



# Инструкция по API платёжного шлюза для мерчанта

# Содержание

---

<b>1. История изменений</b>	<b>3</b>
<b>2. Введение</b>	<b>5</b>
<b>3. Определения и термины</b>	<b>6</b>
<b>4. Описание API</b>	<b>8</b>
4.1 Общие принципы работы API	8
4.2 Требования к Интернет-сайту мерчанта	8
4.3 Подключение к API	10
4.4 Безопасность использования API	10
4.5 Доступные валюты для операций	11
4.6 Типы операций	11
4.7 Типы оплат	11
4.8 Статусы ордера	12
4.9 Статусы транзакций для типа оплаты SBP	13
4.10 Статусы транзакций для типа оплаты CARD (Одностадийный платёж)	13
4.11 Статусы транзакций для типа оплаты CARD (Двухстадийный платёж)	13
4.12 Операции с ордерами	14
4.12.1 POST v1/orders	14
4.12.2 GET v1/orders/{orderId}	18
4.12.3 POST v1/orders/preauth	22
4.12.4 POST v1/orders/complete/{orderId}	25
4.12.5 POST /orders/{orderId}/cancel	30
4.12.6 Создание ордера (заказа) с CoF	32
4.13 Операции с возвратами	34
4.13.1 POST v1/refunds	34
4.13.2 GET v1/refunds/{refundId}	37
4.14 Уведомления (callback) от Платежного шлюза	39
4.15 Типы отправляемых уведомлений (callback) от Платежного шлюза	39

# Содержание

---

4.15.1 PaymentResponse	40
4.15.2 Уведомление RefundResponse	42
4.16 Интеграция с тестовой средой Платежного шлюза	44
4.16.1 Настройки для подключения к тестовой среде (Песочница)	44
4.16.2 Отправка запросов к API Платежного шлюза (Песочница)	46
4.17 Тестирование API	47
4.17.1 Тестовые кейсы	47
4.17.2 Реквизиты тестовой банковской карты	49
4.18 Интеграция с PRODUCTION средой Платежного шлюза	49
4.18.1 Настройки для подключения к PRODUCTION среде Платежного шлюза	49
4.18.2 Отправка запросов к API Платежного шлюза (PRODUCTION)	51
5 Фискализация в соответствии с 54-ФЗ	52
5.1 Общее описание функциональности фискализация в соответствии с 54-ФЗ	52
5.2 Создание ордера с товарной корзиной	52
5.3 Создание возврата с товарной корзиной	57
5.4 Операции с фискальными чеками	58
5.4.1 GET /receipts/{orderId}/status	58
<b>Приложение 1. Описание объекта ошибки Платежного шлюза, возвращаемого через API</b>	<b>62</b>
<b>Приложение 2. Расшифровка кодов ответов по ошибкам от Платежного шлюза</b>	<b>63</b>

# 1. История изменений

---

Версия	Описание изменений	Дата
1.0.0	Первая версия документа	14.03.2022
1.0.1	<ul style="list-style-type: none"><li>Исключены типы валют "USD", "EUR" из таблицы "Доступные валюты для операций"</li><li>В главе "Безопасность использования API" добавлено описание дополнительной аутентификации сайта мерчанта через заголовок запроса Merchant-Authorization</li><li>В главе "Безопасность использования API" добавлен пункт Пример указания заголовков при обращении к API Платежного шлюза"</li><li>Скорректированы шаги по пункту "Шаги по подключению" в разделе "Подключение к API"</li><li>Скорректировано описание раздела "Уведомления (callback) от Платежного шлюза"</li><li>Дополнен раздел "Определения и термины"</li></ul>	01.09.2022
1.0.2	<ul style="list-style-type: none"><li>Дополнен раздел "Определения и термины"</li><li>В раздел "Типы оплат" добавлен тип оплаты (CARD) и добавлена колонка с описанием сценариев оплаты</li><li>Актуализирован объект ошибки, отдаваемый Платежным шлюзом</li><li>Добавлено описание ошибок по операциям оплаты по картам</li><li>Добавлены тестовые кейсы по сценариям оплаты картой</li><li>Раздел "Статусы транзакций" разделен на два новых раздела "Статусы транзакций для типа оплаты SBP (Оплата с помощью Системы быстрых платежей)", "Статусы транзакций для типа оплаты CARD (Оплата по банковской карте)"</li><li>В разделе "PaymentResponse" для параметра "type" в колонке "Формат" добавлено новое значение "card"</li></ul>	23.09.2022
1.0.3	Актуализированы адреса и параметры подключения к "Песочнице"	11.10.2022
1.0.4	<ul style="list-style-type: none"><li>Обновлено описание запросов для получения access_token на тестовой и PRODUCTION среде</li><li>Обновлено описание запроса к API Платежного шлюза на тестовой среде и добавлено описание для отправки запросов к API на PRODUCTION среде</li><li>Во входящие параметры эндпоинта по созданию ордера (POST /orders) добавлен и описан новый параметр "returnUrl"</li><li>Обновлены примеры запросов к API и ответов от шлюза</li></ul>	20.12.2022
1.0.5	<ul style="list-style-type: none"><li>В разделе "Приложение 1. Описание объекта ошибки Платежного шлюза, возвращаемого через API" обновлено описание объекта ошибки, возвращаемого Платежным шлюзом</li><li>В разделе "Приложение 2. Расшифровка кодов ответов по ошибкам от Платежного шлюза" обновлено описание кодов ошибок и расшифровки ошибок, возвращаемых Платежным шлюзом</li></ul>	26.01.2023

# 1. История изменений

---

Версия	Описание изменений	Дата
1.0.6	<ul style="list-style-type: none"><li>Для эндпоинта POST v1/orders добавлено уточнение про идемпотентность эндпоинта</li><li>Актуализировано значение термина "Merchant-Authorization"</li><li>Выполнены незначительные исправления и уточнения по всему документу</li></ul>	10.03.2023
1.0.7	<ul style="list-style-type: none"><li>Добавлен новый раздел «Реквизиты тестовой банковской карты».</li><li>Выполнены незначительные исправления и уточнения по всему документу</li></ul>	20.06.2023
1.0.8	<ul style="list-style-type: none"><li>В раздел «Определения и термины» добавлен новый термин «Двухстадийный платёж»</li><li>Обновлен раздел «Статусы ордера»</li><li>Обновлен раздел «Статусы транзакций для типа оплаты CARD»</li><li>Добавлен новый раздел «Статусы транзакций для типа оплаты CARD (Двухстадийный платёж)»</li><li>Добавлено раздел с описанием эндпоинта POST v1/orders/preauth для создания ордера с предавторизацией (двухстадийный платеж)</li><li>Добавлен раздел с описанием эндпоинта POST v1/orders/complete/{orderId} для завершения ранее предавторизованного ордера</li><li>Добавлен раздел с описанием эндпоинта POST /orders/{orderId}/cancel для отмены ордера</li><li>Выполнены незначительные исправления и уточнения по всему документу</li></ul>	04.07.2023
1.0.9	<ul style="list-style-type: none"><li>В раздел 3 «Определения и термины» — добавлено новое определение для связок CoF</li><li>В раздел 4.12.1 «POST v1/orders» — добавлены новые входные параметры запроса, необходимые для создания/оплаты связками CoF</li><li>Добавлен новый раздел 4.12.6 «Создание ордера (заказа) с CoF»</li><li>Раздел «Уведомления(callback) от Платежного шлюза» — добавлены новые параметры для связок CoF в тело уведомления «PaymentResponse»</li><li>Приложение 2. «Расшифровка кодов ответов по ошибкам от Платежного шлюза» — добавлены новые коды ошибок при создании/оплаты связками CoF</li></ul>	15.03.2024
1.0.10	<ul style="list-style-type: none"><li>Раздел 3 «Определения и термины» — добавлены новые определения.</li><li>Добавлен новый раздел 5 «Фискализация в соответствии с 54-ФЗ».</li><li>Приложение 2. «Расшифровка кодов ответов по ошибкам от Платежного шлюза» — добавлены новые коды ошибок, которые могут возникнуть при передаче в заказе товарной корзины и при запросе статуса чеков.</li></ul>	29.03.2024

## 2. Введение

---

Данный документ описывает работу API (Application protocol interface) Платежного шлюза. API предназначено для интеграции ресурса мерчанта и Платежного шлюза с целью предоставления сервисов по безналичной оплате товаров и услуг и выполнения других видов операций на стороне ресурса мерчанта.

## 3. Определения и термины

---

**Система (IPSP)** — Платежный шлюз на стороне Банка, предоставляющий услуги интернет-эквайринга (обработки электронных платежей)

**Банк** — Банк ВТБ (ПАО)

**Покупатель (клиент)** — физическое лицо, совершающее операции по оплате товаров и услуг мерчанта в интернет магазине мерчанта

**Application protocol interface (API)** — программный интерфейс для взаимодействия с Системой

**Мерчант (Merchant) (ТСП)** — торгово-сервисное предприятие, продающее товары или оказывающее услуги через интернет-сайт. Мерчант подключается к API Платежного шлюза для осуществления приема электронных платежей и осуществления иных операций

**Ресурс** — электронная торговая площадка мерчанта в сети Интернет (интернет-сайт мерчанта)

**Bearer** — одна из распространенных схем аутентификации на основе использования токенов безопасности, передаваемых в заголовках запросов

**Менеджер доступа API** — хост на стороне Банка, который при прохождении аутентификации предоставляет токены безопасности (access\_token)

**access\_token** — токен безопасности, выдаваемый менеджером доступа Платежного шлюза и используемый для аутентификации при обращении к платежного шлюзу

**Merchant-Authorization** — необязательный HTTP-заголовок, который используется в запросах к API для дополнительной аутентификации ресурса мерчанта для тех случаев, когда у мерчанта подключено к Платежному шлюзу более одного ресурса. В качестве значения заголовка Merchant-Authorization при обращении к API Платежного шлюза передается идентификатор ресурса, выданный представителем Платежного шлюза мерчанту после регистрации ресурсов на стороне Платежного шлюза.

**Ордер** — заказ, создаваемый в платежном шлюзе для его последующей оплаты. Каждый ордер имеет уникальные идентификаторы на стороне мерчанта и системы

**Транзакция** — совокупность последовательных операций на стороне Платежного шлюза в рамках выполнения операции оплаты или возврата по ордеру. Каждой транзакции присваивается уникальный идентификатор

**Система быстрых платежей (СБП)** — система быстрых платежей

**Refund** — частичный или полный возврат денежных средств на карту покупателя в случае его отказа от получения товара (услуги) или его возврата. Операция возврата денежных средств выполняется после списания денежных средств со счета покупателя

**Reversal** — отмена операции оплаты (снятие блокировки с денежных средств на карте покупателя)

**Платёжная форма** — WEB-страница, которая используется покупателем для ввода реквизитов платежа. URL адрес платежной формы всегда уникален и передается мерчанту в параметре "payUrl» в ответе на запрос на создание нового ордера

**Callback (колбэк)** — сообщение-уведомление от Платежного шлюза к системе мерчанта о статусе выполнения оплаты или возврата платежа

**Песочница** — тестовая среда Платежного шлюза для проверки интеграции и работы API

**Production** — продуктивная среда Платежного шлюза

**Банк-эквайер** — банк, который реализует и эксплуатирует платёжный шлюз

**Банк-эмитент** — банк, выпустивший банковскую карту клиента

**НУЦ** — национальный удостоверяющий центр

**БИН (BIN)** — банковский идентификационный номер, часть номера, расположенного на пластиковой карте. Первые 6 цифр номера карты

**Номер банковской карты (Primary Account Number) (PAN)** — номер банковской карты (обычно 16 цифр). Первые 6 в номере карты - это как раз BIN

**Международная платежная система (МПС)** — системы, обслуживающие платежи по картам (VISA, Mastercard, МИР)

**3D-secure (3DS)** — протокол дополнительной аутентификации держателя карты

**Одностадийный платёж** — операция по оплате товаров / услуг, совершённая через интернет с использованием банковских карт, не требующая дополнительного подтверждения (один запрос инициирует сразу блокирование и списание денежных средств с карты)

**Двухстадийный платёж** — операция по оплате товаров / услуг, совершённая через интернет с использованием банковских карт, требующая дополнительного подтверждения.

Двухстадийный механизм работы позволяет разделить процесс на проверку платёжеспособности банковской карты (авторизация) и списание денежных средств (финансовое подтверждение). Первый запрос инициирует проверку платёжеспособности банковской карты и блокирование денежных средств на счёте клиента, второй запрос инициирует списание. Максимальный срок холдирования денежных средств зависит от правил платежной системы

**returnUrl** — URL-адрес фронт системы ТСП для переадресации клиента в случае успешной / неуспешной оплаты

**CoF (credential on file)** — уникальный защищенный токен, который генерируется на стороне Платежного шлюза для последующей связки номера банковской карты (PAN) покупателя с его идентификатором в системе мерчанта. В качестве идентификатора покупателя может использоваться его логин в системе мерчанта или другой принятый в системе мерчанта идентификатор

**ОФД** — оператор фискальных данных

**Фискализация** — функциональность по созданию (печати) чеков в соответствии с 54-ФЗ с использованием онлайн-кассы внешнего оператора фискальных данных (ОФД)

**ФД** — фискальные данные

**ФН** — фискальный накопитель

## 4. Описание API

---

### 4.1 Общие принципы работы API

В качестве прикладного протокола передачи данных используется защищенный протокол HTTPS. Запросы по протоколу HTTP (без шифрования) отклоняются менеджером доступа API и при обращении к API Платежного шлюза.

В API заложено использование архитектурных принципов взаимодействия REST: для взаимодействия с API Платежного шлюза используются POST и GET-запросы, при этом, в теле POST-запроса указываются параметры в формате JSON, а в GET-запросе параметры запроса указываются в URL. Ответом на GET или POST-запрос всегда будет результат в формате JSON.

Одной из основных сущностей при работе с платежным шлюзом является ордер (заказ), создание которого инициирует мерчант со стороны своего ресурса перед тем, как покупатель будет готов произвести оплату товаров / услуг, находящихся в корзине покупателя на сайте мерчанта. Без создания ордера в платежном шлюзе покупатель не сможет произвести его оплату, а мерчант не сможет выполнить возврат или другие операции, связанные с ордером.

Срок жизни ордера задается мерчантом через соответствующий параметр в эндпоинте для создания ордера. Если мерчант не указал срок жизни ордера при его создании, то срок жизни ордера задается в виде стандартного значения (20 минут). В рамках операций по оплате или возврату по ордеру на стороне Платежного шлюза создаются транзакции.

### 4.2 Требования к Интернет-сайту мерчанта

1. Сайт Интернет-магазина должен иметь статический IP адрес и размещаться на платном хостинге.
2. Сайт Интернет-магазина не должен содержать страниц и ссылок на URL-адреса Интернет-магазинов мерчанта другого содержания.
3. Сайт Интернет-магазина не должен содержать баннеров, не соответствующих специфике деятельности мерчанта.
4. Все внутренние ссылки сайта должны быть рабочими и адекватно обрабатываемыми.
5. Все страницы Интернет-магазина мерчанта, которые связаны с реализацией мерчантом Товаров должны находиться под единым доменным именем.
6. Витрина Интернет-магазина не должна содержать информацию и / или Товары, запрещенные к распространению, в соответствии с законодательством Российской Федерации и Правилами Платежных систем.
7. Витрина Интернет-магазина должна быть на русском языке или иметь русскоязычный вариант изложения информации.
8. В Витрине Интернет-магазина мерчанта должна быть представлена клиентам следующая информация:
  - данные о мерчанте (наименование, адрес, ИНН, контактный телефон, электронная почта и т.д.);
  - регистрационная форма для клиента (данные о клиенте, формирующем заказ);
  - информацию с логотипами Платежных систем, карты которых принимаются мерчантом в соответствии с Договором;

- информацию о Товарах, продаваемых / совершаемых / предоставляемых мерчантом (перечень Товаров, их описание, цены и т.п);
  - информацию о порядке оформления и оплаты Заказа и / или Товара с использованием реквизитов карт;
  - условия доставки Товара;
  - ограничения на экспортную деятельность (при их наличии);
  - разъяснение для покупателей о политике информационной безопасности, применяемой на мерчанте;
  - разъяснение для покупателей процедуры безопасной передачи по каналам связи конфиденциальной информации Держателей, применяемой на мерчанте;
  - информация о мерах по обеспечению безопасности Операций с использованием реквизитов Карт;
  - информацию о процедуре возврата Товаров / отказа от работ / услуг, оплаченных при помощи реквизитов карты, а также порядке возврата денежных средств держателю карты по операциям возврата товаров /отказа от работ / услуг;
  - информацию о процедуре отмены операции оплаты заказа, совершенной с использованием реквизитов карты, а также о порядке возврата денежных средств Держателю карты по отмененным заказам.
9. Перечень Товаров, представленный мерчантом Банку, должен соответствовать перечню Товаров, указанному в Заявлении на регистрацию мерчанта и фактически реализуемых через Интернет-магазин.
  10. Вид деятельности Интернет-магазина должен соответствовать указанному в Заявлении виду деятельности мерчанта.
  11. Для лицензируемой деятельности в Витрине Интернет-магазина должна быть размещена информация о лицензиях, разрешениях от производителя, правообладателя или государственных органов на реализуемые Товары.
  12. В Витрине Интернет-магазина не должно быть неработающих страниц, ссылок, направление которых не соответствует их названию, а также страниц / ссылок, перенаправляющих на другие сайты (без явного указания перенаправления).
  13. Витрина Интернет-магазина должна находиться в рабочем состоянии, содержать исчерпывающую информацию о предлагаемых Товарах.

