

Программный интерфейс сервиса Яндекс.Такси для Служб Такси

Справочник разработчика

10.04.2014

Яндекс

Программный интерфейс сервиса Яндекс.Такси для Служб Такси. Справочник разработчика. Версия 1.12.0

Дата сборки документа: 10.04.2014.

Этот документ является составной частью технической документации Яндекса.

Сайт справки к сервисам Яндекса: <http://help.yandex.ru>

© 2008—2014 ООО «ЯНДЕКС». Все права защищены.

Предупреждение об исключительных правах

Яндексу (а также указанному им правообладателю) принадлежат исключительные права на все результаты интеллектуальной деятельности и приравненные к ним средства индивидуализации, используемые при разработке, поддержке и эксплуатации сервиса Программный интерфейс сервиса Яндекс.Такси для Служб Такси. К таким результатам могут относиться, но не ограничиваясь указанными, программы для ЭВМ, базы данных, изображения, тексты, другие произведения, а также изобретения, полезные модели, товарные знаки, знаки обслуживания, коммерческие обозначения и фирменные наименования. Эти права охраняются в соответствии с Гражданским кодексом РФ и международным правом.

Вы можете использовать сервис Программный интерфейс сервиса Яндекс.Такси для Служб Такси или его составные части только в рамках полномочий, предоставленных вам Пользовательским соглашением сервиса Программный интерфейс сервиса Яндекс.Такси для Служб Такси или специального соглашения.

Нарушение требований по защите исключительных прав правообладателя влечет за собой дисциплинарную, гражданско-правовую, административную или уголовную ответственность в соответствии с российским законодательством.

Контактная информация

ООО «ЯНДЕКС»

<http://www.yandex.ru>

Тел.: +7 495 739 7000

Email: pr@yandex-team.ru

Главный офис: 119021, Россия, г. Москва, ул. Льва Толстого, д. 16

Содержание

О справочнике	4
Подключение к Яндекс.Такси	5
Программный интерфейс	5
Запросы синхронизации данных	7
Запрос тарифов	7
Запрос тарифов (устаревший формат)	18
Запрос профилей водителей	24
Запрос черного списка телефонных номеров	28
Обновление статусов водителей	29
Запросы обработки заказов	32
Предложение заказа	32
Закрепление заказа	40
Сообщение о готовности водителя выполнить заказ	47
Обновление статуса заказа	48
Отмена заказа	50
Отзывы о заказах	51
Запрос стоимости заказа	52
Релизы	54

О справочнике

Сервис Яндекс.Такси предоставляет Службам Такси возможность подключиться к централизованной системе обработки заказов такси.

Сервис предоставляет API, с помощью которого Службы Такси и Яндекс.Такси обмениваются информацией о заказах, тарифах и водителях.

Настоящий документ содержит описание API Яндекс.Такси, используемого для обмена информацией со Службами Такси.

Целевой аудиторией документа являются разработчики HTTP-серверов, обслуживающих диспетчерские системы Служб Такси, а также разработчики Яндекс.Такси.

Структура документа

Документ включает следующие разделы:

1. [Подключение к Яндекс.Такси](#): описание порядка подключения Службы Такси к сервису Яндекс.Такси.
2. [Программный интерфейс](#): общее описание формата запросов и ответов API.
3. [Запросы синхронизации данных](#): описание запросов, с помощью которых Служба Такси предоставляет Яндекс.Такси актуальные данные о водителях и используемых тарифах, а также черный список телефонов клиентов.
4. [Запросы обработки заказов](#): описание запросов, с помощью которых Яндекс.Такси и Служба Такси обмениваются информацией о поступающих и выполняемых заказах.

Соглашения

В документе приняты следующие соглашения:

- **Протокол запросов.** Для отправки и обработки запросов рекомендуется использовать протокол HTTPS. Допускается также использовать протокол HTTP без шифрования.
- **Формат URL запроса.** Примеры значений параметров, задаваемых в URL, приведены в удобном для чтения виде. В реальных запросах URL должны быть [корректно закодированы](#).

Например, для параметра `address` в качестве значения предлагается указать строку «Москва, ул. Таганская, дом 5».

В закодированном виде этот параметр будет выглядеть так:

```
address=%D0%9C%D0%BE%D1%81%D0%BA%D0%B2%D0%B0%2C+%D1%83%D0%BB.%D0%A2%D0%B0%D0%B3%D0%B0%D0%BD%D1%81%D0%BA%D0%B0%D1%8F%2C+5
```

- **Адрес сервера Яндекс.Такси.** В примерах справочника используется адрес боевого сервера Яндекс.Такси (работающего с реальными заказами клиентов).

Терминология

Служба Такси

Компания, которая предоставляет услуги такси и зарегистрирована в сервисе Яндекс.Такси.

Клиент

Пользователь Яндекс.Такси.

Водитель

Работник Службы Такси, непосредственно выполняющий заказы клиентов.

Заказ

Информация о требуемых услугах такси, переданная клиентом сервису Яндекс.Такси для поиска подходящей машины.

Срочный заказ

Заказ на ближайшее время, от 10 до 25 минут с момента поступления заказа.

Несрочный заказ

Заказ на отдаленное время (промежуток между поступлением заказа и указанным временем подачи машины больше 25 минут). По запросу Службы Такси допустимый промежуток может быть увеличен.

Подключение к Яндекс.Такси

Чтобы начать обрабатывать заказы клиентов, Служба Такси должна пройти два этапа подготовки:

1. Получить идентификаторы на сервисе (передаются при заключении договора с Яндексом):
 - `clid` — уникальный идентификатор Службы Такси;
 - `apikey` — API-ключ для доступа к программному интерфейсу.
2. Протестировать корректность обработки всех необходимых запросов. Проверки проводятся в порядке, описанном ниже.

Сервер Службы Такси должен:

1. правильно реагировать на запросы [тарифов](#), [профилей водителей](#) и [черного списка телефонов](#);
2. своевременно отправлять сервису данные об [изменении статусов водителей](#);
3. правильно обрабатывать запросы с [предложением заказа](#);
4. своевременно уведомлять Яндекс.Такси об [изменении статусов заказов](#).

Сотрудники Яндекса делают Службу Такси доступной пользователям Яндекс.Такси после того, как успешно протестирована обработка реальных заказов.

Адреса серверов Яндекс.Такси

Яндекс.Такси поддерживает работу двух серверов для взаимодействия со Службами Такси:

- Тестовый сервер предназначен для общего тестирования запросов к Яндекс.Такси.
Адрес тестового сервера: <https://ymsh.taxi-partners-test.mobile.yandex.net>.
- Боевой сервер работает с реальными данными о заказах и Службах Такси. Доступен только после успешного тестирования.

Адрес боевого сервера: <https://ymsh.mobile.yandex.net/>.

Программный интерфейс

API Яндекс.Такси использует протокол HTTP/HTTPS. Для обмена информацией между сервисом и Службой Такси используются запросы GET и POST.

Запросы API можно разделить на две группы:

- [Запросы синхронизации данных](#). Используются для синхронизации данных о тарифах, профилей водителей и статусов водителей в базах Яндекс.Такси и Службы такси.
- [Запросы обработки заказов](#). Используются для передачи данных о поступающих и выполняющихся заказах между Яндекс.Такси и Службой Такси.

Параметры запроса передаются в URL запроса и/или в XML-документе, содержащемся в теле запроса.

Ответ на запрос состоит из кода HTTP-ответа и тела ответа. Некоторые ответы могут не содержать тела, но код ответа всегда отражает результат обработки запроса.

Допустимые коды HTTP-ответа описаны отдельно для каждого запроса.

Запросы синхронизации данных

Сервис Яндекс.Такси ведет базу данных водителей и тарифов, а также поддерживает черный список телефонов клиентов. Соответствующая информация регулярно запрашивается на сервере Службы Такси. Служба Такси, в свою очередь, отправляет сервису данные о текущей занятости водителей по мере того, как водители освобождаются и принимают заказы.

Если запросы синхронизации данных перестали обрабатываться корректно, данные Яндекс.Такси о машинах и тарифах Службы Такси могут потерять актуальность. Обнаружив проблемы с синхронизацией данных, сотрудники Яндекса могут отключить Службу Такси от сервиса.

Реализованы следующие запросы синхронизации:

1. **Запрос тарифов.** Яндекс.Такси запрашивает у сервера Службы Такси актуальный список тарифов. Запрос производится с периодом 1 час.
2. **Запрос профилей водителей.** Яндекс.Такси запрашивает у сервера Службы Такси профили всех имеющихся водителей (общие сведения об имени, возрасте, используемой машине и т. п.). Запрос производится с периодом 1 час.
3. **Запрос черного списка телефонных номеров.** Яндекс.Такси запрашивает у сервера Службы Такси список телефонов клиентов, которых Служба Такси считает недобросовестными. Запрос производится с периодом 1 час.
4. **Обновление статусов водителей.** Сервер Службы Такси отправляет запрос к Яндекс.Такси, сообщая об изменении статуса одного или нескольких водителей Службы Такси.

Примечание:

Чтобы машины Службы Такси были доступны с помощью сервиса Яндекс.Такси, запросы тарифов, профилей водителей и обновления статусов водителей должны быть успешно выполнены хотя бы один раз.

Черные списки телефонов клиентов поддерживать необязательно.

Запрос тарифов

Яндекс.Такси отправляет запрос тарифов на сервер Службы Такси. Сервер обрабатывает запрос и возвращает ответ с XML-документом, содержащим список тарифов.

Внимание!

Яндекс.Такси переходит на новый формат тарифов. В связи с этим для запроса тарифов нужно подготавливать ответ как в [старом формате](#), так и в текущем.

Запрос

URL запроса

Запрос отправляется на адрес, предоставленный Службой Такси (отличный от адреса для тарифов в старом формате).

Параметры запроса

Параметры запроса отсутствуют.

Тело запроса

Тело запроса отсутствует.

Ответ

Описание кодов ответа

Код	Название	Пояснение
200	Ok	Запрос выполнен успешно
400	Bad Request	Запрос сформирован с ошибками (не хватает параметров, значение параметров указано неверно)
403	Forbidden	Ограничение в доступе. Передана неправильная пара "clid/apikey"
500	Internal Server Error	Внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

- 200 OK — всегда обозначает, что запрос выполнен полностью без ошибок;
- 400 Bad Request — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно, либо XML-документ в теле запроса сформирован неправильно.
- 403 Forbidden — в ответах сервера Яндекс.Такси всегда обозначает, что передана неправильная пара «идентификатор Службы Такси / API-ключ».
- 500 Internal Server Error — всегда обозначает, что возникла внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

Тело ответа содержит список тарифов Службы Такси.

Каждый тариф в списке разделен на наборы временных интервалов, для которых различаются цены и набор услуг. Например, расценки по тарифу могут различаться в дневное время будней, ночное время будней и выходные.

Для каждого набора временных интервалов отдельно описываются все доступные услуги и стоимость услуг. Также есть возможность указать набор услуг, доступных по тарифу в любое время.

Ниже приведен формат ответа на запрос тарифов. Элементы и их атрибуты приведены с примерами значений.

Многоточие после закрывающего тега XML-элемента означает, что ответ может содержать несколько таких элементов, следующих один за другим.

Все элементы XML-документа обязательны, если не указано иначе.

```
<?xml version="1.0" encoding="UTF-8"?>
<Tariffs>
  <Tariff>
    <Id>1</Id>
    <Name Lang="ru">Супер-Эконом Плюс</Name>
    <Intervals>
      <TariffChoice>start</TariffChoice>
      <Interval>
        <Schedule>
          <Span>
            <Days>
              <Item>Monday</Item>
              <Item>Tuesday</Item>
              <Item>Wednesday</Item>
            </Days>
            <TimeInterval>
              <Start>09:00</Start>
              <End>20:59</End>
            </TimeInterval>
          </Span>
          ...
        </Schedule>
        <FreeRoute>
          <Services>
```



```

<Service Type="taximeter" CalcRule="sum">
  <TaximeterCalc>
    <MinPrice Type="once">400</MinPrice>
    <StopSpeed Unit="km/h">5</StopSpeed>
    <Meter Type="distance">
      <Areas>
        <Item>suburb</Item>
      </Areas>
      <Price>5</Price>
      <Per Unit="kilometer">1</Per>
    </Meter>
    <Meter Type="idle_time">
      <Price>13</Price>
      <Per Unit="minute">1</Per>
    </Meter>
    ...
  </TaximeterCalc>
  ...
</Service>
<Service Type="continue_transfer">
  <TransferLocations>
    <Item>nao</Item>
    <Item>neao</Item>
    <Item>eao</Item>
    <Item>seao</Item>
  </TransferLocations>
  <TaximeterCalc>
    <Condition>
      <Meter Type="city_remoteness">
        <Value When="lt" Unit="kilometer">10</Value>
      </Meter>
    </Condition>
    <Meter Type="distance">
      <Areas>
        <Item>suburb</Item>
      </Areas>
      <Price>15</Price>
      <Per Unit="kilometer">1</Per>
    </Meter>
  </TaximeterCalc>
  <TaximeterCalc>
    <Condition>
      <Meter Type="city_remoteness">
        <Value When="gte" Unit="kilometer">10</Value>
      </Meter>
    </Condition>
    <Meter Type="distance">
      <Areas>
        <Item>suburb</Item>
      </Areas>
      <Price>25</Price>
      <Per Unit="kilometer">1</Per>
    </Meter>
  </TaximeterCalc>
  ...
</Service>
<Service Type="other">
  <Name Lang="ru">Помощь в переносе багажа, за единицу</Name>
  <MinPrice>50</MinPrice>
</Service>
<Service Type="child_chair">
  <MinPrice>100</MinPrice>
</Service>
<Service Type="waiting">
  <FreeTime Unit="minute">5</FreeTime>
</Service>
...
</Services>
<Comment Lang="ru">Перевозка животных осуществляется только при наличии

```

```

специально оборудованной сумки.</Comment>
</FreeRoute>
<FixedRoutes>
  <FixedRoute>
    <Routes>
      <Route>
        <Area Type="source">svo</Area>
        <Area Type="destination">cao</Area>
        <MinPrice>1250</MinPrice>
      </Route>
      ...
    </Routes>
    <Services>
      <Service Type="taximeter" CalcRule="sum">
        <TaximeterCalc>
          <Meter Type="time">
            <Prepaid Unit="second">7200</Prepaid>
            <Price>12</Price>
            <Per Unit="second">60</Per>
          </Meter>
          ...
        </TaximeterCalc>
      </Service>
      ...
    </Services>
    <Comment Lang="ru">При встрече в зале аэропорта с табличкой, стоянка
оплачивается отдельно клиентом и не включается в стоимость поездки!</Comment>
  </FixedRoute>
</FixedRoutes>
</Interval>
...
</Intervals>
</Tariff>
...
</Tariffs>

```

Ниже описаны элементы, которые могут быть включены в тело ответа на запрос информации о тарифах от Яндекс.Такси. Для элементов, содержащих значения, приведены примеры.

