫВ
    
```

Запишем Объект:

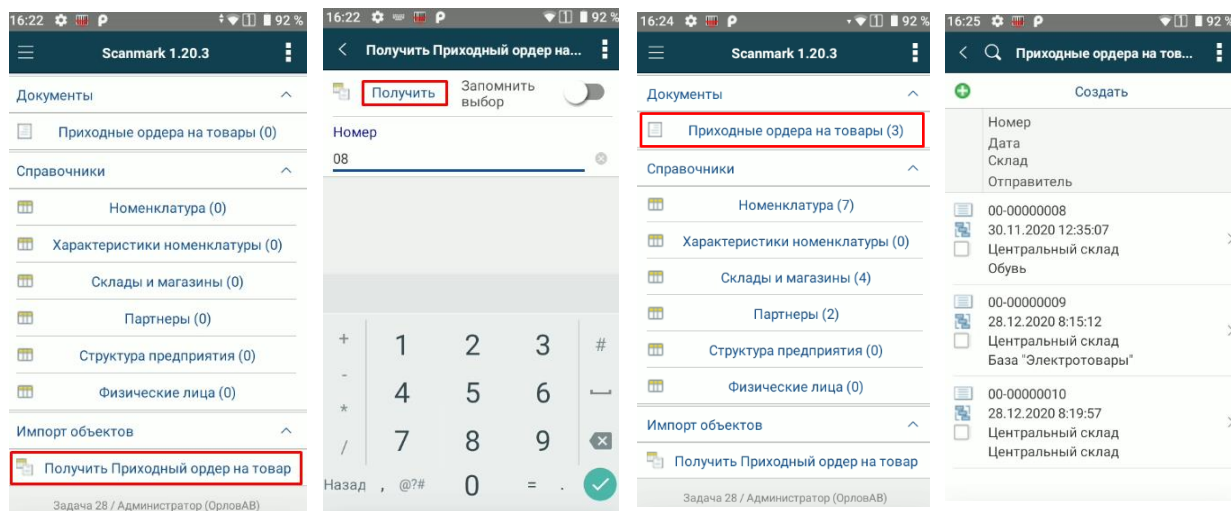


На ТСД загрузим созданную структуру данных:

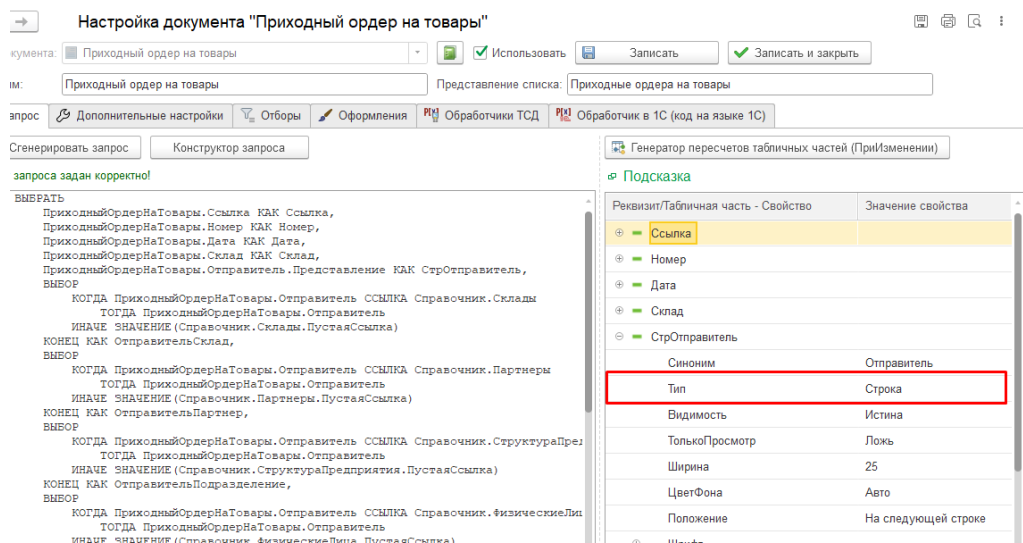
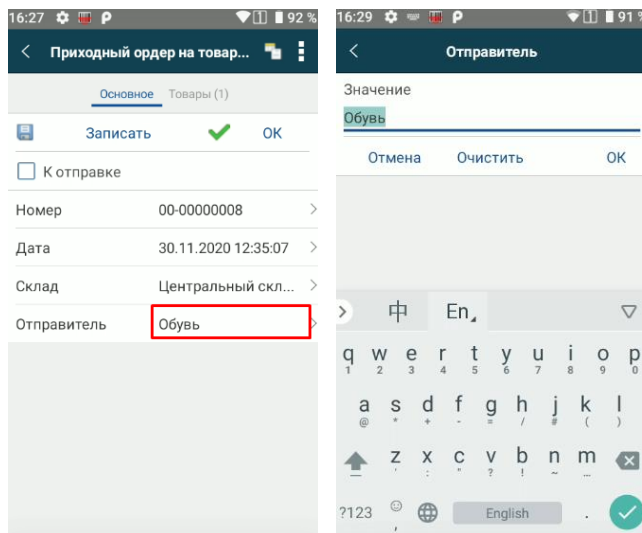


С помощью объекта «Получить Приходный order на товары» загрузим несколько документов из 1С:Предприятие:

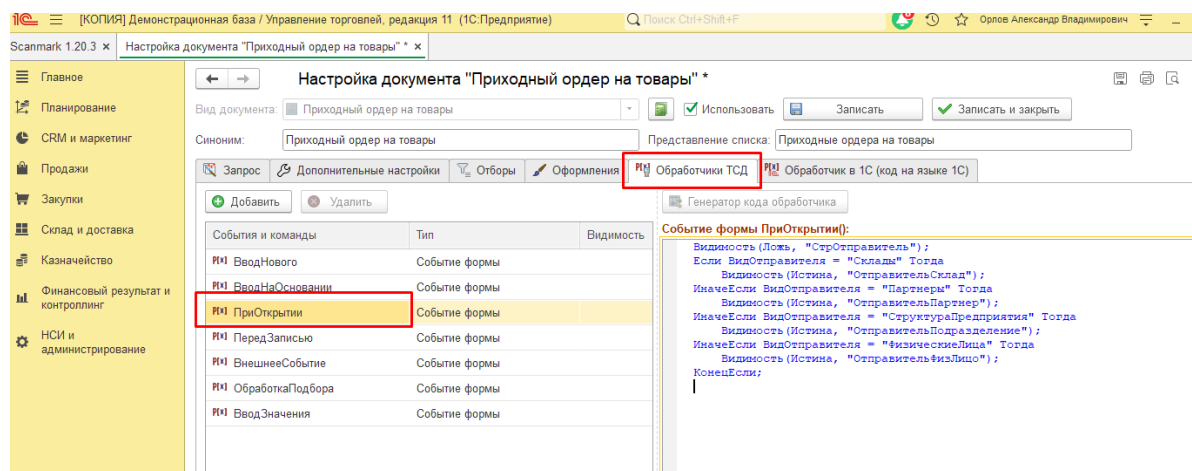
Номер	Дата	Отправитель	Статус	Расписание	Ответственный	Склад
00-00000065	15.06.2017	Электронприбор	Принят	Приобретение товаров и у...	Федоров Борис Михайло...	Склад бытовой
00-00000001	08.06.2018	1С-СОФТ ООО	Принят	Приобретение товаров и у...	Соколов Максим Игоревич	Продуктовая
00-00000002	08.06.2018	1С-СОФТ ООО	Принят	Приобретение товаров и у...	Соколов Максим Игоревич	Продуктовая
00-00000003	08.06.2018	1С-СОФТ ООО	Принят	Приобретение товаров и у...	Соколов Максим Игоревич	Продуктовая
00-00000001	03.06.2020	База "Продукты"	Принят	Заказ поставщику ТД00-0...	Афонин Геннадий Алекс...	Центральный
00-00000002	30.06.2020	База "Электротовары"	Принят	Заказ поставщику ТД00-0...	Афонин Геннадий Алекс...	Центральный
00-00000003	07.09.2020	База "Электротовары"	Принят	Заказ поставщику ТД00-0...	Афонин Геннадий Алекс...	Центральный
00-00000004	07.09.2020	База "Электротовары"	Принят	Заказ поставщику ТД00-0...	Афонин Геннадий Алекс...	Центральный
00-00000005	30.11.2020	Обувь	Принят	Приобретение товаров и у...	Афонин Геннадий Алекс...	Центральный
00-00000006	30.11.2020	Обувь	Принят	Приобретение товаров и у...	Афонин Геннадий Алекс...	Центральный
00-00000007	30.11.2020	Обувь	Принят	Приобретение товаров и у...	Афонин Геннадий Алекс...	Центральный
00-00000008	30.11.2020	Обувь	Принят	Приобретение товаров и у...	Афонин Геннадий Алекс...	Центральный
00-00000009	28.12.2020	База "Электротовары"	Принят	Заказ поставщику ТД00-0...	Афонин Геннадий Алекс...	Центральный
00-00000010	28.12.2020	Центральный склад	Принят	Заказ поставщику ТД00-0...	Афонин Геннадий Алекс...	Центральный



Откроем первый документ и попробуем выбрать значение «Отправитель». Как и было задано в настройках документа «Приходный order на товары» обработки описание типа = «Строка»:



А нам нужен реквизит с выбором из «правильного» справочника. Откроем в настройку документа «Приходный порядок на товары» обработки, перейдем на закладку «Обработчики ТСД», в списке с левой стороны перейдем на обработчик события «ПриОткрытии» и в поле справа зададим код:



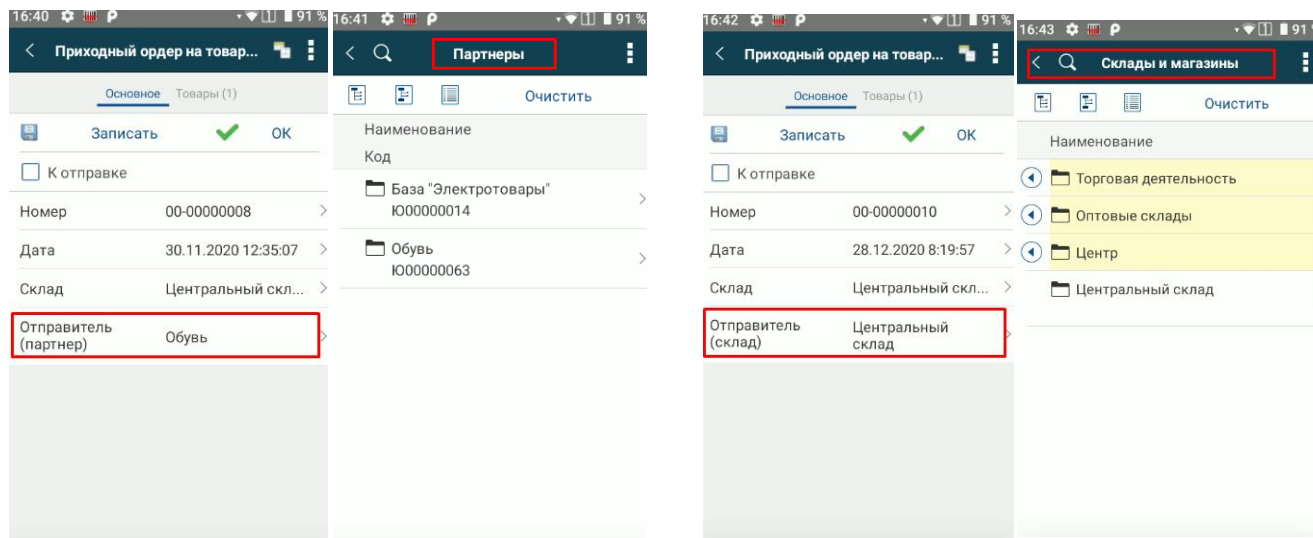
Видимость (Ложь, "СтрОтправитель");  
Если ВидОтправителя = "Склады" Тогда  
Видимость (Истина, "ОтправительСклад");  
ИначеЕсли ВидОтправителя = "Партнеры" Тогда

```

Видимость (Истина, "ОтправительПартнер");
ИначеЕсли ВидОтправителя = "СтруктураПредприятия" Тогда
    Видимость (Истина, "ОтправительПодразделение");
ИначеЕсли ВидОтправителя = "ФизическиеЛица" Тогда
    Видимость (Истина, "ОтправительФизЛицо");
КонецЕсли;

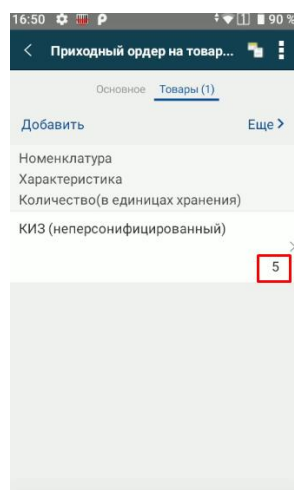
```

Сохраним настройку документа, на ТСД обновим структуру данных. Проверим выбор значений реквизита «Отправитель» из «корректного» справочника:



Мы получили желаемый результат!