## 4.3 Подключение к API

Шаги по подключению ресурса мерчанта:

1. Пройти предварительные организационные шаги для подключения к платежному шлюзу (заключение договора с банком, подключение услуги интернет-эквайринга, регистрация в Системе быстрых платежей (СБП) и т.д.).
2. Получить тестовые credentials для подключения к тестовой среде («Песочнице»).
3. Проверить работу интеграции с API Платежного шлюза на тестовой среде («Песочнице») путем выполнения тест-кейсов, которые приведены в этой инструкции в разделе «Тестирование API».
4. После успешного выполнения тестовых кейсов необходимо предоставить ответственным лицам от Платежного шлюза результаты выполнения тестирования.
5. Получить credentials для подключения к Production среде API Платежного шлюза, идентификатор ресурса для дополнительной аутентификации на стороне Платежного шлюза.
6. Настроить взаимодействие с Production средой Платежного шлюза на стороне ресурса мерчанта. Предоставить ответственным лицам от Платежного шлюза URL-адреса для отправки уведомлений от Платежного шлюза (необязательный пункт).

## 4.4 Безопасность использования API

При обращении к Менеджеру доступа Платежного шлюза и к API Платежного шлюза должно устанавливаться защищенное протоколом SSL / TLS соединение.

**!** Для предотвращения появления ошибки с SSL / TLS сертификатом необходимо скачать и добавить в хранилище корневых доверенных сертификатов сервера, на котором расположен ресурс, сертификаты Национального удостоверяющего центра (НУЦ) в случае их отсутствия!

- Корневой сертификат: [http://rostelecom.ru/cdp/rootca\\_ssl\\_rsa2022.crt](http://rostelecom.ru/cdp/rootca_ssl_rsa2022.crt)
- Промежуточный сертификат: [http://rostelecom.ru/cdp/subca\\_ssl\\_rsa2022.crt](http://rostelecom.ru/cdp/subca_ssl_rsa2022.crt)

Для аутентификации ресурса мерчанта при обращении к API Платежного шлюза используется:

1. Схема аутентификации Bearer с передачей в HTTP-заголовке Authorization значения динамического access\_token, полученного в Менеджере доступа.
2. Дополнительный HTTP-заголовок X-IBM-Client-Id, в котором указывается client\_id, присвоенный ресурсу мерчанта (значение заголовка указывается в нижнем регистре и без указания домена!).

Данный заголовок следует указывать только при обращении к API Платежного шлюза (указывать этот заголовок при обращении к Менеджеру доступа не следует)!

3. Дополнительный (необязательный) HTTP-заголовок Merchant-Authorization, в котором передается идентификатор ресурса, если у мерчанта подключено более одного ресурса к Платежному шлюзу.

В запросах к API по каждому ресурсу мерчанта используется свой уникальный идентификатор ресурса, однозначно идентифицирующий ресурс мерчанта в Платежном шлюзе.

Данный заголовок следует указывать только при обращении к API Платежного шлюза (указывать этот заголовок при обращении к Менеджеру доступа не следует)!

При обращении к Менеджеру доступа Платежного шлюза используются следующие параметры, получаемые мерчантом после регистрации:

- `client_id`
- `client_secret`

**!** Значения параметров `client_id`, `client_secret` и HTTP-заголовков `X-IBM-Client-Id`, `Merchant-Authorization` используются для аутентификации ресурса мерчанта и должны держаться мерчантом в секрете от третьих лиц! Не указывайте, пожалуйста, необезличенные этих параметров и заголовков в документации, инструкциях, примерах кода и т.д.!

Подробные примеры для запроса `access_token` через Менеджер доступа Платежного шлюза для тестовой и `production` среды приведены в соответствующих главах данного руководства.

## 4.5 Доступные валюты для операций

Код валюты	Наименование валюты
RUB	Российский рубль

## 4.6 Типы операций

Тип	Описание
orders	Операции с ордерами
refunds	Операции с возвратами

## 4.7 Типы оплат

Тип	Описание	Сценарии оплаты
SBP	Оплата с помощью Системы быстрых платежей (СБП)	<ul style="list-style-type: none"><li>• QR-код для последующей оплаты клиентом в мобильном приложении интернет-банка генерируется в системе Мерчанта (если при создании ордера в объекте <code>returnPaymentData</code> было передано значение "sbp")</li><li>• QR-код для последующей оплаты клиентом в мобильном приложении интернет-банка генерируется на Платежной форме после нажатия на кнопку "СБП"</li><li>• Если клиент перешел на Платежную форму с мобильного устройства и нажал на кнопку "СБП", то он сразу переходит в мобильное приложение интернет-банка без генерации QR-кода</li></ul>
CARD	Оплата по банковской карте	Оплата производится на Платежной форме путем ввода реквизитов банковской карты после редиректа клиента по уникальному адресу Платежной форме, переданному в ответе запроса на создание ордера в параметре <code>payUrl</code>

## 4.8 Статусы ордера

Используемые статусы ордера в системе приведены в таблице ниже:

Статус	Описание	Финальный статус
CREATED	Ордер создан. Первичный статус ордера после его создания	—
PAID	Ордер полностью оплачен	—
REFUNDED	По ордеру совершен полный возврат	Да
PARTIALLY_REFUNDED	По ордеру совершен частичный возврат. Новые возвраты по ордеру возможны в пределах оставшейся суммы	—
EXPIRED	Срок жизни ордера истек (срок, который был задан при создании ордера или срок действия ордера по умолчанию, если срок жизни ордера не был указан при его создании)	Да
CANCELED	Ордер отменен мерчантом. Переход ордера в статус "CANCELED" возможен только в том случае, если ордер находится в статусе "CREATED" и по ордеру отсутствуют связанные транзакции вне зависимости от их статуса	Да
PENDING	Сумма ордера захолдирована. Статус применим только для двустадийной оплаты по банковской карте	—
VOIDED	Холдирование денежных средств по ордеру отменено. Статус применим только для двустадийной оплаты по банковской карте	Да

## 4.9 Статусы транзакций для типа оплаты SBP

Оплата с помощью Системы быстрых платежей

Статус	Описание	Финальный статус
NEW	Новая транзакция	—
SBP_PAYMENT_CREATED	По транзакции инициирована оплата через СБП. Ожидается получение статуса выполнения оплаты на стороне СБП	—
SBP_REFUND_CREATED	По транзакции инициирован возврат на стороне СБП. Ожидается получение статуса выполнения возврата на стороне СБП	—
RECONCILED	По транзакции успешно выполнен платеж в СБП	Да
DECLINED	Транзакция отклонена	Да

## 4.10 Статусы транзакций для типа оплаты CARD (Одностадийный платёж)

Оплата по банковской карте при одностадийном платеже

Статус	Описание	Финальный статус
NEW	Новая транзакция	—
DECLINED	Транзакция отклонена	Да
CONFIRMED	Транзакция подтверждена	Да

## 4.11 Статусы транзакций для типа оплаты CARD (Двухстадийный платёж)

Оплата по банковской карте при двухстадийном платеже

Статус	Описание	Финальный статус
NEW	Новая транзакция	—
AUTHORIZED	Транзакция авторизована (успешно выполнено холдирование суммы ордера)	—
DECLINED	Транзакция отклонена	Да
CONFIRMED	Транзакция подтверждена	Да
REVERSED	Транзакция холдирования отменена мерчантом	Да

## 4.12 Операции с ордерами

### 4.12.1 POST v1/orders

POST v1/orders – запрос для создания нового ордера (заказа). На каждую новую оплату создается новый ордер. Данный запрос является идемпотентным (при отправке запроса с указанием ранее использовавшегося значения параметра "orderId" будет возвращен ответ с информацией по ранее созданному ордеру).

#### Параметры запроса POST v1/orders

<b>Merchant-Authorization</b>	Дополнительный (необязательный) HTTP-заголовок, в котором передается идентификатор ресурса, если у мерчанта подключено более одного ресурса к Платежному шлюзу	Url / body / header	Header	Тип n/a	Формат n/a
<b>orderId</b> Обязательный	Уникальный идентификатор ордера в системе мерчанта	Url / body / header	Body	Тип String	Формат Byte
<b>orderId</b> Обязательный	Наименование ордера в системе мерчанта	Url / body / header	Body	Тип String (255)	Формат Byte
<b>expire</b>	Срок жизни ордера. Указывается с обязательным указанием timezone (обозначение зоны для нулевого смещения UTC) и указанием 3-х знаков для обозначения миллисекунд. Пример: 2023-10-12T12:02:16.883Z. Если параметр и его значение не было передано при создании ордера, то срок жизни ордера будет задан автоматически путем прибавления 20 минут к времени создания ордера	Url / body / header	Body	Тип String (32)	Формат date-time
<b>returnPaymentData</b>	Вернуть данные для осуществления платежа дополнительными способами оплаты	Url / body / header	Body	Тип String	Формат enum: - sbp
<b>customer</b>	Данные о клиенте	Url / body / header	Body	Тип Object	Формат n/a
<b>customerid</b> Условно обязательный	Идентификатор покупателя для конкретного сайта мерчанта в системе мерчанта. <b>Параметр обязателен для передачи при создании заказа с COF</b>	Url / body / header	Body	Тип String (255)	Формат Byte
<b>email</b> Условно обязательный	Электронный адрес клиента (в формате email) <b>Параметр обязателен для передачи при создании заказа с COF</b>	Url / body / header	Body	Тип String (255)	Формат Byte
<b>phone</b>	Телефонный номер покупателя (в формате 79009990099)	Url / body / header	Body	Тип String (11) or null	Формат Byte

<b>amount</b>	Объект суммы
Обязательный	Url / body / header Body Тип Object Формат n/a
<b>value</b>	Сумма. Указывается в виде целой и (опционально) дробной части длиной в два знака после разделителя (точки). Пример: 1055.20
Обязательный	Url / body / header Body Тип Number Формат minimum: 0; maximum: 10000000000
<b>code</b>	Валюта
Обязательный	Url / body / header Body Тип String Формат enum: - RUB - USD - EUR

<b>returnUrl</b>	URL-адрес фронт системы ТСП для переадресации клиента в случае успешной / неуспешной оплаты
	Url / body / header Body Тип String Формат Byte

<b>binding</b>	Объект данных о связке CoF. Объект обязателен для передачи только при создании заказа с COF
Условно обязательный	Url / body / header Body Тип Object Формат n/a
<b>bindingType</b>	Тип связки. Возможные типы связок: RECURRING - Рекарринговые платежи INSTALLMENT - Установочный платеж COMMON - Пользовательский платеж <b>Параметр обязателен для передачи только при создании заказа с COF</b>
Условно обязательный	Url / body / header Body Тип String Формат enum: RECURRING, INSTALLMENT, COMMON
<b>operationType</b>	Параметр, указывающий на то, какой тип операции будет проводиться со стороны инициатора (мерчанта). Возможные типы операции: MIT (Merchant Initialized Transaction) - повторные операции, которые инициируются мерчантом; CIT (Customer Initialized Transaction) - повторные операции, которые инициируются плательщиком <b>Параметр обязателен для передачи только при создании заказа с COF</b>
Условно обязательный	Url / body / header Body Тип String Формат enum: MIT, CIT
<b>bindingId</b>	Уникальный идентификатор связки CoF на стороне Платежного шлюза. <b>Параметр обязателен для передачи только при необходимости оплаты заказа с помощью ранее созданной связки CoF</b>
Условно обязательный	Url / body / header Body Тип String(64) Формат Byte

#### Пример запроса POST v1/orders

```
{
  "orderId": "ORDER10000000001",
  "orderName": "Заказ №10000000101",
  "expire": "2022-12-31T01:09:44Z",
  "amount": {
    "value": 10.56,
    "code": "RUB"
  }
}
```

## Параметры ответа POST v1/orders

<b>type</b>	Тип объекта (ORDER)
Обязательный	Тип String      Формат enum: -ORDER
<hr/>	
<b>object</b>	Объект ордера
Обязательный	Тип Object      Формат n/a
<b>orderId</b>	Уникальный идентификатор ордера в системе мерчанта
Обязательный	Тип String (64)      Формат Byte
<b>orderCode</b>	Уникальный внешний идентификатор ордера
Обязательный	Тип String (64)      Формат Byte
<b>expire</b>	Срок жизни ордера. В формате: 2022-10-12T12:02:16.883Z
Обязательный	Тип String (32)      Формат date-time
<b>payUrl</b>	Адрес платежной формы для редиректа клиента на страницу оплаты (в формате FQDN)
Обязательный	Тип String (255)      Формат Byte
<b>preparedPayments</b>	Дополнительные данные для осуществления оплаты
	Тип Array      Формат n/a
<b>type</b>	Тип оплаты
Обязательный	Тип String      Формат enum: - sbp - invoice
<b>object</b>	Объект для оплаты
Обязательный	Тип Object      Формат n/a
<b>url</b>	Ссылка для оплаты
Обязательный	Тип String (255)      Формат n/a
<b>amount</b>	Объект суммы
Обязательный	Тип Object      Формат n/a
<b>value</b>	Сумма. Указывается в виде целой и (опционально) дробной части длиной в два знака после разделителя (точки). Пример: 1055.20
Обязательный	Тип Number      Формат minimum: 0 maximum: 10000000000
<b>code</b>	Валюта
Обязательный	Тип String      Формат enum: - RUB - USD - EUR
<b>status</b>	Объект статуса
Обязательный	Тип n/a      Формат n/a
<b>changedAt</b>	Время изменения статуса. В формате: 2022-10-12T12:02:16.883Z
Обязательный	Тип String (32)      Формат date-time
<b>status</b>	

status	value	Валюта	Тип String (255)	Формат Byte
	Обязательный			
	description	Описание кода статуса	Тип String (255)	Формат Byte

#### Пример ответа POST v1/orders

```
{
  "type": "ORDER",
  "object": {
    "orderId": "ORDER10000000001",
    "orderCode": "848bb7b2-6961-4263-85dc-a4df9c377eod",
    "status": {
      "value": "CREATED",
      "description": "CREATED",
      "changedAt": "2022-12-27T12:02:52.483Z"},
    "createdAt": "2022-12-27T12:02:52.483Z",
    "payUrl": "https://pay.vtb.ru/?order-id=848bb7b2-6961-4263-85dc-a4df9c377eod",
    "expire": "2022-12-31T01:09:44.000Z",
    "amount": {
      "value": 10.56,
      "code": "RUB"}
  }
}
```

#### 4.12.2 GET v1/orders/{orderId}

GET v1/orders/{orderId} – запрос для получения информации по ордеру по его ID.