<Tariffs>

Корневой элемент описания тарифов.

<Tariff>

Элемент, содержащий описание отдельного тарифа.

<Id>

```
<Id>1</Id>
```

Идентификатор тарифа (присваивается Службой Такси). Не длиннее 32 символов, может содержать только латинские буквы и цифры.

<Name>

```
<Name Lang="ru">Супер-Эконом Плюс</Name>
```

Название тарифа. Обязательно хотя бы на русском языке.

Язык, на котором указано содержимое элемента, должен быть указан в атрибуте `Lang` согласно стандарту [ISO 3166-1 alpha-2](#).

<Intervals>

Элемент, описывающий расценки тарифа.

Обязательный элемент.

<TariffChoice>

```
<TariffChoice>start</TariffChoice>
```

Элемент указывает, как следует выбирать тариф поездки: ориентируясь на время подачи машины или на момент оплаты.

Допустимые значения:

- «start» — цена рассчитывается по расценкам, активным на момент начала поездки;
- «end» — цена рассчитывается по расценкам, активным на момент окончания поездки;
- «transition» — цена поездки включает в себя минимальную стоимость поездки на момент подачи машины и стоимость дальнейшего движения, которая рассчитывается в соответствии с расценками интервалов времени, в течение которых была совершена поездка.

<Interval>

Условия тарифа для определенного набора временных интервалов. Описание тарифа должно содержать хотя бы один элемент <Interval>.

<Schedule>

Время действия данной части тарифа.

В элементе следует описать временные промежутки, к которым привязаны расценки тарифа. Каждый элемент — сочетание дней недели (вложенный элемент <Days>) и времени (вложенный элемент <TimeInterval>).

Описание каждого тарифа должно включать элементы , которые в совокупности описывают все 24 часа каждого дня недели и дополнительного «праздничного» дня.

Например, время с 9:00 до 18:00 по будним дням следует описывать так:

```
<Span>
  <Days>
    <Item>Monday</Item>
    <Item>Tuesday</Item>
    <Item>Wednesday</Item>
    <Item>Thursday</Item>
    <Item>Friday</Item>
  </Days>
  <TimeInterval>
    <Start>09:00</Start>
    <End>17:59</End>
  </TimeInterval>
</Span>
```

Таким образом, остальные элементы (которые могут быть включены в другой элемент <Interval> внутри того же элемента <Tariff>) должны покрывать время с 00:00 до 08:59 и с 18:00 до 23:59 в будние дни, а все выходные и «праздничный» день.

<Days>

```
<Days>
  <Item>Monday</Item>
  <Item>Tuesday</Item>
  <Item>Wednesday</Item>
</Days>
```

Список дней, для которых действуют расценки (соответствующее время суток следует указывать в элементе <TimeInterval>). Каждый день должен быть указан во вложенном элементе <Item>.

Допустимые значения элемента `<Item>`:

- «Monday» — понедельник;
- «Tuesday» — вторник;
- «Wednesday» — среда;
- «Thursday» — четверг;
- «Friday» — пятница;
- «Saturday» — суббота;
- «Sunday» — воскресенье;
- «Holiday» — праздничные дни.

Пример указания временного интервала приведен в [описании элемента ``](#).

`<TimeInterval>`

```
<TimeInterval>
  <Start>09:00</Start>
  <End>20:59</End>
</TimeInterval>
```

Время действия тарифа в дни, перечисленные в элементе `<Days>`.

Время задается двумя вложенными элементами, `<Start>` и `<End>`, указывающими на начало и конец временного отрезка.

Пример указания временного интервала приведен в [описании элемента ``](#).

`<FreeRoute>`

Элемент, в котором должен быть описан тариф на обычные поездки (трансферы следует описывать в элементе `<FixedRoutes>`).

`<Services>`

Услуги, предлагаемые в обычных поездках.

`<Service>`

Отдельная услуга, которая может быть предоставлена клиенту. Для каждого элемента с описанием услуги обязателен атрибут `Type`, обозначающий вид услуги.

Ограничение:

Элементы `<Service>` с различными значениями атрибутов должны следовать в том же порядке, в котором эти значения перечислены ниже.

Допустимые значения атрибута `Type`:

- «taximeter» — базовая услуга перевозки. Элемент `<Service>` с таким значением атрибута должен содержать элемент `<TaximeterCalc>` с вложенными элементами `<MinPrice>` и `<Meter>`, а также атрибут `CalcRule`.

Атрибут `CalcRule` указывает, каким образом следует обрабатывать суммы, рассчитанные по правилам, описанным в различных элементах `<Meter>`. Допустимые значения атрибута:

- «sum» — в цену поездки следует включить сумму результатов по всем счетчикам (например, 5 рублей в минуту и 10 рублей за километр дадут для 10 минут и 2 километров общий результат в 70 рублей);
- «max» — в цену следует включить только максимальный результат счетчика (например, 7 рублей в минуту и 20 рублей за километр дадут для 10 минут и 4 километров результат в 80 рублей — учитывается только максимальная сумма).

Внимание!

Элемент `<Service>` с атрибутом `Type="taximeter"` **необязателен** для элемента `<FixedRoutes>`.

- «paid_dispatch» — платный выезд (отдельная плата за вызов машины). Элемент `<Service>` с таким значением атрибута должен содержать элемент `<TaximeterCalc>` с вложенным элементом `<MinPrice>` и/или `<Meter>`.
- «continue_transfer» — продолжение трансфера после конечной точки, указанной в элементе `<Route>`, или начало трансфера перед формальной отправной точкой.

Например, клиенту нужен трансфер из аэропорта Домодедово в Одинцово, а тариф регламентирует трансферы только в АО Москвы. Услуга показывает, что при выезде из Москвы следует включить счетчик и рассчитывать добавочную стоимость согласно ему.

Зоны, из которых можно продолжить трансфер, перечислены [ниже](#).

Элемент `<Service>` с атрибутом `Type="continue_transfer"` должен содержать элемент `<TaximeterCalc>` со вложенными элементами `<MinPrice>` и `<Meter>`.

- «waiting» — ожидание клиента. Элемент `<Service>` с таким значением атрибута может содержать элементы: `<FreeTime>`, `<Per>`, `<Price>`. Элементы `<Per>` и `<Price>` описывают стоимость платного ожидания и являются необязательными.
- «conditioner» — кондиционер в машине. Элемент `<Service>` с таким значением атрибута должен содержать элемент `<MinPrice>`.
- «no_smoking» — некурящий водитель. Элемент `<Service>` с таким значением атрибута должен содержать элемент `<MinPrice>`.
- «child_chair» — детское кресло в машине. Элемент `<Service>` с таким значением атрибута должен содержать элемент `<MinPrice>`.
- «universal» — подача машины с кузовом «универсал». Элемент `<Service>` с таким значением атрибута должен содержать элемент `<MinPrice>`.
- «animal_transport» — перевозка животных. Элемент `<Service>` с таким значением атрибута должен содержать элемент `<MinPrice>`.
- «other» — дополнительная услуга, предоставляемая Службой Такси. Чтобы указать несколько дополнительных услуг, включите в XML несколько элементов `<Service>` с атрибутом `Type="other"`. Такие элементы `<Service>` должны содержать элементы `<Name>` и `<MinPrice>`.

<TaximeterCalc>

Элемент, содержащий правило подсчета цены поездки. Правило расчета может быть основным и дополнительным (выполняться при выполнении условия).

Дополнительное правило подсчета цены поездки должно содержать элемент `<Condition>`. Тогда правило расчета цены поездки выполняется при выполнении всех условий `Condition` одновременно

<MinPrice>

```
<MinPrice Type="once">400</MinPrice>
```

Минимальная цена поездки.

<Meter>

Правило, по которому рассчитывается цена поездки в зависимости от времени или расстояния.

Тип счетчика следует задавать атрибутом `Type`. Значение атрибута может указывать на один из следующих способов расчета:

- «time» — плата за время в пути;
- «distance» — плата за пройденное расстояние;
- «idle_time» — плата за время простоя (когда скорость падает ниже значения `<StopSpeed>`);
- «driving_distance» — плата за расстояние, которое было пройдено со скоростью выше `<StopSpeed>`.

<Price>

```
<Price>12</Price>
```

Цена за минимальную единицу времени или расстояния. Валюта устанавливается автоматически для каждого города.

<Per>

```
<Per Unit="second">60</Per>
```

Время или расстояние, оцененное в элементе `<Price>`. Единицы измерения должны быть заданы в атрибуте `Unit`.

Допустимые значения атрибута:

- «second» — секунды;
- «minute» — минуты;
- «hour» — часы;
- «meter» — метры;
- «kilometer» — километры.

<Prepaid>

```
<Prepaid Unit="second">1800</Prepaid>
```

Предоплаченное время или расстояние поездки. Единицы измерения должны быть заданы в атрибуте `Unit`.

Допустимые значения атрибута:

- «second» — секунды;
- «minute» — минуты;
- «hour» — часы;
- «meter» — метры;
- «kilometer» — километры.

<Areas>

```
<Areas>
  <Item>suburb</Item>
</Areas>
```

Список всех географических зон, в которых должен действовать данный счетчик. Каждая зона должна быть указана в отдельном вложенном элементе `<Item>`.

Поддерживаемые зоны:

- «city» — город;
- «suburb» — пригород;
- [зоны, поддерживаемые только в Москве.](#)

<Condition>

Элемент, содержащий описание условия, при выполнении которого действует правило расчета цены поездки (элемент <Meter> элемента <Condition>).

```
<Condition>
  <Meter Type="city_remoteness">
    <Value When="lt" Unit="kilometer">10</Value>
  </Meter>
</Condition>
```

<Meter>

```
<Meter Type="city_remoteness">
```

Тип условия. Имеет атрибут Type, возможные значения которого следующие:

- «city_remoteness» — расстояние поездки;
- «time» — время поездки;
- «price» — стоимость поездки.

<Value>

```
<Value When="lt" Unit="kilometer">10</Value>
```

Семантика условия. Например, элемент <Value>, показанный выше описывает ситуацию когда пройденное расстояние меньше 10 километров. Элемент имеет атрибуты:

- When — оператор условия;
- Unit — единицы измерения условия.

Возможные значения атрибута When:

- «lt» — означает «меньше чем»;
- «gte» — означает «больше или равно».

Возможные значения атрибута Unit:

- «second» — секунды;
- «minute» — минуты;
- «hour» — часы;
- «meter» — метры;
- «kilometer» — километры.

<StopSpeed>

```
<StopSpeed Unit="km/h">5</StopSpeed>
```

Скорость, при которой машина считается простаивающей. Единицы измерения следует задавать в атрибуте Unit, возможные значения которого:

- «km/h» — километры в час;
- «m/s» — метры в секунду.

Элемент <StopSpeed> обязателен, если в элементе <TaximeterCalc> присутствуют элементы <Meter> с атрибутом Type="idle_time" или Type="driving_distance".

Если скорость машины равна заданному значению или меньше его, расчет переключается на правило `<Meter>` с атрибутом `Type="idle_time"` (если оно есть).

Если скорость машины выше заданного значения, расчет переключается на правило `<Meter>` с атрибутом `Type="driving_distance"` (если оно есть).

<Name>

Название дополнительной услуги, предоставляемой Службой Такси. Используется и обязательно (хотя бы на русском языке) только если для элемента `<Service>` предоставлен атрибут `Type="other"`.

Язык, на котором указано содержимое элемента, должен быть указан в атрибуте `Lang` согласно стандарту [ISO 3166-1 alpha-2](#).

<MinPrice>

```
<MinPrice>100</MinPrice>
```

Цена услуги.

<FreeTime>

Бесплатное время ожидания клиента. Единицы измерения следует указывать в атрибуте `Unit`. Допустимые значения атрибута:

- «second» — секунды;
- «minute» — минуты;
- «hour» — часы.

Нужное количество секунд или минут должно быть указано в значении элемента. Пример для 30 минут бесплатного ожидания:

```
<FreeTime Unit="second">1800</FreeTime>
```

<TransferLocations>

```
<TransferLocations>
  <Item>eao</Item>
  <Item>neao</Item>
</TransferLocations>
```

Географические зоны, из которой можно продолжить трансфер. Каждая зона должна быть указана в отдельном вложенном элементе `<Item>`.