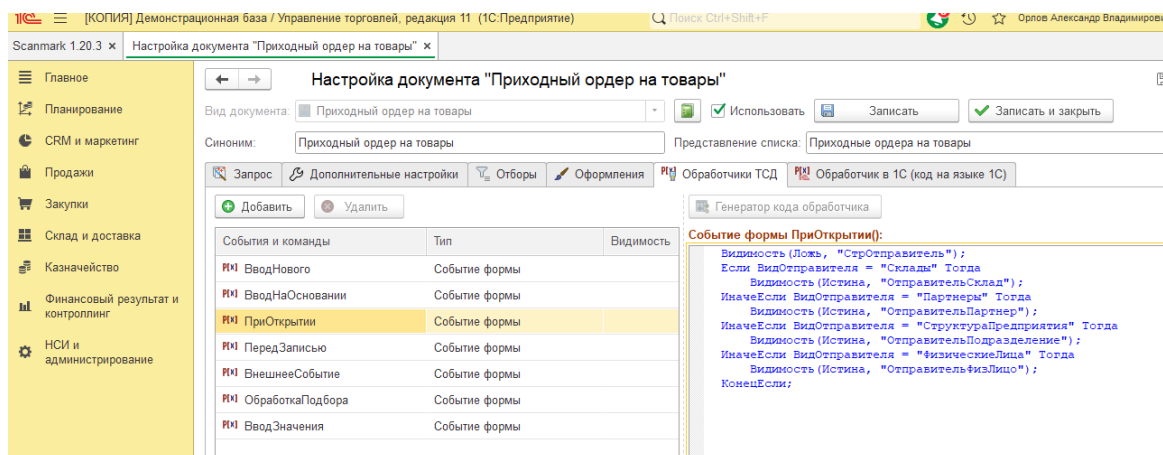
Также проверим представление реквизита «Количество» табличной части «Товары», у которого мы изменили описание типа данных на Число(15,0), т.е. ожидаем увидеть значения без дробной части:



## Задача 29: Пример управления отображением элементов на форме на ТСД.

В этой задаче за основу используем настройку, созданную в предыдущей задаче - «Задача 28: Пример работы с составным типом данных, преобразование типа данных».

В Задаче 28 уже был использован метод управления отображением данных Видимость (<Видимость>, <ИмяРеквизита>, <ИмяТабличнойЧасти>):

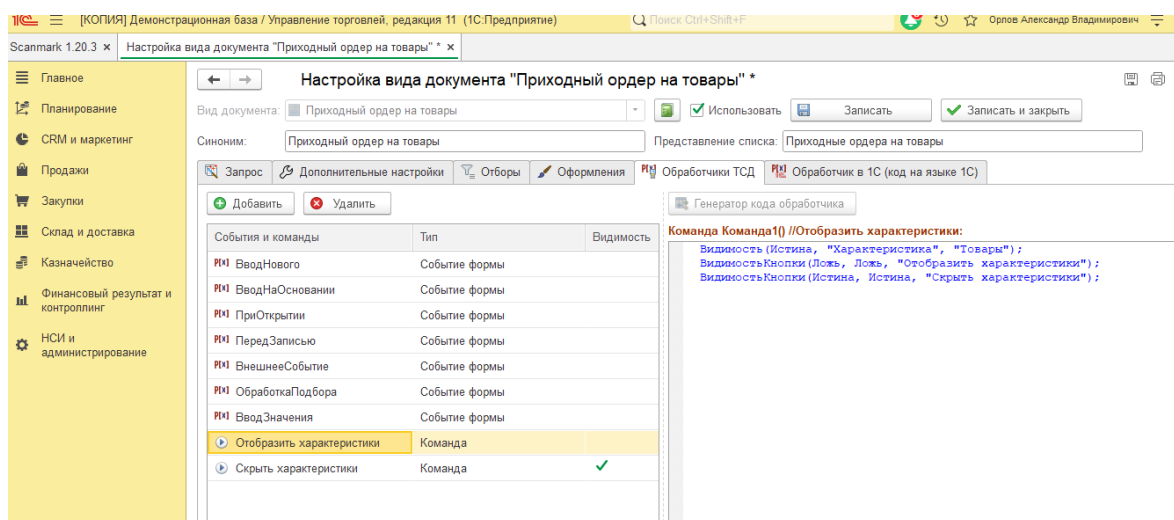


В обработке в настройке документа «Приходный ордер на товары» на закладке «Обработчики ТСД» добавим команды «Отобразить характеристики» и «Скрыть характеристики». Для команды «Отобразить характеристики» снимем флаг «Видимость». Для команды «Отобразить характеристики» напомним код:

```
Видимость(Истина, "Характеристика", "Товары");  
ВидимостьКнопки(Ложь, Ложь, "Отобразить характеристики");  
ВидимостьКнопки(Истина, Истина, "Скрыть характеристики");
```

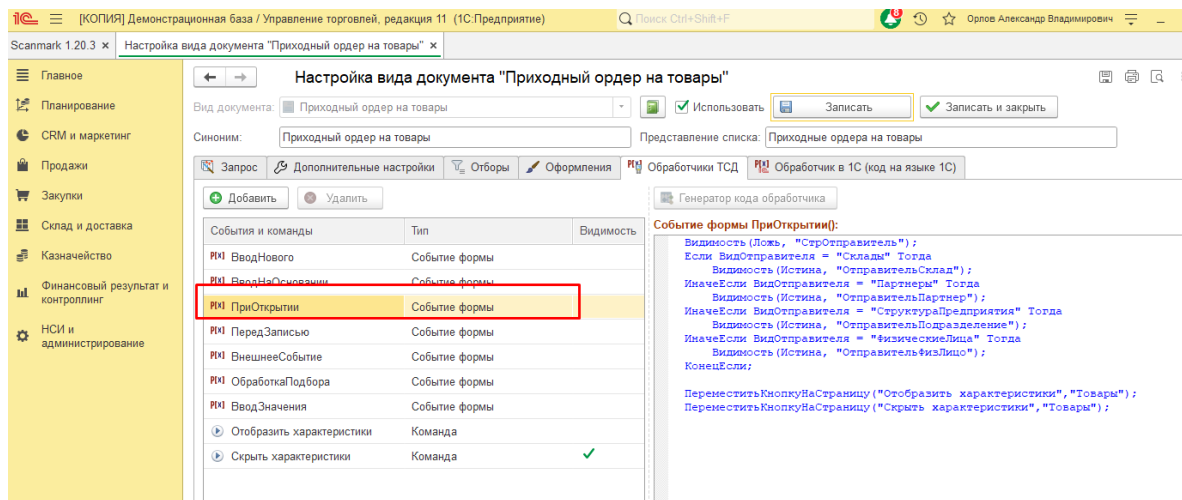
Для команды «Скрыть характеристики» напомним код:

```
Видимость(Ложь, "Характеристика", "Товары");  
ВидимостьКнопки(Истина, Истина, "Отобразить характеристики");  
ВидимостьКнопки(Ложь, Ложь, "Скрыть характеристики");
```

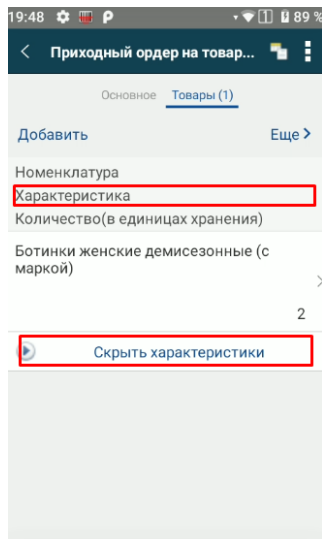


На ТСД в форме документа «Приходный ордер на товары» созданные элементы-кнопки по умолчанию будут отображены на странице «Основное». Для пользователя удобнее будет, если эти кнопки будут расположены на странице «Товары», т.к. поле «Характеристика», видимостью которого мы хотим управлять находится именно на этой странице. Чтобы элементы-кнопки формы были расположены на странице «Товары» в событии «ПриОткрытии»

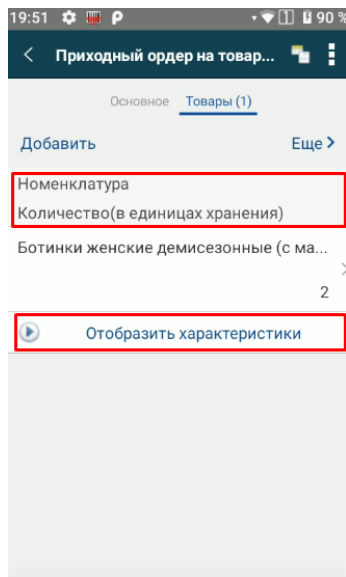
```
ПереместитьКнопкуНаСтраницу("Отобразить характеристики", "Товары");
ПереместитьКнопкуНаСтраницу("Скрыть характеристики", "Товары");
```



Запишем настройку. На ТСД обновим структуру данных. Проверим новый функционал: Откроем любой документ «Приходный ордер на товары», перейдем на закладку «Товары»:



Нажмем на «Скрыть характеристики». Результат – скрылась колонка «Характеристика», скрылась кнопка «Скрыть характеристики», появилась кнопка «Отобразить характеристики»:



Если нажать на «Отобразить характеристики», то видимость элементов формы вернется в исходное состояние.