#### Параметры запроса GET v1/orders/{orderId}

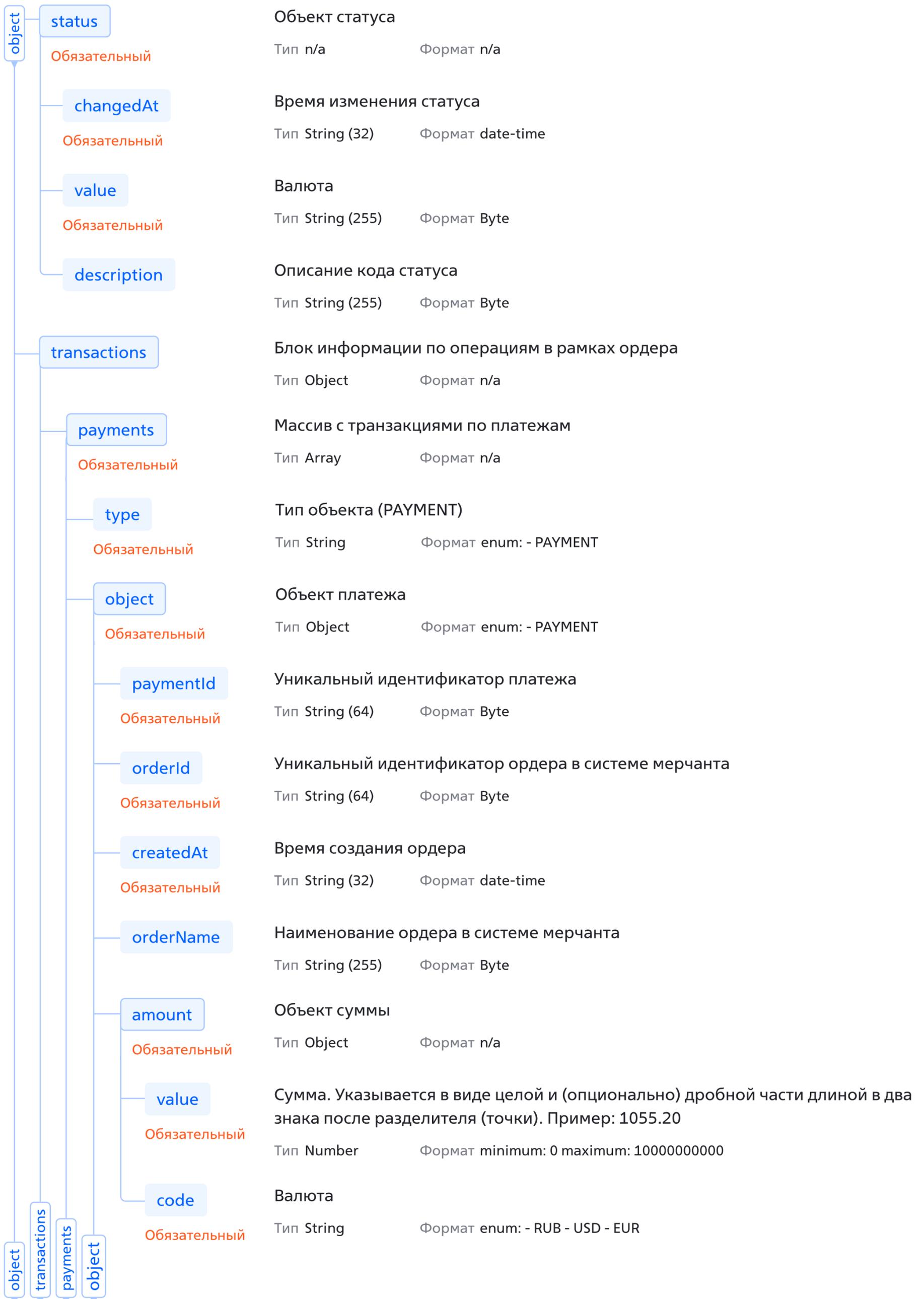
Merchant-Authorization	Дополнительный (необязательный) HTTP-заголовок, в котором передается идентификатор ресурса, если у мерчанта подключено более одного ресурса к Платежному шлюзу	Url / body / header	Header	Тип n/a	Формат n/a
orderId	Уникальный идентификатор ордера в системе мерчанта	Url / body / header	URL	Тип String (64)	Формат Byte
Обязательный					

#### Пример запроса GET v1/orders/{orderId}

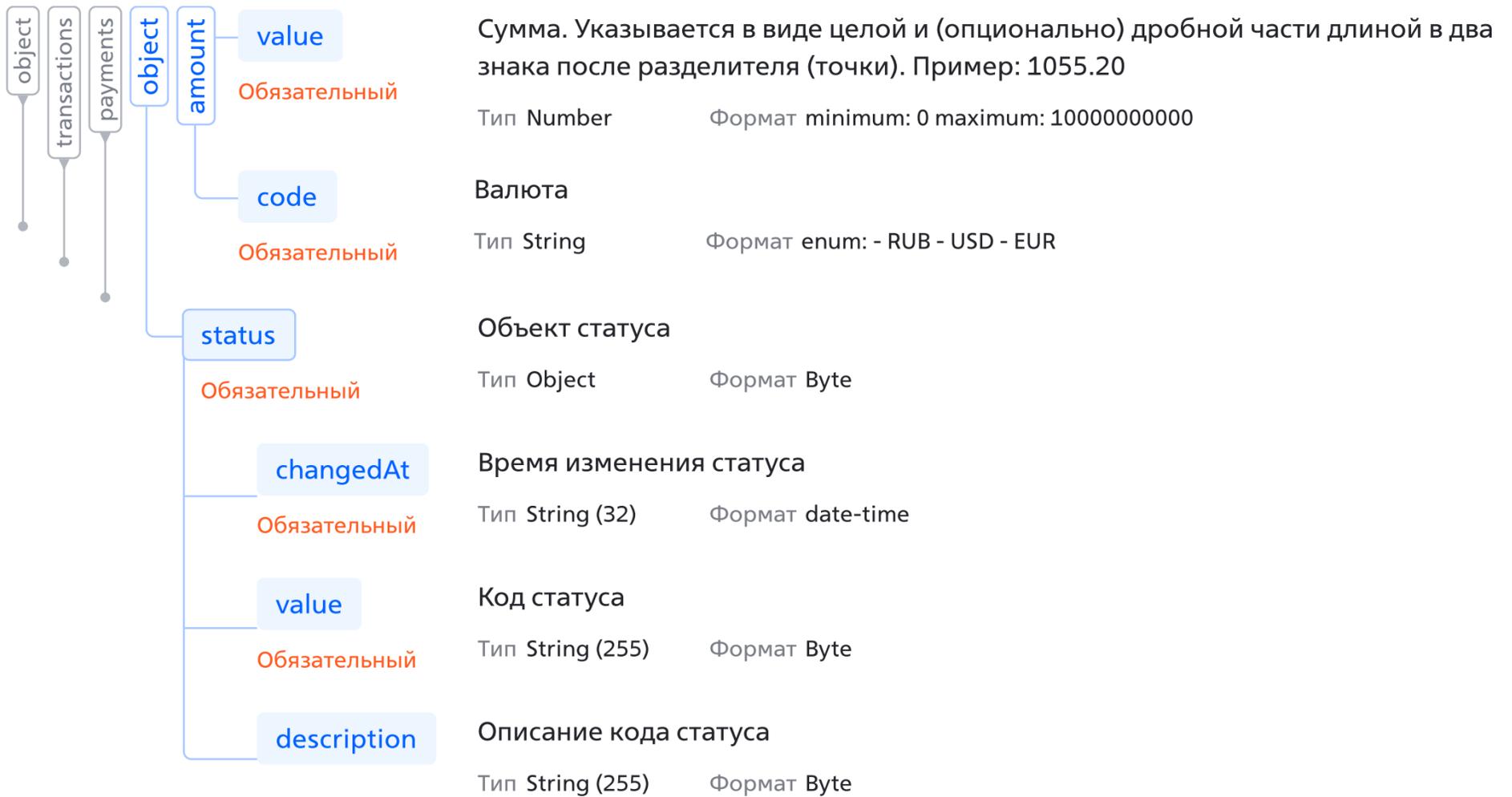
GET v1/orders/{orderId}

## Параметры ответа GET v1/orders/{orderId}

<b>type</b>	Тип объекта (ORDER)
Обязательный	Тип String      Формат enum: -ORDER
<b>object</b>	Объект ордера
Обязательный	Тип Object      Формат n/a
<b>orderId</b>	Уникальный идентификатор ордера в системе мерчанта
Обязательный	Тип String (64)      Формат Byte
<b>orderCode</b>	Уникальный внешний идентификатор ордера
Обязательный	Тип String (64)      Формат Byte
<b>changedAt</b>	Время создания ордера
Обязательный	Тип String (32)      Формат date-time
<b>expire</b>	Срок жизни ордера
Обязательный	Тип String (32)      Формат date-time
<b>payUrl</b>	Адрес платежной формы для редиректа клиента на страницу оплаты (в формате FQDN)
Обязательный	Тип String (255)      Формат Byte
<b>preparedPayments</b>	Дополнительные данные для осуществления оплаты
	Тип Array      Формат n/a
<b>type</b>	Тип оплаты
Обязательный	Тип String      Формат enum: - sbp - invoice
<b>object</b>	Объект для оплаты
Обязательный	Тип Object      Формат n/a
<b>url</b>	Ссылка для оплаты
Обязательный	Тип String (255)      Формат n/a
<b>amount</b>	Объект суммы
Обязательный	Тип Object      Формат n/a
<b>value</b>	Сумма. Указывается в виде целой и (опционально) дробной части длиной в два знака после разделителя (точки). Пример: 1055.20
Обязательный	Тип Number      Формат minimum: 0 maximum: 10000000000
<b>code</b>	Валюта
Обязательный	Тип String      Формат enum: - RUB - USD - EUR







### Пример ответа GET v1/orders/{orderId}

```
{ "type": "ORDER",
  "object": {
    "orderId": "ORDER10000000001",
    "orderCode": "848bb7b2-6961-4263-85dc-a4df9c377eod",
    "status": {
      "value": "REFUNDED",
      "description": "REFUNDED",
      "changedAt": "2022-12-27T12:02:52.483Z"},
    "createdAt": "2022-12-27T12:02:52.483Z",
    "payUrl": "https://pay.vtb.ru/?order-id=848bb7b2-6961-4263-85dc-a4df9c377eod",
    "expire": "2022-12-31T01:09:44.000Z",
    "amount": { "value": 10.56, "code": "RUB"},
    "transactions": {
      "payments": [{
        "type": "PAYMENT",
        "object": {
          "paymentId": "03492e87-c6f9-411b-23d8-4d715744f519",
          "orderId": "ORDER10000000001",
          "orderName": "Заказ №10000000101",
          "createdAt": "2022-12-27T12:03:45.149Z",
          "amount": { "value": 10.56, "code": "RUB"},
          "status": {
            "value": "CONFIRMED",
            "description": "CONFIRMED",
            "changedAt": "2022-12-27T12:11:50.660Z"},
          "paymentData": { "type": "card" }
        }
      }],
      "refunds": [{
        "type": "REFUND",
        "object": {
          "refundId": "REFUND100000000001",
          "paymentId": "REFUND100000000001",
          "paymentCode": "913328dc-d749-43e4-8c61-17de55be1e21",
          "orderId": "ORDER10000000001",
          "orderCode": "848bb7b2-6961-4263-85dc-a4df9c377eod",
          "createdAt": "2022-12-27T12:05:34.719Z",
          "status": {
            "value": "CONFIRMED",
            "description": "CONFIRMED",
            "changedAt": "2022-12-27T12:11:50.660Z"},
          "amount": { "value": 10.56, "code": "RUB" }
        }
      }
    ]
  }
}
```

### 4.12.3 POST v1/orders/preauth

POST v1/orders/preauth - запрос для создания ордера с предавторизацией (двухстадийный платёж). Данный запрос является идемпотентным (при отправке запроса с указанием ранее использовавшегося значения параметра "orderId" будет возвращен ответ с информацией по ранее созданному ордеру). Максимальный срок холдирования денежных средств зависит от правил платежной системы

! Для возможности создания ордеров с предавторизацией (двухстадийный платеж) необходимо получить специальное разрешение на стороне Платежного шлюза!

#### Параметры запроса POST v1/orders/preauth

<b>Merchant-Authorization</b>	Дополнительный (необязательный) HTTP-заголовок, в котором передается идентификатор ресурса, если у мерчанта подключено более одного ресурса к Платежному шлюзу	Url / body / header	Header	Тип n/a	Формат n/a
<b>orderId</b> Обязательный	Уникальный идентификатор ордера в системе мерчанта	Url / body / header	Body	Тип String	Формат Byte
<b>orderId</b> Обязательный	Наименование ордера в системе мерчанта	Url / body / header	Body	Тип String (255)	Формат Byte
<b>expire</b>	Срок жизни ордера. Указывается с обязательным указанием timezone (обозначение зоны для нулевого смещения UTC) и указанием 3-х знаков для обозначения миллисекунд. Пример: 2023-10-12T12:02:16.883Z. Если параметр и его значение не было передано при создании ордера, то срок жизни ордера будет задан автоматически путем прибавления 20 минут к времени создания ордера	Url / body / header	Body	Тип String (32)	Формат date-time
<b>customer</b>	Данные о клиенте	Url / body / header	Body	Тип Object	Формат n/a
<b>email</b>	Электронный адрес клиента (в формате email)	Url / body / header	Body	Тип String (255)	Формат Byte
<b>amount</b> Обязательный	Объект суммы	Url / body / header	Body	Тип Object	Формат n/a
<b>value</b> Обязательный	Сумма. Указывается в виде целой и (опционально) дробной части длиной в два знака после разделителя (точки). Пример: 1055.20	Url / body / header	Body	Тип Number	Формат minimum: 0; maximum: 10000000000
<b>code</b> Обязательный	Валюта	Url / body / header	Body	Тип String	Формат enum: - RUB - USD - EUR

returnUrl

URL-адрес фронт системы ТСП для переадресации клиента в случае успешной / неуспешной оплаты

Url / body / header Body

Тип String

Формат Byte

### Пример запроса POST v1/orders/preauth

```
{
  "orderId": "ORDER10000000001",
  "orderName": "Заказ №10000000101",
  "expire": "2022-12-31T01:09:44Z",
  "amount": {
    "value": 10.56,
    "code": "RUB"
  }
}
```

## Параметры ответа POST v1/orders/preauth

type

Тип объекта (ORDER)

Обязательный

Тип String

Формат enum: -ORDER

object

Объект ордера

Обязательный

Тип Object

Формат n/a

orderId

Уникальный идентификатор ордера в системе мерчанта

Обязательный

Тип String (64)

Формат Byte

orderCode

Уникальный внешний идентификатор ордера

Обязательный

Тип String (64)

Формат Byte

expire

Срок жизни ордера. В формате: 2022-10-12T12:02:16.883Z

Обязательный

Тип String (32)

Формат date-time

payUrl

Адрес платежной формы для редиректа клиента на страницу оплаты (в формате FQDN)

Обязательный

Тип String (255)

Формат Byte

amount

Объект суммы

Обязательный

Тип Object

Формат n/a

value

Сумма. Указывается в виде целой и (опционально) дробной части длиной в два знака после разделителя (точки).  
Пример: 1055.20

Обязательный

Тип Number

Формат minimum: 0 maximum: 10000000000

code

Валюта

Обязательный

Тип String

Формат enum: - RUB - USD - EUR

object	<b>status</b>	Объект статуса	Тип n/a	Формат n/a
	Обязательный			
	<b>changedAt</b>	Время изменения статуса. В формате: 2022-10-12T12:02:16.883Z	Тип String (32)	Формат date-time
	Обязательный			
	<b>value</b>	Валюта	Тип String (255)	Формат Byte
	Обязательный			
	<b>description</b>	Описание кода статуса	Тип String (255)	Формат Byte

### Пример ответа POST v1/orders/preauth

```
{
  "type": "ORDER",
  "object": {
    "orderId": "ORDER10000000001",
    "orderCode": "848bb7b2-6961-4263-85dc-a4df9c377eod",
    "status": {
      "value": "CREATED",
      "description": "CREATED",
      "changedAt": "2022-12-27T12:02:52.483Z"
    },
    "createdAt": "2022-12-27T12:02:52.483Z",
    "payUrl": "https://pay.vtb.ru/?order-id=848bb7b2-6961-4263-85dc-a4df9c377eod",
    "expire": "2022-12-31T01:09:44.000Z",
    "amount": {
      "value": 10.56,
      "code": "RUB"
    }
  }
}
```

## 4.12.4 POST v1/orders/complete/{orderId}

Запрос для завершения ранее предавторизованного ордера

### Параметры запроса POST v1/orders/complete/{orderId}

<b>Merchant-Authorization</b>	Необязательный HTTP-заголовок, который используется в запросах к API для дополнительной аутентификации ресурса мерчанта для тех случаев, когда у мерчанта подключено к Платежному шлюзу более одного ресурса (сайта).	Url / body / header Header	Тип String	Формат n/a
<b>orderId</b> Обязательный	Уникальный идентификатор ордера в системе мерчанта	Url / body / header URL	Тип String (64)	Формат Byte
<b>amount</b>	Объект суммы	Url / body / header Body	Тип Object	Формат n/a
<b>value</b> Обязательный	Сумма	Url / body / header Body	Тип Number	Формат minimum: 0 maximum: 10000000000
<b>code</b> Обязательный	Валюта	Url / body / header Body	Тип String	Формат enum: - RUB

