# Руководство по API на основе протокола HTTP

# Введение

Для организации технического взаимодействия транспортной компании «Magic Trans» с клиентами и партнёрами реализован программный интерфейс (далее: API), работающий на основе протокола HTTP.

Данное руководство ориентировано в первую очередь на технических специалистов, имеющих навыки программирования и знание основ протокола HTTP.

# Общие сведения

API использует протокол HTTP для транспортировки данных. Используется два типа запросов: GET или POST (предпочтительнее последний). Все запросы являются синхронными и не зависят друг от друга. Весь функционал API разделён на методы и сгруппирован в модули.

# Протокол запроса к API

Шаблон URL для запроса к API выглядит следующим образом:

***#BASE\_URL#/#MODULE#/#METHOD# . #FORMAT#***

*#BASE\_URL#* – путь к API (на текущий момент: **http://magic-trans.ru/api/v1/**, v1 – это версия API, в дальнейшем может меняться, но обратная совместимость будет сохранена)

*#MODULE#* – символьный код модуля, уникален в системе. Список доступных модулей отражён в таблице 1.

*#METHOD#* – символьный код вызываемого метода, уникален в рамках одного модуля.

*#FORMAT#* – формат возвращаемых данных. Система может отвечать на запросы либо в формате JSON, либо XML. По умолчанию – JSON. Если нет необходимости указывать этот параметр, то точка на конце URL также **не нужна**!

Примеры:

http://magic-trans.ru/api/v1/module/method – ответит в формате JSON

http://magic-trans.ru/api/v1/module/method.json – ответит в формате JSON

http://magic-trans.ru/api/v1/module/method.xml – ответит в формате XML

# Протокол ответа

Объект, описывающий ответ, всегда содержит 4 ключа:

Таблица 1. Ключи ответа API

|  |  |  |
| --- | --- | --- |
| **Ключ** | **Тип** | **Примечания** |
| status | boolean | true в случае успешного исполнения запроса, false – в случае ошибки. |
| error | integer | код ошибки, возможные значения описаны в таблице 2. Если ошибок нет, то значение 0. |
| message | string | Текст с описанием ошибки. |
| result | mixed | Результат выполнения запроса. Формат ответа зависит от метода и запроса. |

При получении данных в формате XML ответ оборачивается в тег <response/>.

Пример ответа в формате JSON:

{"status":true,"error":0,"message":null,"result":null}

Пример ответа в формате XML:

<response>

 <status>true</status>

 <error>0</error>

 <message/>

 <result/>

</response>

# Коды ошибок

Таблица 2. Коды ошибок

|  |  |
| --- | --- |
| **Код ошибки** | **Описание** |
| 0 | Нет ошибок. |
| 1 | Неизвестная ошибка. |
| 2 | Модуль не найден. |
| 3 | Метод не найден. |
| 100 | По заданному направлению перевозка грузов не осуществляется. |
| 101 | Не указан город. |
| 102 | Город не найден. |
| 103 | Не указаны обязательные параметры груза. |
| 104 | Индивидуальный расчет. |
| 200 | Не указан номер трека. |
| 201 | Номер трека не найден. |

# Список доступных модулей

Таблица 3. Доступные модули

|  |  |
| --- | --- |
| **Символьный код** | **Описание** |
| dictionary | Справочники. Организует выдачу типовых данных, необходимых для расчета. |
| delivery | Организует расчёт доставки исходя из входных параметров. |

# Модуль справочников.

## Получение списка городов.

Код метода: **getCityList**

Пример URL: **http://magic-trans.ru/api/v1/dictionary/getCityList**

**Параметры:**

|  |  |  |
| --- | --- | --- |
| **Код параметра** | **Обяз.** | **Возможные значения** |
| terminal | нет | Y - выводить только те города, которые являются терминаламиN - обратноене задано - все |
| delivery | нет | Y - выводить только те города, в которые осуществляется доставкаN - обратноене задано - все |
| deliverycities | нет | Y - выводить только те города, в которые осуществляется доставка. В результатах выполнения будут только параметры id и name. |
| name | нет | Фильтрация по названию города. Можно передавать часть названия для поиска города. |

**Описание возвращаемого результата:**

Массив объектов, описывающих города. Каждый объект содержит ключи:

|  |  |
| --- | --- |
| **Код параметра** | **Возможные значения** |
| id | Числовой идентификатор города в системе «Magic Trans». |
| name | Название города. |
| delivery | Флаг (Y/N), показывающий, осуществляется ли доставка в данный город. |
| terminal | Флаг (Y/N), показывающий, является ли город терминалом. |