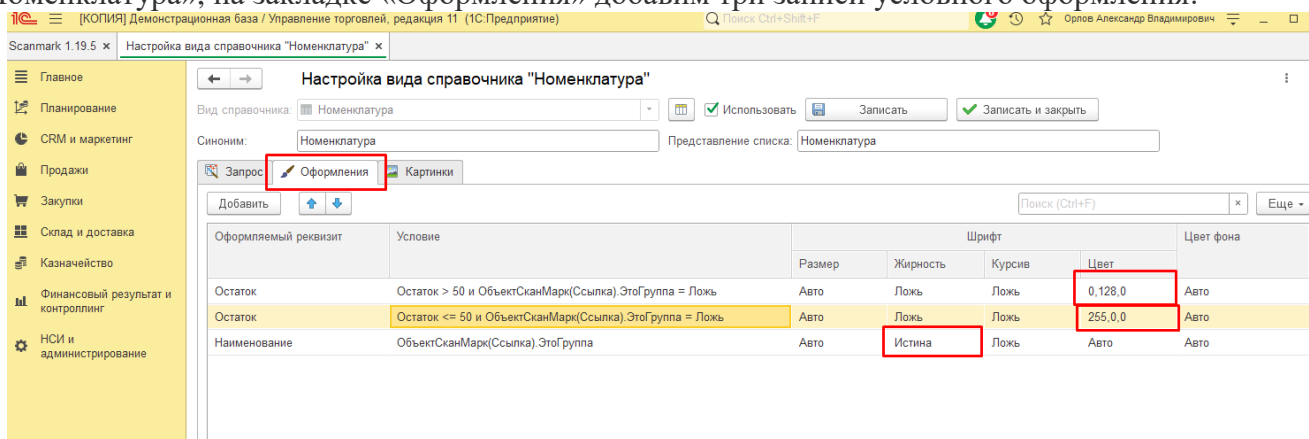
# Условное оформление

ПО «Scanmark» поддерживает условное оформление для списков документов и справочников, а также форм объектов справочников, документов, команд, печати этикеток и своих наборов данных. Настройка условного оформления осуществляется на закладках «Оформление» объектов в настройках обработки «Scanmark».

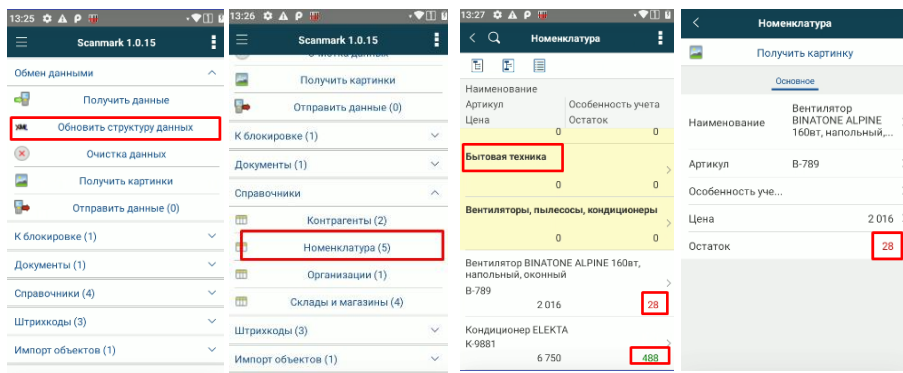
При добавлении условных оформлений в настройках объектов следует обратить внимание, что применение условных оформлений связано с расчетом условий. Это занимает значительные ресурсы, особенно в табличных частях форм объектов с большим количеством записей, что может быть причиной значительных задержек.

## Пример условного оформления справочника «Номенклатура».

Для реализации примера в 1С:Предприятие за основу возьмём задачу 8: На ТСД на основании документа «Заказ клиента» оформить, подобрать коды маркировки DataMatrix GS1 и отправить документ «Реализация товаров и услуг». Откроем настройку справочника «Номенклатура», на закладке «Оформления» добавим три записи условного оформления:

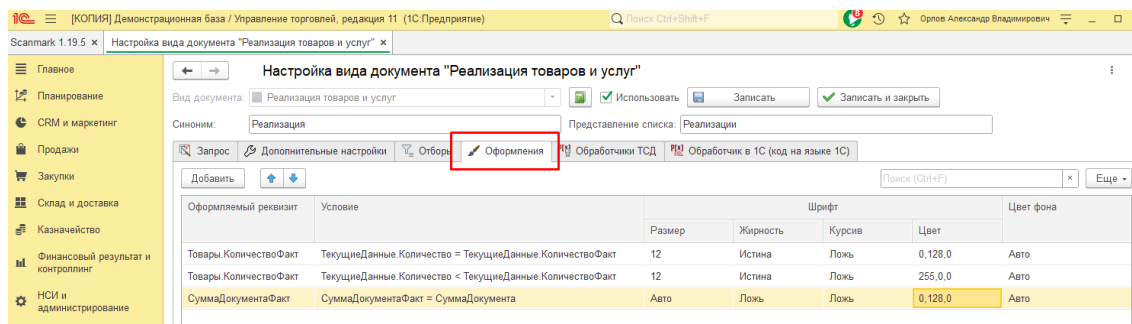


Обновим структуру данных в приложении «Scanmark» и проверим, как теперь выглядит условное оформление в формах списка и элемента справочника «Номенклатура»:



## Пример условного оформления документа «Реализация».

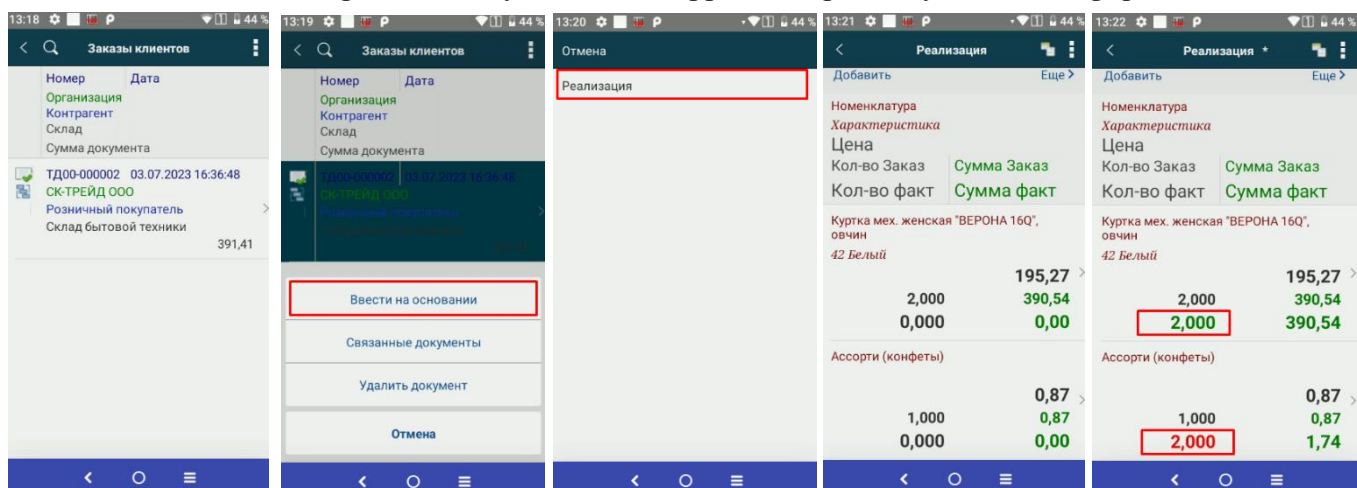
В 1С:Предприятие откроем настройку документа «Реализация». На закладке «Оформления» добавим записи условного оформления:



Обратите внимание на код в колонке «Условие» табличной части «Оформления» - обращение к реквизиту табличной части осуществляется по формату «ТекущиеДанные.<ИмяРеквизита>», где «ТекущиеДанные» - оператор обращения к записи табличной части оформляемого реквизита, <ИмяРеквизита> - имя оформляемого реквизита табличной части.

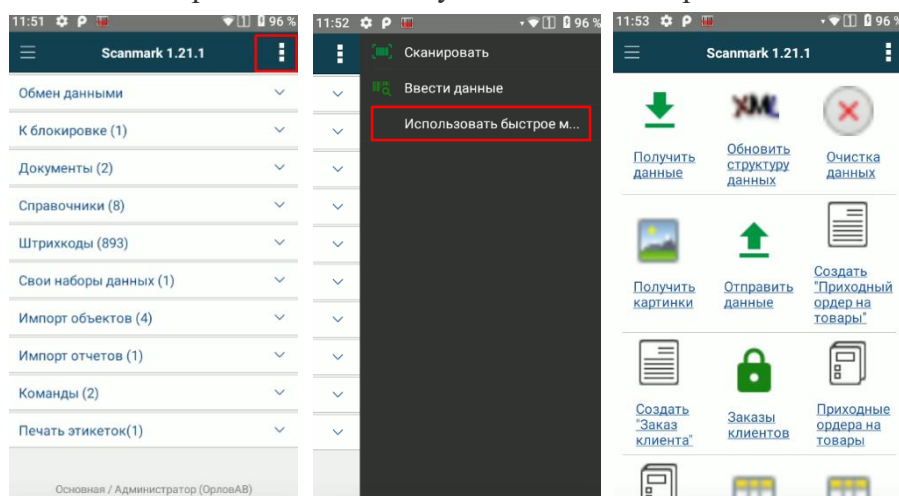
Запишем настройку документа. На ТСД обновим структуру.

