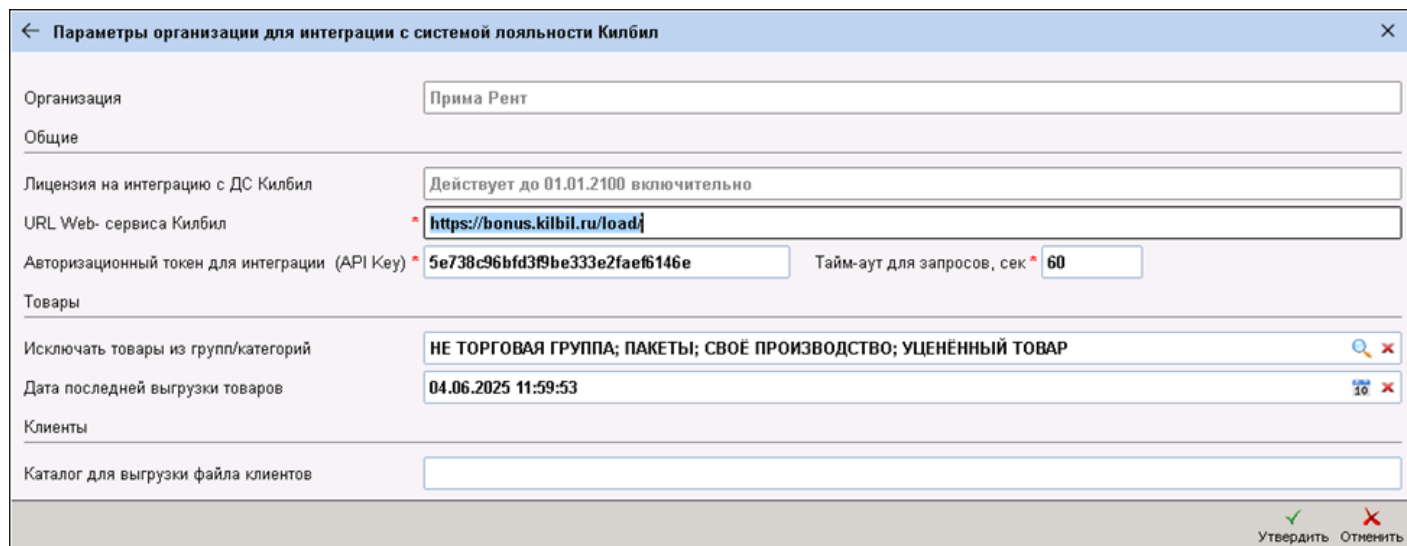


# Kilbil

- Настройка
- Сценарий идентификации покупателя
- Запрос промокода
- Процесс Расчёт скидок
- Процесс Возврат товара
- Список запросов в KilBil
- Выгрузка справочника товаров в KilBil

# Настройка

В карточке компании, кнопка "Интеграция с внешними ДС", пункт меню Kilbil, заполняются параметры в разделе "Общие":



← Параметры организации для интеграции с системой лояльности Килбил ×

Организация:

Общие

Лицензия на интеграцию с ДС Килбил:

URL Web- сервиса Килбил:

Авторизационный токен для интеграции (API Key):  Тайм-аут для запросов, сек:

Товары

Исключать товары из групп/категорий:

Дата последней выгрузки товаров:

Клиенты

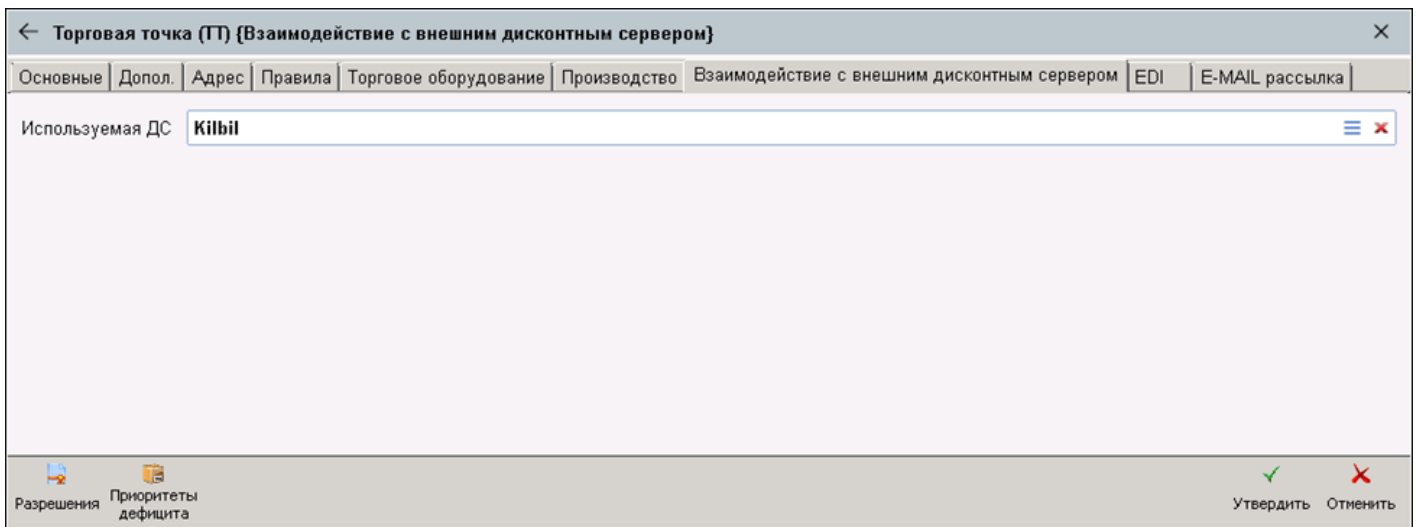
Каталог для выгрузки файла клиентов:

Утвердить Отменить

- URL Web-сервиса Kilbil
- Авторизационный токен для интеграции (API Key)
- Тайм-аут для запросов, сек

Авторизационный токен допускает выгрузку из базы данных товаров и партнеров (покупателей).

В карточке структурного подразделения (торговой точки) заполняется параметр "Используемая ДС".



Обмен касс магазина с Kilbil включается только в том случае, если параметр заполнен будет.

В карточках точек продаж (POS) заполняются:

- Авторизационный токен для касс (API Key)
- Тайм-аут для запросов в секундах. Это время ожидания ответа от Kilbil на запрос кассы. Если за указанное время Kilbil не ответит, то касса будет обрабатывать эту ситуацию как ошибку обмена (отсутствие связи).

В настройке интерфейса кассира следует привязать к горячим клавишам две команды: '*Идентификация покупателя в ДС*', '*Запрос промокода*'. Эти команды используются для запуска соответствующих процессов.

# Сценарий идентификации покупателя

Процесс запускается по команде 'Идентификация покупателя в ДС'. По окончании процесса касса получает идентификатор покупателя. Если идентификатор не будет заполнен, то считается, что покупатель не участвует в программе лояльности.

На экране отображается форма с данными покупателя, информационной строкой, кнопками для выбора действий.

Идентификация покупателя

Номер телефона  
79024090105

Номер дисконтной карты  
2670009264277

Фамилия  
Желтышева

Имя  
Александра

Отчество  
Юрьевна

Дата рождения  
02.01.1959

Общее количество баллов  
2000

Покупатель является участником программы лояльности

Поиск    Регистрация    Подтвердить телефон    Отмена    Выбрать

Поля:

- Номер телефона (начинается с 7)
- Номер карты – можно сканировать ШК, можно ввести ручную часть номера
- Фамилия

- Имя
- Отчество
- Дата рождения
- Число баллов (для просмотра)
- Строка с сообщением для кассира

Кнопки:

- Поиск
- Регистрация
- Отмена
- Подтвердить телефон
- Выбрать

Вся дальнейшая работа внутри процесса выполняется в данной форме. Форма закрывается только при выходе из процесса. Промежуточные результаты отображаются в полях формы и в информационной строке.

Сначала кассир вводит один из идентификаторов покупателя: номер телефона или номер карты. Номер телефона начинается с цифры 7.