Поддерживаемые зоны:

- «city» — город;
- «suburb» — пригород;
- [зоны, поддерживаемые только в Москве](#).

<Comment>

```
<Comment Lang="ru">Перевозка животных осуществляется только при
наличии специально оборудованной сумки.</Comment>
```

Комментарий к тарифам на обычные поездки. Обязательный элемент.

Комментарий — это одно законченное предложение, заканчивающееся корректным знаком препинания (точкой, восклицательным знаком и т. д.).

Язык, на котором указано содержимое элемента, должен быть указан в атрибуте `Lang` согласно стандарту [ISO 3166-1 alpha-2](#).

<FixedRoutes>

Элемент, в котором должен быть описан тариф на трансферы (обычные поездки следует описывать в элементе [<FreeRoute>](#)).

<FixedRoute>

Описание условий обслуживания группы трансферов (для трансферов с разными доступными услугами)

<Routes>

Элемент, включающий описания всех трансферов, входящих в группу.

<Route>

Описание отдельного трансфера. Обязательно включает начальную и конечную точки трансфера, а также его минимальную цену.

```
<Route>
  <Area Type="source">svo</Area>
  <Area Type="destination">cao</Area>
  <MinPrice>1250</MinPrice>
</Route>
```

<Area>

Точка отправления или прибытия.

<MinPrice>

Минимальная цена трансфера (без дополнительных услуг).

<Services>

Услуги, предоставляемые при трансферах, которые перечислены в элементе [<Routes>](#). Содержит элементы [<Service>](#), которые описывают услуги аналогично описанию [услуг для обычных поездок](#).

В случае отсутствия элемента [<Service>](#) с атрибутом `Type="taximeter"` цена трансфера соответствует минимальной цене (элемент [<MinPrice>](#) элемента [<Route>](#)).

<Comment>

```
<Comment Lang="ru">При встрече в зале аэропорта с табличкой,
стоянка оплачивается отдельно клиентом и не включается в стоимость
поездки!</Comment>
```

Комментарий к тарифам на трансферы. Обязательный элемент (комментарий должен быть заполнен хотя бы на русском языке).

Комментарий — это одно законченное предложение, заканчивающееся корректным знаком препинания (точкой, восклицательным знаком и т. д.).

Язык, на котором указано содержимое элемента, должен быть указан в атрибуте `Lang` согласно стандарту [ISO 3166-1 alpha-2](#).

Список географических зон в Москве

Для указания в элементах <Area> и <Areas> Москва разделена на географические зоны. Ниже приведен их список и соответствующие им значения.

- «mkad» — МКАД
- «sao» — ЮАО
- «zelenogradsky» — Зеленоградский АО
- «seao» — ЮВАО
- «wao» — ЗАО
- «nwao» — СЗАО
- «swao» — ЮЗАО
- «nao» — ЦАО
- «eao» — ВАО
- «troitsky» — Троицкий АО
- «novomoskovsky» — Новомосковский АО
- «neao» — СВАО
- «cao» — ЦАО
- «yaroslavsky» — Ярославский вокзал
- «paveletsky» — Павелецкий вокзал
- «kiyevsky» — Киевский вокзал
- «vko» — аэропорт Внуково
- «kursky» — Курский вокзал
- «rizhsky» — Рижский вокзал
- «savyolovsky» — Савёловский вокзал
- «kazansky» — Казанский вокзал
- «svo» — аэропорт Шереметьево
- «dme» — аэропорт Домодедово
- «belorussky» — Белорусский вокзал
- «leningradsky» — Ленинградский вокзал

Запрос тарифов (устаревший формат)

Яндекс.Такси отправляет запрос тарифов на сервер Службы Такси. Сервер обрабатывает запрос и возвращает ответ с XML-документом, содержащим список тарифов.

Внимание!

Яндекс.Такси переходит на новый формат тарифов. В связи с этим для запроса тарифов нужно подготавливать ответ как в [новом формате](#), так и в старом.

Запрос

URL запроса

Запрос отправляется на адрес, предоставленный Службой Такси.

Параметры запроса

Параметры запроса отсутствуют.

Тело запроса

Тело запроса отсутствует.

Ответ

Допускаются следующие коды HTTP-ответа:

- 200 OK — всегда обозначает, что запрос выполнен полностью без ошибок;
- 400 Bad Request — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно, либо XML-документ в теле запроса сформирован неправильно.
- 403 Forbidden — в ответах сервера Яндекс.Такси всегда обозначает, что передана неправильная пара «идентификатор Службы Такси / API-ключ».
- 500 Internal Server Error — всегда обозначает, что возникла внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

Тело ответа содержит список тарифов Службы Такси.

Каждый тариф в списке разделен на наборы временных интервалов, для которых различаются цены и набор услуг. Например, расценки по тарифу могут различаться в дневное время будней, ночное время будней и выходные.

Для каждого набора временных интервалов отдельно описываются все доступные услуги и стоимость услуг. Также есть возможность указать набор услуг, доступных по тарифу в любое время.

Ниже приведен формат ответа на запрос тарифов. Элементы и их атрибуты приведены с примерами значений.

Многоточие после закрывающего тега XML-элемента означает, что ответ может содержать несколько таких элементов, следующих один за другим.

Все элементы XML-документа обязательны, если не указано иначе.

```
<?xml version="1.0" encoding="UTF-8"?>
<Tariffs>
  <Tariff>
    <Id>t000001</Id>
    <Name>Супер-Эконом Плюс</Name>
    <Description>
      <Interval name="Ночь">
        <Schedule>
          <Span>
            <Interval start="23:00" end="07:00" />
            <Weekday>1</Weekday>
            <Weekday>2</Weekday>
            ...
          </Span>
          ...
        </Schedule>
        <Special>В период новогодних праздников действует наценка 200%!</Special>
        <City>
          <MinPrice>350</MinPrice>
          <Currency>руб</Currency>
          <Included>20 мин</Included>
          <Other>11 руб/мин</Other>
        </City>
        <Transfer>
          <Destination type="airport" name="Внуково">
            <Source type="district" name="Любой, кроме ЮВАО">1500</Source>
```

```

...
</Destination>
...
<Unit name="встреча с табличкой">200 руб</Unit>
...
<Other>Тарифы на такси в аэропорт действуют в пределах МКАД, далее
тарифицируются согласно тарифам по городу.</Other>
</Transfer>
<Suburb>
  <DriveMKAD>+10 руб/км</DriveMKAD>
  <DriveAfterMKAD>+15 руб/км, в пределах 10 км от города, далее +25 руб/
км</DriveAfterMKAD>
  <StartAfterMKAD>100 руб в пределах 10 км, далее +5 руб/км</
StartAfterMKAD>
  <Unit name="езда по магистрали">+5 руб/км</Unit>
  ...
  <Other>При поездке в область и обратно в рамках одной поездки вся поездка
оплачивается по городскому тарифу.</Other>
</Suburb>
<OverPrice>
  <Unit name="курение в салоне">200 руб</Unit>
  ...
  <Other>В случае повреждения или загрязнения салона автомобиля или
багажного отдела по вине пассажира, клиент обязан оплатить стоимость ремонта или
химчистки.</Other>
</OverPrice>
  <Other>Для постоянных клиентов существует гибкая система скидок!</Other>
</Interval>
...
<CommonOverPrice>
  <Unit name="перевозка животных">200 руб</Unit>
  ...
  <Other>Плата за отказ от поданной машины – минимальная стоимость поездки.</
Other>
</CommonOverPrice>
</Description>
</Tariff>
...
</Tariffs>

```

Ниже описаны элементы, которые могут быть включены в тело ответа на запрос информации о тарифах от Яндекс.Такси. Для элементов, содержащих значения, приведены примеры.

<Tariffs>

Корневой элемент описания тарифов.

<Tariff>

Элемент, содержащий описание отдельного тарифа.

<Id>

```
<Id>t0000001</Id>
```

Идентификатор тарифа (присваивается Службой Такси). Не длиннее 32 символов, может содержать только латинские буквы и цифры.

<Name>

```
<Name>Супер-Эконом Плюс</Name>
```

Название тарифа.

<Description>

Элемент, описывающий расценки тарифа.

<Interval>

```
<Interval name="выходные">
```

Условия тарифа для определенного набора временных интервалов. В ответе должен содержаться хотя бы один элемент <Interval>.

Атрибут:

- name — название интервала.

<Schedule>

Время действия данной части тарифа. Необязательный элемент.

Время действия тарифа.

```
<!-- Тариф действует с 14 до 15 по понедельникам -->
<Span>
  <Interval start="14:00" end="14:59" />
  <Weekday>1</Weekday>
</Span>

<!-- Круглые сутки по вторникам и средам -->
<Span>
  <Interval start="00:00" end="23:59" />
  <Weekday>2</Weekday>
  <Weekday>3</Weekday>
</Span>

<!-- С 7 до 8 и с 17 до 18 по четвергам и пятницам -->
<Span>
  <Interval start="07:00" end="07:59" />
  <Interval start="17:00" end="17:59" />
  <Weekday>4</Weekday>
  <Weekday>5</Weekday>
</Span>
```

Каждый элемент указывает на временные интервалы <Interval> в каждый из дней <Weekday>: все указанные интервалы относятся ко всем перечисленным дням. Если интервалы пересекаются (например, с 7:00 до 10:00 и с 9:00 до 11:00), они будут объединены в один (с 7:00 до 11:00). Таким образом можно тарифицировать регулярные промежутки времени, например, вечера будней.

Чтобы тарифицировать разные временные промежутки для разных дней (утро понедельника и вечер вторника), используйте несколько элементов , указав в каждом из них нужный интервал и день недели.

<Interval>

```
<Interval start="23:00" end="07:00" />
```

Интервал времени в пределах суток.

Если начало интервала указано позже, чем конец, то интервал разбивается на два: от полуночи до более раннего времени, и от более позднего времени до полуночи.

Таким образом, в примере указано два интервала: от полуночи до 7 часов и от 23 часов до полуночи.

Внимание!

Интервалы, указанные в рамках одного , не должны пересекаться. Например, интервалы с 6 до 8 часов и с 8 до 9 нельзя совместить так:

```
<Interval start="06:00" end="08:00" />
<Interval start="08:00" end="09:00" />
```

Правильно:

```
<Interval start="06:00" end="07:59" />
<Interval start="08:00" end="09:00" />
```

Атрибуты:

- start — начало интервала;
- end — конец интервала.

<Weekday>

```
<Weekday>1</Weekday>
```

Номер для недели (1 соответствует понедельнику, 2 — вторнику и т. д.).

<Special>

```
<Special>В период новогодних праздников действует наценка 200%!  
</Special>
```

Спецпредложение или другая важная информация. Необязательный элемент.

<City>

Условия перевозки по городу.

<MinPrice>

```
<MinPrice>350</MinPrice>
```

Стоимость минимального заказа

<Currency>

```
<Currency>руб</Currency>
```

Валюта, в которой указывается стоимость.

<Included>

```
<Included>20 мин</Included>
```

Объем услуг, включенных в минимальный заказ.

<Other>

```
<Other>11 руб/мин</Other>
```

Стоимость сверх минимального заказа и другие условия. Необязательный элемент.

<Transfer>

Поездки с фиксированной стоимостью (от и до аэропортов, между аэропортами и прочее). Необязательный элемент.

<Destination>

```
<Destination type="airport" name="Внуково">
```

Условия трансфера до определенного пункта назначения. Каждый элемент <Destination> может включать несколько элементов <Source>, описывающих возможное место отправления для трансфера.

Атрибуты:

- type — тип места назначения. Возможные значения:
 - «airport» — аэропорт;
 - «district» — район города;
 - «train» — железнодорожный вокзал.
- name — название места назначения.

<Source>

```
<Source type="district" name="Любой, кроме ЮВАО">1500</Source>
```

Значение элемента содержит цену трансфера в валюте, указанной в элементе <Currency>. Пункт назначения трансфера описывается в родительском элементе <Destination>, пункт отправления — в атрибутах элемента.

Если требуется указать несколько пунктов отправления с одинаковой ценой, допускается создать один элемент <Source>, описав все требуемые пункты отправления в атрибуте name (см. пример).

Атрибуты:

- type — тип места отправления. Возможные значения:
 - «airport» — аэропорт;
 - «district» — район города;
 - «train» — железнодорожный вокзал.
- name — название места отправления в свободной форме.

<Unit>

```
<Unit name="встреча с табличкой">200 руб</Unit>
```

Значение элемента — цена услуги, общей для всех поездок с фиксированной стоимостью.

Атрибут:

- name — название услуги.

<Other>

```
<Other>Тарифы на такси в аэропорт действуют в пределах МКАД, далее тарифицируются согласно тарифам по городу.</Other>
```

Комментарии ко всем поездкам с фиксированной стоимостью.

<Suburb>

Условия перевозки за чертой города. Необязательный элемент.

<DriveMKAD>

```
<DriveMKAD>+10 руб/км</DriveMKAD>
```

Наценка за движение по МКАДу.

<DriveAfterMKAD>

```
<DriveAfterMKAD>+15 руб/км, в пределах 10 км от города, далее +25 руб/км</DriveAfterMKAD>
```

Наценка за движение за МКАДом.

<StartAfterMKAD>

```
<StartAfterMKAD>100 руб в пределах 10 км, далее +5 руб/км</StartAfterMKAD>
```

Наценка за подачу машины за МКАД.

<Unit>

```
<Unit name="езда по магистрали">+5 руб/км</Unit>
```

Другие услуги, одинаковые для всех поездок за чертой города.

Атрибут:

- name — название услуги.