На ТСД на основании документа «Заказ клиента» создадим новый документ «Реализация». В новом документе «Реализация» пробуем изменять значения «Кол-во факт» в табличной части «Товары», чтобы убедиться в корректной работе условного оформления:



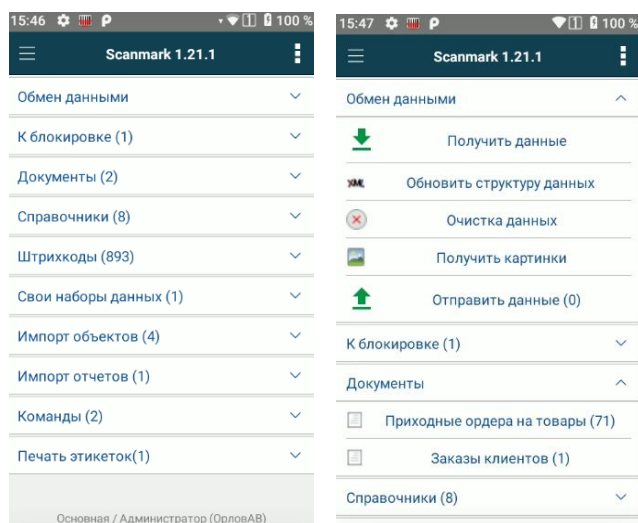
# Настройка основного меню

Приложение «Scanmark» поддерживает использование двух видов меню основного окна: обычное и быстрое. В зависимости от личных предпочтений пользователь выбирает вид меню. Также возможно переключение между видами меню через контекстное меню:

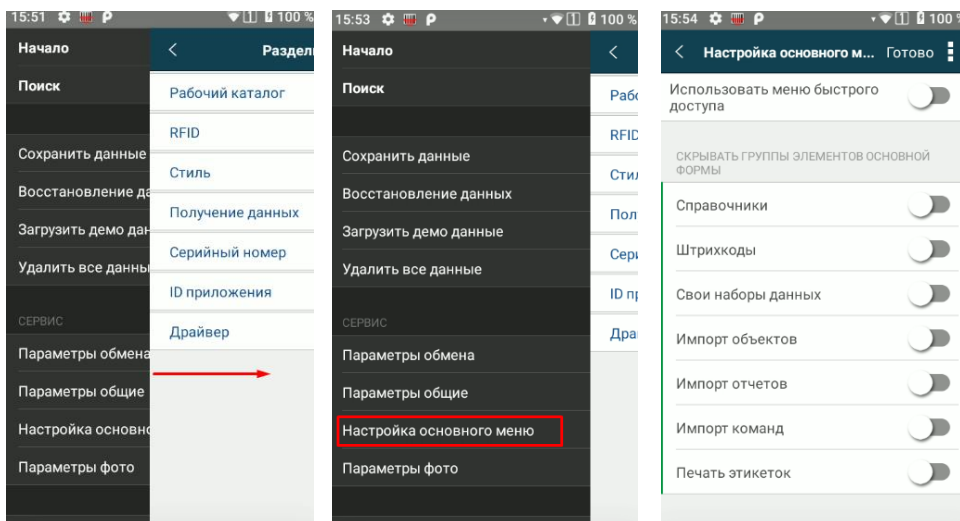


## Обычное меню (по умолчанию).

Этот вид меню представляет из себя иерархичный список, сгруппированный по типам элементов:

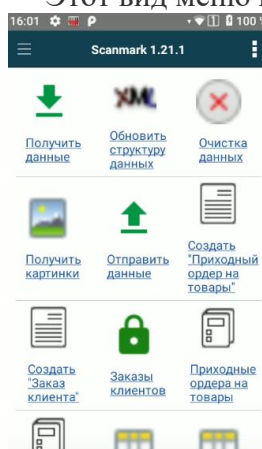


Настройка обычного меню позволяет скрыть\отобразить видимость типов групп элементов:



## Быстрое меню.

Этот вид меню представляет из себя «плиточный» тип без иерархии и группировок:

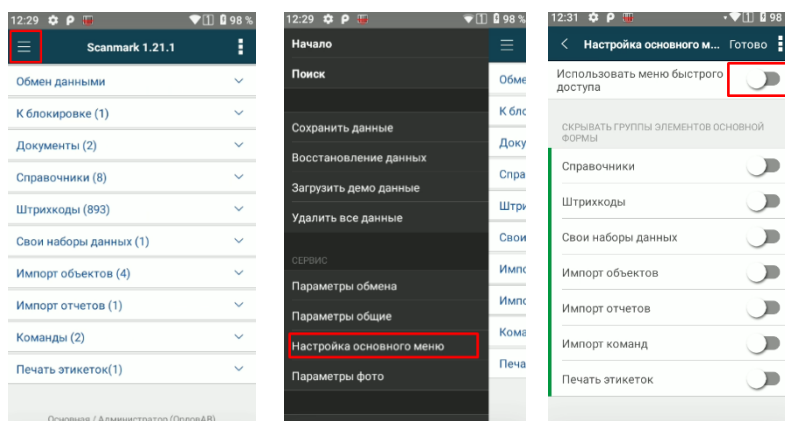


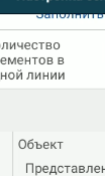
Настройки быстрого меню для элементов приложения позволяют установить:



- количество элементов в одной линии;
- видимость;
- надпись;
- порядок.


Кроме того, в быстром меню, в отличие от обычного меню, доступны команды создания новых документов.

Откройте настройку основного меню:










12:36   99%

**Настройка основного м...** **Готово** 

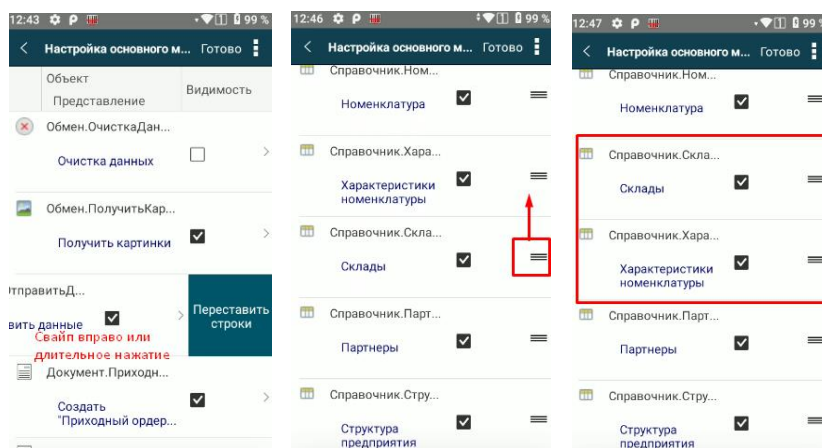
Заполнить по умолчанию

Количество элементов в одной линии 3  

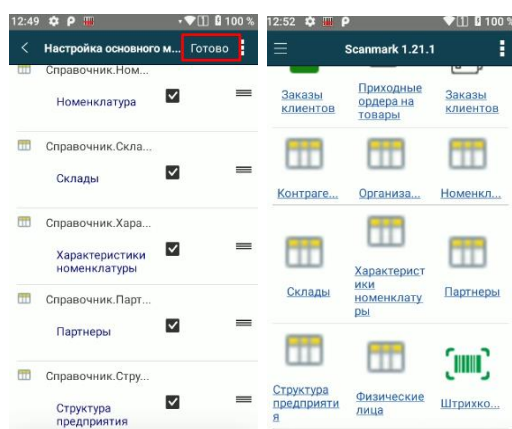
[Еще >](#)

Объект	Представление	Видимость
 Обмен.ОбновитьСтр...	Обновить структуру данных	<input checked="" type="checkbox"/>
 Обмен.ОчисткаДан...	Очистка данных	<input type="checkbox"/>
 Обмен.ПолучитьКар...	Получить карточки	<input checked="" type="checkbox"/>

Задайте порядок:



Запишите и примените сделанные изменения:



# Особенности объектов в «Scanmark»

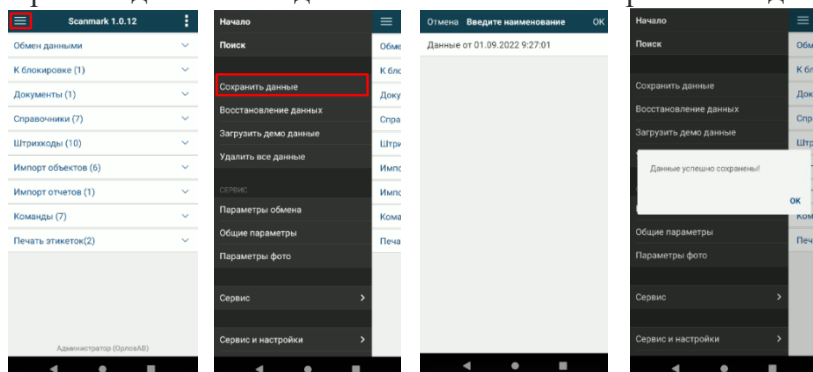
---

Объекты данных для приложения «Scanmark» создаются при помощи обработки «Scanmark» из состава ПО «Scanmark». При кажущейся схожести описанных объектов данных «Scanmark» на объекты метаданных в конфигураторе 1С 8, свойства объектов, созданных таким образом, не содержат привычное «дерево» реквизитов и табличных частей. Чтобы получить данные свойств объекта в приложении «Scanmark» необходимо использовать функцию ОбъектСканмарк (см. раздел Процедуры и функции обработчиков «Scanmark»).

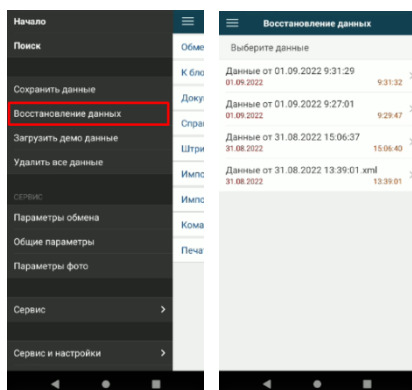
# Сохранение и восстановление данных в приложении «Scanmark»

В мобильном приложении «Scanmark» есть возможность сохранить состояние текущей базы, а потом в случае необходимости восстановить сохраненные данные.

Для сохранения данных в главном окне нужно вызвать основное меню и выбрать команду «Сохранить данные». Задайте наименование сохраняемых данных и нажмите ОК.

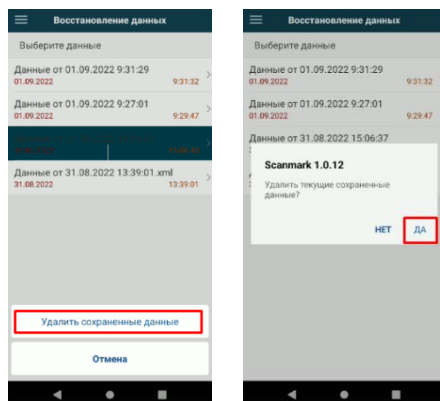


Если по каким то причинам необходимо восстановить сохраненные данные, то в главном окне нужно вызвать основное меню и выбрать команду «Восстановление данных». Выберите требуемые сохраненные данные.



**ВНИМАНИЕ!** При восстановлении данных текущие данные будут полностью удалены.

Удалить сохраненные возможно через контекстное меню (долгое нажатие на строке таблицы с сохраненными данными) и выберите «Удалить текущие сохраненные данные» после чего подтвердите удаление.