После ввода телефона выполняется запрос в КВ. КВ по номеру телефона ищет в своей БД запись участника программы. Если поиск завершается успешно, то КВ в ответе на запрос возвращает информацию об участнике программы. Касса заполняет остальные поля формы данными из ответа.

## Основные сценарии работы кассира

### *1. Покупатель является участником программы лояльности*

Если покупатель говорит, что он является участником программы лояльности и называет номер телефона. Кассир вводит номер и нажимает клавишу 'Enter'. Касса отправляет в КВ запрос. КВ по номеру телефона (или номеру карты) ищет в своей БД запись участника программы. Если поиск завершается успешно, то КВ в ответе на запрос возвращает информацию об участнике программы. Касса заполняет остальные поля формы данными из ответа. Далее проверяется наличие значения в поле Телефон. Если поле не заполнено, то в информационной строке отображается тест: *'Требуется подтверждение телефона'*. Если

телефон заполнен, то в информационной строке появляется сообщение *'Покупатель является участником программы лояльности'*. Если КВ присылает отрицательный ответ (подходящая запись в БД не обнаружена), то в информационной строке отображается ответ из КВ.

Далее кассир анализирует информацию на экране и действует по одному из следующих сценариев.

- *1.1. Участие подтверждено, телефон имеется. В строке сообщения находится текст 'Покупатель является участником программы лояльности'*

Кассир нажимает кнопку *'Выбрать'*. Форма закрывается. Идентификатор покупателя определён.

- *1.2. Участие подтверждено, не указан телефон. В строке сообщения находится текст 'Требуется подтверждение телефона'*

Кассир вводит номер телефона и нажимает кнопку *'Подтвердить телефон'*.

Сначала касса отправляет запрос для проверки, что данный телефон ещё не зарегистрирован. Если ответ положительный, то в информационной строке отображается текст *'Покупатель с таким телефоном уже зарегистрирован'*. Выполняется возврат в форму. Кассир либо вводит другой номер телефона, либо отменяет идентификацию.

Если ответ отрицательный (т.е. в базе КВ нет регистрации данного телефона), то касса отправляет в КВ запрос для подтверждения номера. Если ответ отрицательный, то в информационную строку записывается сообщение об ошибке. Выполняется возврат в форму.

В случае положительного ответа отображается новая форма для запроса кода. В форме одно поле *'Введите код'* и две кнопки: *'Проверить'*, *'Отмена'*. Покупатель называет код, полученный по sms, кассир вводит этот код и нажимает *'проверить'*. Касса отправляет запрос на проверку полученного кода. КВ возвращает результат проверки.

Если коды совпадают, то касса отправляет в КВ запрос на изменение данных покупателя.

Если ответ положительный, то в информационной строке отображается *'Телефон подтверждён'*. Выполняется возврат в основную форму. В информационной строке появляется сообщение *'Покупатель является участником программы лояльности'*.

Если получен отрицательный ответ на запрос изменения данных покупателя, то выполняется возврат в форму, в информационной строке отображается сообщение об ошибке.

Если код, названный покупателем, не совпадают с кодом, полученным по запросу, то в форме для запроса отображается *'неверный код'*. Кассир может повторно ввести код и нажать *'проверить'*. Если кассир нажимает *'Отмена'*, то выполняется возврат в основную форму без подтверждения телефона. В информационной строке остаётся текст *'Требуется подтверждение телефона'*.

- *1.3. Участие не подтверждено. Текст в информационной строке содержит описание ошибки*

Дальнейшие действия либо по ветке 2, либо по ветке 3.

## ***2. Покупатель не является участником программы лояльности и не хочет участвовать***

Кассир выбирает *'Отмена'*. Форма закрывается, идентификация завершается без заполнения идентификатора покупателя.

## ***3. Покупатель не является участником программы лояльности и хочет зарегистрироваться***

Кассир заполняет в форме остальные параметры. Обязательно должны быть заполнены телефон, Имя, Фамилия. Отчество и дата рождения не обязательны к заполнению. Номер карты для регистрации не нужен и будет очищен.