<Other>

```
<Other>При поездке в область и обратно в рамках одной поездки вся поездка оплачивается по городскому тарифу.</Other>
```

Комментарии о движении за чертой города.

<OverPrice>

Общие услуги для временного интервала. Необязательный элемент.

<Unit>

```
<Unit name="курение в салоне">200 руб</Unit>
```

Стоимость отдельной услуги, общей для временного интервала. Название услуги указывается в атрибуте name.

Атрибут:

- name — название услуги.

<Other>

```
<Other>В случае повреждения или загрязнения салона автомобиля или багажного отдела по вине пассажира, клиент обязан оплатить стоимость ремонта или химчистки.</Other>
```

Комментарии к услугам.

<Other>

```
<Other>Для постоянных клиентов существует гибкая система скидок!</Other>
```

Общий комментарий к тарифу. Необязательный элемент.

<CommonOverPrice>

Услуги, общие для всех интервалов тарифа. Необязательный элемент.

<Unit>

```
<Unit name="перевозка животных">200 руб</Unit>
```

Стоимость услуги, общей для всех интервалов тарифа. Название услуги указывается в атрибуте name.

Атрибут:

- name — название услуги.

<Other>

```
<Other>Плата за отказ от поданной машины — минимальная стоимость поездки.</Other>
```

Общий комментарий к тарифу.

Запрос профилей водителей

Яндекс.Такси отправляет запрос профилей водителей на сервер Службы Такси. Сервер обрабатывает запрос и возвращает ответ с XML-документом, содержащим профили водителей.

Запрос

URL запроса

Запрос отправляется по адресу, предоставленному Службой Такси.

Параметры запроса

Параметры запроса отсутствуют.

Тело запроса

Тело запроса отсутствует.

Ответ

Допускаются следующие коды HTTP-ответа:

- 200 OK — всегда обозначает, что запрос выполнен полностью без ошибок;
- 400 Bad Request — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно, либо XML-документ в теле запроса сформирован неправильно.
- 403 Forbidden — в ответах сервера Яндекс.Такси всегда обозначает, что передана неправильная пара «идентификатор Службы Такси / API-ключ».
- 500 Internal Server Error — всегда обозначает, что возникла внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

Тело ответа содержит профили водителей Службы Такси.

Профиль каждого водителя состоит из двух частей: личные данные водителя и характеристики машины, которую водитель использует.

Ниже приведен формат ответа на запрос информации о водителях. Элементы и их атрибуты приведены с примерами значений.

Многоточие после окончания XML-элемента означает, что ответ может содержать несколько таких элементов, следующих один за другим.

Все элементы XML-документа обязательны, если не указано иначе.

```
<?xml version="1.0" encoding="UTF-8"?>
<Cars>
  <Car>
    <Uuid>ua65451</Uuid>
    <RealClid>rc651</RealClid>
    <RealName>Старое Черное Такси</RealName>
    <RealWeb>http://oldblacktaxi.ru</RealWeb>
    <RealScid>450</RealScid>
    <Tariff>t000001</Tariff>
    ...
    <DriverDetails>
      <DisplayName>Иванов Иван Иванович</DisplayName>
      <Phone>+79022617513</Phone>
      <Age>1965</Age>
    </DriverDetails>
    <CarDetails>
      <Model>Mercedes Grand Vitara</Model>
      <Age>1998</Age>
      <Color>черный</Color>
      <CarNumber>K221PT150</CarNumber>
      <Permit>00708</Permit>
      <Require name="has_conditioner">yes</Require>
      <Require name="no_smoking">no</Require>
      <Require name="child_chair">0.6-4</Require>
      <Require name="animal_transport">yes</Require>
      <Require name="universal">no</Require>
      <Require name="check">no</Require>
    </CarDetails>
  </Car>
  ...
</Cars>
```

Ниже описаны элементы, которые могут быть включены в тело ответа на запрос информации о тарифах от Яндекс.Такси. Для элементов, содержащих значения, приведены примеры.

<Cars>

Корневой элемент XML-документа с информацией о водителях.

<Car>

Элемент, содержащий описание отдельного водителя и его машины.

<Uuid>

```
<Uuid>ua65451</Uuid>
```

Идентификатор водителя. Присваивается Службой Такси.

Каждый идентификатор водителя должен быть уникален в рамках Службы Такси. Также идентификаторы должны быть не длиннее 32 символов, и должны состоять только из латинских букв и цифр.

<RealClid>

```
<RealClid>rc651</RealClid>
```

Идентификатор таксопарка, который участвует в Яндекс.Такси при посредничестве Службы Такси. Необязательный элемент.

Если Служба Такси выступает посредником для нескольких независимых таксопарков, машины этих таксопарков считаются принадлежащими Службе Такси. Для всех таких машин клиентские приложения отображают логотип Службы Такси.

Чтобы машины независимых таксопарков отображались с собственными логотипами таксопарков, Служба Такси должна:

1. Назначить каждому таксопарку уникальный идентификатор. Идентификатор должен быть не длиннее 32 символов, и должен состоять только из латинских букв и цифр.
2. Передать Яндекс.Такси идентификаторы таксопарков с соответствующими файлами логотипов.
3. В ответе на запрос профилей водителей указывать идентификаторы таксопарков, в которых работают водители, в элементах `<RealClid>`.

<RealName>

```
<RealName>Старое Черное Такси</RealName>
```

Название таксопарка, чей идентификатор указан в значении `<RealClid>`, на русском языке. Обязательный элемент.

<RealWeb>

```
<RealWeb>http://oldblacktaxi.ru</RealWeb>
```

Адрес сайта таксопарка, чей идентификатор указан в значении `<RealClid>`. Необязательный элемент.

Адрес должен быть указан вместе с префиксом протокола (в примере — `http://`). Также адрес не должен заканчиваться слэшем: например, адрес `http://taxi.ru/` обработан не будет.

<RealScid>

Номер получателя платежа в Яндекс.Деньгах. Необязательный элемент.

Если разные водители Службы Такси приписаны к различным юридическим лицам, вы можете указать нужный номер получателя платежа для каждого водителя.

<Tariff>

```
<Tariff>t0000001</Tariff>
```

Идентификатор тарифа.

Для одного водителя может быть указано несколько тарифов. Первым в списке должен идти основной тариф, затем перечисляются тарифы, по которым Водитель готов взять заказ, если более высокий тариф не устраивает клиента.

<DriverDetails>

Информация о водителе. Необязательный элемент.

<DisplayName>

```
<DisplayName>Иванов Иван Иванович</DisplayName>
```

Полные фамилия, имя и отчество водителя на русском языке.

<Phone>

```
<Phone>+79022617513</Phone>
```

Номер телефона водителя в формате +7XXXXXXXXXX.

<Age>

```
<Age>1965</Age>
```

Год рождения водителя.

<CarDetails>

Элемент, содержащий характеристики машины.

<Model>

```
<Model>Mercedes Grand Vitara</Model>
```

Марка и модель автомобиля в наиболее близком к оригинальному написанию.

<Age>

```
<Age>1998</Age>
```

Год выпуска автомобиля. Необязательный элемент.

<Color>

```
<Color>черный</Color>
```

Цвет машины на русском языке. Необязательный элемент.

<CarNumber>

```
<CarNumber>K221PT150</CarNumber>
```

Государственный номер и регион регистрации автомобиля. Необязательный элемент.

Все буквы номера — заглавные буквы кириллицы.

<Permit>

```
<Permit>00708</Permit>
```

Номер разрешения на перевозки. Необязательный элемент.

Расположение номера на московском разрешении:



Внимание!

Номер разрешения проверяется в паре с номером автомобиля (элемент **<CarNumber>**). Разрешение считается отсутствующим, если:

- номер автомобиля или разрешения отсутствует;
- номер автомобиля или разрешения указан в неверном формате;
- база разрешений не содержит разрешения с указанным номером для заданного номера автомобиля.

<Require>

```
<Require name="child_chair">0.11-4</Require>
<Require name="check">yes</Require>
```

Характеристика машины. Каждая характеристика описывается ровно одним элементом **<Require>**. Значение элемента интерпретируется в зависимости от значения атрибута name (атрибут содержит идентификатор характеристики):

- name="animal_transport": перевозка животных.
 - «yes» — в машине можно перевозить животных;
 - «no» — животных перевозить нельзя.
- name="check": строгая отчетность при оплате.
 - «yes» — водитель может выписать квитанцию об оплате заказа (документ строгой отчетности);
 - «no» — водитель не может выписать квитанцию.
- name="child_chair": детское кресло.
 - «возраст ребенка» — детское кресло есть, предназначено для детей указанного возраста. Возраст указывается в виде промежутка между минимальным и максимальным значением, например: «0.11-4».

Возраст меньше года указывается в формате «0.<количество месяцев>» (11 месяцев следует записать как «0.11»). Возраст больше года указывается как целое число лет.

- «no» — детского кресла нет.
- name="has_conditioner": автомобильный кондиционер.
 - «yes» — в машине есть кондиционер;
 - «no» — кондиционера нет.
- name="no_smoking": разрешено ли курение.
 - «yes» — водитель не курит, пассажирам курить запрещено;
 - «no» — пассажирам курить разрешено.
- name="universal": кузов «универсал».
 - «yes» — машина с кузовом «универсал»;
 - «no» — машина с другим кузовом.

Запрос черного списка телефонных номеров

Каждая Служба Такси может поддерживать черный список телефонных номеров. Если черный список поддерживается, сервер Службы Такси должен возвращать данные списка по запросу Яндекс.Такси. Сервис консолидирует все полученные черные списки и учитывает их, формируя запросы о поступивших заказах, как срочных, так и несрочных.

Если черный список поддерживать не планируется, следует сообщить об этом сотрудникам Яндекса, и запрос черного списка не будет проверяться.

Яндекс.Такси отправляет запрос черного списка телефонных номеров на сервер Службы Такси. Сервер обрабатывает запрос и возвращает ответ с XML-документом, содержащим черный список.

Запрос

URL запроса

Запрос отправляется по адресу, предоставленному Службой Такси.

Параметры запроса

Параметры запроса отсутствуют.

Тело запроса

Тело запроса отсутствует.

Ответ

Допускаются следующие коды HTTP-ответа:

- 200 OK — всегда обозначает, что запрос выполнен полностью без ошибок;
- 400 Bad Request — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно, либо XML-документ в теле запроса сформирован неправильно.
- 403 Forbidden — в ответах сервера Яндекс.Такси всегда обозначает, что передана неправильная пара «идентификатор Службы Такси / API-ключ».
- 500 Internal Server Error — всегда обозначает, что возникла внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

Тело ответа содержит номера телефонов клиентов из черного списка Службы Такси.

Ниже приведен формат ответа на запрос черного списка телефонных номеров. Элементы и их атрибуты приведены с примерами значений.

Многоточие после окончания XML-элемента означает, что ответ может содержать несколько таких элементов, следующих один за другим.

```
<?xml version="1.0" encoding="UTF-8"?>
<Blacklist>
  <Entry phone="+71234567890" />
  ...
</Blacklist>
```

Ниже описаны элементы, которые могут быть включены в тело ответа на запрос информации о тарифах от Яндекс.Такси. Для элементов, содержащих значения, приведены примеры.

<Blacklist>

Корневой элемент XML-документа с черным списком телефонных номеров.

<Entry>

```
<Entry phone="+71234567890" />
```

Элемент, содержащий отдельный номер телефона из черного списка в атрибуте phone.

Атрибут:

- phone — номер телефона, в формате +7XXXXXXXXXX.

Обновление статусов водителей

Сервер Службы Такси отправляет запрос обновления статусов водителей сервису Яндекс.Такси. Сервис обрабатывает запрос и возвращает результат обработки в коде HTTP-ответа.

Сведениями о статусе одного водителя передаются с помощью GET-запроса, без использования тела. Сведения о нескольких водителях передаются с помощью XML-документа, включаемого в тело POST-запроса.

Примечание:

Получив GET-запрос, сервер Яндекс.Такси извлекает информацию о статусе водителя только из URL.

Получив POST-запрос, сервер Яндекс.Такси извлекает информацию о статусах водителей из тела запроса, игнорируя соответствующие параметры URL.

Запрос

Формат URL запроса

```
https://ymsh.mobile.yandex.net/1.x/carstatus ?
clid=1822049
& apikey=283bbb6e157a4a529ac06c101aa15020
& [uuid=<идентификатор водителя>]
& [status= free | busy | verybusy ]
```

Параметр	Значение	Описание
Обязательные параметры		
clid	<идентификатор Службы Такси>	Идентификатор Службы Такси, назначенный сервисом Яндекс.Такси.
apikey	<API-ключ>	API-ключ Службы Такси, выданный сервисом Яндекс.Такси.
Дополнительные параметры		
uuid	<идентификатор водителя>	Идентификатор водителя, назначенный Службой Такси.

Параметр	Значение	Описание
		<p>Параметр может использоваться, если в запросе передается статус только одного водителя.</p> <p>Представляет собой буквенно-цифровую последовательность не длиннее 32 символов.</p>
status	<API-ключ>	<p>Статус водителя.</p> <p>Параметр может использоваться, если в запросе передается статус только одного водителя.</p> <p>Возможные значения:</p> <ul style="list-style-type: none"> «free» — «свободен», водитель предположительно не занят, но нельзя с определенностью сказать, возьмется ли он за заказ. Яндекс.Такси следует отослать предложение заказа и дожидаться подтверждения. «busy» — «занят», водитель занят выполнением заказа. «verybusy» — «очень занят», водитель занят и освободится нескоро (взял следующий заказ в очередь, заканчивает работу, планирует перерыв и т. п.). <p>Обязательно поддерживать только статусы «free» и «busy». Остальные статусы Служба Такси поддерживает по своему усмотрению (чем больше используемых статусов, тем точнее результаты поиска машин с помощью сервиса Яндекс.Такси).</p>

Тело запроса

Ниже приведен формат тела запроса с отчетом об изменении статусов водителей. Элементы и их атрибуты приведены с примерами значений. Для элементов, содержащих значения, приведены примеры.

```
<?xml version="1.0" encoding="UTF-8"?>
<CarsStatus>
  <Cars>
    <Car uuid="ua65451" status="ready">
    </Car>
    ...
  </Cars>
</CarsStatus>
```

Ниже описаны элементы, которые могут быть включены в тело запроса с информацией о статусах водителей.