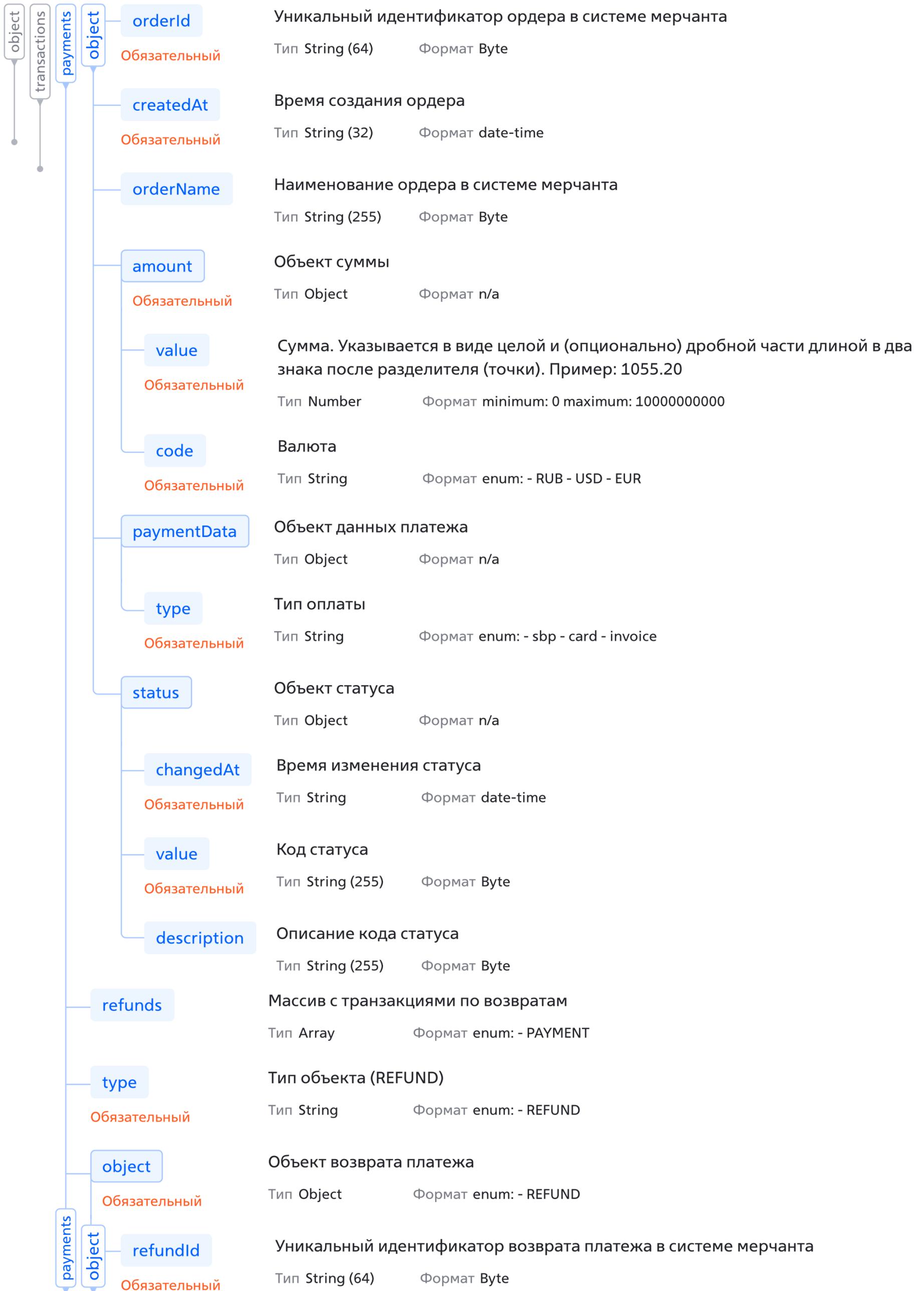
#### Пример запроса POST v1/orders/complete/{orderId}

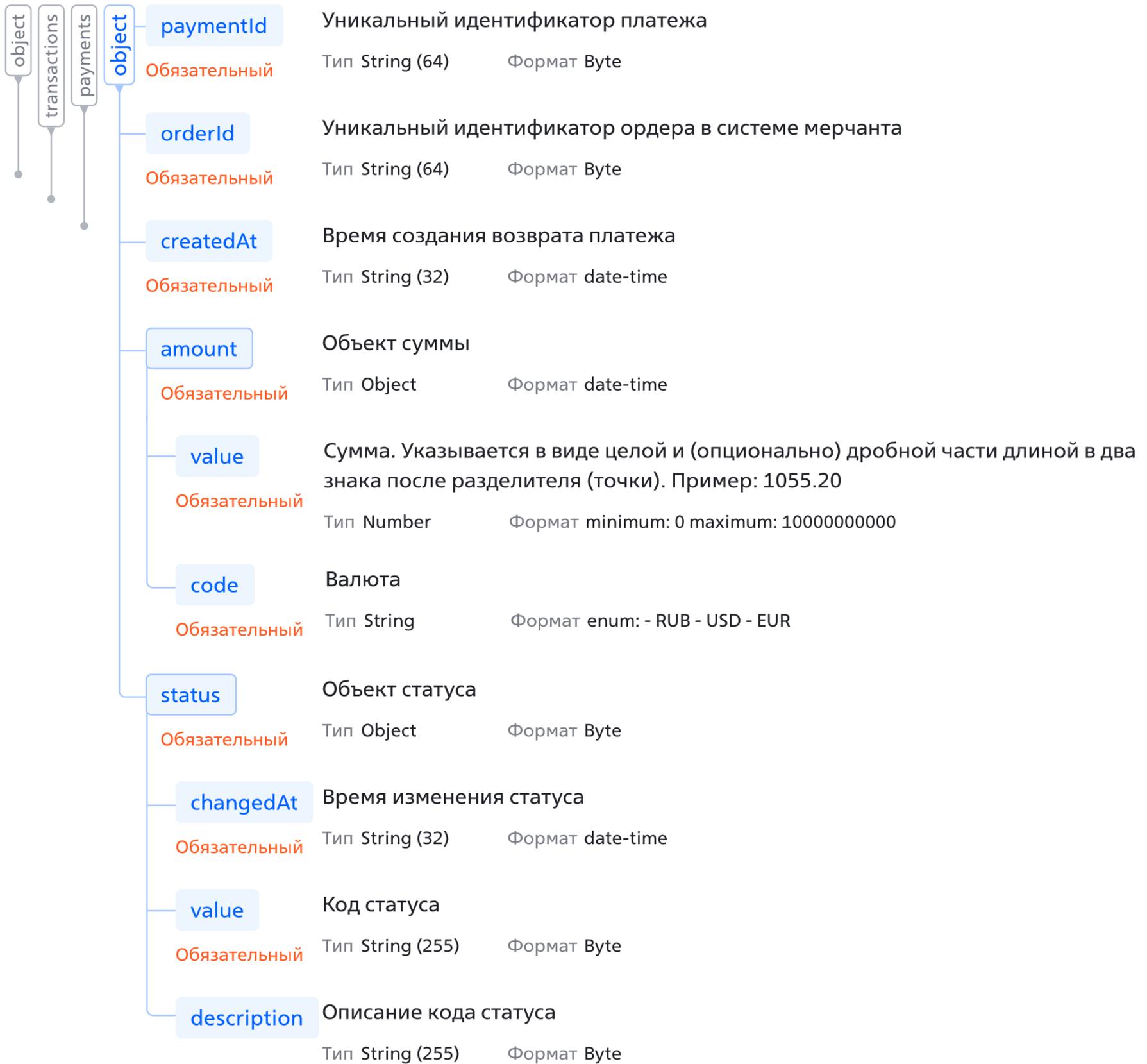
```
{
  "amount": {
    "value": 10.56,
    "code": "RUB"
  }
}
```

### Параметры ответа POST v1/orders/complete/{orderId}

<b>type</b> Обязательный	Тип объекта (ORDER)	Тип String	Формат enum: -ORDER
<b>object</b> Обязательный	Объект ордера	Тип Object	Формат n/a
<b>orderId</b> Обязательный	Уникальный идентификатор ордера в системе мерчанта	Тип String (64)	Формат Byte
<b>orderCode</b> Обязательный	Уникальный внешний идентификатор ордера	Тип String (64)	Формат Byte

object	changedAt	Обязательный	Время создания ордера	Тип String (32)	Формат date-time				
	expire	Обязательный	Срок жизни ордера	Тип String (32)	Формат date-time				
	payUrl	Обязательный	Адрес платежной формы для редиректа клиента на страницу оплаты (в формате FQDN)	Тип String (255)	Формат Byte				
	amount	Обязательный	Объект суммы	Тип Object	Формат n/a				
			value	Обязательный	Сумма. Указывается в виде целой и (опционально) дробной части длиной в два знака после разделителя (точки). Пример: 1055.20	Тип Number	Формат minimum: 0 maximum: 10000000000		
			code	Обязательный	Валюта	Тип String	Формат enum: - RUB - USD - EUR		
	status	Обязательный	Объект статуса	Тип n/a	Формат n/a				
			changedAt	Обязательный	Время изменения статуса	Тип String (32)	Формат date-time		
			value	Обязательный	Валюта	Тип String (255)	Формат Byte		
			description		Описание кода статуса				
	transactions	Обязательный	Блок информации по операциям в рамках ордера	Тип Object	Формат n/a				
			payments	Обязательный	Массив с транзакциями по платежам	Тип Array	Формат n/a		
					type	Обязательный	Тип объекта (PAYMENT)	Тип String	Формат enum: - PAYMENT
					object	Обязательный	Объект платежа	Тип Object	Формат enum: - PAYMENT
					paymentId	Обязательный	Уникальный идентификатор платежа	Тип String (64)	Формат Byte
object	Обязательный	orderId	Обязательный	Уникальный идентификатор ордера в системе мерчанта	Тип String (64)	Формат Byte			





### Пример ответа POST v1/orders/complete/{orderId}

```
{
  "type": "ORDER",
  "object": {
    "orderId": "ORDER10000000001",
    "orderCode": "848bb7b2-6961-4263-85dc-a4df9c377eod",
    "status": {
      "value": "REFUNDED",
      "description": "REFUNDED",
      "changedAt": "2022-12-27T12:02:52.483Z"
    },
  },
  "createdAt": "2022-12-27T12:02:52.483Z",
  "payUrl": "https://pay.vtb.ru/?order-id=848bb7b2-6961-4263-85dc-a4df9c377eod",
  "expire": "2022-12-31T01:09:44.000Z",
  "amount": {
    "value": 10.56,
    "code": "RUB"},
  "transactions": {
    "payments": [{
      "type": "PAYMENT",
      "object": {
        "paymentId": "03492e87-c6f9-411b-23d8-4d715744f519",
        "orderId": "ORDER10000000001",
        "orderName": "Заказ №10000000101",
        "createdAt": "2022-12-27T12:03:45.149Z",
        "amount": {
          "value": 10.56,
          "code": "RUB"},
      }
    ]
  },
  "status": {
    "value": "CONFIRMED",
    "description": "CONFIRMED",
    "changedAt": "2022-12-27T12:11:50.660Z"},
  "paymentData": {
    "type": "card"}
  }
}
```

## 4.12.5 POST /orders/{orderId}/cancel

Запрос для отмены заказа по его ID

### Параметры запроса POST /orders/{orderId}/cancel

<b>Merchant-Authorization</b>	Необязательный HTTP-заголовок, который используется в запросах к API для дополнительной аутентификации ресурса мерчанта для тех случаев, когда у мерчанта подключено к Платежному шлюзу более одного ресурса (сайта)			
Url / body / header	Header	Тип String	Формат n/a	
<b>orderId</b>	Уникальный идентификатор заказа в системе мерчанта			
Обязательный	Url / body / header	URL	Тип String (64)	Формат Byte

#### Пример запроса POST /orders/{orderId}/cancel

POST /orders/{orderId}/cancel

### Параметры ответа POST /orders/{orderId}/cancel

<b>type</b>	Тип объекта (ORDER)	
Обязательный	Тип String	Формат enum: -ORDER

<b>object</b>	Объект заказа	
Обязательный	Тип Object	Формат n/a
<b>orderId</b>	Уникальный идентификатор заказа в системе мерчанта	
Обязательный	Тип String (64)	Формат Byte
<b>orderCode</b>	Уникальный внешний идентификатор заказа	
Обязательный	Тип String (64)	Формат Byte
<b>changedAt</b>	Время создания заказа	
Обязательный	Тип String (32)	Формат date-time
<b>expire</b>	Срок жизни заказа	
Обязательный	Тип String (32)	Формат date-time
<b>payUrl</b>	Адрес платежной формы для редиректа клиента на страницу оплаты (в формате FQDN)	
Обязательный	Тип String (255)	Формат Byte

object	amount	Объект суммы	Тип Object	Формат n/a
	Обязательный			
	value	Сумма. Указывается в виде целой и (опционально) дробной части длиной в два знака после разделителя (точки). Пример: 1055.20	Тип Number	Формат minimum: 0 maximum: 10000000000
	Обязательный			
	code	Валюта	Тип String	Формат enum: - RUB - USD - EUR
	Обязательный			
	status	Объект статуса	Тип n/a	Формат n/a
	Обязательный			
	changedAt	Время изменения статуса	Тип String (32)	Формат date-time
	Обязательный			
value	Валюта	Тип String (255)	Формат Byte	
Обязательный				
description	Описание кода статуса	Тип String (255)	Формат Byte	

### Пример ответа POST /orders/{orderId}/cancel

```
{
  "type": "ORDER",
  "object": {
    "orderId": "16877832140400",
    "orderCode": "7ce50ee3-fd82-4e8a-9927-bbc60467d2a3",
    "status": {
      "value": "CANCELED",
      "description": "CANCELED",
      "changedAt": "2023-07-07T12:09:50.661Z"
    },
    "createdAt": "2023-07-07T12:09:50.661Z",
    "payUrl": "https://pay.vtb.ru/?order-id=7ce50ee4-fd82-4e8a-8827-bbc60467d2a3",
    "expire": "2023-07-07T12:29:50.661Z",
    "amount": {
      "value": 1.0,
      "code": "RUB"
    }
  },
}
```

## 4.12.6 Создание ордера (заказа) с CoF

### Создание ордера (заказа) с CoF

Создание связок CoF и оплата заказов с их использованием возможна только для ордеров с одностадийной оплатой (одностадийный платеж).

Для создания связок и оплаты с помощью ранее созданных связок используется эндпоинт POST v1/orders (см. описание параметров запроса в разделе «4.12.1 POST v1/orders» данной инструкции)

**!** Для возможности создания и оплаты заказов с использованием связок CoF необходимо получить специальное разрешение на стороне Платежного шлюза!

Операции с использованием ранее сохраненных карт (CoF) предусматривают первичную операцию, во время которой данные карты сохраняются, и последующие операции, которые осуществляются с использованием уже сохраненных ранее данных карты.

Первичная операция проводится с обязательным получением согласия плательщика на сохранение реквизитов карты для проведения последующих операций.

Первичная операция CoF проводится с верификацией с использованием CVC2 и 3D-Secure.

Повторные операции могут осуществляться после успешного проведения первичной операции с обязательным согласием плательщика на дальнейшее использование данных карты.

Повторные операции проводятся без процедуры повторной верификации, если это позволяет протокол работы процессинга.

Повторная верификация может быть включена по требованию процессинга или мерчанта, желающего защитить также и повторные транзакции.

## Разновидности операций оплаты с помощью CoF

**Customer Initialized Transaction (CIT)** — повторные операции, которые инициируются плательщиком. Мерчант ведет базу своих клиентов (покупателей), присваивая им уникальные номера (customerId), а Платежный шлюз хранит в зашифрованном виде номера банковских карт, соответствующих номерам клиентов (customerId) данного мерчанта.

При оплате очередного заказа клиент получает предложение от системы мерчанта запомнить карту для следующих покупок с специального сервиса.

Данный сервис мерчанта дает возможность постоянному клиенту мерчанта совершать новые покупки без ввода данных карты, подтверждая оплату только указанием кода CVC2.

Однако, для последующих покупок по сохраненной карте CVC2 обычно запрашиваться не будет.

**Merchant Initialized Transaction (MIT)** — повторные операции, которые инициируются мерчантом.

## Сценарии проведения операций MIT (иницированных ТСП)

**Установочный платеж** — выполняется при выполнении первого установочного платежа (как правило 3D-Secure-аутентифицированного), который является основой для последующих периодических (рекуррентных) платежей без участия держателя банковской карты по сохраненным на стороне ТСП реквизитам карты, либо платежей, инициированных держателем банковской карты с использованием сохраненных на стороне ТСП реквизитов банковской карты. Например, если во время выполнения платёжной операции держателю банковской карты было предложено сохранить данные банковской карты для последующего использования, или предполагается выполнение в дальнейшем периодических платежей по договоренности с держателем банковской карты.