Далее выполняются действия по проверке телефона в базе КВ и его подтверждения. Подробно эти действия описаны в ветке 1.2. Алгоритм действий такой же, но после успешного подтверждения телефона регистрация продолжается. Касса отправляет запрос на регистрацию. Если в ответе будет ошибка, то отображается текст ошибки в информационной строке. Выполняется возврат в основную форму. В случае положительного ответа форма закрывается. Идентификатор покупателя определён.

# Запрос промокода

Процесс запускается по команде *'Запрос промокода'*. По окончании процесса касса получает название промокода. Если название промокода не заполнено, то считается, что промокода нет.

На экране отображается форма с одним полем *'Промокод'* и двумя кнопками *'Подтвердить'* и *'Отмена'*. Никакие проверки введённой строки не выполняются.

Промокод будет применён в запросах в КВ для расчёта скидок.

Никаких ограничений на доступ к команде нет. Доступность команды не зависит от наличия обмена с КВ или идентификации покупателя.

Промокод на чек может быть только один. Повторный ввод промокода удаляет предыдущий.

# Процесс Расчёт скидок

Данный процесс является неотъемлемой частью основных кассовых процессов заполнения и оплаты чека.

Условно процесс делится на четыре этапа:

- Регулярный расчёт скидок по чеку
- Запрос у покупателя количества баллов для оплаты чека
- Финальный расчёт чека
- Создание чека в КВ

Этапы процесса выполняются при следующих условиях:

- включён обмен с КВ (заполнены параметры в карточке точки продаж, имеется связь с КВ)
- магазин является участником программы лояльности (заполнен параметр в карточке магазина)
- чек не является оформлением покупки по интернет-заказу (кассир не запрашивал номер интернет-заказа)

Для запроса баллов и расчёта скидок за баллы дополнительно требуется успешная идентификация покупателя. Если покупатель не идентифицирован как участник программы лояльности, то баллы не запрашиваются.

## Регулярный расчёт скидок по чеку

После каждого изменения содержания чека (добавление и удаление товаров, изменение количества) касса отправляет в КВ запрос на расчёт чека. Число баллов в запросе равно 0, чтобы КВ при расчёте скидок не учитывал баллы. В случае положительного ответа от КВ касса разбирает ответ и изменяет фактические цены товаров. Если КВ не ответил или вернул отрицательный ответ, то цены не меняются.

При отсутствии связи с КВ на экране красным цветом загорается соответствующий сигнальный символ. Если связь прервалась после предыдущего успешного запроса, то на

экране высвечивается сообщение для кассира *'Сервер лояльности недоступен'*. Если на предыдущий запрос также не было ответа, то повторно сообщение не высвечивается.

## Запрос баллов

Выполняется только при условии успешной Идентификации покупателя.

В процессе Идентификации КВ вернул число баллов. Эти баллы можно потратить для оплаты чека – на товары может быть рассчитана дополнительная скидка в рублях, равная числу баллов.

После перехода в режим оплаты на экране появляется окно с предложением ввести количество баллов для оплаты чека. По умолчанию в поле находится имеющееся значение баллов. Кассир по согласованию с покупателем может уменьшить это число. В форме имеются две кнопки: *'Подтвердить'* и *'Отмена'*. Если кассир нажал *'Подтвердить'*, то число баллов для оплаты принимается равным введённому значению. Выбор кнопки *'Отмена'* означает отказ от списания баллов.

## Финальный расчёт чека

Выполняется только при условии успешной Идентификации покупателя, после запроса баллов. Касса отправляет в КВ финальный запрос для расчёта скидок. В запросе должен быть заполнен тег с числом баллов, чтобы КВ при расчёте скидок учитывал баллы. Значение берётся из ответа на предыдущий запрос расчета чека.

В случае положительного ответа от КВ касса разбирает ответ и изменяет фактические цены товаров. Если КВ не ответил или вернул отрицательный ответ, то касса ничего не делает.

Если покупатель отказывается от оплаты и возвращается в редактирование чека, то следует сбросить расчёт с учётом баллов. Для этого касса отправляет в КВ запрос на расчёт чека с нулевым количеством баллов.

## Подтверждение оплаты чека

Выполняется после полного оформления чека, когда покупатель оплатил покупку и чек отправлен на ФР.

Касса отправляет в КВ запрос для подтверждения покупки.