**Пример HTTP**: <http://magic-trans.ru/api/v1/dictionary/getCityList/>

**Пример Python**:

import requests

import json

url = 'http://magic-trans.ru/api/v1/dictionary/getCityList/'

params = {}

response = requests.post(*url*=url, *params*=params)

response = response.json()

print(response)

## Получение параметров для доставки по городу.

Код метода: **CityDelivery**

Пример URL: **http://magic-trans.ru/api/v1/delivery/CityDelivery**

|  |  |  |
| --- | --- | --- |
| **Код параметра** | **Обяз.** | **Возможные значения** |
| from | да | Идентификатор города отправленияСм. метод dictionary/getCityList |
| to | да | Идентификатор города полученияСм. метод dictionary/getCityList |

**Пример HTTP**: <https://magic-trans.ru/api/v1/delivery/CityDelivery/?from=121&to=128>

**Пример Python**:

import requests

import json

url = 'https://magic-trans.ru/api/v1/delivery/CityDelivery/'

params = {

    'from': str(121),

    'to': str(128),

}

response = requests.post(*url*=url, *params*=params)

response = response.json()

print(response)

**Описание возвращаемого результата:**

Массив объектов, описывающих параметры груза и цену по каждому параметру. Ключи для массивов ‘to’ и ‘from’ одинаковые. Каждый объект содержит ключи:

|  |  |
| --- | --- |
| **Код параметра** | **Возможные значения** |
| from | Массив, содержащий параметры по городу отправления груза. |
| to | Массив, содержащий параметры по городу приема груза. |
| └ id | Числовой идентификатор города в системе «Мейджик Транс». |
| └ name | Название города в системе «Мейджик Транс». |
| └ PROPERTY\_M\_5 | Цена доставки по городу для грузов меньше 5кг. |
| └ PROPERTY\_M\_40 | Цена доставки по городу для грузов больше 5кг и меньше 40кг. |
| └ PROPERTY\_M\_50\* | \*Данный параметр оставлен в целях совместимости, используйте PROPERTY\_M\_40 или PROPERTY\_M\_60. |
| └ PROPERTY\_M\_60 | Цена доставки по городу для грузов больше 40кг и меньше 60кг. |
| └ PROPERTY\_M\_100 | Цена доставки по городу для грузов больше 60кг и меньше 100кг. |
| └ PROPERTY\_M\_150 | Цена доставки по городу для грузов больше 100кг и меньше 150кг. |
| └ PROPERTY\_M\_200\* | \*Данный параметр оставлен в целях совместимости, используйте PROPERTY\_M\_250 |
| └ PROPERTY\_M\_250 | Цена доставки по городу для грузов больше 150кг и меньше 250кг. |
| └ PROPERTY\_M\_350 | Цена доставки по городу для грузов больше 250кг и меньше 350кг. |
| └ PROPERTY\_M\_500 | Цена доставки по городу для грузов больше 350кг и меньше 500кг. |
| └ PROPERTY\_M\_750 | Цена доставки по городу для грузов больше 500кг и меньше 750кг. |
| └ PROPERTY\_M\_1000 | Цена доставки по городу для грузов больше 750кг и меньше 1000кг. |
| └ PROPERTY\_M\_1250 | Цена доставки по городу для грузов больше 1000кг и меньше 1250кг. |
| └ PROPERTY\_M\_1500 | Цена доставки по городу для грузов больше 1250кг и меньше 1500кг. |
| └ PROPERTY\_M\_2000 | Цена доставки по городу для грузов больше 1500кг и меньше 2000кг. |
| └ PROPERTY\_M\_2500 | Цена доставки по городу для грузов больше 2000кг и меньше 2500кг. |
| └ PROPERTY\_M\_3000 | Цена доставки по городу для грузов больше 2500кг и меньше 3000кг. |
| └ PROPERTY\_M\_5000 | Цена доставки по городу для грузов больше 3000кг и меньше 5000кг. |
| └ PROPERTY\_M\_7000 | Цена доставки по городу для грузов больше 5000кг и меньше 7000кг. |
| └ PROPERTY\_M\_10000 | Цена доставки по городу для грузов больше 7000кг и меньше 10000кг. |
| └ PROPERTY\_M\_20000 | Цена доставки по городу для грузов больше 10000кг и меньше 20000кг. |
| └ PROPERTY\_V\_01 | Цена доставки по городу грузов объемом меньше 0.1м3. |
| └ PROPERTY\_V\_02 | Цена доставки по городу грузов объемом больше 0.1м3 и меньше 0.2м3. |
| └ PROPERTY\_V\_03 | Цена доставки по городу грузов объемом больше 0.2м3 и меньше 0.3м3. |
| └ PROPERTY\_V\_05 | Цена доставки по городу грузов объемом больше 0.3м3 и меньше 0.5м3. |
| └ PROPERTY\_V\_07 | Цена доставки по городу грузов объемом больше 0.5м3 и меньше 0.7м3. |
| └ PROPERTY\_V\_1 | Цена доставки по городу грузов объемом больше 0.7м3 и меньше 1м3. |
| └ PROPERTY\_V\_1\_5 | Цена доставки по городу грузов объемом больше 1м3 и меньше 1.5м3. |
| └ PROPERTY\_V\_2 | Цена доставки по городу грузов объемом больше 1.5м3 и меньше 2м3. |
| └ PROPERTY\_V\_3 | Цена доставки по городу грузов объемом больше 2м3 и меньше 3м3. |
| └ PROPERTY\_V\_4 | Цена доставки по городу грузов объемом больше 3м3 и меньше 4м3. |
| └ PROPERTY\_V\_6 | Цена доставки по городу грузов объемом больше 4м3 и меньше 6м3. |
| └ PROPERTY\_V\_8 | Цена доставки по городу грузов объемом больше 6м3 и меньше 8м3. |
| └ PROPERTY\_V\_12 | Цена доставки по городу грузов объемом больше 8м3 и меньше 12м3. |
| └ PROPERTY\_V\_14 | Цена доставки по городу грузов объемом больше 12м3 и меньше 14м3. |
| └ PROPERTY\_V\_16 | Цена доставки по городу грузов объемом больше 14м3 и меньше 16м3. |
| └ PROPERTY\_V\_30 | Цена доставки по городу грузов объемом больше 16м3 и меньше 30м3. |
| └ PROPERTY\_V\_35 | Цена доставки по городу грузов объемом больше 30м3 и меньше 35м3. |
| └ PROPERTY\_V\_40 | Цена доставки по городу грузов объемом больше 35м3 и меньше 40м3. |
| └ PROPERTY\_V\_86 | Цена доставки по городу грузов объемом больше 40м3 и меньше 86м3. |
| └ PROPERTY\_dim\_04\_04\_04 | Цена доставки по городу грузов габаритами до 0,4/0,4/0,4 д/ш/в (м). |
| └ PROPERTY\_dim\_05\_05\_05 | Цена доставки по городу грузов габаритами до 0,5/0,5/0,5 д/ш/в (м). |
| └ PROPERTY\_dim\_15\_09\_1\* | \*Данный параметр оставлен в целях совместимости, используйте PROPERTY\_dim\_15\_12\_12 |
| └ PROPERTY\_dim\_15\_09\_12\* | \*Данный параметр оставлен в целях совместимости, используйте PROPERTY\_dim\_15\_12\_12 |
| └ PROPERTY\_dim\_15\_12\_12 | Цена доставки по городу грузов габаритами до 1,5/1,2/1,2 д/ш/в (м). |
| └ PROPERTY\_dim\_15\_12\_16 | Цена доставки по городу грузов габаритами до 1,5/1,2/1,6 д/ш/в (м). |
| └ PROPERTY\_dim\_3\_17\_16 | \*Данный параметр оставлен в целях совместимости, используйте PROPERTY\_dim\_3\_17\_18 |
| └ PROPERTY\_dim\_3\_17\_18 | Цена доставки по городу грузов габаритами до 3/1,7/1,8 д/ш/в (м). |
| └ PROPERTY\_dim\_4\_205\_2 | Цена доставки по городу грузов габаритами до 4/2,05/2 д/ш/в (м). |
| └ PROPERTY\_dim\_5\_22\_22 | Цена доставки по городу грузов габаритами до 5/2,2/2,2 д/ш/в (м). |
| └ PROPERTY\_dim\_6\_24\_24 | Цена доставки по городу грузов габаритами до 6/2,4/2,4 д/ш/в (м). |
| └ PROPERTY\_dim\_13\_24\_25 | Цена доставки по городу грузов габаритами до 13/2,4/2,5 д/ш/в (м). |

## Получение списка упаковок.

Код метода: **getPackageList**

Пример URL: **http://magic-trans.ru/api/v1/dictionary/getPackageList**

Данный метод не принимает никаких параметров.

