**Страница разработчика | Проект «Продажа со склада поставщика»**

Данный документ описывает API, используемое для интеграции между системой поставщика и проектом «продажа со склада поставщика».

Для начала работы с API необходимо [**создать склад поставщика**](https://suppliers.wildberries.ru/marketplace-pass/warehouses).
Токен для работы с сервисами можно получить в [**настройках поставщика**](https://suppliers.wildberries.ru/marketplace-pass/api-access).

API Stock&Price

Поставщик присылает один раз все текущие остатки по всем SKU, затем в оперативном режиме присылает остатки только по тем SKU, по которым были изменения.

Передача остатков на складах поставщика:

**Production URL** https://wbxgate.wildberries.ru/stocks
**Тестовый URL** https://wbxgate-test.wildberries.ru/stocks

**Авторизация:**

Не требуется передавать заголовок X-Auth-Token. Токен передается в теле запроса.

POST - запрос

Тело запроса:

{

 "token": string, // ВАШ\_ТОКЕН

 "data": [

 {

 "nmId": int,

 "stocks": [

 {

 "chrtId": int,

 "price": int,

 "quantity": int,

 "storeId": int

 },

 {

 "chrtId": int,

 "price": int,

 "quantity": int,

 "storeId": int

 },

 {

 "chrtId": int,

 "price": int,

 "quantity": int,

 "storeId": int

 }

 ]

 },

 {

 "nmId": int,

 "stocks": [

 {

 "chrtId": int,

 "price": int,

 "quantity": int,

 "storeId": int

 },

 {

 "chrtId": int,

 "price": int,

 "quantity": int,

 "storeId": int

 },

 {

 "chrtId": int,

 "price": int,

 "quantity": int,

 "storeId": int

 },

 {

 "chrtId": int,

 "price": int,

 "quantity": int,

 "storeId": int

 },

 {

 "chrtId": int,

 "price": int,

 "quantity": int,

 "storeId": int

 }

 ]

 }

 ]

}

* “token” - токен полученный при регистрации
* “data” - массив c пересылаемыми данными
* “nmId” - идентификатор номенклатуры, по которой присылаются остатки
* “stocks” - массив с информацией по остаткам, цене и складам для каждой chrtId (SKU)
* “chrtId” - идентификатор размера, по которому присылаются остатки
* “price” - цена товара
* “quantity” - реальное количество единиц товара
* “storeId” - идентификатор склада

**Дополнение 1:**

* Код размера (chrtId)
* Артикул WB (nmId)

Данные поля можно получить на странице с [**перечнем номенклатуры поставщика**](https://suppliers.wildberries.ru/analytics/nomenclatures).

**Дополнение 2:**

* ID склада (storeId)

Данное поле можно получить в [**настройках склада поставщика**](https://suppliers.wildberries.ru/marketplace-pass/warehouses).

В рамках одного артикула (nmId) необходимо всегда присылать актуальные остатки по всем связанным **SKU** (chrtId).

При успешной заливке остатков:
\*RESPONSE STATUS == 200\*

При неуспешной заливке:
\*RESPONSE STATUS != 200\* и \*RESPONSE:\*

{

 "error": "error message text"

 "errorDetails": "details text"

}

API Orders

Интерфейс для работы со сборочными заданиями (заказами).

**Production URL** https://suppliers-orders.wildberries.ru
**Тестовый URL** https://suppliers-orders-test.wildberries.ru

**Авторизация:**

При вызове каждого метода должен передаваться заголовок X-Auth-Token с выданным токеном.

1. Получение списка заказов:

1.1. Получить заказы в промежутке времени:

GET /api/v1/orders?date\_start=2020-09-14T00:00:00+03:00&date\_end=2020-09-15T00:00:00+03:00

**Параметры:**
"date\_start" - дата в формате rfc3339 с которой нужно получить заказы
"date\_end" - дата в формате rfc3339 по которую нужно получить заказы

1.2. Получить все заказы позже указанного времени:

GET /api/v1/orders?date\_start=2020-09-14T00:00:00+03:00

**Параметры:**
"date\_start" - дата в формате rfc3339 с которой нужно получить заказы

**Формат ответа:**

[

 {

 "order\_id": string, // Идентификатор заказа (32 символа макс.)

 "wb\_wh\_id": int64, // Идентификатор склада WB, на которой заказ должен быть доставлен

 "items": [

 {

 "rid": string, // (32 символа макс.)

 Уникальный идентификатор вещи, разный у одинаковых chrt\_id, по ним

 надо отчитываться об отменах и подборе

 “status”: int8,

 "chrt\_id": int64, // Идентификатор артикула

 "total\_price": int64, // Стоимость товара с учетом скидок в копейках!

 "sticker": {

 "wbsticker\_encoded": string(15), // Закодированное значение стикера (представляется на этикетке в формате Code-128)

 "wbsticker\_id": int64, // Идентификатор этикетки

 "wbsticker\_svg\_base64": string(5000), // Полное представление этикетки в векторном формате

 "wb\_sticker\_zpl": string(300), // Полное представление этикетки в формате ZPL (горизонтальное положение)

 "wb\_sticker\_zpl\_v": string(300), // Полное представление этикетки в формате ZPL (вертикальное положение)

 "wbsticker\_id\_parts": {

 "A": string(15), // Первая часть идентификатора этикетки (для печати подписи)

 "B": string(15) // Вторая часть идентификатора этикетки

 }

 }

 }

 ]

 }

]

В случае успеха возвращается HTTP статус 200 (204 - нет заказов)

В случае неудачи возвращается статус отличный от 200 (400 не передан временной диапазон, 500 проблемы на стороне нашего сервиса)

2. Обновление статуса заказа:

PUT - Запрос

Тело запроса:

[

 {

 "order\_id": string, // Идентификатор заказа (32 символа макс.)