Если КВ вернул ошибку, то повторяем расчет чека и ещё раз подтверждение покупки. Если опять ошибка, то чек сохраняется в списке неотправленных чеков.

Если связь с КВ отсутствует, то чек также сохраняется в списке неотправленных чеков.

*Список неотправленных чеков нужен только для тех чеков, по которым был выполнен расчёт скидок, покупатель оплатил чек, но информация о покупке в КВ не была передана. Т.е. КВ расчёты выполнил, но запрос на создание покупки не получил.*

# Процесс Возврат товара

Покупатель оформляет возврат товаров в кассе. Предполагается, что это отказ от части позиций, без добавления новых позиций.

Кассир вводит реквизиты чека продажи (фискальный номер, дата, номер смены). Поиск чека сначала выполняется в базе кассы. Если чек не обнаружен, то выполняется поиск в ЦБД. Цель поиска - определить реквизиты чека продажи в Domino для формирования номера этого чека в KB.

При создании чека в KB номер чека был создан по следующему шаблону: <Код структурного подразделения> - <код кассы> - <номер смены> - <дата чека> - <номер чека>.

Выполняется запрос покупки в KB. Если KB вернул чек, то работаем с ним. В противном случае (например, нет связи с KB) считаем, что чек продажи не определён. Для оформления возврата нужны номера позиций в чеке. В запросе указываются изменённые позиции.

После окончания денежных расчётов касса отправляет в KB запрос для оформления возврата и списания бонусов. Запрос создаётся только в том случае, если известен чек продажи. Если возврат создан без чека продажи, то невозможно заполнить идентификатор чека в KB, и, следовательно, запрос не отправляется.

Ответ из KB заносится в протокол. Касса продолжает работу в обычном режиме.

# Список запросов в KiVBil

## Общие положения по обмену с KB

<https://ru.sailplay.dev/reference/ogranicheniya-api>

Запуск запроса: [https://bonus.kilbil.ru/load/{название\\_функции}?h={ключ}](https://bonus.kilbil.ru/load/{название_функции}?h={ключ})

Тело запроса должно соответствовать формату JSON.

Время ожидания ответа от KB указывается в карточке компании. Касса отправляет запрос и ждёт ответа указанное время. Если KB не отвечает, то данная ситуация считается ошибкой 'Сервер лояльности недоступен'. Касса продолжает работу без обмена с KB.

Если в ответе на запрос приходит код 200, то ответ положительный. Любые другие значения кода ответа и отсутствие ответа трактуются как отрицательный ответ.

Действия при получении отрицательного ответа зависят от запроса и места вызова.

		Действие
Запрос	Место	ошибка
searchclient	Открытие смена	Выставить
	связь	отсутствует'
	Начало чека	Ничего
	при не	отсутствии
	связи	связи
searchclient	Формат	Отобразить
	идентификации	ответ
	покупательской	информационной
	строке	строке

		Отобразить Формат ответ <i>askconfirmphone</i> идентификации <i>checkconfirmphone</i> покупательской информационной строке
		Отобразить Формат ответ идентификации <i>addclient</i> покупательской информационной строке
	При <i>process</i> состоянии связь чека отсутствует'	Выставить состояние 'связь отсутствует'
	В конце чека	Выставить состояние связь отсутствует'
	<i>confirmsale</i> После оплаты чек в список неотправленных чеков	Выставить состояние 'связь отсутствует'
	Запрос <i>getdetailsbyid</i> состояние связь продажи отсутствует'	Выставить состояние 'связь отсутствует'

Тексты запросов и подробные ответы протоколируются в логах кассы.

## Действия в случае отсутствия связи с КВ

В карточке компании указывается время ожидания ответа на запрос к КВ. По истечении указанного времени считается, что связь с КВ отсутствует. На экране появляется сообщение для кассира об отсутствии связи с программой лояльности. Соответствующий сигнальный символ в левом верхнем углу экрана загорается красным цветом. Сообщение о

потере связи отображается только один раз – при первом неудачном запросе. Для последующих запросов сообщение не выводится. После восстановления связи лампочка загорается зеленым цветом.