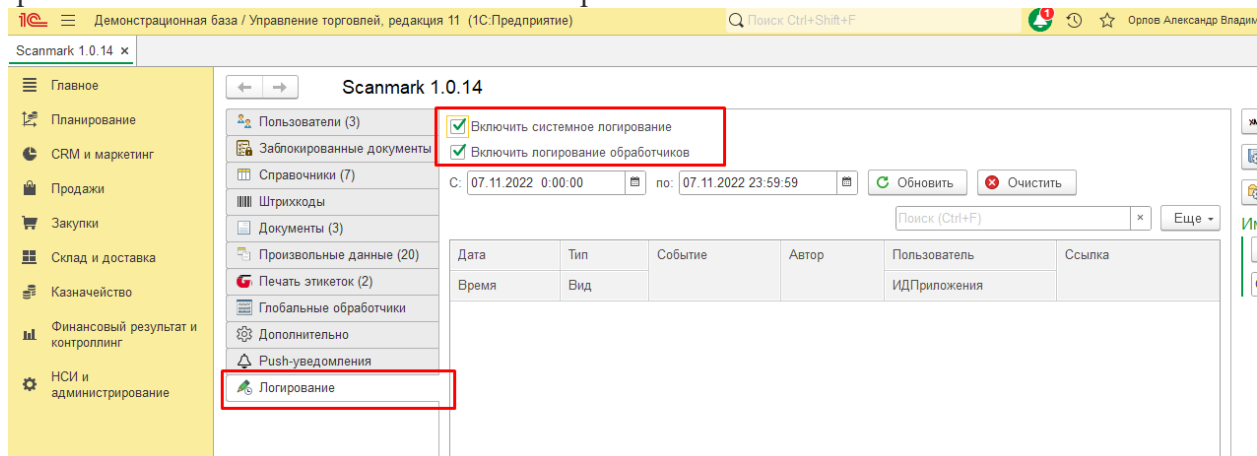


# Логирование

В ПО «Scanmark» реализована возможность ведения лога событий как в системе 1С:Предприятие так и на стороне мобильного приложения.

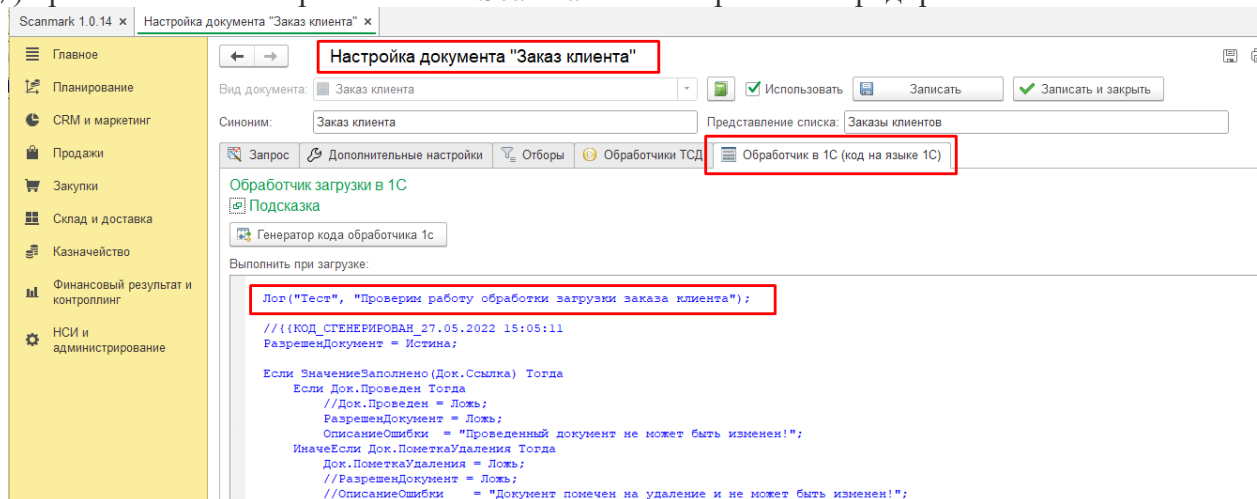
## Логирование на стороне 1С:Предприятие.

В 1С:Предприятие в основном окне обработки «Scanmark» откройте закладку «Логирование». Включите или отключите логирование.



Установите флажок «Включить системное логирование» для автоматической записи событий обмена с приложениями «Scanmark».

Установите флажок «Включить логирование обработчиков» для записи событий (метод `Лог()`) при выполнении обработчиков «Scanmark» на стороне 1С:Предприятие:



Описание метода `Лог()` смотрите в разделе Процедуры и функции обработчиков «Scanmark».

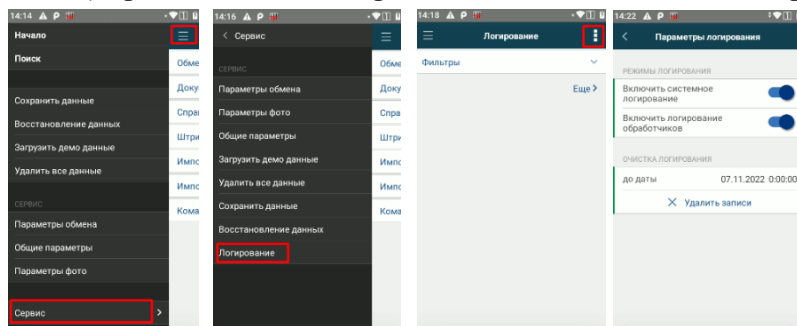
## Логирование на стороне мобильного приложения.

В приложении «Scanmark» в основном меню откройте окно «Логирование». В окне «Логирование» откройте «Параметры». Включите или отключите логирование.

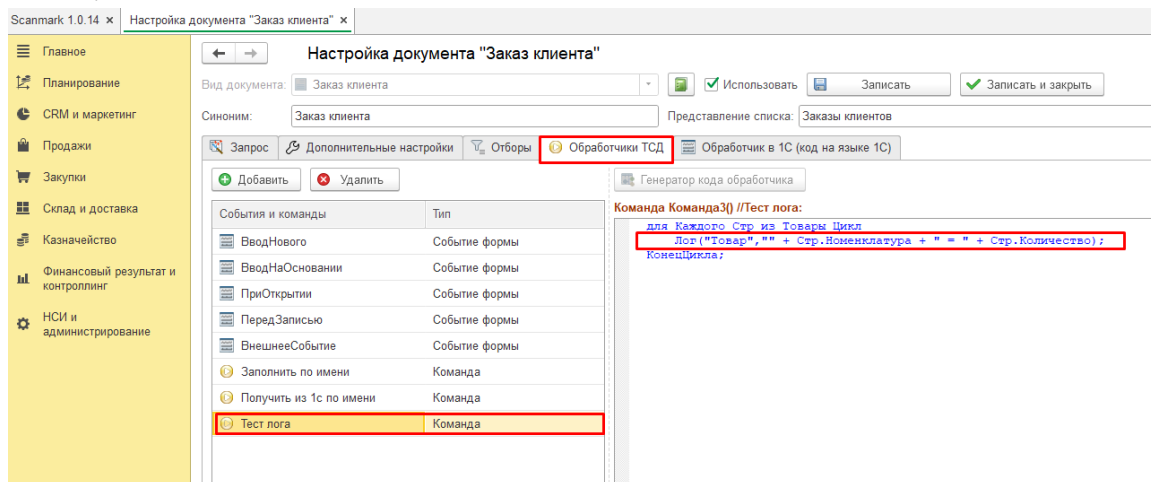
Установите флажок «Включить системное логирование» для автоматической записи событий приложения «Scanmark» и обмена с 1С:Предприятие.

Установите флажок «Включить логирование обработчиков» для записи событий (метод

Log()) при выполнении обработчиков «Scanmark» на стороне мобильного приложения.



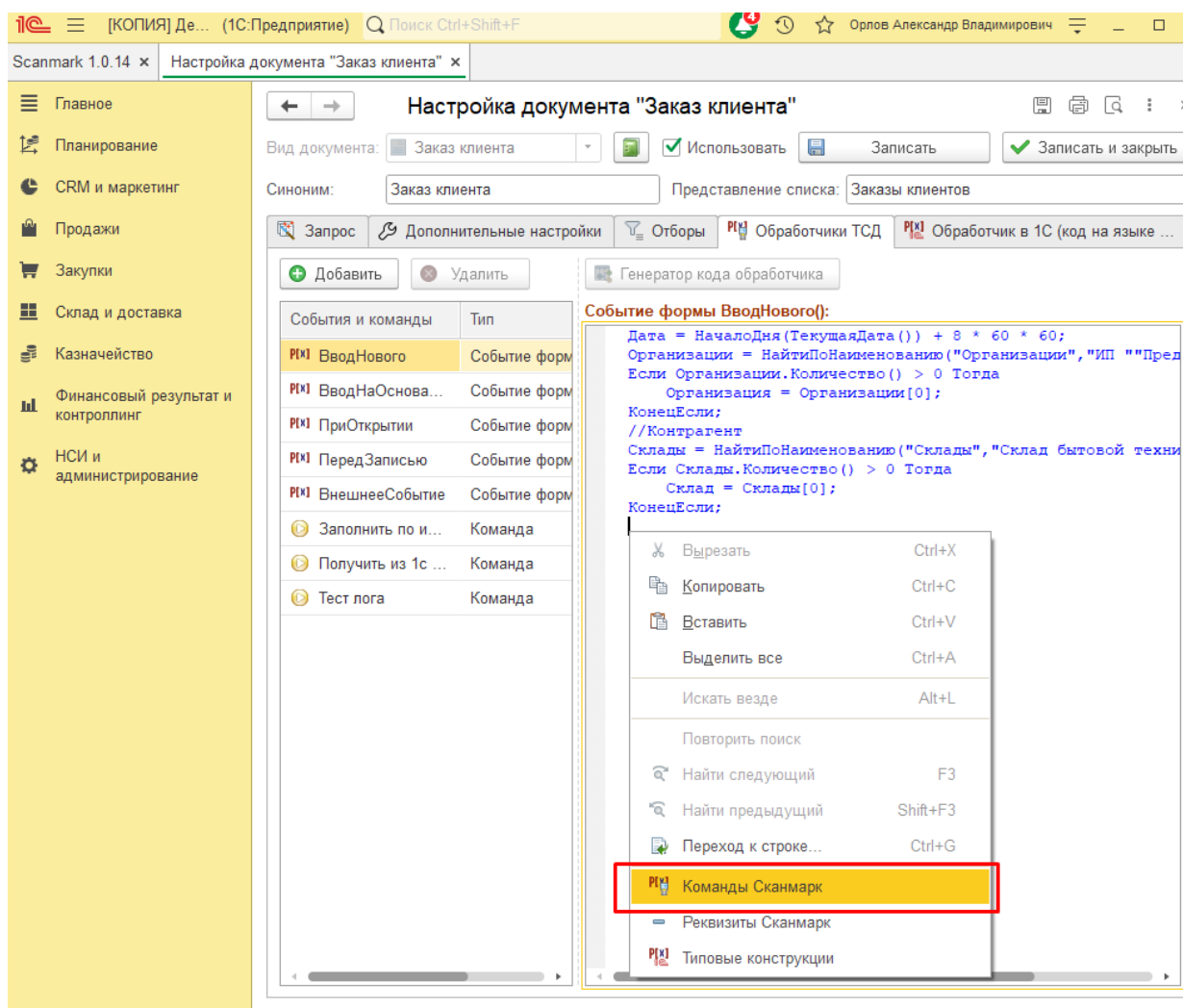
Описание метода Log() смотрите в разделе Процедуры и функции обработчиков «Scanmark».

