☆ Home / API Marketplace / Conversion APIs / Exchange Rates Data API



# Exchange Rates Data API

Real-Time, Intraday & Historical Market Data API.

<u>API info</u>

**Documentation** 

# **Exchange Rates Data API Reference**

This API is organized around REST. Our API has predictable resource-oriented URLs, accepts form-encoded request bodies, returns JSON-encoded responses, and uses standard HTTP response codes, authentication, and verbs.

# **Just Getting Started?**

Check out our development quickstart guide.

# <u>Authentication</u>

Exchange Rates Data API uses API keys to authenticate requests. You can view and manage your API keys in the <u>Accounts page</u>.

Your API keys carry many privileges, so be sure to keep them secure! Do not share your secret API keys in publicly accessible areas such as GitHub, client-side code, and so forth.

All requests made to the API must hold a custom HTTP header named "apikey". Implementation differs with each programming language. Below are some samples.

All API requests must be made over HTTPS. Calls made over plain HTTP will fail. API requests without authentication will also fail.

# **Endpoints**

# > GET /convert

Currency conversion endpoint, which can be used to convert any amount from one currency to another. In order to convert currencies, please use the API's convert endpoint, append the from and to parameters and set them to your preferred base and target currency codes.

# **Parameters**

# amount (required)

The amount to be converted.

Location: Query, Data Type: string

# from (required)

The three-letter currency code of the currency you would like to convert from.



Location: Query, Data Type: string

You are subscribed to Free Plan.



# Categories

# **API Provider**



# Developer links

- Support
- Privacy Policy

Report abuse

#### to (required)

The three-letter currency code of the currency you would like to convert to.

Location: Query, Data Type: string

#### date (optional)

Specify a date (format YYYY-MM-DD) to use historical rates for this conversion.

Location: Query, Data Type: string

#### **GET** /convert

```
var myHeaders = new Headers();
1
     myHeaders.append("apikey", "5kBzBGbLo9TRgNDDwIspgUGkV0ix5I7k");
2
3
    var requestOptions = {
4
      method: 'GET',
5
      redirect: 'follow',
6
     headers: myHeaders
7
    };
8
9
    fetch("https://api.apilayer.com/exchangerates_data/convert?to={to}&from={frc
10
       .then(response => response.text())
11
```

\*\* A word enclosed with curly brackets "{ }" in the code means that it is a parameter and it should be replaced with your own values when executing. (also overwriting the curly brackets).

#### Returns

Below is a sample response from the endpoint

```
"date": "2018-02-22",
    "historical": "",
    "info": {
        "rate": 148.972231,
        "timestamp": 1519328414
     },
    "query": {
        "amount": 25,
        ""Saan": "CDD"
```

# > GET /fluctuation

Fluctuation endpoint you will be able to retrieve information about how currencies fluctuate on a day-to-day basis. To use this feature, simply append a start\_date and end\_date and choose which currencies (symbols) you would like to query the API for. Please note that the maximum allowed timeframe is 365 days.

# Parameters

# end\_date (required)

The end date of your preferred timeframe.

Location: Query, Data Type: string



# start\_date (required)

The start date of your preferred timeframe.

Location: Query, Data Type: string

### base (optional)

Enter the three-letter currency code of your preferred base currency.

Location: Query, Data Type: string

# symbols (optional)

Enter a list of comma-separated currency codes to limit output currencies.

Location: Query, Data Type: string

#### **GET** /fluctuation

```
var myHeaders = new Headers();
1
     myHeaders.append("apikey", "5kBzBGbLo9TRgNDDwIspgUGkV0ix5I7k");
2
3
    var requestOptions = {
4
      method: 'GET',
5
      redirect: 'follow',
6
      headers: myHeaders
7
    };
8
9
    fetch("https://api.apilayer.com/exchangerates_data/fluctuation?start_date={s
10
       .then(response => response.text())
11
```

\*\* A word enclosed with curly brackets "{ }" in the code means that it is a parameter and it should be replaced with your own values when executing. (also overwriting the curly brackets).

#### Returns

Below is a sample response from the endpoint

```
"base": "EUR",
    "end_date": "2018-02-26",
    "fluctuation": true,
    "rates": {
        "JPY": {
            "change": 0.0635,
            "change_pct": 0.0483,
            "end_rate": 131.651142,
```

# > GET /latest

Returns real-time exchange rate data updated every 60 minutes, every 10 minutes or every 60 seconds.

# **Parameters**

# base (optional)

Enter the three-letter currency code of your preferred base currency.

Location: Query, Data Type: string



symbols (optional)

Enter a list of comma-separated currency codes to limit output currencies.

Location: Query, Data Type: string

#### **GET /latest**

```
var myHeaders = new Headers();
1
     myHeaders.append("apikey", "5kBzBGbLo9TRgNDDwIspgUGkV0ix5I7k");
2
3
     var requestOptions = {
4
      method: 'GET',
5
      redirect: 'follow',
6
      headers: myHeaders
7
    };
8
9
     fetch("https://api.apilayer.com/exchangerates_data/latest?symbols={symbols}&
10
       .then(response => response.text())
11
```

\*\* A word enclosed with curly brackets "{ }" in the code means that it is a parameter and it should be replaced with your own values when executing. (also overwriting the curly brackets).

#### Returns

Below is a sample response from the endpoint

```
{
    "base": "USD",
    "date": "2021-03-17",
    "rates": {
        "EUR": 0.813399,
        "GBP": 0.72007,
        "JPY": 107.346001
    },
    "success": true,
    "timestame": 1510206206
```

# > GET /symbols

Returning all available currencies.

# **Parameters**

No parameters.