Все элементы XML-документа обязательны, если не указано иначе.

<CarsStatus>

Корневой элемент XML-документа с информацией о статусах водителей.

<Cars>

Элемент, содержащий информацию о статусах водителей.

<Car>

```
<Car uuid="ua65451" status="ready">
```

Элемент, атрибуты которого содержат информацию о статусе отдельного водителя.

Атрибуты:

- `uuid` — идентификатор водителя, назначаемый Службой Такси;
- `status` — статус водителя, из списка возможных значений:
 - «free» — «свободен», водитель предположительно не занят, но нельзя с определенностью сказать, возьмется ли он за заказ. Яндекс.Такси следует отослать [предложение заказа](#) и дожидаться [подтверждения](#).
 - «busy» — «занят», водитель занят выполнением заказа.
 - «verybusy» — «очень занят», водитель занят и освободится нескоро (взял следующий заказ в очередь, заканчивает работу, планирует перерыв и т. п.).

Ответ

Ответ Яндекс.Такси на отчет об изменении статусов водителей не содержит тела. Результат обработки описывается кодом HTTP-ответа.

Особым значением для такого ответа обладают следующие коды:

- 203 `Non-Authoritative Information` — возвращается в ответ на запрос с информацией о нескольких водителях, если только некоторые из указанных водителей присутствуют в базе данных Яндекс.Такси.
- 404 `Not Found` — возвращается, если ни один указанный в запросе водитель не был найден в базе данных Яндекс.Такси.

Остальные коды HTTP-ответа несут тот же смысл, что и для других запросов:

- 200 `OK` — всегда обозначает, что запрос выполнен полностью без ошибок;
- 400 `Bad Request` — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно, либо XML-документ в теле запроса сформирован неправильно.
- 403 `Forbidden` — в ответах сервера Яндекс.Такси всегда обозначает, что передана неправильная пара «идентификатор Службы Такси / API-ключ».
- 500 `Internal Server Error` — всегда обозначает, что возникла внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

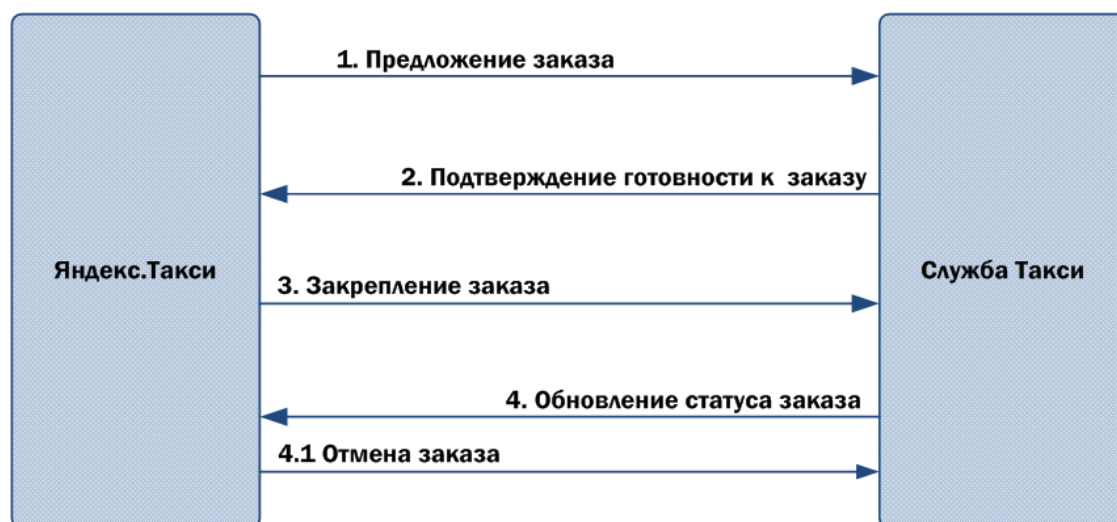
Запросы обработки заказов

Каждый заказ, поступивший в Яндекс.Такси, обрабатывается в определенном порядке:

1. Сервис Яндекс.Такси получает заказ от пользователя. Заказ **предлагается** Службам Такси в зависимости от срочности:
 - Если заказ срочный, сервис самостоятельно подбирает водителей для каждой Службы Такси и предлагает согласовать заказ с одним из них.
 - Если заказ несрочный, Службы Такси могут согласовать его с любым из своих водителей (Яндекс.Такси не подбирает подходящих водителей).
2. Служба Такси должна прислать **подтверждение** того, что найден подходящий водитель. Заказ закрепляется за той Службой Такси, которая пришлет подтверждение первой.
3. Получив подтверждение, Яндекс.Такси **закрепляет заказ за водителем**.
4. После того, как водитель приступил к заказу, сервер Службы Такси оповещает Яндекс.Такси о каждом **изменении статуса заказа**.
 1. Яндекс.Такси может **отменить предложенный заказ** в любой момент до подачи машины.

Ниже приведена общая схема обработки заказов.

Схема обработки заказа



Предложение заказа

Яндекс.Такси отправляет Службе Такси запрос с предложением заказа.

- Если заказ срочный, сервис перечисляет в запросе подобранных для заказа водителей. Срочный заказ можно закрепить только за водителями из этого списка.

Примечание:

Яндекс.Такси может присылать несколько запросов об одном и том же срочном заказе. Заказ может быть передан любому из указанных в этих запросах водителей.

- Если заказ несрочный, Служба Такси должна выбрать водителя самостоятельно, учитывая перечисленные в запросе требования:
 - тариф (элемент запроса `<Tariff>`);
 - необходимые характеристики машины (элемент `<Requirements>`).

Если заказ принимается, Служба Такси должна ответить [подтверждением](#) сервису Яндекс.Такси. Если заказ принят не будет, на запрос отвечать не следует.

Совет:

Несрочные заказы не обязательно подтверждать сразу. Такие заказы можно предлагать водителям как срочные, незадолго до времени подачи — водителям будет легче планировать свое время.

Запрос

Формат URL запроса

```
<адрес, предоставленный Службой Такси>/1.x/requestcar
```

Тело запроса

Ниже приведен формат тела запроса с предложением срочного заказа и списком водителей, подобранных сервисом Яндекс.Такси. Сервис гарантирует, что все водители, включенные в список, были получены с помощью [запроса информации о водителях Службы Такси](#), и для каждого из них хотя бы один раз был получен новый статус в [запросе обновления статусов водителей](#).

Все элементы XML-документа обязательно присутствуют в запросе, если не указано иначе.

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Orderid>trui323f323r2332r</Orderid>
  <Tariffs>
    <Tariff>t000001</Tariff>
    ...
  </Tariffs>
  <Cars>
    <Car>
      <Uuid>v990</Uuid>
      <Dist>8</Dist>
      <Time>25</Time>
      <Tariff>t000001</Tariff>
      <MapHref>http://static-maps.yandex.ru/1.x/?lang=ru-RU&key=<API-ключ
Ян-
декс.Карт>&key=<API-ключ>&key=<API-ключ>&key=<API-ключ>&key=<API-ключ>&key=<API-ключ>
1%2Cpm2bm</MapHref>
    </Car>
    ...
  </Cars>
  <Recipient blacklisted="no" loyal="yes" />
  <Source>
    <FullName>Россия, Москва, улица Тверская, 7, подъезд 4</FullName>
    <ShortName>Тверская, 7</ShortName>
    <ClosestStation>Парк культуры</ClosestStation>
    <Point>
      <Lon>55.75847</Lon>
      <Lat>37.61195</Lat>
    </Point>
    <Airport>
```

```

    <Flight>SU-119</Flight>
    <Terminal>B</Terminal>
  </Airport>
  <Country>
    <CountryName>Россия</CountryName>
    <Locality>
      <LocalityName>Москва</LocalityName>
      <Thoroughfare>
        <ThoroughfareName>улица Тверская</ThoroughfareName>
        <Premise>
          <PremiseNumber>7г</PremiseNumber>
        </Premise>
      </Thoroughfare>
    </Locality>
  </Country>
</Source>
<Destinations>
  <Destination order="1">
    <FullName>Россия, Москва, улица Моховая, 7, подъезд 2</FullName>
    <ShortName>Моховая, 7</ShortName>
    <Point>
      <Lon>55.75447</Lon>
      <Lat>37.61295</Lat>
    </Point>
    <Country>
      <CountryName>Россия</CountryName>
      <Locality>
        <LocalityName>Москва</LocalityName>
        <Thoroughfare>
          <ThoroughfareName>улица Моховая</ThoroughfareName>
          <Premise>
            <PremiseNumber>7, корпус 3</PremiseNumber>
          </Premise>
        </Thoroughfare>
      </Locality>
    </Country>
  </Destination>
  ...
</Destinations>
<BookingTime type="notlater">2012-02-13T13:01</BookingTime>
<Requirements>
  <Require name="has_conditioner">yes</Require>
  <Require name="no_smoking">no</Require>
  <Require name="child_chair">4</Require>
  <Require name="animal_transport">yes</Require>
  <Require name="universal">no</Require>
  <Require name="check">no</Require>
  <Require name="yamoney">yes</Require>
  <Require name="coupon">300</Require>
</Requirements>
<Comments>Чтобы водитель не слушал шансон.</Comments>
</Request>

```

Ниже описаны элементы, которые могут быть включены в тело запроса с информацией о статусах водителей.

<Request>

Корневой элемент описания заказа.

<Orderid>

```
<Orderid>trui323f323r2332r</Orderid>
```

Уникальный идентификатор заказа, формируется сервисом Яндекс.Такси.

<Tariffs>

Список тарифов, по которым заказ может быть принят.

<Tariff>

```
<Tariff>t000001</Tariff>
```

Идентификатор одного из тарифов, по которому заказ может быть принят.

<Cars>

Список водителей, которых сервис Яндекс.Такси подобрал для срочного заказа. Необязательный элемент.

Элемент не задается для несрочных заказов: Служба Такси должна подобрать водителя самостоятельно.

<Car>

Информация об одном из подобранных водителей.

<Uuid>

```
<Uuid>v990</Uuid>
```

Уникальный идентификатор водителя в рамках Службы Такси.

<Dist>

```
<Dist>8</Dist>
```

Расстояние от водителя до клиента по кратчайшему маршруту в километрах.

<Time>

```
<Time>25</Time>
```

Время следования по кратчайшему маршруту с учетом пробок, в минутах (целое число).

<Tariff>

```
<Tariff>t000001</Tariff>
```

Тариф, по которому данному водителю предлагается принять запрос клиента.

<MapHref>

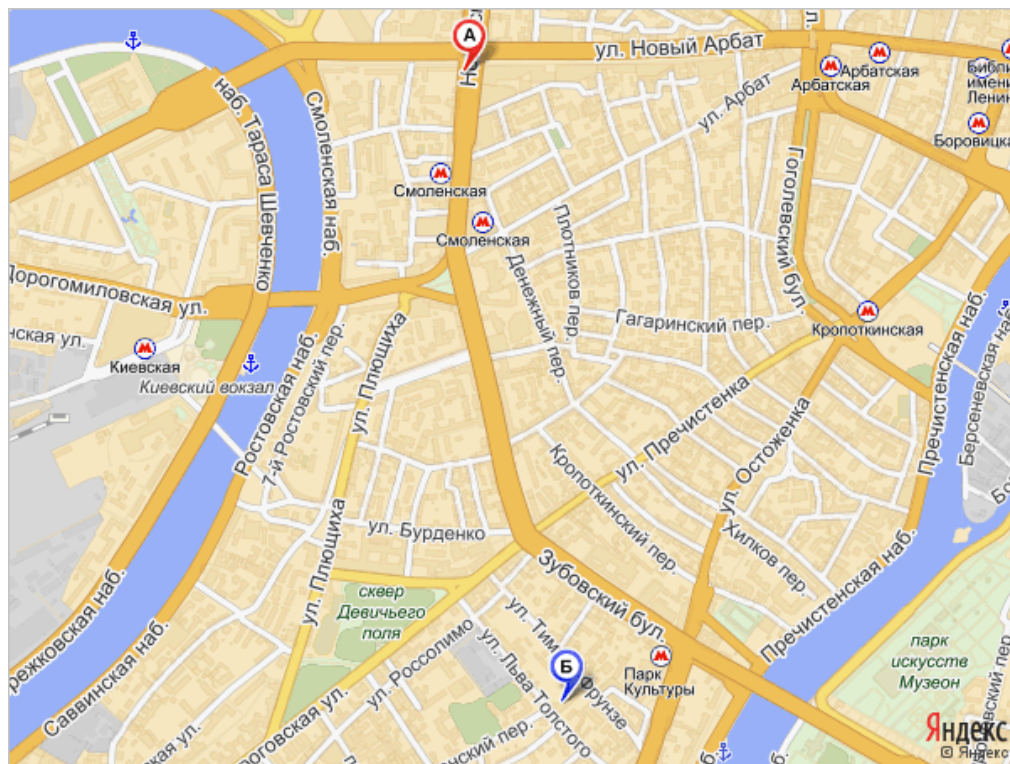
```
<MapHref>http://static-maps.yandex.ru/1.x/?lang=ru-RU&key=<API-ключ
Яндекс.Карт>&l=map&pt=37.582962%2C55.751961%2Cpm2am~37.587937%2C
55.733771%2Cpm2bm</MapHref>
```

Ссылка на статическую карту, на которой отмечено положение водителя и место подачи машины. С помощью карты водитель может быстро оценить расстояние до места подачи и решить, стоит ли принимать заказ.

Примечание:

Символ «&» в ссылке заменен на HTML-код символа, «&», чтобы значение элемента не противоречило спецификации XML.

Пример карты:



<Recipient>