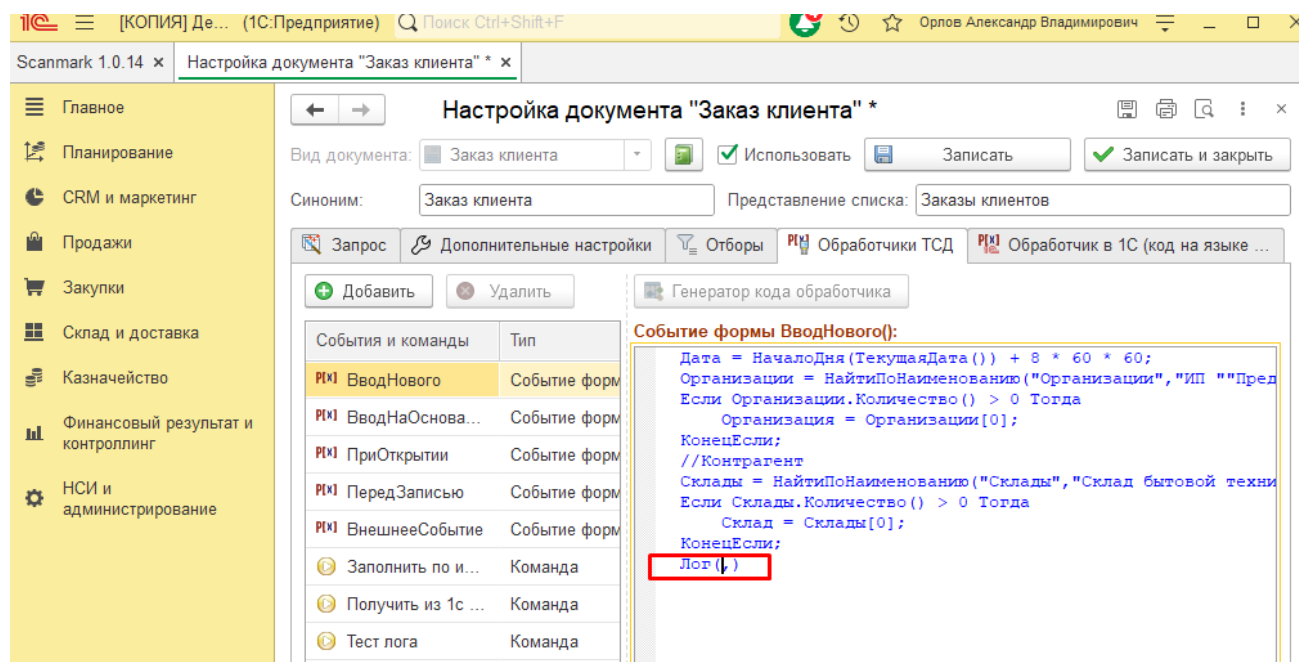
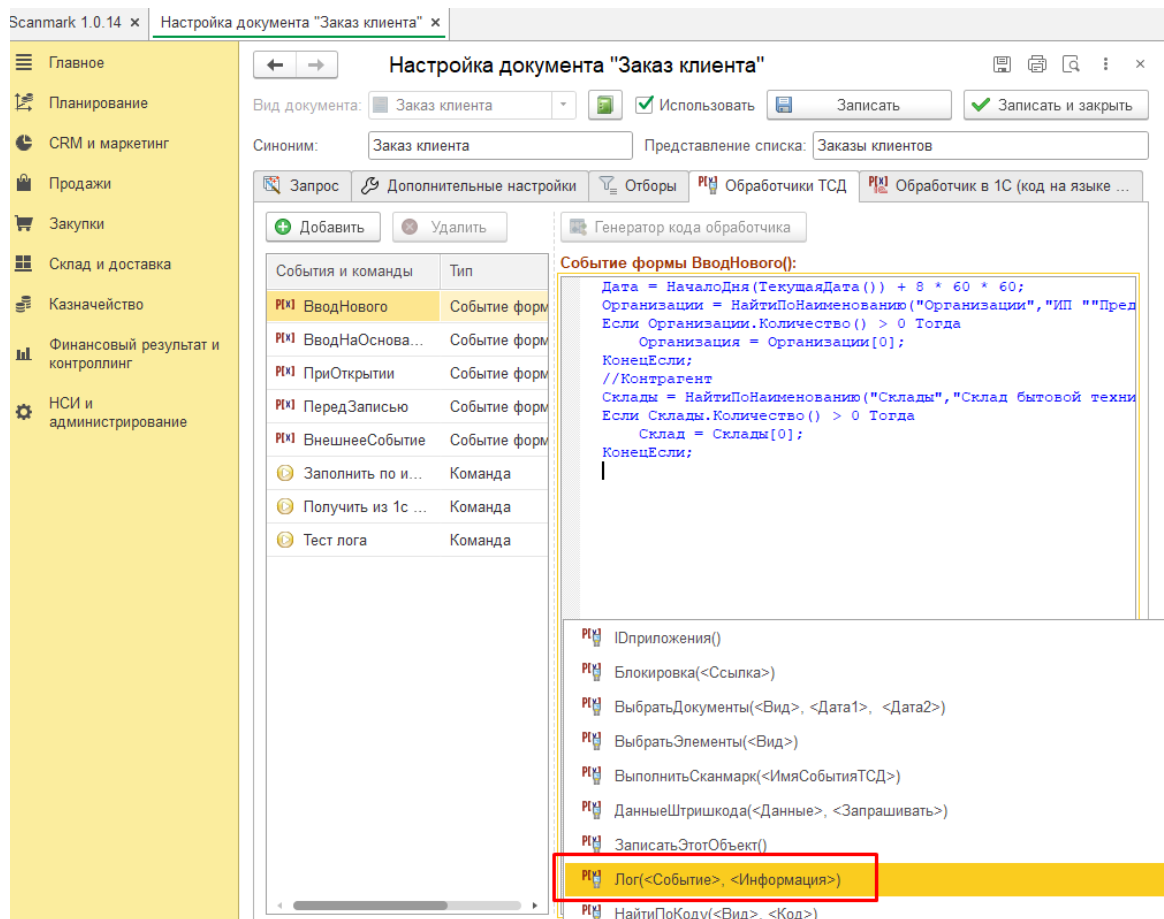


# Процедуры и функции обработчиков «Scanmark»

## В помощь при настройке

Вставка в редакторах обработчиков процедур и функции «Scanmark» доступны через контекстное меню. Кроме этого, можно воспользоваться вставкой через контекстное меню реквизитов дерева формы и некоторые типовые конструкции (например, организация цикла Пока <условие> Цикл <операторы> КонецЦикла;).





## IDприложения

**Синтаксис:**

IDприложения()

**Параметры:**

Нет

**Возвращаемое значение:**

Тип: [Строка](#).

**Описание:**

Возвращает ID приложения. Если приложение имеет доступ к получению серийного номера ТСД, то ID приложения равен этому серийному номеру. Иначе будет сгенерирован уникальный идентификатор.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## PushУведомление

**Синтаксис:**

PushУведомление(<Текст>, <Заголовок>)

**Параметры:**

<Текст> (обязательный)

Тип: [Строка](#).

<Заголовок> (необязательный)

Тип: [Строка](#).

Значение по умолчанию [<пустая строка>](#).

**Описание:**

Отправляет уведомление пользователю, от имени которого выполняется данная процедура.

**Доступность:**

Обработчик в 1С (код на языке 1С).

## Блокировка

**Синтаксис:**

Блокировка(<Ссылка>)

**Параметры:**

<Ссылка> (обязательный)

Тип: [ДокументСсылка](#).

**Возвращаемое значение:**

Тип: [Структура](#).

Структура с параметрами:

    Блокирован (тип [Булево](#));

    Пользователь (тип [Строка](#));

    IDПриложения (тип [Строка](#));

    ДатаБлокировки (тип [Дата](#)).

**Описание:**

Возвращает структуру значений с информацией о блокировке документа.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

ВвестиЗначение

**Синтаксис:**

ВвестиЗначение(<ПараметрыЗначения>, <ДополнительныеПараметры>)

**Параметры:**

<ПараметрыЗначения> (обязательный)

Тип: Структура.

<ДополнительныеПараметры> (необязательный)

Тип: Структура, Неопределено.

Значение по умолчанию Неопределено.

**Возвращаемое значение:**

Нет.

**Описание:**

Вызов диалога для ввода значения определенного типа, заданного в ПараметрыЗначения. ПараметрыЗначения имеет предопределенный состав ключей, состав которых зависит от типа вводимого значения. Предопределенный метод формы ВводЗначения (<Результат>, <Данные>) служит для обработки выбранного значения.

**Описание структуры «ПараметрыЗначения»:**

**Вопрос (Булево)**

Ключи и описание значений:

Ключ: "Тип", Значение: "Булево";

Ключ: "Подсказка", Значение: произвольная строка, текст вопроса.

**Строка**

Ключи и описание значений:

Ключ: "Тип", Значение: "Строка";

Ключ: "Значение", Значение: произвольная строка (значение по умолчанию);

Ключ: "Подсказка", Значение: произвольная строка, заголовок диалогового окна.

**Число**

Ключи и описание значений:

Ключ: "Тип", Значение: "Число";

Ключ: "Значение", Значение: произвольное число (значение по умолчанию);

Ключ: "Подсказка", Значение: произвольная строка, заголовок диалогового;

Ключ: "Длина", Значение: длина вводимого числа (целое число);

Ключ: "Точность", Значение: число знаков после десятичной точки (целое число).

**Дата**

Ключи и описание значений:

Ключ: "Тип", Значение: "Дата";

Ключ: "Значение", Значение: произвольная дата (значение по умолчанию);

Ключ: "Подсказка", Значение: произвольная строка, заголовок диалогового окна;

Ключ: "Время", Значение: включать выбор времени (Булево).

**Список значений**

Ключи и описание значений:

Ключ: "Тип", Значение: "СписокЗначений";

Ключ: "Значение", Значение: коллекция значений (СписокЗначений).

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## Видимость

**Синтаксис:**

Видимость(<Видимость>, <ИмяРеквизита>, <ИмяТабличнойЧасти>)

**Параметры:**

<Видимость> (обязательный)

Тип: Булево.