**Рекарринговые платежи (RECURRING)** — иницируется ТСП и выполняются по длительному поручению клиента в течение неопределенного срока.

Используется при выполнении самих периодических (рекуррентных) платежей, иницированных ТСП без участия держателя карты по сохраненным на стороне ТСП реквизитам банковской карты.

## **Сценарии проведения операций СІТ (иницированных покупателем)**

**Установочный платеж** — выполняется при выполнении первого установочного платежа (как правило 3D-Secure-аутентифицированного), который является основой для последующих периодических (рекуррентных) платежей без участия держателя банковской карты по сохраненным на стороне ТСП реквизитам карты, либо платежей, иницированных держателем банковской карты с использованием сохраненных на стороне ТСП реквизитов банковской карты. Например, если во время выполнения платёжной операции держателю банковской карты было предложено сохранить данные банковской карты для последующего использования, или предполагается выполнение в дальнейшем периодических платежей по договоренности с держателем банковской карты.

**Common** — пользовательский платеж, при котором ТСП использует ранее сохраненные покупателем реквизиты банковской карты.

Используется при выполнении платежей, иницированных держателем карты с использованием сохраненных на стороне ТСП реквизитов карты.

Например, держатель карты в каком-либо платежном приложении указывает ссылку на сохраненную карту и не указывает никаких других ее реквизитов (CVV2/CVC2/CVP2).

## 4.13 Операции с возвратами

### 4.13.1 POST v1/refunds

POST v1/refunds – запрос для создания возврата платежа. Возврат платежа может быть как полным, так и частичным (Платежный шлюз во время выполнения частичного возврата определяет оставшуюся сумму для возврата).

#### Параметры запроса POST v1/refunds

<b>Merchant-Authorization</b>	Дополнительный (необязательный) HTTP-заголовок, в котором передается идентификатор ресурса, если у мерчанта подключено более одного ресурса к Платежному шлюзу	Url / body / header	Header	Тип n/a	Формат n/a
<b>refundId</b> Обязательный	Уникальный идентификатор возврата платежа в системе мерчанта	Url / body / header	Body	Тип String (64)	Формат Byte
<b>paymentId</b> Обязательный	Уникальный идентификатор платежа	Url / body / header	Body	Тип String (64)	Формат Byte
<b>amount</b> Обязательный	Объект суммы	Url / body / header	Body	Тип Object	Формат n/a
<b>value</b> Обязательный	Сумма. Указывается в виде целой и (опционально) дробной части длиной в два знака после разделителя (точки). Пример: 1055.20	Url / body / header	Body	Тип Number	Формат minimum: 0; maximum: 10000000000
<b>code</b> Обязательный	Валюта	Url / body / header	Body	Тип String	Формат enum: - RUB - USD - EUR

#### Пример запроса POST v1/refunds

```
{
  "refundId": "REFUND100000000001",
  "paymentId": "03492e87-c6f9-411b-23d8-4d715744f519",
  "amount": {
    "value": 10.56,
    "code": "RUB"
  }
}
```

## Параметры ответа POST v1/refunds

<b>type</b>	Тип объекта (REFUND)
Обязательный	Тип String      Формат enum: - REFUND
<hr/>	
<b>object</b>	Объект возврата платежа
Обязательный	Тип Object      Формат n/a
<b>refundId</b>	Уникальный идентификатор возврата платежа в системе мерчанта
Обязательный	Тип String (64)      Формат Byte
<b>paymentId</b>	Уникальный идентификатор транзакции платежа
Обязательный	Тип String (64)      Формат Byte
<b>paymentCode</b>	Уникальный внешний идентификатор транзакции для выполнения операции возврата
Обязательный	Тип String (64)      Формат Byte
<b>orderId</b>	Уникальный идентификатор ордера в системе мерчанта
Обязательный	Тип String (64)      Формат Byte
<b>orderCode</b>	Уникальный внешний идентификатор ордера
Обязательный	Тип String (64)      Формат Byte
<b>createdAt</b>	Время создания возврата платежа. В формате: 2022-10-12T12:02:16.883Z
Обязательный	Тип String (32)      Формат date-time
<b>amount</b>	Объект суммы
Обязательный	Тип Object      Формат n/a
<b>value</b>	Сумма. Указывается в виде целой и (опционально) дробной части длиной в два знака после разделителя (точки). Пример: 1055.20
Обязательный	Тип Number      Формат minimum: 0 maximum: 10000000000
<b>code</b>	Валюта
Обязательный	Тип String      Формат enum: - RUB - USD - EUR
<b>status</b>	Объект статуса
Обязательный	Тип Object      Формат n/a
<b>changedAt</b>	Время изменения статуса. В формате: 2022-10-12T12:02:16.883Z
Обязательный	Тип String (32)      Формат date-time
<b>value</b>	Код статуса
Обязательный	Тип String (255)      Формат Byte
<b>description</b>	Описание кода статуса
Обязательный	Тип String (255)      Формат Byte

### Пример ответа POST v1/refunds

```
{
  "type": "REFUND",
  "object": {
    "refundId": "REFUND100000000001",
    "paymentId": "REFUND100000000001",
    "paymentCode": "913328dc-d749-43e4-8c61-17de55be1e21",
    "orderId": "ORDER100000000001",
    "orderCode": "848bb7b2-6961-4263-85dc-a4df9c377eod",
    "createdAt": "2022-12-27T12:05:34.719Z",
    "status": {
      "value": "NEW",
      "description": "NEW",
      "changedAt": "2022-12-27T12:05:34.719Z"
    },
    "amount": {
      "value": 10.56,
      "code": "RUB"
    }
  }
}
```

## 4.13.2 GET v1/refunds/{refundId}

GET v1/refunds/{refundId} – запрос для поиска информации по возврату платежа по его ID

### Параметры запроса GET v1/refunds/{refundId}

Merchant-Authorization

HTTP-заголовок для аутентификации сайта. Значение заголовка соответствует идентификатору сайта в Платежном шлюзе и выдается представителем Платежного шлюза.

Url / body / header Header Тип n/a

Формат n/a

refundId

Уникальный идентификатор возврата платежа в системе мерчанта

Обязательный

Url / body / header URL Тип String (64)

Формат Byte

#### Пример запроса GET v1/refunds/{refundId}

GET v1/refunds/{refundId}

### Параметры ответа GET v1/refunds/{refundId}

type

Тип объекта (REFUND)

Обязательный

Тип String

Формат n/a

object

Объект возврата платежа

Обязательный

Тип Object

Формат n/a

refundId

Уникальный идентификатор возврата платежа в системе мерчанта

Обязательный

Тип String (64)

Формат Byte

paymentId

Уникальный идентификатор транзакции платежа

Обязательный

Тип String (64)

Формат Byte

paymentCode

Уникальный внешний идентификатор транзакции для выполнения операции возврата

Обязательный

Тип String (64)

Формат Byte

orderId

Уникальный идентификатор ордера в системе мерчанта

Обязательный

Тип String (64)

Формат Byte

orderCode

Уникальный внешний идентификатор ордера

Обязательный

Тип String (64)

Формат Byte

createdAt

Время создания возврата платежа. В формате: 2022-10-12T12:02:16.883Z

Обязательный

Тип String (32)

Формат date-time

object

object	<b>amount</b>	Объект суммы	Тип Object	Формат n/a
	Обязательный			
	<b>value</b>	Сумма. Указывается в виде целой и (опционально) дробной части длиной в два знака после разделителя (точки). Пример: 1055.20	Тип Number	Формат minimum: 0 maximum: 10000000000
	Обязательный			
	<b>code</b>	Валюта	Тип String	Формат enum: - RUB - USD - EUR
	Обязательный			
	<b>status</b>	Объект статуса	Тип Object	Формат n/a
Обязательный				
<b>changedAt</b>	Время изменения статуса. В формате: 2022-10-12T12:02:16.883Z	Тип String (32)	Формат date-time	
Обязательный				
<b>value</b>	Код статуса	Тип String (255)	Формат Byte	
Обязательный				
<b>description</b>	Описание кода статуса	Тип String (255)	Формат Byte	
Обязательный				

#### Пример ответа GET v1/refunds/{refundId}

```
{
  "type": "REFUND",
  "object": {
    "refundId": "REFUND100000000001",
    "paymentId": "REFUND100000000001",
    "paymentCode": "913328dc-d749-43e4-8c61-17de55be1e21",
    "orderId": "ORDER100000000001",
    "orderCode": "848bb7b2-6961-4263-85dc-a4df9c377eod",
    "createdAt": "2022-12-27T12:05:34.719Z",
    "status": {
      "value": "CONFIRMED",
      "description": "CONFIRMED",
      "changedAt": "2022-12-27T12:05:34.719Z"
    },
    "amount": {
      "value": 10.56,
      "code": "RUB"
    }
  }
}
```

## 4.14 Уведомления (callback) от Платежного шлюза

Платежный шлюз предоставляет функционал по отправке ресурсу мерчанта уведомлений (callback) по статусам совершенных операций, которые были выполнены через Платежный шлюз.

Настроенное получение уведомлений (callback) от Платежного шлюза на стороне ресурса мерчанта позволяют не запрашивать GET-запросом (по несколько раз) информацию по операции, выполненной на стороне Платежного шлюза.

Для получения уведомлений (callback) от Платежного шлюза необходимо предоставить представителю Платежного шлюза специальный URL-адрес на стороне ресурса мерчанта, по которому будет настроен прием и обработка уведомлений (callback) от Платежного шлюза.

URL-адрес можно предоставить во время регистрации на стороне Платежного шлюза или сообщить представителю Платежного шлюза позже для его последующего добавления в настройки Платежного шлюза.

В Платежном шлюзе одному ресурсу мерчанта может соответствовать только один адрес для получения уведомлений (callback).

## 4.15 Типы отправляемых уведомлений (callback) от Платежного шлюза

Тип	Описание
PaymentResponse	Результат оплаты
RefundResponse	Результат возврата оплаты

## 4.15.1 PaymentResponse

### Пример PaymentResponse

```
{
  "type": "PAYMENT",
  "object": {
    "paymentId": "269b0e7f-cec9-490f-9873-dcbb9852da42",
    "paymentCode": "a851f018-961e-4d1b-9302-2c1211a30753",
    "orderId": "65413254",
    "orderCode": "c44a8e04-5f62-4533-b000-ffd483e4ee8b",
    "createdAt": "2022-11-18T03:49:21.094150",
    "description": "Заказ №65413254",
    "amount": {
      "value": 954.00,
      "code": "RUB"
    },
    "paymentData": {
      "type": "CARD"
    },
    "Status": {
      "changedAt": "2022-11-18T03:49:57.883811",
      "value": "CONFIRMED",
      "description": "ONFIRMED"
    }
  }
}
```

### Описание структуры уведомления PaymentResponse

<b>type</b>	Тип объекта (PAYMENT)
Обязательный	Тип string      Формат enum: - PAYMENT
<hr/>	
<b>object</b>	Объект платежа
Обязательный	Тип Object      Формат n/a
<b>paymentId</b>	Идентификатор транзакции, передаваемый партнеру
Обязательный	Тип String (64)      Формат Byte
<b>paymentCode</b>	Уникальный внешний идентификатор транзакции
Обязательный	Тип String      Формат Byte
<b>orderId</b>	Уникальный идентификатор заказа в системе мерчанта
Обязательный	Тип String (64)      Формат Byte
<b>orderCode</b>	Уникальный внешний идентификатор заказа
Обязательный	Тип String      Формат Byte
<b>createdAt</b>	Время создания заказа
Обязательный	Тип String (32)      Формат date-time

object	description	Дополнительная информация	Тип String (255)	Формат Byte	
	amount	Объект суммы	Тип Object	Формат n/a	
	Обязательный				
	value	Границы значений суммы	Тип number	Формат minimum: 0 maximum: 10000000000	
	Обязательный				
	code	Валюта	Тип String	Формат enum: RUB	
	Обязательный				
	paymentData	Объект данных платежа	Тип Object	Формат n/a	
	type	Тип обработки платежа	Тип String	Формат enum: sbp, card	
	Обязательный				
	bindingId	Уникальный идентификатор связки CoF на стороне Платежного шлюза	Url / body / header Body	Тип String(64) or null	Формат Byte
	bindingType	Тип связки CoF. Возможные типы связок: recurring - рекарринговые платежи installment - установочный платеж common - пользовательский платеж	Url / body / header Body	Тип String or null	Формат enum: recurring, installment, common
	bindingState	Статус связки CoF. Возможные значения: active - активная deactivated - деактивированная	Url / body / header Body	Тип String or null	Формат enum: active, deactivated
	status	Объект статуса	Тип Object	Формат n/a	
Обязательный					
changedAt	Время изменения статуса	Тип String (32)	Формат date-time		
Обязательный					
value	Код статуса	Тип String (255)	Формат Byte		
Обязательный					
description	Описание кода статуса	Тип String (255)	Формат Byte		

## 4.15.2 Уведомление RefundResponse

### Пример RefundResponse

```
{
  "type": "REFUND",
  "object": {
    "refundId": "65546412-2",
    "paymentId": "65546412-2",
    "paymentCode": "c765dec8-ec7f-4231-86b7-e24c2a7830c8",
    "orderId": "65546412",
    "orderCode": "bb587bb1-cc4e-231e-a7ad-0e57644b491e",
    "createdAt": "2022-11-08T11:22:54.054205",
    "amount": {
      "value": 1250.0,
      "code": "RUB"
    },
    "Status": {
      "changedAt": "2022-11-20T06:05:34.703188",
      "value": "CONFIRMED",
      "description": "CONFIRMED"
    }
  }
}
```

## Описание структуры уведомления RefundResponse

<b>type</b>	Тип объекта (REFUND)
Обязательный	Тип String      Формат enum: - REFUND
<hr/>	
<b>object</b>	Объект возврата платежа
Обязательный	Тип Object      Формат n/a
<b>refundId</b>	Уникальный идентификатор возврата платежа в системе мерчанта
Обязательный	Тип String (64)      Формат Byte
<b>paymentId</b>	Идентификатор транзакции на возврат, передаваемый партнеру
Обязательный	Тип String (64)      Формат Byte
<b>paymentCode</b>	Уникальный внешний идентификатор транзакции, которая выполняет возврат
Обязательный	Тип String (64)      Формат Byte
<b>orderId</b>	Уникальный идентификатор ордера в системе мерчанта
Обязательный	Тип String (64)      Формат Byte
<b>orderCode</b>	Уникальный внешний идентификатор ордера
Обязательный	Тип String (64)      Формат Byte
<b>createdAt</b>	Время создания возврата платежа
Обязательный	Тип String (32)      Формат date-time
<b>amount</b>	Объект суммы
Обязательный	Тип Object      Формат n/a
<b>value</b>	Границы значений суммы
Обязательный	Тип Number      Формат minimum: 0 maximum: 10000000000
<b>code</b>	Валюта
Обязательный	Тип String      Формат enum: - RUB
<b>status</b>	Объект статуса
Обязательный	Тип Object      Формат n/a
<b>changedAt</b>	Время изменения статуса
Обязательный	Тип String (32)      Формат date-time
<b>value</b>	Код статуса
Обязательный	Тип String (255)      Формат Byte
<b>description</b>	Описание кода статуса
	Тип String (255)      Формат Byte

## 4.16 Интеграция с тестовой средой Платежного шлюза

Перед подключением к Production среде API мерчанту необходимо ознакомиться с инструкций по работе с API и успешно выполнить тестовые кейсы на тестовой среде Платежного шлюза ("Песочнице").

После успешного выполнения тестовых кейсов мерчант должен прислать результаты выполнения тестирования ответственным лицам от Платежного шлюза для анализа готовности к интеграции.

После выполнения проверки результатов тестирования на стороне Платежного шлюза принимается решение о возможности подключения сайта мерчанта к Production среде Платежного шлюза.

### 4.16.1 Настройки для подключения к тестовой среде ("Песочница")

Перед отправкой каждого нового запроса к API тестовой среды Платежного шлюза ("Песочница") необходимо запросить access\_token по URL-адресу менеджера доступа.

Адрес менеджера доступа для получения access\_token:

<https://auth.bankingapi.ru/auth/realms/kubernetes/protocol/openid-connect/token>

Срок действия access\_token составляет 300 секунд (5 минут).

Access\_token можно использовать множество раз в течение его срока жизни, который указывается в значении параметра "expires\_in" (в секундах) в ответе на запрос на получение access\_token.

POST <https://auth.bankingapi.ru/auth/realms/kubernetes/protocol/openid-connect/token> – запрос для получения access\_token

### Параметры запроса

 В запросе на получение access\_token необходимо использовать формат x-www-form-urlencoded для передачи формы с параметрами!