Связь с КВ проверяется в начале чека запросом *searchklient* для телефона 70000000000. Если связь отсутствует, то при работе с данным чеком запросы в КВ не посылаются.

Если связь с КВ пропала при создании покупки (был успешно выполнен *processsale*, но неудача при *confirmsale*), то чек (параметры запроса) сохраняется в отдельном списке для последующей отправки.

Передача неотправленных чеков выполняется при закрытии смены и в специальном пункте меню.

## Информация об участниках (клиентах) программы лояльности

Запись клиента имеет два уникальных идентификатора: номер карты, номер телефона. Для старых клиентов номер карты будет заполнен, а номер телефона может быть пустым.

В Домино требуется проверять уникальность идентификаторов при создании нового клиента. Ограничимся проверкой только номера телефона. Поскольку остальные уникальные идентификаторы при регистрации нового клиента можно не заполнять, то так и будем поступать.

Ввод адреса электронной почты, объединение клиентов будут выполняться только в личном кабинете КВ. <https://help.kilbil.ru/93-1075-614--kak-dobavit-klienta-cherez-lichnyj-kabinet/>

## Запрос ‘Получить информацию о клиенте’ *searchklient*

<https://bonus.kilbil.ru/load/searchklient>

Запрос:

Параметр	описание
----------	----------

search_mode	0 - поиск по № телефона 2 - поиск по № карты
search_value	значение, по которому производится поиск, если по номеру телефона, это должно быть 11 символов первая 7-ка

Ответ:

Параметр	Описание
result_code	код результата, 0 - успешно
client_id	id клиента, null если не найден

bonus_balance	количество бонусов у клиента
card_code	номер карты, если нет
phone	номер телефона клиента, null если не заполнено
birth_date	дата рождения клиента, null если не заполнено
first_name	имя клиента, null если не заполнено
middle_name	отчество клиента, null если не заполнено
last_name	фамилия клиента, null если не заполнено

	максимальная сумма списания на текущий чек без учета
max_balance_out	баланса клиента, параметр нужен для дальнейшего использования
	в processsale

## Запрос 'Отправить код подтверждения' *askconfirmphone*

<https://bonus.kilbil.ru/load/askconfirmphone>

Запрос:

Параметр	Описание
client_id	id клиента
phone	номер телефона клиента 11 символов, первая 7

	тип шаблона сообщения
0 -	сообщение с пин-кодом подтверждения для регистрации клиента
1 -	сообщение с пин-кодом, например для списания бонусов

Ответ:

Параметр	Описание
success	код выполнения
result_code	код результата, 0 - успешно

## Запрос 'Проверить код подтверждения' *checkconfirmphone*

<https://bonus.kilbil.ru/load/checkconfirmphonecode>

Запрос:

Параметр	Описание
----------	----------

client_id	id клиента
phone	номер телефона клиента 11 символов, первая 7
code	код подтверждения

Ответ:

Параметр	Описание
success	код выполнения
result_code	код результата, где 0 - успешно

Запрос 'Добавление (обновление) клиента'  
*addclient*

<https://bonus.kilbil.ru/load/addclient>

Запрос:

Параметр	Описание
client_id	id клиента

phone	номер телефона клиента 11 символов, первая 7, клиент не будет активирован без номера телефона
-------	--------------------------------------------------------------------------------------------------------------------------------------

Ответ:

Параметр	Описание
success	код выполнения
client_id	id клиента
result_code	код результата, 0 - успешно

## Расчёт скидок

Для промежуточных расчётов применяется запрос processsale. Непосредственно перед оплатой следует запросить число баллов и отправить финальный запрос processsale для учёта баллов.

Подтверждение покупки в KB выполняется отдельным запросом.