 "items": [

 {

 "rid": string, // (32 символа макс.)

 Уникальный идентификатор вещи, разный у одинаковых chrt\_id,

 по ним надо отчитываться об отменах и подборе

 “status”: int8

 }

 ]

 }

]

В случае успеха возвращается HTTP статус 200

В случае неудачи возвращается статус отличный от 200 (400 проблемы с валидацией переданных данных, 500 проблемы на стороне нашего сервиса)

"status": int8 (0,1,2,3)

|  |  |
| --- | --- |
| 0 | - Новый заказ |
| 1 | - Принял заказ |
| 2 | - Подобран |
| 3 | - Не подобран |

Status&Error

**Production URL** https://marketplace-remotewh.wildberries.ru
**Тестовый URL** https://marketplace-remotewh-test.wildberries.ru

**Авторизация:**

При вызове каждого метода должен передаваться заголовок X-Auth-Token с выданным токеном.

Метод получения статуса товаров по сборочным заданиям/заказам

GET /api/public/v1/supply\_tasks/status

Параметры:
- order\_ids - список идентификаторов сборочных заданий/заказов, по которым необходимо получить статус товаров (максимум 50 идентификаторов в одном запросе) через запятую, обязательный при отсутствии **date\_start** и **date\_end**
- date\_start, date\_end - период, за который необходимо получить все изменения статусов, даты в формате **rfc3339**, обязательный при отсутствии **task\_ids**

Механика работы параметров:
- Если запрашивается статус только по task\_ids, то по заданию будет возвращен последний актуальный на текущий момент статус.
- Если в запросе указаны date\_start, date\_end (как при наличии task\_ids, так и без), то будут возвращены все изменения статуса, которые произошли в указанный период. task\_ids в данном случае выступает в качестве дополнительного фильтра, т.е. можно получить статусы за период по конкретным заданиям.

Формат ответа:

В случае успеха статус 200:

[

 {

 "order\_id": string, // идентификатор сборочного задания/заказа

 "items": [

 {

 "rid": string, // уникальный идентификатор вещи

 "chrt\_id": int, // идентификатор артикула

 "status": int, // код статуса

 "price": float, // цена

 "date": string // дата статуса в формате rfc3339

 },

 ...

 ] // массив статусов по каждому RID

 },

 ...

]

Коды статусов:

|  |  |
| --- | --- |
| 0 | - Информация о статусе отсутствует |
| 1 | - Отмена заказа |
| 2 | - Клиент получил товар |
| 3 | - Клиент вернул/отказался от товара |
| 4 | - Товар поступил на склад WB |

Список статусов может быть расширен в будущем.
После статусов отмены, получения, возврата или отказа от товара можно прекращать отслеживать статус по заказу. В дальнейшем возвраты необходимо отслеживать, используя нижеописанный метод.

В случае неудачи статус отличный от 200:

{

 "success": bool, // флаг успешности операции

 "errors": string // текст ошибки

}

Метод получения списка возвратов

GET /api/public/v1/supply\_tasks/returns

Параметры:
- date\_start - дата в формате **rfc3339**, с которой нужно получить возвраты, обязательный
- date\_end - дата в формате **rfc3339**, по которую нужно получить возвраты, обязательный

Формат ответа:

В случае успеха статус 200:

[

 {

 "order\_id": string, // идентификатор сборочного задания

 "items": [

 {

 "rid": string, // уникальный идентификатор вещи

 "chrt\_id": int, // идентификатор артикула

 "date": string // дата статуса в формате rfc3339

 },

 ...

 ] // массив RID, которые были возвращены

 },

 ...

]

В случае неудачи статус отличный от 200:

{

 "success": bool, // флаг успешности операции

 "errors": string // текст ошибки

}

Метод уведомления об ошибке

POST /api/public/v1/suppliers/errors/add

Request

{

 "source": string, // источник ошибки

 "error": string, // текст ошибки (произвольная строка)

 "error\_type": string, // тип ошибки (произвольная строка)

 "additional": json // json с доп. информацией (произвольный формат)

}

Возможные варианты source:

* orders - ошибки, связанные с сервисом заказов
* stocks - ошибки, связанные с сервисом остатков
* items\_status - ошибки, связанные с сервисом получения статуса товаров и списка возвратов

Формат ответа:

В случае успеха статус 200:

{

 "success": bool // флаг успешности операции

}

В случае неудачи статус отличный от 200:

{

 "success": bool, // флаг успешности операции

 "errors": string // текст ошибки

}

Запуск интеграции

Для перевода интеграции в режим промышленной эксплуатации вам нужно использовать токен, который указан в [**настройках поставщика**](https://suppliers.wildberries.ru/marketplace-pass/api-access).

Перед началом получения информации о заказах необходимо:

* Убедиться в успешной загрузке данных складов;
* Убедиться в успешной загрузке данных об остатках;