Параметр	Url/body/header	Описание
grant_type	body	Статичный параметр. Указывается значение "client_credentials"
client_id	body	Указывается значение client_id, полученное от представителя Платежного шлюза
client_secret	body	Указывается значение client_secret, полученное от представителя Платежного шлюза



## 4.16.2 Отправка запросов к API Платежного шлюза ("Песочница")

После получения `access_token` (через запрос к Менеджеру доступа тестовой среды) возможно перейти к шагу по отправке запроса к API Платежного шлюза на тестовой среде ("Песочница").

Адрес для обращения к эндпоинтам API на тестовой среде ("Песочница"):

[https://hackaton.bankingapi.ru/api/smb/efcp/e-commerce/api/v1/{наименование\\_эндпоинта}](https://hackaton.bankingapi.ru/api/smb/efcp/e-commerce/api/v1/{наименование_эндпоинта})

При отправке запроса к API Платежного шлюза в запрос необходимо включить HTTP-заголовок "Authorization", дополнительные HTTP-заголовки для идентификации клиента и его сайта ("X-IBM-Client-Id", "Merchant-Authorization") и заголовок для указания типа передаваемого содержимого ("Content-Type").

### HTTP-заголовки для использования в запросах к API

Заголовок	Описание
X-IBM-Client-Id	Указывается значение <code>client_id</code> , полученное от представителя Платежного шлюза. Пример: <code>ext.cmm.3e9d5564-dae7-4b66-b740-63ed7782e155</code>  <div style="background-color: #f4a460; padding: 5px; border-radius: 10px;">Значение необходимо прописать в нижнем регистре, без домена <code>@ext.vtb.ru</code>: например, <code>EXT.ORG1sdcsiocsy@ext.vtb.ru</code> -&gt; <code>ext.org1sdcsiocsy</code></div>
Merchant-Authorization	Значение заголовка соответствует уникальному идентификатору ресурса мерчанта. Пример: <code>Merchant-Authorization = 5d761993-d444-450a-ad9e-097fa52055ff</code>  <div style="background-color: #f4a460; padding: 5px; border-radius: 10px;">Передача заголовка и его значения обязательна только в том случае, если у мерчанта подключено более одного ресурса к Платежному шлюзу</div>
Authorization	Указывается значение полученного Bearer <code>access_token</code> в результате запроса к Менеджеру доступа. Пример: <code>Authorization: Bearer AQIC5wM2LY4SfczCimpZsQyW3PP3NZXNPHHzDLQDZ1qSDcE.*AAJTSQACMDIAAIN LABQtNDk1NDQwMDQ2MTkyMTIzNTM3NwACUzEAAjA</code>
Content-Type	Тип передаваемого содержимого. Указывается значение <code>"application/json"</code>

### Примеры использования HTTP-заголовков

#### Пример тела запроса

```
curl --location --request POST 'https://hackaton.bankingapi.ru/api/smb/efcp/e-commerce/api/v1/orders' \
--header 'X-IBM-Client-Id: ext.cmm.3e9d5564-dae7-4b66-b740-63ed7782e155' \
--header 'Merchant-Authorization: 5d761993-d444-450a-ad9e-097fa52055ff' \
--header 'Authorization: Bearer AQIC5wM2LY4Sfcx7ug0mf6PKkx1BR4M5U8ift6Poi_OqsE.*AAJTSQACMDIAAINLABMtnTAyNDY0NTQ0MjA3MjQ1Mjc5AAJTMQAC MDE.*' \
--header 'Content-Type: application/json' \
--data-raw '{
.....
}'
```

## 4.17 Тестирование API

### 4.17.1 Тестовые кейсы

#### Оплата через СБП

1. Выполнить запрос регистрации ордера по эндпоинту [POST / orders](#)
2. Перейти на платежную форму по полученной ссылке в параметре "payurl"
3. Нажать на кнопку СБП для отображения QR-кода.
4. Провести оплату по QR-коду, выполнив запрос [POST / qr/success](#)
5. Получить результат оплаты в callback-уведомлении
6. Если не получено callback-уведомление, выполнить запрос информации по ордеру через эндпоинт [GET / orders/{orderId}](#)

#### Полный возврат оплаты через СБП

1. Выполнить запрос регистрации ордера [POST /orders](#)
2. Перейти на платежную форму по полученной ссылке в параметре "payurl"
3. Нажать на кнопку СБП для отображения QR-кода
4. Провести оплату по QR-коду, выполнив запрос [POST /qr/success](#)
5. Получить результат оплаты в callback-уведомлении
6. Если не получено callback-уведомление, выполнить запрос информации по ордеру [GET /orders/{orderId}](#)
7. Выполнить запрос на возврат [POST /refunds](#), указав полную сумму ордера
8. Получить результат возврата в callback-уведомлении
9. Если не получено callback-уведомление, выполнить запрос информации по возврату [GET /refunds/{refundId}](#)

#### Частичный возврат оплаты через СБП

1. Выполнить запрос регистрации ордера [POST /orders](#)
2. Перейти на платежную форму по полученной ссылке в параметре "payurl"
3. Нажать на кнопку СБП для отображения QR-кода
4. Провести оплату по QR-коду, выполнив запрос [POST / qr/success](#)
5. Получить результат оплаты в callback-уведомлении.
6. Если не получено callback-уведомление, выполнить запрос информации по ордеру [GET /orders/{orderId}](#)
7. Выполнить запрос на возврат [POST /refunds](#), указав часть суммы ордера
8. Получить результат возврата в callback-уведомлении
9. Если не получено callback-уведомление, выполнить запрос информации по возврату [GET /refunds/{refundId}](#)

#### Оплата по карте

1. Выполнить запрос регистрации ордера по эндпоинту [POST /orders](#)
2. Перейти на платежную форму по полученной ссылке в параметре "payurl"
3. Выполнить оплату по тестовой карте на платежной форме
4. Получить результат оплаты в callback-уведомлении
5. Если не получено callback-уведомление, выполнить запрос информации по ордеру через эндпоинт [GET / orders/{orderId}](#)

## Полный возврат оплаты по карте

1. Выполнить запрос регистрации ордера [POST / orders](#)
2. Перейти на платежную форму по полученной ссылке в параметре "payurl"
3. Выполнить оплату по тестовой карте на платежной форме
4. Получить результат оплаты в callback-уведомлении
5. Если не получено callback-уведомление, выполнить запрос информации по ордеру [GET /orders/{orderId}](#)
6. Выполнить запрос на возврат [POST /refunds](#), указав полную сумму ордера
7. Получить результат возврата в callback-уведомлении
8. Если не получено callback-уведомление, выполнить запрос информации по возврату [GET /refunds/{refundId}](#)

## Частичный возврат оплаты по карте

1. Выполнить запрос регистрации ордера [POST /orders](#)
2. Перейти на платежную форму по полученной ссылке в параметре "payurl"
3. Выполнить оплату по тестовой карте на платежной форме на платежной форме
4. Получить результат оплаты в callback-уведомлении
5. Если не получено callback-уведомление, выполнить запрос информации по ордеру [GET /orders/{orderId}](#)
6. Выполнить запрос на возврат [POST /refunds](#), указав часть суммы ордера
7. Получить результат возврата в callback-уведомлении
8. Если не получено callback-уведомление, выполнить запрос информации по возврату [GET /refunds/{refundId}](#)

## Тестовые эндпоинты ("заглушки") для имитации обработки оплаты ордера на стороне Системы быстрых платежей (СБП)

Для имитации работы Системы быстрых платежей (СБП) в тестовой среде ("Песочнице") предусмотрены запросы, которые могут выполнить имитацию успешной или неуспешной оплаты на стороне СБП, URL-адрес для отправки запросов к заглушкам СБП: <https://qr-efcp.bankingapi.ru/>

Совет: значение URL для оплаты в системе быстрых платежей можно посмотреть в консоли разработчика в веб-браузере (после генерации QR-кода на платежной форме), нажав клавишу F12.

Во вкладке "Network" в списке будет выведен объект "sbp-link", внутри которого в параметре "link" можно узнать URL для подстановки в заглушку для имитации успешной или неуспешной оплаты со стороны СБП,

## POST / qr / success

POST /qr/success – тестовый эндпоинт для имитации проведения успешной оплаты на стороне СБП (используется только для Песочницы).

### Пример запроса

```
{
  "qrCodeLink": "https://qr.nspk.ru/7030303567957146?type=01&bank=10000000005"
}
```

### Пример ответа

```
{
  "sent": true,
  "qrId": "7030303567957146"
}
```

## POST / qr / fail

POST / qr / fail – тестовый эндпоинт для имитации проведения неуспешной оплаты на стороне СБП (используется только для Песочницы).

### Пример запроса

```
{
  "qrCodeLink": "https://qr.nspk.ru/7030303567957146?type=01&bank=100000000005"
}
```

### Пример ответа

```
{
  "sent": true,
  "qrId": "7030303567957146"
}
```

## 4.17.2 Реквизиты тестовой банковской карты

Номер карты: 4111 1111 1111 1111

Дата истечения срока действия: 12 / 24

Проверочный код на обратной стороне карты (CVV / CVC): 123

## 4.18 Интеграция с PRODUCTION средой Платежного шлюза

### 4.18.1 Настройки для подключения к PRODUCTION среде Платежного шлюза

Перед отправкой каждого нового запроса к API Платежного шлюза необходимо запросить `access_token` по URL-адресу менеджера доступа.

Адрес менеджера доступа для получения `access_token`: <https://open.api.vtb.ru:443/passport/oauth2/token>

Срок действия `access_token` не имеет фиксированного значения, но составляет примерно 170 секунд.

`Access_token` можно использовать множество раз в течение его срока жизни, который указывается в значении параметра `"expires_in"` (в секундах) в ответе на запрос на получение `access_token`

Обратите внимание, что длина полученного значения `access_token` для PRODUCTION-среды отличается от длины `access_token` на тестовой среде в меньшую сторону.

POST <https://open.api.vtb.ru:443/passport/oauth2/token>

POST <https://open.api.vtb.ru:443/passport/oauth2/token> – запрос для получения `access_token`

## Параметры запроса

! В запросе на получение access\_token необходимо использовать формат x-www-form-urlencoded для передачи формы с параметрами!

Параметр	Url/body/header	Описание
grant_type	body	Статичный параметр. Указывается значение "client_credentials"
client_id	body	Указывается значение client_id, полученное от представителя Платежного шлюза
client_secret	body	Указывается значение client_secret, полученное от представителя Платежного шлюза

## Параметры ответа

Параметр	Описание
access_token	Значение полученного access_token. Используется в HTTP заголовке Authorization при отправке запроса к API
expires_in	Время жизни access_token в секундах
token_type	Тип access_token. Возвращается значение "Bearer"

### Пример тела запроса

```
*** Формат тела запроса x-www-form-urlencoded!!!!
grant_type=client_credentials
client_id=ext.cmm.3e9d5564-dae7-4b66-b740-63ed7782e155@ext.vtb.ru
client_secret=Pq(29!KoOjeAb6hs
```

### Пример ответа

```
{
  "access_token":
  "AQIC5wM2LY4SfcxA7ug0mf6PKkx1BR4M5U8ift6Poi_OqsE.*AAJTSQACMDIAAINLABMtNTAyNDY0NTQ0MjA3MjQ1Mjc5AAJTMQAC
  MDE.*",
  "expires_in": 173,
  "token_type": "Bearer"
}
```

## 4.18.2 Отправка запросов к API Платежного шлюза (PRODUCTION)

После получения `access_token` (через запрос к Менеджеру доступа PRODUCTION среды) возможно перейти к шагу по отправке запроса к API Платежного шлюза на PRODUCTION среде.

Адрес для обращения к эндпоинтам API на PRODUCTION среде: [https://gw.api.vtb.ru/openapi/smb/efcp/e-commerce/v1/{наименование\\_эндпоинта}](https://gw.api.vtb.ru/openapi/smb/efcp/e-commerce/v1/{наименование_эндпоинта})

При отправке запросов к API Платежного шлюза в запрос необходимо включать следующие HTTP-заголовки, указанные в таблице ниже.

При отправке запроса к API Платежного шлюза в запрос необходимо включить HTTP-заголовок "Authorization", дополнительные HTTP-заголовки для идентификации клиента и его сайта ("X-IBM-Client-Id", "Merchant-Authorization") и заголовок для указания типа передаваемого содержимого ("Content-Type").

### HTTP-заголовки для использования в запросах к API

Заголовок	Описание
X-IBM-Client-Id	Указывается значение <code>client_id</code> , полученное от представителя Платежного шлюза. Пример: <code>ext.cmm.3e9d5564-dae7-4b66-b740-63ed7782e155</code> <div style="background-color: #f4a460; padding: 5px; border-radius: 10px; margin-top: 10px;">Значение необходимо прописать в нижнем регистре, без домена @ext.vtb.ru: например, EXT.ORG1sdcsiocsy@ext.vtb.ru -&gt; ext.org1sdcsiocsy</div>
Merchant-Authorization	Значение заголовка соответствует уникальному идентификатору ресурса мерчанта. Пример: <code>Merchant-Authorization = 5d761993-d444-450a-ad9e-097fa52055ff</code> <div style="background-color: #f4a460; padding: 5px; border-radius: 10px; margin-top: 10px;">Передача заголовка и его значения обязательна только в том случае, если у мерчанта подключено более одного ресурса к Платежному шлюзу.</div>
Authorization	Указывается значение полученного Bearer <code>access_token</code> в результате запроса к Менеджеру доступа. Пример: <code>Authorization: Bearer AQIC5wM2LY45fczCimpZsQyW3PP3NZXNPHHzDLQDZ1qSDcE.*AAJTSQACMDIAAINLABQtNDk1NDQwMDQ2MTkyMTIzNTM3NwACUzEAAjA</code>
Content-Type	Тип передаваемого содержимого. Указывается значение "application/json"

### Примеры использования HTTP-заголовков

#### Пример тела запроса

```
curl --location --request POST 'https://gw.api.vtb.ru/openapi/smb/efcp/e-commerce/v1/orders' \  
--header 'X-IBM-Client-Id: ext.cmm.3e9d5564-dae7-4b66-b740-63ed7782e155' \  
--header 'Merchant-Authorization: 5d761993-d444-450a-ad9e-097fa52055ff' \  
--header 'Authorization: Bearer AQIC5wM2LY45fcx7ug0mf6PKkx1BR4M5U8ift6Poi_OqsE.*AAJTSQACMDIAAINLABMnNTAyNDY0NTQ0MjA3MjQ1Mjc5AAJTMQAC MDE.*' \  
--header 'Content-Type: application/json' \  
--data-raw '{  
.....  
'
```

## 5. Фискализация в соответствии с 54-ФЗ

---

### 5.1 Общее описание функциональности фискализация в соответствии с 54-ФЗ

С 1 июля 2017 года вступили в силу требования в рамках новой редакции закона 54-ФЗ об обязательном применении онлайн-касс для интернет-магазинов. Поправки к закону обязывают онлайн-магазины использовать специальную кассовую технику для отправки фискальных данных в Федеральную налоговую службу (ФНС).

Интернет-магазины могут использовать различные (как облачные, так и физические) кассы для создания фискальных чеков, но для онлайн торговли в основном используется схема взаимодействия с облачными кассами и сторонними сервисами фискализации, которые предоставляют такие услуги.

Платежный шлюз предоставляет функциональность печати фискальных чеков (прихода и возврата прихода) путем взаимодействия с внешними операторами фискальных данных (ОФД)

Для использования функциональности фискализации в личном кабинете Интернет-эквайринга для ресурса (сайта) в разделе «Настройка ресурсов» в подразделе «Настройка сервиса фискализации (54-ФЗ)» должны быть установлены настройки оператора фискальных данных (ОФД).

Поддерживаемые Платежным шлюзом операторы фискальных данных (ОФД): Первый ОФД.

Поддерживаемые типы фискальных чеков: «Приход» и «Возврат прихода».

Поддерживаемые Платежным шлюзом версии ФФД: 1.2

Фискализация выполняется только по ордерам без предавторизации (одностадийный платеж).

Фискализация для ордеров с предавторизацией (двухстадийный платеж) не поддерживается.

Отправка фискального чека реализуется только на email покупателя, который был передан мерчантом вместе с товарной корзиной в запросе на создание ордера. Если email покупателя не был передан, то фискальный будет отправлен оператором фискальных данных на email, который установлен мерчантом в настройках фискализации в Личном кабинете Интернет-эквайринга.