<ИмяРеквизита> (обязательный)

Тип: [Строка](#).

<ИмяТабличнойчасти> (необязательный)

Тип: [Строка](#), [Неопределено](#).

Значение по умолчанию [Неопределено](#).

**Возвращаемое значение:**

Нет.

**Описание:**

Устанавливает видимость элемента диалога в форме ТСД.

[Истина](#) - поле отображается, [Ложь](#) - поле не отображается.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Команда), настройках Свои наборы данных, настройках Печать этикеток.

## ВидимостьКнопки

**Синтаксис:**

ВидимостьКнопки (<ВидимостьНаФорме>, <ВидимостьВДополнительномМеню>, <ЗаголовокКнопки>)

**Параметры:**

<ВидимостьНаФорме> (обязательный)

Тип: [Булево](#).

<ВидимостьВДополнительномМеню> (обязательный)

Тип: [Булево](#).

<ЗаголовокКнопки> (обязательный)

Тип: [Строка](#).

**Возвращаемое значение:**

Нет.

**Описание:**

Устанавливает видимость Кнопки (Команды) в форме и дополнительном меню ТСД.

[Истина](#) - кнопка отображается, [Ложь](#) - кнопка не отображается.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Команда), настройках Свои наборы данных, настройках Печать этикеток.

## ВыбратьДокументы

**Синтаксис:**

ВыбратьДокументы(<Вид>, <Дата1>, <Дата2>)

**Параметры:**

<Вид> (обязательный)

Тип: [Строка](#).

<Дата1> (необязательный)

Тип: [Дата](#), [Неопределено](#).

Значение по умолчанию [Неопределено](#).

<Дата2> (необязательный)

Тип: [Дата](#), [Неопределено](#).

Значение по умолчанию [Неопределено](#).

**Возвращаемое значение:**

Тип: [Массив](#).

Создает массив, содержащий все ссылки на документы вида <Вид> за период с <Дата1> по <Дата2>.

Если параметр <Дата1> опущен, то выборка начинается с самого первого существующего в системе документа.

Если параметр <Дата2> опущен, то выборка заканчивается самым последним существующим в системе документом.

**Описание:**

Возвращает массив ссылок на документы заданного вида.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## ВыбратьЭлементы

**Синтаксис:**

ВыбратьЭлементы(<Вид>)

**Параметры:**

<Вид> (обязательный)

Тип: [Строка](#).

**Возвращаемое значение:**

Тип: [Массив](#).

Создает массив, содержащий все ссылки на элементы справочника вида <Вид>.

**Описание:**

Возвращает массив ссылок на элементы справочника заданного вида.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## ВыполнитьСканмарк

**Синтаксис:**

ВыполнитьСканмарк(<ИмяСобытияТСД>)

**Параметры:**

<ИмяСобытияТСД> (обязательный)

Тип: [Строка](#).

**Возвращаемое значение:**

Нет.

**Описание:**

Позволяет использовать код обработчика ТСД события или команды, заданного в этом же объекте, т.е. метод «ВыполнитьСканмарк» в коде обработчика ТСД заменяет код, взятый из кода обработчика ТСД (<ИмяСобытияТСД>).

Пример использования см. «Задача: Пример использования метода «ВыполнитьСканмарк».

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток.

## Данные



**Синтаксис:**

Данные

**Параметры:**

Нет

**Возвращаемое значение:**

Тип: [Неопределено](#), [Строка](#), [Массив](#).

Данные события.

**Описание:**

Содержит данные, полученные от внешнего события на ТСД (сканирование с помощью сканера штрихкода, сканирование с помощью камеры, ввод данных через пункт меню «Ввод данных», сканирование с помощью сканера RFID).

Если параметр внешнего события (Событие) = «RFID», тогда тип «Данные» = [Массив\(n, 3\)](#). Такой массив содержит n-количество полученных меток (от 0 до n-1) в виде фиксированного набора данных из 3-х значений:

**EPC**, Тип: [Строка](#)

**TID**, Тип: [Строка](#)

**USER**, Тип: [Строка](#)

Если параметр внешнего события (Событие) <> «RFID», тогда тип «Данные» = [Строка](#).

**Доступность:**

Обработчик ТСД <ВнешнееСобытие> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток.

## ДанныеШтрихкода

**Синтаксис:**

ДанныеШтрихкода(<Данные>, <Запрашивать>)

**Параметры:**

<Данные> (обязательный)

Тип: [Строка](#).

<Запрашивать> (необязательный)

Тип: [Булево](#).

Значение по умолчанию [Ложь](#).

**Возвращаемое значение:**

Тип: [ТаблицаЗначений](#).

Таблица значений заданная настройками структуры данных в разделе «Штрихкоды».

**Описание:**

Возвращает таблицу значений с данными соответствующими значению реквизита «Штрихкод» настройки в разделе «Штрихкоды». Если таблицу значений не содержит строку и параметр <Запрашивать> = [Истина](#), то будет предложено загрузить данные по штрихкоду из учетной системы.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## ДокументОснование

**Синтаксис:**

ДокументОснование

**Параметры:**

Нет

**Возвращаемое значение:**

Тип: [Ссылка](#), [Неопределено](#).

Ссылка на документ основание (если документ создан через «Ввести на основании»).

**Описание:**

Возвращает [Ссылку](#) на документ основание (если документ создан через «Ввести на основании») или [Неопределено](#).

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы.

## ЗаписатьЭтотОбъект

**Синтаксис:**

ЗаписатьЭтотОбъект()

**Параметры:**

Нет

**Описание:**

Записывает текущий документ сразу же после выполнения кода обработчика.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы.

## Источник

**Синтаксис:**

Источник

**Параметры:**

Нет

**Возвращаемое значение:**

Тип: [Строка](#).

Источник события.

**Описание:**

Содержит информацию об источнике от внешнего события на ТСД (сканирование с помощью сканера штрихкода, сканирование с помощью камеры, ввод данных через пункт меню «Ввод данных», сканирование с помощью сканера RFID).

**Доступность:**

Обработчик ТСД <ВнешнееСобытие> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток.

**Пример:**

scanner, camera, keyboard.

## КОтправке

**Синтаксис:**

КОтправке

**Параметры:**

Нет

**Возвращаемое значение:**

Тип: [Булево](#).

**Описание:**

Свойство объекта, полученного функцией «ОбъектСканмарк()» для типа объекта [ДокументСсылка](#), указывающий на установленный признак отправки документа.

**Доступность:**

Обработчики ТСД формы документа, обработчики ТСД произвольных данных (Объекты, Отчеты, Команды), обработчики <ПриИзменении> реквизита форма на ТСД.

## Лог

**Синтаксис:**

Лог(<Событие>, <Информация>)

**Параметры:**

<Событие> (обязательный)

Тип: [Строка](#).

<Информация> (необязательный)

Тип: [Строка](#).

**Возвращаемое значение:**

Нет.

**Описание:**

Записывает информацию о событии. Для приложения и для обработки в 1С:Предприятие ведутся свои собственные независимые журналы событий.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных», обработчики обработки 1С:Предприятие.

## НайтиПоКоду

**Синтаксис:**

НайтиПоКоду(<Вид>, <Код>)

**Параметры:**

<Вид> (обязательный)

Тип: [Строка](#).

<Код> (обязательный)

Тип: [Строка](#), [Число](#).

**Возвращаемое значение:**

Тип: [Массив](#).

Создает массив, содержащий ссылки на элементы справочника вида <Вид> с кодом <Код>.

**Описание:**

Осуществляет поиск элементов по коду.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## НайтиПоНаименованию

**Синтаксис:**

НайтиПоНаименованию(<Вид>, <Наименование>)

**Параметры:**

<Вид> (обязательный)

Тип: [Строка](#).

<Наименование> (обязательный).

Тип: [Строка](#).

**Возвращаемое значение:**

Тип: [Массив](#).

Создает массив, содержащий ссылки на элементы справочника вида <Вид> с наименованием <Наименование>. В параметре <Наименование> допускается использование «%» и «?» для поиска наименования по подобию.

**Описание:**

Осуществляет поиск элементов по наименованию.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## НайтиПоРеквизиту

**Синтаксис:**

НайтиПоРеквизиту(<Вид>, <ИмяРеквизита>, <Значение>)

**Параметры:**

<Вид> (обязательный)

Тип: [Строка](#).

<ИмяРеквизита> (обязательный)

Тип: [Строка](#).

<Значение> (обязательный)

Тип: [Строка](#), [Число](#), [Булево](#), [Дата](#), [СправочникСсылка](#), [ДокументСсылка](#).

**Возвращаемое значение:**

Тип: [Массив](#).

Создает массив, содержащий ссылки на элементы справочника вида <Вид> и значением <Значение> реквизита <Реквизит>.

**Описание:**

Осуществляет поиск элементов по значению реквизита.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## НайтиПоУникальномуИдентификатору

**Синтаксис:**

НайтиПоУникальномуИдентификатору(<УникальныйИдентификатор>)

**Параметры:**

<УникальныйИдентификатор> (обязательный)

Тип: [Строка](#).

**Возвращаемое значение:**

Тип: [Массив](#).

Создает массив, содержащий ссылки на документы и/или элементы справочников по строковому

представлению <УникальныйИдентификатор>.

**Описание:**

Осуществляет поиск ссылок документов и элементов справочников по строковому представлению Уникального Идентификатора.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## ОбъектСканмарк

**Синтаксис:**

ОбъектСканмарк(<Ссылка>)

**Параметры:**

<Ссылка> (обязательный)

Тип: [СправочникСсылка](#), [ДокументСсылка](#).

**Возвращаемое значение:**

Тип: [Структура](#), [Неопределено](#).

Структура с реквизитами и табличными частями, заданными настройками структуры данных в разделах «Справочники» и «Документы». Тип табличных частей [ТаблицаЗначений](#). Если передана пустая ссылка или тип значения <Ссылка> не является описанным типом [СправочникСсылка](#), [ДокументСсылка](#), то будет возвращено [Неопределено](#).

**Описание:**

Возвращает структуру реквизитов и табличных частей элемента справочника или документа.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## ОписаниеОшибки

**Синтаксис:**

ОписаниеОшибки

**Параметры:**

нет

**Возвращаемое значение:**

Тип: [Строка](#).

Описание ошибки.

**Описание:**

Возвращает текстовое представление ошибки. Пользователь может задать свой текст описания ошибки, который будет передан на ТСД при отправке результата выполнения кода обработчика 1С.