**Пример HTTP**: https://magic-trans.ru/api/v1/dictionary/getPackageList/

**Пример Python**:

import requests

import json

url = 'https://magic-trans.ru/api/v1/dictionary/getPackageList/'

params = {}

response = requests.post(*url*=url, *params*=params)

response = response.json()

print(response)

**Описание возвращаемого результата:**

Массив объектов, описывающих упаковку. Каждый объект содержит ключи:

|  |  |
| --- | --- |
| **Код параметра** | **Возможные значения** |
| id | Числовой идентификатор упаковки в системе «Magic Trans». |
| name | Название вида упаковки. |

# Модуль расчёта.

## Детальный расчёт

Код метода: **calculate**

Пример URL: **http://magic-trans.ru/api/v1/calculate**

**Параметры:**

|  |  |  |
| --- | --- | --- |
| **Код параметра** | **Обяз.** | **Описание** |
| from | да | Идентификатор города отправления.См. метод dictionary/getCityList |
| to | да | Идентификатор города назначения. См. метод dictionary/getCityList |
| regime | нет | Режимный груз.Y – даN – нет (по умолчанию) |
| price | нет | Объявленная стоимость груза, руб. |
| package | нет | Идентификатор дополнительной упаковки.См. метод dictionary/getPackageList |
| items | да | Массив объектов с описанием параметров позиций груза. Каждый объект должен содержать ключи:  |
| └ count | нет | Количество позиций. По умолчанию – 1. |
| └ weight | да | Вес одной позиции, кг |
| └ volume | да\* | Объём груза одной позиции, м3 |
| └ length | да\* | Длина позиции, м |
| └ width | да\* | Ширина позиции, м |
| └ height | да\* | Высота позиции, м |

\* Необходимо указывать либо объём, либо длину, ширину и высоту. Если указаны габариты, то объём будет рассчитан автоматически из приведённых данных.

**Пример HTTP**: [https://magic-trans.ru/api/v1/delivery/calculate/?from=464&to=216&price=10000&items[0][count]=4&items[0][weight]=20&items[0][volume]=1](https://magic-trans.ru/api/v1/delivery/calculate/?from=464&to=216&price=10000&items%5b0%5d%5bcount%5d=4&items%5b0%5d%5bweight%5d=20&items%5b0%5d%5bvolume%5d=1)

**Пример Python**:

import requests

import json

url = 'http://magic-trans.ru/api/v1/delivery/calculate'

params = {

    'from': str(121),

    'to': str(128),

    'items[0][weight]': '133',

    'items[0][volume]': '1',

}

response = requests.post(*url*=url, *params*=params)

response = response.json()

print(response)

**Описание возвращаемого результата:**

Объект, содержащий ключи:

|  |  |
| --- | --- |
| **Код параметра** | **Возможные значения** |
| price | Общая стоимость перевозки, руб. |
| min | Минимальный срок доставки, дней. |
| max | Максимальный срок доставки, дней. |
| routes | Массив объектов, описывающих маршрут перевозки. В каждом объекте содержатся ключи: |
| └ from  | Объект, описывающий город отправления. Объект содержит ключи: |
|  └ id | Идентификатор города. |
|  └ name | Название города. |
| └ to  | Объект, описывающий город назначения. Объект содержит ключи: |
|  └ id | Идентификатор города. |
|  └ name | Название города. |
| └ price  | Стоимость перевозки по этому маршруту, руб. |
| └ min | Минимальный срок доставки, дней. |
| └ max | Максимальный срок доставки, дней. |

**Отслеживание груза**

Код метода: **calculate**

Пример URL: **http://magic-trans.ru/api/v1/getOrderInfo**

**Параметры:**

|  |  |  |
| --- | --- | --- |
| **Код параметра** | **Обяз.** | **Описание** |
| track | нет | Номер трека. |
| act | нет | Номер акта |

* **один из параметров должен быть обязательно заполнен**

**Пример HTTP**: <https://magic-trans.ru/api/v1/delivery/getOrderInfo/?track=TR010746247MT>

**Пример Python**:

import requests

import json

url = 'http://magic-trans.ru/api/v1/delivery/getOrderInfo'

params = {

    'track': 'TR010746247MT',

}

response = requests.post(*url*=url, *params*=params)

response = response.json()

print(response)

**Описание возвращаемого результата:**

Объект, содержащий ключи:

|  |  |
| --- | --- |
| **Код параметра** | **Возможные значения** |
| track | Номер трека. |
| date | Дата и время создания акта. |
| act | Номер акта. |
| marking | Маркировка груза. |
| status | Статус груза. |
| location | Текущее расположение. |
| estimation\_date | Ориентировочная дата получения груза. |
| date\_of\_issue | Дата выдачи груза |
| delivery\_date | Дата доставки на терминал |