### 5.2 Создание ордера с товарной корзиной

Требования к формированию запроса на создание ордера с товарной корзиной

- Товарная корзина передаётся внутри отдельного объекта «bundle» в запросе на создание ордера (POST v1/orders);
- Если объект товарной корзины «bundle» не был передан при создании ордера (при этом у ресурса мерчанта имеются активные настройки фискализации), то в этом случае товарная корзина будет сформирована на основании настроек типовой фискализации, указанных в Личном кабинете Интернет-эквайринга по ресурсу (сайту), в рамках которого создается ордер.
- Сумма всех товарных позиций корзины должна быть равна сумме ордера;
- Все параметры товарной корзины валидируются на соответствие требуемому формату и длине;
- В случае невыполнения хотя бы одного из указанных выше условий ордер считается неправильно сформированным, в результате чего Платежный шлюз вернет ошибку.

## Параметры объекта товарной корзины «bundle» в запросе на создание ордера POST v1/orders

<b>bundle</b>	Объект товарной корзины		
Url / body / header Body	Тип Object	Формат n/a	
<hr/>			
<b>fiscalInfo</b>	Объект с параметрами фискализации		
Url / body / header Body	Тип Object	Формат n/a	
<b>clientEmail</b>	Email покупателя для отправки фискального чека		
Url / body / header Body	Тип String (255)	Формат Byte	
<hr/>			
<b>items</b>	Массив с экземплярами товарной корзины		
Обязательный	Url / body / header Body	Тип Array	
<b>positionId</b>	Порядковый номер позиции единицы товара. ФН имеет аппаратное ограничение 32768 байт для каждого финансового документа в двоичной форме. Это ограничение размера документа предусматривает около 250 позиций товарной корзины в чеке. ФН отклонит создание чека, если будет превышено ограничение 32768 байт. Рекомендуется использовать не более 250 позиций в корзине		
Обязательный	Url / body / header Body	Тип Integer	Формат minimum: 1, maximum: 250
<b>name</b>	Наименование товара, работы, услуги, платежа, выплаты, иного предмета расчета		
Обязательный	Url / body / header Body	Тип String (255)	Формат Byte
<b>code</b>	Уникальный код товара (Артикул)		
	Url / body / header Body	Тип String (255)	Формат Byte
<b>description</b>	Описание единицы товара		
	Url / body / header Body	Тип String (255)	Формат Byte
<b>shippable</b>	Признак доставляемости товара		
	Url / body / header Body	Тип Boolean	
<b>price</b>	Цена за единицу товара, работы, услуги, платежа, выплаты, иного предмета расчета с учетом скидок и наценок. Цена в рублях: — целая часть не более 8 знаков; — дробная часть не более 2 знаков. Максимальное значение цены — 42 949 672.95. При этом произведение цены и количество/веса (price*quantity) позиции должно быть не больше максимального значения цены позиции		
Обязательный	Url / body / header Body	Тип Number	Формат minimum: 0, maximum: 42949672.95

items	measure	Обязательный	Единицы измерения количества предмета расчета. Может принимать одно из значений: 0 — Применяется для предметов расчета, которые могут быть реализованы поштучно или единицами; 10 — Грамм; 11 — Килограмм; 12 — Тонна; 20 — Сантиметр; 21 — Дециметр; 22 — Метр; 30 — Квадратный сантиметр; 31 — Квадратный дециметр; 32 — Квадратный метр; 40 — Миллилитр; 41 — Литр; 42 — Кубический метр; 50 — Киловатт час; 51 — Гигакалория; 70 — Сутки (день); 71 — Час; 72 — Минута; 73 — Секунда; 80 — Килобайт; 81 — Мегабайт; 82 — Гигабайт; 83 — Терабайт; 255 — Применяется при использовании иных единиц измерения	Url / body / header Body	Тип String (255)	Формат enum: 0, 10, 11, 12, 20, 21, 22, 30, 31, 32, 40, 41, 42, 50, 51, 70, 71, 72, 73, 80, 81, 82, 83, 255
	quantity	Обязательный	Количество/вес. Правила валидации: — Целая часть не более 5 знаков; — Дробная часть не более 3 знаков; — Максимальное значение – 99 999.999	Url / body / header Body	Тип Number	Формат minimum: 0; maximum: 99999.999
	taxParams	Обязательный	Объект с параметрами налогообложения	Url / body / header Body	Тип Object	
	taxType	Обязательный	Ставка налогообложения. Устанавливает номер налога в ККТ. Перечисление со значениями: «none» – без НДС; «vat0» – НДС по ставке 0%; «vat10» – НДС чека по ставке 10%; «vat110» – НДС чека по расчетной ставке 10/110; «vat20» – НДС чека по ставке 20%; «vat120» – НДС чека по расчетной ставке 20/120.	Url / body / header Body	Тип String	Формат enum: none, vat0, vat10, vat110, vat20, vat120
	taxSum		Сумма налога позиции в рублях. — целая часть не более 8 знаков; — дробная часть не более 2 знаков.	Url / body / header Body	Тип Number	Формат minimum: 0; maximum: 99999999.99
	paymentType		Признак способа расчёта. Возможные значения: «full_prepayment» — предоплата 100%. Полная предварительная оплата до момента передачи предмета расчета; «prepayment» — предоплата. Частичная предварительная оплата до момента передачи предмета расчета; «advance» — аванс; «full_payment» — полный расчет. Полная оплата, в том числе с учетом аванса (предварительной оплаты) в момент передачи предмета расчета; «partial_payment» — частичный расчет и кредит. Частичная оплата предмета расчета в момент его передачи с последующей оплатой в кредит; «credit» — передача в кредит. Передача предмета расчета без его оплаты в момент его передачи с последующей оплатой в кредит; «credit_payment» — оплата кредита. Оплата предмета расчета после его передачи с оплатой в кредит (оплата кредита);	Url / body / header Body	Тип String	Формат enum: full_prepayment, prepayment, advance, full_payment, partial_payment, credit, credit_payment
items						

## paymentSubject

Признак предмета расчёта. Должен принимать одно из значений:

1. О реализуемом товаре, за исключением подакцизного товара и товара, подлежащего маркировке средствами идентификации (наименование и иные сведения, описывающие товар) работе (наименование и иные сведения, описывающие работу);
2. О реализуемом подакцизном товаре, за исключением товара, подлежащего маркировке средствами идентификации (наименование и иные сведения, описывающие товар);
3. О выполняемой работе (наименование и иные сведения, описывающие работу);
4. Об оказываемой услуге (наименование и иные сведения, описывающие услугу);
5. О приеме ставок при осуществлении деятельности по проведению азартных игр;
6. О выплате денежных средств в виде выигрыша при осуществлении деятельности по проведению азартных игр;
7. О приеме денежных средств при реализации лотерейных билетов, электронных лотерейных билетов, приеме лотерейных ставок при осуществлении деятельности по проведению лотерей;
8. О выплате денежных средств в виде выигрыша при осуществлении деятельности по проведению лотерей;
9. О предоставлении прав на использование результатов интеллектуальной деятельности или средств индивидуализации;
10. Об авансе, задатке, предоплате, кредите;
11. О вознаграждении пользователя, являющегося платежным агентом (субагентом), банковским платежным агентом (субагентом), комиссионером, поверенным или иным агентом;
12. О взносе в счет оплаты, пени, штрафе, вознаграждении, бонусе и ином аналогичном предмете расчета;
13. О предмете расчета, не относящемся к предметам расчета, которым может быть присвоено значение от «1» до «11» и от «14» до «26»;
14. О передаче имущественных прав;
15. О внереализационном доходе;
16. О суммах расходов, платежей и взносов, указанных в подпунктах 2 и 3 пункта Налогового кодекса Российской Федерации, уменьшающих сумму налога;
17. О суммах уплаченного торгового сбора;
18. О курортном сборе;
19. О залоге;
20. О суммах произведенных расходов в соответствии со статьей 346.16 Налогового кодекса Российской Федерации, уменьшающих доход;

Url / body / header Body Тип Number Формат enum: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20

## amount

Обязательный

Итоговая стоимость предмета расчета с учетом скидок и наценок (в рублях).

Правила валидации:

- целая часть не более 8 знаков;
- дробная часть не более 2 знаков.

Максимальное значение – 42 949 672.95.

Значение реквизита «стоимость предмета расчета с учетом скидок и наценок» должно быть равно произведению значения реквизита «цена за единицу предмета расчета с учетом скидок и наценок», умноженному на значение реквизита «количество предмета расчета».

Url / body / header Body Тип Number Формат minimum: 0; maximum: 42949672.95

## discount

Сумма скидки

Url / body / header Body Тип Number Формат minimum: 0; maximum: 42949672.95

## userData

Дополнительный реквизит предмета расчета.

Например, любой сквозной идентификатор в системе мерчанта

Url / body / header Body Тип String(255) Формат Byte

## Пример запроса POST v1/orders с объектом товарной корзины «bundle»

```
{
  "orderId": "ORDER10000000083",
  "orderName": "Заказ №10000000083",
  "amount": {
    "value": 60000.0,
    "code": "RUB"
  },
  "bundle": {
    "fiscalInfo": {
      "clientEmail": "client_email@yandex.ru"
    }
  },
  "items": [
    {
      "positionId": 1,
      "name": "Обеденный стол",
      "code": "арт.4957686074846",
      "description": "Стол барный Dorian 11770",
      "shippable": true,
      "price": 20000.0,
      "measure": 255,
      "quantity": 1,
      "taxParams": {
        "taxType": "vat20",
        "taxSum": 4000.0
      },
      "paymentType": "full_prepayment",
      "paymentSubject": 1,
      "amount": 20000.0,
      "discount": 0,
      "userData": "Дополнительный реквизит предмета расчета"
    },
    {
      "positionId": 2,
      "name": "Обеденный стул",
      "code": "арт.49576860745555",
      "description": "Стул барный Miranda",
      "shippable": true,
      "price": 10000.0,
      "measure": 255,
      "quantity": 4,
      "taxParams": {
        "taxType": "vat20",
        "taxSum": 8000.0
      },
      "paymentType": "full_prepayment",
      "paymentSubject": 1,
      "amount": 40000.0,
      "discount": 0,
      "userData": "Дополнительный реквизит предмета расчета"
    }
  ]
}
```

## 5.3 Создание возврата с товарной корзиной

### Требования к формированию запроса на возврат с товарной корзиной

Товарная корзина передаётся внутри отдельного объекта «bundle» в запросе на создание возврата (POST v1/refunds);

Если возврат выполняется на всю сумму ордера (полный возврат), то передавать товарную корзину внутри объекта «bundle» не обязательно! В этом случае Платёжный шлюз при создании чека возврата прихода в ОФД отправит товарную корзину, которая была передана при создании ордера, на который выполняется полный возврат;

При возврате заказа на сумму, отличную от суммы списания (кроме передачи значения «0») обязательно должна передаваться товарная корзина с возвращаемыми позициями.

В случае проведения нескольких возвратов по ордеру с товарной корзиной все они должны осуществляться только по алгоритму возврата с товарной корзиной;

Денежная сумма по возвращаемым позициям в товарной корзине не должна превышать подтвержденную денежную сумму оригинального заказа;

При создании возврата в товарной корзине запрещены для передачи товарные позиции, отсутствующие в оригинальном заказе. Происходит проверка наличия указанного товара в корзине запроса на возврат в изначальном заказе. Необходимо совпадение элементов. Если хотя бы одно из значений не совпадает, считается, что данная товарная позиция отсутствует в оригинальном заказе;

Контроль корректности заполнения параметров по возвращаемым позициям в товарной корзине, их количества и т.д. выполняется на стороне системы мерчанта (проверки, что данная позиция не была возвращена ранее, что для возвращаемой позиции указаны те же самые параметры налогообложения и т.д.);

Все параметры товарной корзины валидируются на соответствие требуемому формату и длине;

В случае невыполнения хотя бы одного из указанных выше условий возврат считается неправильно сформированным, в результате чего Платёжный шлюз вернет ошибку.

### Параметры объекта товарной корзины "bundle" в запросе на создание возврата POST v1/refunds

Формат объекта «bundle» в запросе на возврат POST v1/refunds полностью соответствует формату одноименного объекта, описанному в разделе «5.2 Создание ордера с товарной корзиной»

## Пример запроса POST v1/refunds с объектом товарной корзины «bundle»

```
{  "refundId": "REFUND10000000083-1",
  "paymentId": "488e7417-8df6-4804-8f43-0181816e4c12",
  "amount": {
    "value": 60000.0,
    "code": "RUB"
  },
  "bundle": {
    "fiscalInfo": {
      "clientEmail": "client_email@yandex.ru"
    }
  },
  "items": [
    {
      "positionId": 2,
      "name": "Обеденный стул",
      "code": "арт.49576860745555",
      "description": "Стул барный Miranda",
      "shippable": true,
      "price": 10000.0,
      "measure": 255,
      "quantity": 4,
      "taxParams": {
        "taxType": "vat20",
        "taxSum": 8000.0
      },
      "paymentType": "full_prepayment",
      "paymentSubject": 1,
      "amount": 40000.0,
      "discount": 0,
      "userData": "Дополнительный реквизит предмета расчета"
    }
  ]
}
```

## 5.4 Операции с фискальными чеками

### 5.4.1 GET /receipts/{orderId}/status

GET /receipts/{orderId}/status — запрос для получения статуса чеков по ордеру по идентификатору ордера в системе мерчанта.

В ответе на запрос возвращается массив с фискальными чеками по данному ордеру по всем типам операций (приход, возврат прихода).

Для каждого чека в массиве возвращается его статус и фискальные данные.

В случае отсутствия фискальных чеков по ордеру вернется ошибка.

#### Параметры запроса GET v1/receipts/{orderId}/status

##### Merchant-Authorization

Дополнительный (необязательный) HTTP-заголовок, в котором передается идентификатор ресурса, если у мерчанта подключено более одного ресурса к Платежному шлюзу

Url / body / header Header Тип n/a

Формат n/a

##### orderId

Уникальный идентификатор ордера в системе мерчанта

Обязательный

Url / body / header URL Тип String (64)