**Доступность:**

Обработчик 1С (код на языке 1С).

## ОткрытьПодбор

**Синтаксис:**

ОткрытьПодбор(<ПараметрыПодбора>)

**Параметры:**

<ПараметрыПодбора> (обязательный)

Тип: [Структура](#).

**Возвращаемое значение:**

Нет.

**Описание:**

Открывает форму подбора справочника, заданного в `ПараметрыПодбора`. `ПараметрыПодбора` имеет предопределенный состав ключей. Предопределенный метод формы `ОбработкаПодбора` (`<Источник>`, `<Данные>`) служит для обработки выбора значений.

**Описание структуры «ПараметрыПодбора»:**

Ключи и описание значений:

**Ключ:** "Объект", **Значение:** "Справочник.XXX", где XXX - имя вида справочника;

**Ключ:** "ВыбиратьЭлементы", **Значение:** выбирать среди элементов справочника([Булево](#));

**Ключ:** "ВыбиратьГруппы", **Значение:** выбирать среди групп справочника ([Булево](#));

**Ключ:** "МножественныйВыбор", **Значение:** выбор нескольких значений (Истина) или одного значения (Ложь), после чего окно закрывается ([Булево](#));

**Ключ:** "Источник", **Значение:** произвольная строка, значение параметра возможно использовать в предопределенный методе формы `ОбработкаПодбора` (`<Источник>`, `<Данные>`) ([Строка](#)).

**Доступность:**

Обработчики ТСД, обработчики `<ПриИзменении>` в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных.

## Отправлен

**Синтаксис:**

Отправлен

**Параметры:**

Нет

**Возвращаемое значение:**

Тип: [Булево](#).

**Описание:**

Свойство объекта, полученного функцией «`ОбъектСканмарк()`» для типа объекта [ДокументСсылка](#), указывающий на признак того, что документ был отправлен в 1С:Предприятие хотя бы один раз.

**Доступность:**

Обработчики ТСД, обработчики `<ПриИзменении>` в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «`ПослеПолученияДанных`».

## ПереместитьКнопкуНаСтраницу

**Синтаксис:**

ПереместитьКнопкуНаСтраницу (`<ЗаголовокКнопки>`, `<Имя>`)

**Параметры:**

<ЗаголовокКнопки> (обязательный)

Тип: [Строка](#).

<Имя> (обязательный)

Тип: [Строка](#).

**Возвращаемое значение:**

Нет.

**Описание:**

Перемещает Кнопку (Команду) с именем <ЗаголовокКнопки> на страницу с именем <Имя> в форме ТСД.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Команда), настройках Свои наборы данных, настройках Печать этикеток.

## ПолучитьДанныеИз1с

**Синтаксис:**

ПолучитьДанныеИз1с (<Данные>, <ИмяМетода1с>, <ИмяСобытияТСД>, <ИмяСвоегоНабораДанных>)

**Параметры:**

<Данные> (обязательный)

Тип: [ДокументСсылка](#), [СправочникСсылка](#), [Строка](#), [Число](#), [Булево](#), [Дата](#), [Массив](#), [ТаблицаЗначений](#), [СписокЗначений](#), [Структура](#).

<ИмяМетода1с> (обязательный)

Тип: [Строка](#).

<ИмяСобытияТСД> (необязательный)

Тип: [Строка](#).

<ИмяСвоегоНабораДанных> (необязательный)

Тип: [Строка](#).

**Описание:**

Выполняет запрос на получение объектов и/или «свой набор данных» в форме ТСД из 1С. Метод работает с настройкой «Глобальные обработчики» в настройках обработки «Scanmark».

Порядок использования метода:

1. определить и сформировать набор данных (параметр <Данные>) в форме на ТСД;
2. в 1С:Предприятие в настройках глобальных обработчиков обработки настройки Scanmark на закладке Обработчики метода "ЗапроситьДанные" в 1С добавить поименованный обработчик (параметр <ИмяМетода1с>), в котором прописать код на языке 1С.
3. при выполнении оператора «ПолучитьДанныеИз1с» обработчика в форме ТСД мобильное приложение выполнит обмен с 1С:Предприятие, используя код, заданный в параметре <ИмяМетода1с>. Перед выполнением метода в 1С:Предприятие инициализируются переменная «Данные» (заполняется переданным параметром <Данные>) и «Результат» (по умолчанию [Неопределено](#)). Необходимо набрать в переменную «Результат» данные объектов (должны быть описаны в настройках «Документ» и/или «Справочники»). На выходе из метода набор данных для передачи в ТСД будет получен из переменной «Результат». Допустимые для получения данных объектов из переменной «Результат»: [ДокументСсылка](#), [СправочникСсылка](#), [Массив](#), [ТаблицаЗначений](#), [СписокЗначений](#), [Структура](#).
4. параметр <ИмяСобытияТСД> не является обязательным. Позволяет автоматически выполнить обработчик <ИмяСобытияТСД> в форме ТСД (тип обработчика «Команда») после обмена.
5. параметр <ИмяСвоегоНабораДанных> не является обязательным. Используется только для передачи в «Свой набор данных». Для передачи данных в коде глобального обработчика необходимо инициализировать и заполнить необходимые для передачи наборы данных, соблюдая следующие правила:
  - а. в набор данных «шапки» передаются значения объявленных переменных соответствующих наименованиям реквизитов настройки своего набора данных;
  - б. в наборы данных табличных частей передаются значения из созданных таблиц значений соответствующих наименованиям табличных частей в настройке, а значения колонок из таких таблиц значений соответствующих наименованиям реквизитов табличных частей настройки;
  - в. инициализировать в коде глобального обработчика данных соответствующих настройке не является обязательным – будут переданы только инициализированные

одноименные переменные.

Для большего понимания см. примеры:

- Задача: Пример использования функции «ПолучитьДанныеИз1с»;
- Пересчет табличной части документа «Заказ клиента» после выбора типа цен.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток.

## РазблокироватьДокумент

**Синтаксис:**

РазблокироватьДокумент (<Ссылка>)

**Параметры:**

<Ссылка> (обязательный)

Тип: [ДокументСсылка](#).

**Описание:**

Разблокирует ранее заблокированный документ.

**Доступность:**

Обработчик 1С (код на языке 1С).

## Редактирование

**Синтаксис:**

Редактирование(<Редактирование>, <ИмяРеквизита>, <ИмяТабличнойЧасти>)

**Параметры:**

<Редактирование> (обязательный)

Тип: [Булево](#).

<ИмяРеквизита> (обязательный)

Тип: [Строка](#).

<ИмяТабличнойЧасти> (необязательный)

Тип: [Строка](#), [Неопределено](#).

Значение по умолчанию [Неопределено](#).

**Возвращаемое значение:**

Нет.

**Описание:**

Определяет возможность редактирования значения непосредственно в элементе диалога в форме ТСД.

[Истина](#) – редактирование доступно, [Ложь](#) - поле недоступно.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Команда), настройках Свои наборы данных, настройках Печать этикеток.

## СказатьФразу

**Синтаксис:**

СказатьФразу(<Текст>)

**Параметры:**

<Текст> (обязательный)

Тип: [Строка](#).



**Возвращаемое значение:**

Тип: нет.

**Описание:**

Произносит <Текст>.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток.

## СвойНаборДанных

**Синтаксис:**

СвойНаборДанных (<Вид>, <ИмяРеквизита>, <ПараметрыОтбора>)

**Параметры:**

<Вид> (обязательный)

Тип: [Строка](#).

<ИмяРеквизита> (обязательный)

Тип: [Строка](#).

<ПараметрыОтбора> (необязательный)

Тип: [Структура](#).

**Возвращаемое значение:**

Тип:

Определяется типом реквизита своего набора данных, если <ИмяРеквизита> - это имя реквизита своего набора данных;

[Таблица значений](#), если <ИмяРеквизита> - это имя реквизита табличной части своего набора данных.

**Описание:**

Возвращает значение из своего набора данных.

<Вид> – наименование своего набора данных.

<ИмяРеквизита> - наименование реквизита своего набора данных.

<ПараметрыОтбора> - задает условия поиска в табличной части, ключ структуры определяет имя колонки, по которой будет осуществляться поиск, а значение структуры - искомое значение.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».

## Событие

**Синтаксис:**

Событие

**Параметры:**

Нет

**Возвращаемое значение:**

Тип: [Строка](#).

**Описание:**

Содержит информацию об имени внешнего события на ТСД (сканирование с помощью сканера штрихкода, сканирование с помощью камеры, ввод данных через пункт меню «Ввод данных», сканирование с помощью сканера RFID).

**Пример:**

Barcode, RFID.

**Доступность:**

Обработчик ТСД <ВнешнееСобытие> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток.

## ТекущаяСтраница

**Синтаксис:**

ТекущаяСтраница(<Имя>)

**Параметры:**

<Имя> (необязательный)

Тип: [Строка](#).

Имя страницы, как оно задано в настройках.

**Возвращаемое значение:**

Тип: [Строка](#).

**Описание:**

Устанавливает текущую страницу формы по имени <Имя> (если <Имя> задано).

Возвращает имя текущей страницы как оно задано в настройках после исполнения метода.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Команда), настройках Свои наборы данных, настройках Печать этикеток.

## ТекущиеДанные

**Синтаксис:**

ТекущиеДанные

**Параметры:**

Нет

**Возвращаемое значение:**

Тип: [Строка таблицы значений](#).

Текущая строка табличной части.

**Описание:**

Предоставляет собой текущую (редактируемую) отдельную строку таблицы значений, которая содержит набор свойств в колонках таблицы значений по конкретной строке. Доступ к значению осуществляется по имени колонки

**Пример:**

ТекущаяСтрока.Сумма = ТекущаяСтрока.Цена \* ТекущаяСтрока.Количество;

**Доступность:**

Обработчик ТСД <ПриИзменении> реквизита табличной части в настройках Документы, настройках Произвольные данные (Команда), настройках Свои наборы данных, настройках Печать этикеток.

## ТекущийПользователь

**Синтаксис:**

ТекущийПользователь()

**Параметры:**

Нет

**Возвращаемое значение:**

Тип: [Строка](#).

**Описание:**

Возвращает имя текущего пользователя.

**Доступность:**

Обработчики ТСД, обработчики <ПриИзменении> в настройках Документы, настройках Произвольные данные (Объекты, Отчеты, Команда), настройках Свои наборы данных, настройках Печать этикеток, глобальный обработчик ТСД «ПослеПолученияДанных».