```
<Recipient blacklisted="no" loyal="yes" />
```

Характеристика клиента.

Атрибуты:

- **blacklisted** — признак недобросовестности клиента, определяется по черному списку, формируемому из [черных списков Служб Такси](#).

Возможные значения:

- «yes» — клиент внесен в черный список;
- «no» — клиента нет в черном списке.
- **loyal** — признак постоянного клиента, устанавливается сервисом Яндекс.Такси. Постоянным клиентом считается пользователь, который разместил на сервисе как минимум 10 заказов, из которых как минимум 90% были успешно выполнены.

Возможные значения:

- «yes» — постоянный клиент;
- «no» — пользователь не считается постоянным клиентом.

Атрибуты **blacklisted** и **loyal** могут иметь значение «yes» одновременно: пользователь, попавший в черный список одной Службы Такси, может остаться лояльным Яндекс.Такси в целом.

<Source>

Описание места подачи машины.

<FullName>

```
<FullName>Россия, Москва, улица Тверская, 7, подъезд 4</FullName>
```

Полный адрес места подачи на русском языке, от страны до номера дома. Номер подъезда включается в адрес, если клиент указал подъезд при заказе такси.

<ShortName>

```
<ShortName>Тверская, 7</ShortName>
```

Короткий адрес места подачи машины (улица и номер дома).

<ClosestStation>

```
<ClosestStation>Парк культуры</ClosestStation>
```

Станция метро, ближайшая к месту подачи машины. Элемент отсутствует, если ближайшую станцию метро не удалось определить.

<Point>

Географические координаты места подачи машины. Необязательный элемент.

<Lon>

```
<Lon>55.75847</Lon>
```

Долгота места подачи, дробное число.

<Lat>

```
<Lat>37.61195</Lat>
```

Широта места подачи, дробное число.

<Airport>

Информация о встрече клиента в аэропорте. Необязательный элемент.

<Flight>

```
<Flight>SU-119</Flight>
```

Номер рейса.

<Terminal>

```
<Terminal>B</Terminal>
```

Идентификатор терминала аэропорта.

<Country>

Адрес места подачи, разбитый на элементы. Соответствует адресу, указанному в элементе [<FullName>](#).

<CountryName>

```
<CountryName>Россия</CountryName>
```

Название страны на русском языке.

<Locality>

Населенный пункт.

<LocalityName>

```
<LocalityName>Москва</LocalityName>
```

Название населенного пункта на русском языке.

<Thoroughfare>

Улица (бульвар, проезд и т. п.).

<ThoroughfareName>

```
<ThoroughfareName>улица Тверская</ThoroughfareName>
```

Название улицы на русском языке.

<Premise>

Номер дома (строения, корпуса и т. п.)

<PremiseNumber>

```
<PremiseNumber>7г</PremiseNumber>
```

Номер дома в свободной форме.

<Destinations>

Адреса всех промежуточных остановок и пункта назначения. Необязательный элемент.

<Destination>

```
<Destination order="1">
```

Описание одного места назначения.

Атрибут:

- `order` — порядковый номер остановки в маршруте.

<FullName>

```
<FullName>Россия, Москва, улица Моховая, 7, подъезд 2</FullName>
```

Полный адрес места назначения на русском языке, от страны до номера дома. Номер подъезда включается в адрес, если клиент указал подъезд при заказе такси.

<ShortName>

```
<ShortName>Моховая, 7</ShortName>
```

Короткий адрес места назначения (улица и номер дома).

<Point>

Географические координаты места назначения.

<Lon>

```
<Lon>55.75447</Lon>
```

Долгота места назначения, дробное число.

<Lat>

```
<Lat>37.61295</Lat>
```

Широта места назначения, дробное число.

<Country>

Адрес места назначения, разбитый на элементы. Соответствует адресу, указанному в элементе

[<FullName>](#).

<CountryName>

```
<CountryName>Россия</CountryName>
```

Название страны на русском языке.

<Locality>

Населенный пункт.

<LocalityName>

```
<LocalityName>Москва</LocalityName>
```

Название населенного пункта на русском языке.

<Thoroughfare>

Улица (бульвар, проезд и т.п.)

<ThoroughfareName>

```
<ThoroughfareName>улица Моховая</ThoroughfareName>
```

Название улицы на русском языке.

<Premise>

Номер дома (строения, корпуса и т. п.)

<PremiseNumber>

```
<PremiseNumber>7г</PremiseNumber>
```

Номер дома в свободной форме.

<BookingTime>

```
<BookingTime type="notlater">2012-02-13T13:01</BookingTime>
```

Время подачи такси.

Атрибут:

- `type` — указывает на вид заказа, срочный или несрочный.
 - «notlater» — срочный заказ (до подачи машины меньше 25 минут).
 - «exact» — несрочный заказ (до подачи машины больше 25 минут).

<Requirements>

Дополнительные требования к машине. Необязательный элемент.

<Require>

```
<Require name="child_chair">0.11</Require>
<Require name="card">yes</Require>
```

Требование к машине. Для каждой характеристики указывается отдельный элемент `<Require>`. Значение элемента интерпретируется в зависимости от значения атрибута `name` (атрибут содержит идентификатор характеристики):

- `name="animal_transport"`: перевозка животных.
 - «yes» — в машине должно быть можно перевозить животных.
- `name="check"`: строгая отчетность при оплате.
 - «yes» — водитель должен выписывать квитанцию об оплате заказа (документ строгой отчетности).
- `name="child_chair"`: детское кресло.
 - `<возраст ребенка>` — в машине должно быть детское кресло для ребенка указанного возраста. Возраст указывается в виде промежутка между минимальным и максимальным значением, например: «0.11-4».

Возраст меньше года указывается в формате «0.<количество месяцев>» (11 месяцев следует записать как «0.11»). Возраст больше года указывается как целое число лет.
- `name="has_conditioner"`: автомобильный кондиционер.
 - «yes» — в машине должен быть кондиционер.
- `name="no_smoking"`: разрешено ли курение.
 - «yes» — в машине должно быть запрещено курение;
 - «no» — курение в машине должно быть разрешено.
- `name="universal"`: кузов «универсал».
 - «yes» — машина должна обладать кузовом «универсал».
- `name="yamoney"`: оплата Яндекс.Деньгами.
 - «yes» — водитель должен принимать к оплате Яндекс.Деньги.
- `name="coupon"`: водитель должен будет принять купон сервиса Яндекс.Такси в качестве полной или частичной оплаты заказа.

Допустим, цена заказа составляет 900 рублей, и клиент использует купон на 300 рублей. При успешном завершении заказа (статус [complete](#)):

1. Водитель взимает с клиента разницу между ценой поездки и номиналом купона — 600 рублей.
2. Купон погашается и не может быть использован повторно.
3. Сервис Яндекс.Такси возмещает Службе Такси номинал купона — 300 рублей.

Если заказ не был успешно выполнен (не получил статус [complete](#)), купон к оплате принимать нельзя.

<Comments>

```
<Comments>Чтобы водитель не слушал шансон.</Comments>
```

Комментарии к заказу. Необязательный элемент.

Ответ

Ответ сервера Службы Такси на предложение срочного заказа не должен содержать тела. Результат обработки запроса описывается кодом HTTP-ответа.

Особым значением для ответа на предложение срочного заказа обладают следующие коды:

- 203 Non-Authoritative Information — возвращается в ответ на запрос, перечисляющий нескольких водителей, если только некоторые из указанных водителей присутствуют в базе данных Службы Такси.
- 404 Not Found — возвращается, если ни один указанный в запросе водитель не был найден в базе данных Службы Такси.

Остальные коды HTTP-ответа несут тот же смысл, что и для других запросов:

- 200 OK — всегда обозначает, что запрос выполнен полностью без ошибок;
- 400 Bad Request — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно, либо XML-документ в теле запроса сформирован неправильно.
- 403 Forbidden — в ответах сервера Яндекс.Такси всегда обозначает, что передана неправильная пара «идентификатор Службы Такси / API-ключ».
- 500 Internal Server Error — всегда обозначает, что возникла внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

Закрепление заказа

Яндекс.Такси присылает запрос, закрепляющий заказ за водителем Службы Такси. Чтобы подтвердить закрепление, Служба Такси должна ответить HTTP-кодом 200 (все варианты ответа описаны в разделе [Ответ](#)).

После закрепления заказа сервер Службы Такси должен отправлять Яндекс.Такси [обновления статуса заказа](#).

Запрос

Формат URL запроса

```
<адрес, предоставленный Службой Такси>/1.x/setcar
```

Тело запроса

Ниже приведен формат тела запроса, закрепляющего заказ за водителем, указанным в [сообщении о готовности](#).

Все элементы XML-документа обязательны, если не указано иначе.


```

<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <OrderId>trui323f323r2332r</OrderId>
  <Cars>
    <Car>
      <Uuid>v990</Uuid>
      <Tariff>t000001</Tariff>
      <MapHref>http://static-maps.yandex.ru/1.x/?lang=ru-RU&key=<API-ключ
Яндекс.Карт>&pl=<описание маршрута в виде координат>&l=map&pt=37.57349%2C55.729378%2Cpm2am~37.587937%2C55.733771%2Cpm2bm</MapHref>
    </Car>
  </Cars>
  <ContactInfo>
    <Phones>
      <Phone>+79022617513</Phone>
    </Phones>
    <Name>Васян</Name>
  </ContactInfo>
  <Source>
    <FullName>Россия, Москва, улица Тверская, 7, подъезд 4</FullName>
    <ShortName>Тверская, 7</ShortName>
    <Point>
      <Lon>55.75847</Lon>
      <Lat>37.61195</Lat>
    </Point>
    <Airport>
      <Flight>SU-119</Flight>
      <Terminal>B</Terminal>
    </Airport>
    <Country>
      <CountryName>Россия</CountryName>
      <Locality>
        <LocalityName>Москва</LocalityName>
        <Thoroughfare>
          <ThoroughfareName>улица Тверская</ThoroughfareName>
          <Premise>
            <PremiseNumber>7г</PremiseNumber>
            <PorchNumber>третий от шоссе</PorchNumber>
          </Premise>
        </Thoroughfare>
      </Locality>
    </Country>
  </Source>
  <Destinations>
    <Destination order="1">
      <FullName>Россия, Москва, улица Моховая, 7, подъезд 2</FullName>
      <ShortName>Моховая, 7</ShortName>
      <Point>
        <Lon>55.75447</Lon>
        <Lat>37.61295</Lat>
      </Point>
      <Country>
        <CountryName>Россия</CountryName>
        <Locality>
          <LocalityName>Москва</LocalityName>
          <Thoroughfare>
            <ThoroughfareName>улица Моховая</ThoroughfareName>
            <Premise>
              <PremiseNumber>7, корпус 3</PremiseNumber>
            </Premise>
          </Thoroughfare>
        </Locality>
      </Country>
    </Destination>
    ...
  </Destinations>
  <BookingTime type="exact">2012-02-13T13:01</BookingTime>

```

```

<Requirements>
  <Require name="has_conditioner">yes</Require>
  <Require name="no_smoking">no</Require>
  <Require name="child_chair">4</Require>
  <Require name="animal_transport">yes</Require>
  <Require name="universal">no</Require>
  <Require name="check">no</Require>
  <Require name="yamoney">yes</Require>
  <Require name="coupon">300</Requirement>
</Requirements>
<Comments>Чтобы водитель не слушал шансон.</Comments>
</Request>

```

Ниже описаны элементы, которые могут быть включены в тело запроса с информацией о статусах водителей.

<Request>

Корневой элемент описания заказа.

<Orderid>

```
<Orderid>trui323f323r2332r</Orderid>
```

Уникальный идентификатор заказа, формируется сервисом Яндекс.Такси.

<Cars>

Список водителей, которых сервис Яндекс.Такси подобрал для заказа.

<Car>

Описание одного из подобранных водителей.

<Uuid>

```
<Uuid>v990</Uuid>
```

Уникальный идентификатор водителя.

<Tariff>

```
<Tariff>t000001</Tariff>
```

Тариф, по которому данному водителю предлагается принять запрос клиента.

<MapHref>