## Запрос 'Рассчитать документ'

*processsale*

Параметр	Описание
client_id	id клиента, null если покупатель не идентифицирован
type	тип операции: 0 - продажа, 1 - возврат
bonus	количество списанных бонусов в операции
max_bonus_out	максимально возможная сумма списания бонусов без учета баланса клиента (передается значение выходного параметра max_bill_bonus_out функции searchclient)
good_data	Массив товаров

code	Код товара
vendor_code	Артикул
name	Наименование товара
price	цена
quantity	количество
total	Сумма за позицию
minPrice	Минимальная цена
discounted_price	Цена за позицию, равна price
discounted_total	Сумма за позицию, равна total
move_id	Номер чека
doc_open_dt	Дата чека

return_id	номер чека продажи, по которому проводится возврат (передаётся только для возвратов)
return_date	Дата чека продажи (передаётся только для возвратов)
promo_codes	Список промокодов
coupons	Массив купонов
coupon	промокод

Move\_id потребуется для возврата и он должен быть уникальным. Заполняем из следующих параметров: <Код структурного подразделения> - <код кассы> - <номер смены> - <дата чека> - <номер чека>.

Поскольку применяются только скидки через систему лояльности, то price=discounted\_price, Total= discounted\_total.

Промокод может быть только один. Кассир его ввёл в процессе 'запрос промокода'.

Ответ:

Параметр	Описание
----------	----------

result_code	код результата, где 0 – успешно
bill_data	информация по расчету документа
items	Массив товаров
code	Код товара
discounted_price	Новая цена
price	Цена
bonus_out	Количество списанных бонусов на позицию
bonus_out	Количество списанных бонусов на документ

## Запрос 'Подтвердить документ' *confirmsale*

<https://bonus.kilbil.ru/load/confirmsale>

Подтверждение документа после processsale, после этого в системе лояльности чек считается подтвержденным и участвует в отчетах. Выполняются все начисления и списания бонусов.

Параметр	Описание
----------	----------

move_id	Номер чека
doc_open_dt	Дата чека
doc_dt	Дата закрытия чека

Ответ:

Параметр	Описание
result_code	код результата, где 0 - успешно

## Возврат товаров

Для создания чека возврата применяется запрос 'Рассчитать документ' (processsale с type=1).

Предварительно в базе KB запрашивается чек продажи.

### Запрос 'Получить документ' getdetailingbymove

<https://bonus.kilbil.ru/load/getdetailingbymove>

Параметр	Описание
move_id	Номер чека

Ответ:

Параметр	Описание
----------	----------

result_code	код результата, где 0 - успешно
move_info	общая информация по чеку
oper_type	Тип операции: Ф -продажа
move_detail	Массив товаров
code	Код товара
price	Цена
total	Сумма по позиции
quantity	количество

# Выгрузка справочника товаров в KіlBіl

<https://docs.kilbil.ru/#tag/Nomenclatura/operation/goods>

Предусмотрены два режима выгрузки товаров в KB:

1. Полная выгрузка
2. Инкрементальная выгрузка

Полная выгрузка запускается вручную из меню. Выгружаются все товары в KB.

Товары выгружаются порциями по 2000 товаров.

Инкрементальная выгрузка обрабатывает только изменения товаров с момента предыдущей выгрузки. Запускается либо вручную из меню, либо автоматически планировщиком.

Совместно с карточками товаров выгружаются группы и категории, 4 уровня классификатора.

Запрос:

Параметр	Описание
partial	0 - полная, 1 - инкрементальная

is_group	True - для групп и категорий False - для товаров
delete_mark	True в случае инкрементальной выгрузки
code	Код товара в Домино или код группы/категории
name	Наименование товара или наименование группы/категории
inside	Следующий уровень группы/категории внутри группы или товар внутри категории. Включает параметры is_group, delete_mark, code, name, inside

Ответ:

Параметр	Описание
Success	1 - успешно
result_code	0 - успешно

“ Пример:

```
{  
  "is_group": true,  
  
  "delete_mark": false,  
  
  "code": "1",  
  
  "name": "Название группы",  
  
  "inside": []  
},  
  
  {  
  
    "is_group": true,  
  
    "delete_mark": false,  
  
    "code": "5",  
  
    "name": "Название группы 3",  
  
    "inside": [  
  
      {  
  
        "is_group": false,
```

```
"delete_mark": false,  
  
"code": "706",  
  
"name": "Наименование товара"  
  
},  
  
{  
  
"is_group": false,  
  
"delete_mark": false,  
  
"code": "610",  
  
"name": "название товара"  
  
}  
  
]  
  
}
```