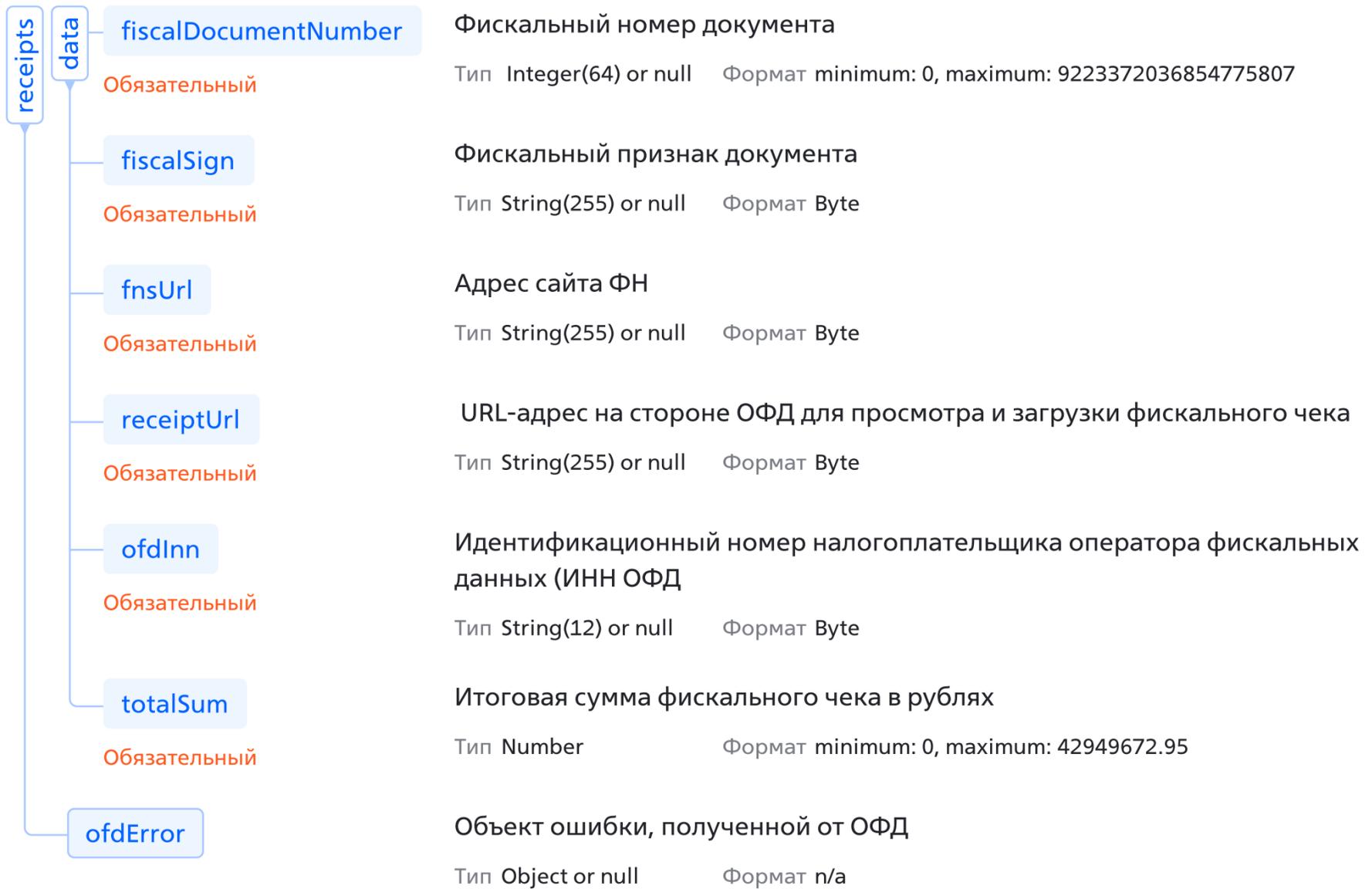
Формат Byte

## Пример запроса GET v1/receipts/{orderId}/status

GET v1/receipts/{orderId}/status

## Параметры ответа GET v1/receipts/{orderId}/status

<b>orderCode</b> Обязательный	Уникальный внешний идентификатор ордера Тип String(64)      Формат Byte
<b>receipts</b> Обязательный	Массив объектов с фискальными чеками Тип Array or null      Формат n/a
<b>uuid</b> Обязательный	Идентификатор фискального чека в ОФД Тип String (64)      Формат UUID
<b>operation</b> Обязательный	Тип операции. Допустимые типы: INCOME – приход; INCOME_RETURN – возврат прихода. Тип String      Формат enum: INCOME, INCOME_RETURN
<b>status</b> Обязательный	Статус фискального чека. Допустимые статусы: NEW - Новый; PRINTED - Напечатан; PRINT_ERROR - Ошибка печати чека. Тип String      Формат enum: NEW, PRINTED, PRINT_ERROR
<b>createdAt</b> Обязательный	Дата и время отправки чека на печать на стороне Платежного шлюза Тип String      Формат Date-time
<b>data</b> Обязательный	Объект с фискальными данными чека из ОФД Тип Object or null      Формат n/a
<b>receiptInShiftNumber</b> Обязательный	Номер чека в смене Тип Integer(32) or null      Формат minimum: 0 maximum: 2147483647
<b>shiftNumber</b> Обязательный	Номер смены Тип Integer(32) or null      Формат minimum: 0 maximum: 2147483647
<b>receiptDatetime</b> Обязательный	Дата и время фискального чека в фискальном накопителе Тип String or null      Формат Date-time
<b>fnNumber</b> Обязательный	Номер фискального накопителя (ФН) Тип String(255) or null      Формат Byte
<b>kktRegId</b> Обязательный	Регистрационный номер контрольно-кассовой техники (ККТ) Тип String(255) or null      Формат Byte



## Пример ответа GET v1/receipts/{orderId}/status

```
{
  "orderId": "string",
  "orderCode": "string",
  "receipts": [
    {
      "uuid": "5ga85f64-5717-4562-b3fc-2c963f66afa6",
      "operation": "INCOME",
      "status": "PRINTED",
      "createdAt": "2024-01-26T09:30:24.962Z",
      "data": {
        "receiptInShiftNumber": 5,
        "shiftNumber": 56,
        "receiptDatetime": "2024-01-26T09:30:50",
        "fnNumber": "9999078902012023",
        "kktRegId": "0180181006059581",
        "fiscalDocumentNumber": 1265,
        "fiscalSign": "31047A876E98",
        "fnsUrl": "www.fns.ru",
        "receiptUrl": "https://consumer.1-ofd.ru/v1?fn=9288000100014915&fp=3004144185&i=108&t=20180522T122800&s=400.00&n=1",
        "ofdInn": "7709364346",
        "totalSum": 400.0
      },
      "ofdError": null
    },
    {
      "uuid": "5ga85f64-5717-4562-b3fc-2c963f66afa6",
      "operation": "INCOME_RETURN",
      "status": "PRINTED",
      "createdAt": "2024-02-13T12:30:13.301",
      "data": {
        "receiptInShiftNumber": 21,
        "shiftNumber": 85,
        "receiptDatetime": "2024-02-13T12:30:26",
        "fnNumber": "9999078903021522",
        "kktRegId": "0180181006059581",
        "fiscalDocumentNumber": 2065,
        "fiscalSign": "31047A876E98",
        "fnsUrl": "www.fns.ru",
        "receiptUrl": "https://consumer.1-ofd.ru/v1?fn=9288000100014915&fp=3004144185&i=108&t=20180522T122800&s=400.00&n=1",
        "ofdInn": "7709364346",
        "totalSum": 400.0
      },
      "ofdError": null
    }
  ]
}
```

# Приложение 1. Описание объекта ошибки Платежного шлюза, возвращаемого через API

В случае возникновения ошибки во время выполнении запроса к API Платежный шлюз вернет стандартный HTTP-код ошибки, расшифровку ошибки, а также объект бизнес-ошибки, если она произошла.

## Пример объекта ошибки

```
{
  "error": {
    "id": "10000002",
    "message": " OPERATION_NOT_SUPPORTED_ERROR",
    "description": "Операция не поддерживается",
    "traceId": "42a383016f62836f"
  }
}
```

## Описание объекта ошибки

<b>error</b>	Объект ошибки
	Тип Object      Формат n/a
<b>id</b>	Уникальный идентификатор ошибки
Обязательный	Тип String (64)      Формат n/a
<b>message</b>	Сообщение об ошибке
Обязательный	Тип String (255)      Формат n/a
<b>description</b>	Человекочитаемый текст ошибки
	Тип String (255)      Формат n/a
<b>traceId</b>	Уникальный идентификатор для обращения в техподдержку
Обязательный	Тип String (255)      Формат Byte

## Приложение 2. Расшифровка кодов ответов по ошибкам от Платежного шлюза

Error code	HTTP-код	Error message	Описание ошибки для мерчанта и покупателя
10000001	500	INTERNAL_ERROR	Внутренняя ошибка сервиса
10000002	402	OPERATION_NOT_SUPPORTED_ERROR	Операция не поддерживается
10000003	500	TEMPORARY_ERROR	Временная ошибка
10000007	400	REQUEST_VALIDATION_ERROR	Ошибка валидации запроса
10000008	402	INCORRECT_AMOUNT	Неправильная сумма
10000009	404	MERCHANT_SITE_NOT_FOUND	Сайт не найден
10000010	402	TRANSACTION_NOT_FOUND	Транзакция не найдена
10000012	500	INCORRECT_PARENT_TRANSACTION	Неправильно указана родительская транзакция
10000013	400	CARD_EXPIRED_ERROR	Срок действия карты истек
10000014	400	MERCHANT_DISABLED	Мерчант не активен
10000017	400	ORDER_PAYED	Заказ уже оплачен
10000018	402	TRANSACTION_IN_PROCESS	Транзакция в процессе
10000021	402	INCORRECT_CURRENCY	Некорректная валюта
10000023	400	TRANSACTION_EXPIRED	Срок жизни транзакции истек
10000024	400	ORDER_EXPIRED_ERROR	Срок жизни заказа истек
10000025	400	ORDER_REFUNDED_ERROR	Заказ уже был возвращен
10000026	402	PARTIAL_REFUND_NOT_SUPPORTED	Частичный возврат для данного мерчанта не поддерживается

<b>Error code</b>	<b>HTTP-код</b>	<b>Error message</b>	<b>Описание ошибки для мерчанта и покупателя</b>
10000027	400	INCORRECT_MERCHANT_SETTINGS	Некорректные настройки мерчанта
10000028	400	INCORRECT_CLIENT_ID	Некорректные аутентификационные данные клиента
10000029	400	TRANSACTION_TYPE_NOT_FOUND	Некорректный тип транзакции
10000030	400	ORDER_TYPE_NOT_FOUND	Некорректный тип заказа
10000031	500	UNKNOWN_ERROR	Неизвестная ошибка
10000032	402	QR_GENERATION_ERROR	Платеж отклонен. Попробуйте еще раз
10100001	500	INCORRECT_JEXL_PARAMETERS_EXCEPTION	Внутренняя ошибка при обработке транзакции
10100002	500	INCORRECT_JEXL_SCRIPT_EXCEPTION	Внутренняя ошибка при обработке транзакции
10100003	500	ROUTE_LOOP_EXCEPTION	Внутренняя ошибка при обработке транзакции
10100003	500	ROUTE_LOOP_EXCEPTION	Внутренняя ошибка при обработке транзакции
10100004	500	INCORRECT_ROUTE_EXCEPTION	Внутренняя ошибка при обработке транзакции
10100005	500	EMPTY_ROUTE_EXCEPTION	Внутренняя ошибка при обработке транзакции
10100006	500	DATA_ACCESS_EXCEPTION	Внутренняя ошибка при обработке транзакции
10200001	402	SBP_NOT_PERMITTED	Платеж отклонен. Проверьте реквизиты платежа
10200002	402	SBP_BAD_REQUEST	Ошибка исполнения платежа

<b>Error code</b>	<b>HTTP-код</b>	<b>Error message</b>	<b>Описание ошибки для мерчанта и покупателя</b>
10200003	402	SBP_UNPROCESSABLE_ENTITY	Ошибка формирования платежа
10200004	402	SBP_INTERNAL_SERVER_ERROR	Внутренняя ошибка сервиса
10200005	402	SBP_BAD_GATEWAY	Внутренняя ошибка сервиса
10300001	400	CRYPTOGRAPHY_EXCEPTION	Внутренняя ошибка
10300002	400	PAN_VALIDATION_EXCEPTION	Платеж отклонен. Проверьте реквизиты платежа
10300003	400	OBSOLETE_KEY_EXCEPTION	Устаревший публичный ключ
10300004	400	DECRYPT_ENCRYPT_EXCEPTION	Устаревший публичный ключ
10400001	402		3DS technical error
10400002	402		3DS technical error
10502001	402	UNSUPPORTED_VERSION	Платеж отклонен. Попробуйте еще раз
10502001	402	UNSUPPORTED_LANGUAGE	Платеж отклонен. Попробуйте еще раз
10502001	402	UNSUPPORTED_COMMAND	Платеж отклонен. Попробуйте еще раз
10502000	402	AUTHENTICATION_CERTIFICATE_ERROR	Платеж отклонен. Попробуйте еще раз
10502001	402	MESSAGE_PARSING_ERROR	Платеж отклонен. Попробуйте еще раз
10502002	402	SYSTEM_ERROR	Платеж отклонен. Попробуйте еще раз
10502000	402	CRYPTOGRAPHY_ERROR	Платеж отклонен. Попробуйте еще раз
10502002	402	TIMEOUT	Платеж отклонен. Попробуйте еще раз

<b>Error code</b>	<b>HTTP-код</b>	<b>Error message</b>	<b>Описание ошибки для мерчанта и покупателя</b>
10502001	402	PARAMETERS_ERROR	Что то пошло не так
10502001	402	ZERO_AMOUNT	Платеж отклонен. Попробуйте еще раз
10502002	402	ORIGINAL_TRANSACTION_NOT_FOUND	Платеж отклонен. Попробуйте еще раз
10502002	402	DUPLICATE_INVOICE	Платеж отклонен. Попробуйте еще раз
10501004	402	ACCOUNT_NOT_FOUND	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501004	402	WRONG_TRANSACTION	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501004	402	WRONG_AMOUNT	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501004	402	UNABLE_PERFORM_OPERATION	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501006	402	AMOUNT_EXCEED	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501004	402	WRONG_PAYMENT_INFORMATION	Платеж отклонен. Обратитесь в Банк выпустивший карту
10502002	402	WRONG_TERMINAL_ID	Платеж отклонен. Попробуйте еще раз
10501004	402	WRONG_RECEIVER_PAN	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501002	402	CARD_EXPIRED	Платеж отклонен. Проверьте реквизиты платежа
10501005	402	ISSUER_CARD_FAIL	Платеж отклонен. Попробуйте еще раз
10501004	402	UNSUPPORTED_TRANSACTION	Платеж отклонен. Обратитесь в Банк выпустивший карту

<b>Error code</b>	<b>HTTP-код</b>	<b>Error message</b>	<b>Описание ошибки для мерчанта и покупателя</b>
10501004	402	AUTHORITIES_ERROR	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501004	402	CARD_LOST	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501009	402	CARD_STATUS	Платеж отклонен. Проверьте реквизиты платежа
10501004	402	CARD_LIMITED	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501003	402	ISSUER_NOT_AVAILABLE	Платеж отклонен. Попробуйте еще раз
10501007	402	ISSUER_LIMIT_COUNT_FAIL	Платеж отклонен. Превышен лимит. Обратитесь в Банк выпустивший карту
10501007	402	ISSUER_LIMIT_AMOUNT_FAIL	Платеж отклонен. Превышен лимит. Обратитесь в Банк выпустивший карту
10501001	402	PIN_ATTEMPT_EXCEEDED	Платеж отклонен. Проверьте данные карты
10501009	402	UNSUPPORTED_CARD	Платеж отклонен. Проверьте реквизиты платежа
10501005	402	FORMAT_ERROR	Платеж отклонен. Попробуйте еще раз
10502002	402	TRANSACTION_TIMEOUT	Платеж отклонен. Попробуйте еще раз
10502002	402	SYSTEM_ERROR	Что-то пошло не так
10501004	402	ISSUER_CALL	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501004	402	WRONG_CAVV	Платеж отклонен. Обратитесь в Банк выпустивший карту

Error code	HTTP-код	Error message	Описание ошибки для мерчанта и покупателя
10501001	402	WRONG_CVV	Платеж отклонен. Проверьте данные карты
10501009	402	CARD_HOLD_EXPIRED	Платеж отклонен. Проверьте реквизиты платежа
10501004	402	ISSUER_CARD_HOLD_FAIL	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501005	402	SYSTEM_ERROR	Платеж отклонен. Попробуйте еще раз
10502002	402	ORIGINAL_OPERATION_NOT_FOUND	Платеж отклонен. Попробуйте еще раз
10501004	402	UNABLE_PERFORM_OPERATION	Платеж отклонен. Обратитесь в Банк выпустивший карту
10501005	402	WRONG_SESSION	Платеж отклонен. Попробуйте еще раз
10900001	400	Incorrect customerid	Некорректный идентификатор покупателя в связке
19200001	400	Incorrect bindingid	Некорректный идентификатор связки
19200002	400	Incorrect operationType	Неверный тип операции
19200003	400	Incorrect bindingType	Неверный тип CoF
19200004	400	Incorrect customerid	Не указан идентификатор покупателя
19200005	400	Incorrect bindingid	Указан некорректный идентификатор связки
19200006	400	Incorrect bindingstate	Состояние связки не позволяет выполнить оплату
10000038	400	BUNDLE_ERROR	Неверный формат параметра name

Error code	HTTP-код	Error message	Описание ошибки для мерчанта и покупателя
10000039	400	BUNDLE_ERROR	Неверный формат параметра code
10000040	400	BUNDLE_ERROR	Неверный формат параметра description
10000041	400	BUNDLE_ERROR	Неверный формат параметра price
10000042	400	BUNDLE_ERROR	Неверный формат параметра measure
10000043	400	BUNDLE_ERROR	Неверный формат параметра quantity
10000044	400	BUNDLE_ERROR	Неверный формат параметра taxSum
10000045	400	BUNDLE_ERROR	Неверный формат параметра amount
10000046	400	BUNDLE_ERROR	Неверный формат параметра discount
10000047	400	BUNDLE_ERROR	Неверный формат параметра userData
10000048	400	BUNDLE_ERROR	Сумма товарных позиций в корзине не совпадает с суммой заказа
10000057	400	BUNDLE_ERROR	В заказе отсутствует товарная корзина
10000055	400	BUNDLE_ERROR	Сумма товарной позиции не соответствует сумме в товарной корзине заказа
10000060	400	BUNDLE_ERROR	Неверный формат параметра paymentType
10000061	400	BUNDLE_ERROR	Неверный формат параметра paymentSubject
10000062	400	BUNDLE_ERROR	Некорректные конфигурации настроек фискализации для ресурса
10000064	400	BUNDLE_ERROR	Неверный формат параметра positionId
10000100	404	RECEIPTS_NOT_FOUND	По данному ордеру отсутствуют фискальные чеки
10000033	404	ORDER_NOT_FOUND	Ордер не найден

<b>Error code</b>	<b>HTTP-код</b>	<b>Error message</b>	<b>Описание ошибки для мерчанта и покупателя</b>
1000027	500	INCORRECT_MERCHANT_SETTINGS	Некорректные настройки ресурса мерчанта