```

<MapHref>http://static-maps.yandex.ru/1.x/?lang=ru-RU&key=<API-ключ
Яндекс.Карт>&pl=<описание маршрута в виде
координат>&l=map&pt=37.57349%2C55.729378%2Cpm2am~37.587937%2C55.
733771%2Cpm2bm</MapHref>

```

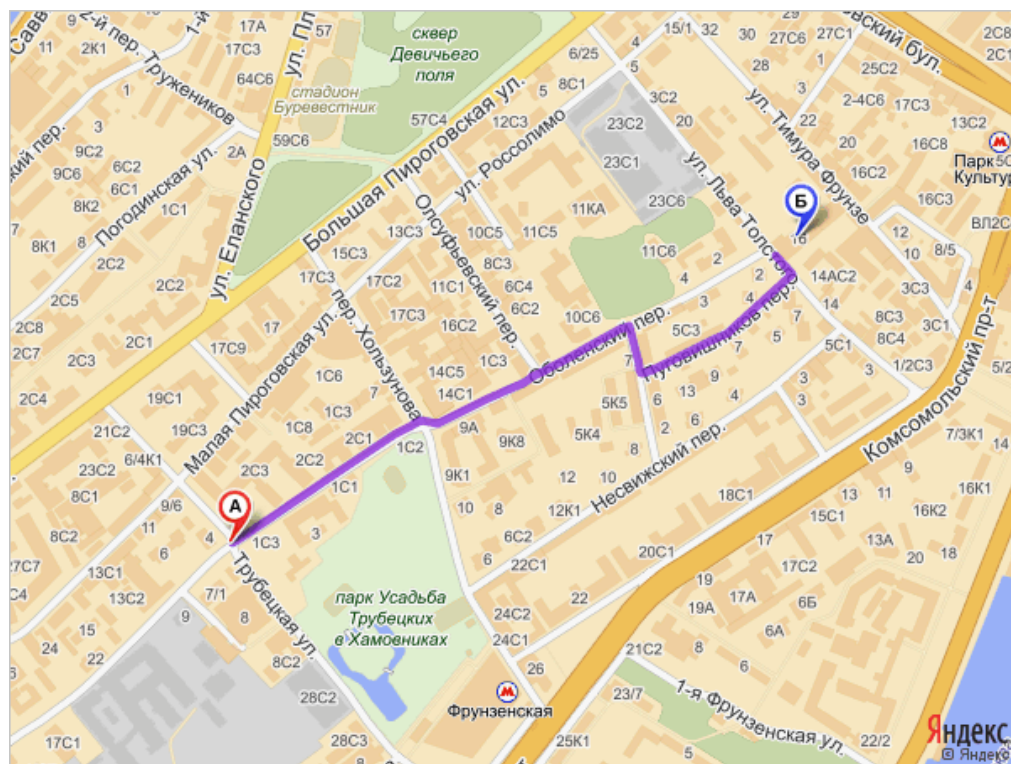
Ссылка на статическую карту, на которой проложен маршрут от текущего положения водителя до места подачи машины. Необязательный элемент.

Примечание:

Символ «&» в ссылке заменен на HTML-код символа, «&», чтобы значение элемента не противоречило спецификации XML.

Ссылка на карту отсутствует, если маршрут проложить не удалось.

Пример карты:



<ContactInfo>

Контактная информация клиента.

<Phones>

Список контактных телефонов клиента.

<Phone>

```
<Phone>+79022617513</Phone>
```

Номер телефона в формате +7XXXXXXXXXX.

<Name>

```
<Name>Васян</Name>
```

Имя в том виде, в котором его предоставил клиент. Необязательный элемент.

<Source>

Описание места подачи машины.

<FullName>

```
<FullName>Россия, Москва, улица Тверская, 7, подъезд 4</FullName>
```

Полный адрес места подачи на русском языке, от страны до номера дома. Номер подъезда включается в адрес, если клиент указал подъезд при заказе такси.

<ShortName>

```
<ShortName>Тверская, 7</ShortName>
```

Короткий адрес места подачи машины (улица и номер дома).

<Point>

Географические координаты места подачи машины. Необязательный элемент.

<Lon>

```
<Lon>55.75847</Lon>
```

Долгота места подачи, дробное число.

<Lat>

```
<Lat>37.61195</Lat>
```

Широта места подачи, дробное число.

<Airport>

Информация о встрече клиента в аэропорте. Необязательный элемент.

<Flight>

```
<Flight>SU-119</Flight>
```

Номер рейса.

<Terminal>

```
<Terminal>B</Terminal>
```

Идентификатор терминала аэропорта.

<Country>

Адрес места подачи, разбитый на элементы. Соответствует адресу, указанному в элементе [<FullName>](#).

<CountryName>

```
<CountryName>Россия</CountryName>
```

Название страны на русском языке.

<Locality>

Населенный пункт.

<LocalityName>

```
<LocalityName>Москва</LocalityName>
```

Название населенного пункта на русском языке.

<Thoroughfare>

Улица (бульвар, проезд и т. п.).

<ThoroughfareName>

```
<ThoroughfareName>улица Тверская</ThoroughfareName>
```

Название улицы на русском языке.

<Premise>

Номер дома (строения, корпуса и т. п.)

<PremiseNumber>

```
<PremiseNumber>7г</PremiseNumber>
```

Номер дома в свободной форме.

PorchNumber

```
<PorchNumber>третий от шоссе</PorchNumber>
```

Номер подъезда в свободной форме. Необязательный элемент.

<Destinations>

Адреса всех промежуточных остановок и пункта назначения. Необязательный элемент.

<Destination>

```
<Destination order="1">
```

Описание одного места назначения.

Атрибут:

- order — порядковый номер остановки в маршруте.

<FullName>

```
<FullName>Россия, Москва, улица Моховая, 7, подъезд 2</FullName>
```

Полный адрес места назначения на русском языке, от страны до номера дома. Номер подъезда включается в адрес, если клиент указал подъезд при заказе такси.

<ShortName>

```
<ShortName>Моховая, 7</ShortName>
```

Короткий адрес места назначения (улица и номер дома).

<Point>

Географические координаты места назначения.

<Lon>

```
<Lon>55.75447</Lon>
```

Долгота места назначения, дробное число.

<Lat>

```
<Lat>37.61295</Lat>
```

Широта места назначения, дробное число.

<Country>

Адрес места назначения, разбитый на элементы. Соответствует адресу, указанному в элементе **<FullName>**.

<CountryName>

```
<CountryName>Россия</CountryName>
```

Название страны на русском языке.

<Locality>

Населенный пункт.

<LocalityName>

```
<LocalityName>Москва</LocalityName>
```

Название населенного пункта на русском языке.

<Thoroughfare>

Улица (бульвар, проезд и т.п.)

<ThoroughfareName>

```
<ThoroughfareName>улица Моховая</ThoroughfareName>
```

Название улицы на русском языке.

<Premise>

Номер дома (строения, корпуса и т. п.)

<PremiseNumber>

```
<PremiseNumber>7г</PremiseNumber>
```

Номер дома в свободной форме.

<BookingTime>

```
<BookingTime type="notlater">2012-02-13T13:01</BookingTime>
```

Время подачи такси.

Атрибут:

- `type` — указывает на вид заказа, срочный или несрочный.
 - «notlater» — срочный заказ (до подачи машины меньше 25 минут).
 - «exact» — несрочный заказ (до подачи машины больше 25 минут).

<Requirements>

Дополнительные требования к машине. Необязательный элемент.

<Require>

```
<Require name="child_chair">0.11</Require>
<Require name="card">yes</Require>
```

Характеристика машины. Каждая характеристика описывается ровно одним элементом `<Require>`. Значение элемента интерпретируется в зависимости от значения атрибута `name` (атрибут содержит идентификатор характеристики):

- `name="animal_transport"`: перевозка животных.
 - «yes» — в машине можно перевозить животных;
 - «no» — животных перевозить нельзя.
- `name="check"`: строгая отчетность при оплате.
 - «yes» — водитель может выписать квитанцию об оплате заказа (документ строгой отчетности);
 - «no» — водитель не может выписать квитанцию.
- `name="child_chair"`: детское кресло.
 - `<возраст ребенка>` — детское кресло есть, предназначено для детей указанного возраста. Возраст указывается в виде промежутка между минимальным и максимальным значением, например: «0.11-4».
 - Возраст меньше года указывается в формате «0.<количество месяцев>» (11 месяцев следует записать как «0.11»). Возраст больше года указывается как целое число лет.
 - «no» — детского кресла нет.
- `name="has_conditioner"`: автомобильный кондиционер.
 - «yes» — в машине есть кондиционер;
 - «no» — кондиционера нет.
- `name="no_smoking"`: разрешено ли курение.
 - «yes» — водитель не курит, пассажирам курить запрещено;
 - «no» — пассажирам курить разрешено.
- `name="universal"`: кузов «универсал».
 - «yes» — машина с кузовом «универсал»;
 - «no» — машина с другим кузовом.
- `name="yamoney"`: оплата Яндекс.Деньгами.
 - «yes» — водитель должен принимать к оплате Яндекс.Деньги.
- `name="coupon"`: водитель должен будет принять купон сервиса Яндекс.Такси в качестве полной или частичной оплаты заказа.

Допустим, цена заказа составляет 900 рублей, и клиент использует купон на 300 рублей. При успешном завершении заказа (статус [complete](#)):

1. Водитель взымает с клиента разницу между ценой поездки и номиналом купона — 600 рублей.
2. Купон погашается и не может быть использован повторно.
3. Сервис Яндекс.Такси возмещает Службе Такси номинал купона — 300 рублей.

Если заказ не был успешно выполнен (не получил статус [complete](#)), купон к оплате принимать нельзя.

`<Comments>`

```
<Comments>Чтобы водитель не слушал шансон.</Comments>
```

Комментарии к заказу. Необязательный элемент.

Ответ

Ответ сервера Службы Такси на закрепление заказа не должен содержать тела. Результат обработки запроса описывается кодом HTTP-ответа.

Особым значением для ответа на закрепление срочного заказа обладают следующие коды:

- 404 Not Found — указанный водитель не найден в базе данных Службы Такси. В этом случае заказ будет закреплен за другим водителем.
- 410 Gone — указанный водитель Службы Такси не может принять заказ. В этом случае заказ будет передан другому водителю.

Остальные коды HTTP-ответа несут тот же смысл, что и для других запросов:

- 200 OK — всегда обозначает, что запрос выполнен полностью без ошибок;
- 400 Bad Request — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно, либо XML-документ в теле запроса сформирован неправильно.
- 403 Forbidden — в ответах сервера Яндекс.Такси всегда обозначает, что передана неправильная пара «идентификатор Службы Такси / API-ключ».
- 500 Internal Server Error — всегда обозначает, что возникла внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

Сообщение о готовности водителя выполнить заказ

Сервер Службы Такси отправляет сервису Яндекс.Такси сообщение о готовности водителя в ответ на [запрос с предложением заказа](#). Сервис обрабатывает запрос и возвращает результат обработки в коде HTTP-ответа.

Если сервис Яндекс.Такси готов закрепить водителя за заказом, Службе Такси отправляется [запрос закрепления](#). Если заказ отдается другой Службе Такси, сервис сообщит об [отмене заказа](#).

Запрос

URL запроса

```
https://ymsh.mobile.yandex.net/1.x/carack ?
clid=<идентификатор Службы Такси>
& apikey=<API-ключ Службы Такси>
& uuid=<идентификатор водителя>
& orderid=<идентификатор заказа>
& [cost=<цена поездки>]
```

Параметр	Значение	Описание
Обязательные параметры		
clid	<идентификатор Службы Такси>	Идентификатор Службы Такси, полученный при регистрации в сервисе Яндекс.Такси.
apikey	<API-ключ>	API-ключ Службы Такси, полученный при регистрации в сервисе Яндекс.Такси.
uuid	<идентификатор водителя>	Идентификатор водителя, назначенный Службой Такси. Должен представлять собой буквенно-цифровую последовательность не длиннее 32 символов.
orderid	<идентификатор заказа>	Идентификатор подтверждаемого заказа (передается в теле запроса с предложением срочного заказа).

Параметр	Значение	Описание
		Представляет собой буквенно-цифровую последовательность не длиннее 32 символов.
Дополнительные параметры		
cost	<цена поездки>	Цена поездки по маршруту, указанному в заказе. Может использоваться только для тех городов, где поддерживается предварительный расчет цены. Значение должно быть задано целым числом, в валюте города, в котором был принят заказ.

Тело запроса

Тело запроса отсутствует.

Ответ

Ответ Яндекс.Такси на сообщение о готовности водителя не содержит тела. Результат обработки запроса описывается кодом HTTP-ответа.

Особым значением для ответа на подтверждение заказа обладают следующие коды:

- 404 Not Found — возвращается, если указанный в запросе водитель или заказ не найден в базе данных Яндекс.Такси.

Также код ответа 404 возвращается, если подобранный Службой Такси водитель не отвечает требованиям заказа (например, предлагается машина эконом-класса несмотря на то, что в заказе указан класс «Комфорт»).

- 410 Gone — возвращается, если указанный заказ уже закреплен за другим водителем.

Внимание!

Получив такой код ответа, Служба Такси должна отменить заказ, не дожидаясь [отдельной команды](#) от сервера Яндекс.Такси.

Остальные коды HTTP-ответа несут тот же смысл, что и для других запросов:

- 200 OK — всегда обозначает, что запрос выполнен полностью без ошибок;
- 400 Bad Request — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно, либо XML-документ в теле запроса сформирован неправильно.
- 403 Forbidden — в ответах сервера Яндекс.Такси всегда обозначает, что передана неправильная пара «идентификатор Службы Такси / API-ключ».
- 500 Internal Server Error — всегда обозначает, что возникла внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

Обновление статуса заказа

Сервер Службы Такси отправляет обновления статуса заказа сервису Яндекс.Такси по мере того, как водитель выполняет закрепленный за ним заказ.

Обновления статуса Служба Такси должна отправлять для всех исполняемых заказов.

Запрос

URL запроса

```
https://ymsh.mobile.yandex.net/1.x/requestconfirm ?
clid=<идентификатор Службы Такси>
& apikey=<API-ключ Службы Такси>
& orderid=<идентификатор заказа>
& status=driving | waiting | transporting | complete | cancelled | failed
& [extra=<дополнительная информация о статусе заказа>]
& [newcar=<идентификатор водителя>]
```

Параметр	Значение	Описание
Обязательные параметры		
clid	<идентификатор Службы Такси>	Идентификатор Службы Такси, полученный при регистрации в сервисе Яндекс.Такси.
apikey	<API-ключ>	API-ключ Службы Такси, полученный при регистрации в сервисе Яндекс.Такси.
orderid	<идентификатор заказа>	Идентификатор заказа, статус которого обновляется (передается в теле запроса с предложением заказа). Представляет собой буквенно-цифровую последовательность не длиннее 32 символов.
status	<driving waiting transporting complete cancelled failed>	Новый статус заказа. Сервер Службы Такси должен передавать статусы только для тех заказов, которые были закреплены за водителями Службы Такси. Возможные значения: <ol style="list-style-type: none"> «driving» — водитель выехал к клиенту. «waiting» — водитель прибыл на место назначения, ожидает клиента. «transporting» — водитель везет клиента к месту назначения. «complete» — заказ выполнен. «cancelled» — заказ отменен клиентом. «failed» — водитель не смог выполнить заказ. Статус заказа может изменяться только в указанном порядке. То есть, для заказа, находящегося в статусе «transporting», нельзя передать статус «waiting». Исключениями из правила являются статусы «cancelled» и «failed», которые могут быть переданы в любой момент.
Дополнительные параметры		
extra	<дополнительная информация о статусе заказа>	Дополнительная информация, значение которой интерпретируется в зависимости от указанного статуса заказа, передаваемого с помощью параметра status: <ul style="list-style-type: none"> «complete» — в параметре следует указать общую цену поездки. «failed» и «cancelled» — в параметре следует указать причину невыполнения заказа. «driving» — в параметре следует указать идентификатор водителя, выполняющего заказ. В остальных случаях параметр extra игнорируется.
newcar	<идентификатор водителя>	Идентификатор нового водителя, назначенного на заказ. Указывается, если по каким-то причинам Служба Такси заменяет водителя, который был закреплен за заказом ранее. Новый водитель должен присутствовать в базе Яндекс.Такси.

Параметр	Значение	Описание
		Назначая нового водителя, статус заказа можно выставить любым — при наличии параметра <code>newcar</code> хронологические ограничения не накладываются.

Тело запроса

Тело запроса отсутствует.

Ответ

Ответ Яндекс.Такси на обновление статуса заказа не содержит тела. Результат обработки запроса описывается кодом HTTP-ответа.

Особым значением для ответа на обновление статуса заказа обладает следующий код:

- 404 Not Found — возвращается, если в базе данных Яндекс.Такси не найден указанный в запросе заказ или водитель.

Остальные коды HTTP-ответа несут тот же смысл, что и для других запросов:

- 200 OK — всегда обозначает, что запрос выполнен полностью без ошибок;
- 400 Bad Request — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно, либо XML-документ в теле запроса сформирован неправильно.
- 403 Forbidden — в ответах сервера Яндекс.Такси всегда обозначает, что передана неправильная пара «идентификатор Службы Такси / API-ключ».
- 500 Internal Server Error — всегда обозначает, что возникла внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

Отмена заказа

Сервис Яндекс.Такси отменяет заказ, отсылая запрос на сервер Службы Такси.

Служба Такси должна ответить сервису следуя описанному ниже формату, а также сообщить об отмене заказа всем своим водителям.

Яндекс.Такси может отменить заказ только до подачи машины (в период между закреплением заказа за водителем и [выставлением статуса заказа](#) «waiting»). Если Яндекс.Такси присылает запрос об отмене заказа после подачи машины, Служба Такси должна вернуть HTTP-ответ с кодом 400 Bad Request.

Запрос

Формат URL запроса

```
https://<адрес, предоставленный Службой Такси>/1.x/cancelrequest ?
orderid=<идентификатор заказа>
& reason=user | assigned | timeout
```

Параметр	Значение	Описание
Обязательные параметры		
orderid	<идентификатор заказа>	Идентификатор отменяемого заказа, назначенный Яндекс.Такси. Представляет собой буквенно-цифровую последовательность не длиннее 32 символов.

Параметр	Значение	Описание
reason	<user assigned timeout>	<p>Причина, по которой заказ отменяется. Возможные значения:</p> <ul style="list-style-type: none"> «user» — отозван клиентом; «assigned» — назначен на водителя другой Службы Такси; «timeout» — ни один водитель не сообщил о готовности выполнить заказ в течение отведенного времени. <p>Отведенное время (порядка 15 минут) устанавливается и регулируется сервисом Яндекс.Такси.</p>

Тело запроса

Тело запроса отсутствует.

Ответ

Ответ сервера Службы Такси на отмену заказа не должен содержать тела. Результат обработки описывается кодом HTTP-ответа.

Особым значением для ответа на отзыв заказа обладает следующий код:

- 404 Not Found — указанный в запросе заказ не найден в базе данных Службы Такси;

Остальные коды HTTP-ответа несут тот же смысл, что и для других запросов:

- 200 OK — всегда обозначает, что запрос выполнен полностью без ошибок;
- 400 Bad Request — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно, либо XML-документ в теле запроса сформирован неправильно.
- 403 Forbidden — в ответах сервера Яндекс.Такси всегда обозначает, что передана неправильная пара «идентификатор Службы Такси / API-ключ».
- 500 Internal Server Error — всегда обозначает, что возникла внутренняя ошибка сервера. Запрос при этом не выполнен и не повлек никаких изменений в статусе заказа.

Отзывы о заказах

Сервис Яндекс.Такси возвращает отзывы клиентов о заказах, сделанных за последние сутки. По запросу Службы Такси возвращаются отзывы о тех заказах, которые были переданы в данную Службу.

Запрос рекомендуется производить не чаще одного раза в час.

Запрос

Формат URL запроса

```
https://ymsh.mobile.yandex.net/1.x/feedbacks ?
clid=<идентификатор Службы Такси>
& apikey=<API-ключ Службы Такси>
```

Параметр	Значение	Описание
Обязательные параметры		
clid	<идентификатор Службы Такси>	Идентификатор Службы Такси, полученный при регистрации в сервисе Яндекс.Такси.
apikey	<API-ключ>	API-ключ Службы Такси, полученный при регистрации в сервисе Яндекс.Такси.

Тело запроса

Тело запроса отсутствует.

Ответ

Тело ответа содержит список отзывов. Каждый отзыв содержит оценку, комментарий или и то, и другое.

Ниже приведен формат ответа на запрос отзывов. Элементы и их атрибуты приведены с примерами значений.

Многоточие после закрывающего тега XML-элемента означает, что ответ может содержать несколько таких элементов, следующих один за другим.

Все элементы XML-документа обязательны, если не указано иначе.

```
<?xml version="1.0" encoding="UTF-8"?>
<Feedbacks>
  <Feedback orderid="trui323f323r2332r" rating="4">Прокатились с ветерком.</
Feedback>
  ...
</Feedbacks>
```

Ниже описаны элементы, которые могут быть включены в тело ответа на запрос отзывов. Для элементов, содержащих значения, приведены примеры.

<Feedbacks>

Корневой элемент списка отзывов.

<Feedback>

Элемент, описывающий отдельный отзыв. Значение элемента — комментарий клиента к заказу, может быть пустым, если элемент содержит атрибут `rating`.

Атрибуты элемента:

orderid

Идентификатор заказа, к которому относится отзыв. (О каждом заказе может быть оставлен только один отзыв.)

rating

Оценка выполнения заказа по 5-балльной шкале. Может отсутствовать, если в значении элемента `<Feedback>` содержится непустой комментарий.

Запрос стоимости заказа

Сервис Яндекс.Такси запрашивает у Службы Такси стоимость предполагаемой поездки по каждому из имеющихся тарифов.

Запрос

Формат URL запроса

```
<адрес, предоставленный Службой Такси>/1.x/cost ?
time=<время начала поездки>
& route=<координаты точек маршрута>
```

Параметр	Значение	Описание
Обязательные параметры		
time	<время начала поездки>	Время начала поездки указывается в формате ISO-8601, с указанием часового пояса. Пример значения: 2012-01-01T13:00:00+04
route	<координаты точек маршрута>	В значении параметра перечисляются точки маршрута, от места подачи машины, до места назначения. Координаты точек указываются как долгота и широта каждой из них, разделенные тильдой: ~.

Параметр	Значение	Описание
		Пример значения с двумя точками маршрута: 37.588628, 55.734046~37.688628, 54.734046

Тело запроса

Тело запроса отсутствует.

Ответ

В теле ответа следует передать стоимость поездки для каждого [тарифа](#), используемого Службой Такси, учитывая требуемое время подачи машины.

Ниже приведен формат ответа на запрос отзывов. Элементы и их атрибуты приведены с примерами значений.

Многоточие после закрывающего тега XML-элемента означает, что ответ может содержать несколько таких элементов, следующих один за другим.

Все элементы XML-документа обязательны, если не указано иначе.

```
<?xml version="1.0" encoding="UTF-8"?>
<Costs>
  <Tariff id="t255">770</Tariff>
  ...
</Costs>
```

Ниже описаны элементы, которые могут быть включены в тело ответа на запрос отзывов. Для элементов, содержащих значения, приведены примеры.

<Costs>

Корневой элемент списка стоимостей.

<Tariff>

Стоимость заказа (целое число в валюте тарифа).

Атрибуты элемента:

id

Идентификатор тарифа, по которому рассчитана стоимость. Все указанные в ответе идентификаторы должны соответствовать [списку тарифов](#) Службы Такси.

Релизы

Номер версии документа состоит из трех чисел. При доработке документа номер версии изменяется по следующим правилам:

1. первое число увеличивается при радикальных переработках описываемого API;
2. второе число увеличивается при обратно-несовместимых изменениях API и обнуляется при изменении первого числа;
3. третье число увеличивается при обратно-совместимых изменениях и обнуляется при изменении второго числа.

Релиз 1.12.2, декабрь 2013 года

Добавлено описание [нового формата тарифов](#).

Релиз 1.12.1, декабрь 2013 года

Удалены некоторые требования к машине, добавлено новое, об оплате поездки [купоном](#).

Релиз 1.12.0, март 2013 года

Все несрочные заказы теперь обрабатываются в рамках запроса [/requestcar](#) (Яндекс.Такси различает срочные и несрочные заказы по требуемому времени подачи машины). Запрос [/requestcardelayed](#) больше не используется. Подробно обработка заказов описана в разделе [Запросы обработки заказов](#).

В профиле водителя теперь можно указать [номер конкретного получателя платежа в Яндекс.Деньгах](#) — например, если водители в рамках одной Службы Такси привязаны к разным юридическим лицам.

Релиз 1.11.12, декабрь 2012 года

Исправлены и дополнены [поддерживаемые характеристики машин](#) и [возможные требования](#) к ним.

Релиз 1.11.11, октябрь 2012 года

Изменен формат запроса [/1.x/requestcar](#): теперь в таком запросе может прийти как срочный, так и несрочный заказ. В запросе несрочного заказа отсутствует элемент `<Cars>`, а элемент `<BookingTime>` всегда содержит атрибут `type` со значением «`exact`».

Все запросы [/1.x/requestcar](#) следует обрабатывать по-прежнему: отправлять [подтверждение](#) и принимать [закрепление заказа](#).

Запрос [/1.x/requestcardelayed](#) также нужно поддерживать, принимая несрочные заказы по прежней схеме.

Релиз 1.11.10, июль 2012 года

Профиль водителя дополнен необязательными элементами `<RealName>` и `<RealWeb>`. Значения этих элементов описывают таксопарк, идентифицированный элементом `<RealClid>`:

- `<RealName>` — название таксопарка на русском языке;
- `<RealWeb>` — адрес сайта таксопарка.

Релиз 1.11.9, июнь 2012 года

Описан [запрос отзывов](#) у Яндекс.Такси, с помощью которого Служба Такси может получить отзывы о закрепленных за Службой заказах, полученные за последние сутки.

В [предложении срочного заказа](#) и в [закреплении несрочного заказа](#) описан элемент `<ClosestStation>` — название станции метро, ближайшей к месту подачи машины.

Описание элементов `<FullName>` в запросах от Яндекс.Такси дополнено: в адресе теперь может быть указан подъезд.

Релиз 1.11.8, май 2012 года

В предложении [срочного заказа](#) и в [закреплении срочного заказа](#) описание водителей дополнено ссылкой на статическую карту. Карты, генерируемые для предложения и закрепления заказа, различаются:

- в предложении заказа элемент `<MapHref>` содержит ссылку на карту, показывающую текущее положение водителя и место подачи машины;
- при закреплении заказа элемент `<MapHref>` содержит ссылку на карту, показывающую оптимальный маршрут проезда до места подачи машины.

Релиз 1.11.7, май 2012 года

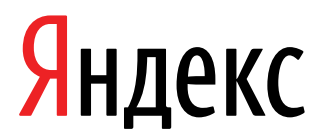
Описание клиента (элемент `<Recipient>` в предложениях [срочного](#) и несрочного заказов) дополнено атрибутом `loyal`, характеризующим лояльность клиента к сервису Яндекс.Такси.

Профиль водителя дополнен необязательным элементом `<RealClid>`, содержащим идентификатор таксопарка, который участвует в Яндекс.Такси при посредничестве Службы Такси. Идентификатор используется для отображения логотипа таксопарка в мобильных приложениях.

Релиз 1.11.6, февраль 2012 года

Новый документ.

Составлен на основе Приложения 3 к договору.



Программный интерфейс сервиса Яндекс.Такси
для Служб Такси
Справочник разработчика

10.04.2014