# **GET /symbols**

```
var myHeaders = new Headers();
1
     myHeaders.append("apikey", "5kBzBGbLo9TRgNDDwIspgUGkV0ix5I7k");
2
3
     var requestOptions = {
4
      method: 'GET',
5
      redirect: 'follow',
6
      headers: myHeaders
7
    };
8
9
    fetch("https://api.apilayer.com/exchangerates_data/symbols", requestOptions)
10
       .then(response => response.text())
11
```



Below is a sample response from the endpoint

```
{
    "success": true,
    "symbols": {
        "AED": "United Arab Emirates Dirham",
        "AFN": "Afghan Afghani",
        "ALL": "Albanian Lek",
        "AMD": "Armenian Dram"
    }
}
```

# > GET /timeseries

Timeseries endpoint lets you query the API for daily historical rates between two dates of your choice, with a maximum time frame of 365 days.

#### **Parameters**

#### end\_date (required)

The end date of your preferred timeframe.

Location: Query, Data Type: string

### start\_date (required)

The start date of your preferred timeframe.

Location: Query, Data Type: string

# base (optional)

Enter the three-letter currency code of your preferred base currency.

Location: Query, Data Type: string

# symbols (optional)

Enter a list of comma-separated currency codes to limit output currencies.

Location: Query, Data Type: string

# **GET** /timeseries

```
var myHeaders = new Headers();
1
     myHeaders.append("apikey", "5kBzBGbLo9TRgNDDwIspgUGkV0ix5I7k");
2
3
     var requestOptions = {
4
      method: 'GET',
5
      redirect: 'follow',
6
      headers: myHeaders
7
    };
8
9
     fetch("https://api.apilayer.com/exchangerates_data/timeseries?start_date={st
10
       .then(response => response.text())
11
```

<sup>\*\*</sup> A word enclosed with curly brackets "{ }" in the code means that it is a parameter and it should be replaced with your own values when executing. (also overwriting the curly brackets).



#### Returns

Below is a sample response from the endpoint

# > GET /{date}

Historical rates are available for most currencies all the way back to the year of 1999. You can query the Fixer API for historical rates by appending a date (format YYYY-MM-DD) to the base URL.

#### **Parameters**

#### date (required)

A date in the past for which historical rates are requested.

Location: Path, Data Type: string

# base (optional)

Enter the three-letter currency code of your preferred base currency.

Location: Query, Data Type: string

# symbols (optional)

Enter a list of comma-separated currency codes to limit output currencies.

Location: Query, Data Type: string

# GET /{date}

```
var myHeaders = new Headers();
1
     myHeaders.append("apikey", "5kBzBGbLo9TRgNDDwIspgUGkV0ix5I7k");
2
3
     var requestOptions = {
4
      method: 'GET',
5
       redirect: 'follow',
6
       headers: myHeaders
7
8
9
     fetch("https://api.apilayer.com/exchangerates_data/{date}?symbols={symbols}&
10
       .then(response => response.text())
11
```

\*\* A word enclosed with curly brackets "{}" in the code means that it is a parameter and it should be replaced with your own values when executing. (also overwriting the curly brackets).

# Returns



Below is a sample response from the endpoint

# **Rate Limiting**

Each subscription has its own rate limit. When you become a member, you start by choosing a rate limit that suits your usage needs. Do not worry; You can upgrade or downgrade your plan at any time. For this reason, instead of starting with a larger plan that you do not need, we can offer you to upgrade your plan after you start with "free" or "gold plan" options and start using the API.

You are currently subscribed to this API with a "". Your subscription has two types of restrictions: "daily" and "monthly" rate limits. These are:

Limit	Allocated	Used *
Daily	250	0
Monthly	250	0

<sup>\*</sup> Updated every 15 minutes.

When you reach a rate limit (both daily and monthly), the service will stop responding and returning the HTTP 429 response status code (Too many requests) for each request with the following JSON string body text.

```
{
"message":"You have exceeded your daily\/monthly API rate limit. Please review and upgra
}
```

A reminder email will be sent to you when your API usage reaches both 80% and 90%, so that you can take immediate actions such as upgrading your plan in order to prevent your application using the API from being interrupted.

You can also programmatically check your rate limit yourself. As a result of each request made to the APILayer, the following 4 fields provide you with all the necessary information within the HTTP Headers.

x-ratelimit-limit-month: Request limit per month

x-ratelimit-remaining-month: Request limit remaining this month

x-ratelimit-limit-day: Request limit per day

x-ratelimit-remaining-day: Request limit remaining today

You can contact our <u>support unit</u> if you need any assistance with your application regarding to handle the returned result by looking at the header information.

# **Error Codes**



APILayer uses standard HTTP response codes to indicate the success or failure of an API request. In general: Codes in the 2xx range indicate success. Codes in the 4xx range indicate a clientside error, which means that failed given the information provided (e.g., a missing parameter, unauthorized access etc.). Codes in the 5xx range indicate an error with APILayer's servers (normally this should'nt happen at all).

If the response code is not 200, it means the operation failed somehow and you may need to take an action accordingly. You can check the response (which will be in JSON format) for a field called 'message' that briefly explains the error reported.

Status Code	Explanation
400 - Bad Request	The request was unacceptable, often due to missing a required parameter.
401 - Unauthorized	No valid API key provided.
404 - Not Found	The requested resource doesn't exist.
429 - Too many requests	API request limit exceeded. See section Rate Limiting for more info.
5xx - Server Error	We have failed to process your request. (You can contact us anytime)

You can always <u>contact for support</u> and ask for more assistance. We'll be glad to assist you with building your product.

See Also



# <u>countrylayer</u>

Data APIs

Reliable real-time country data API for your business.



# <u>coinlayer</u>

Finance APIs

Real-time Crypto Currency
Exchange Rates JSON Rest API.



# NLP API

Al & Machine Learning APIs

Enterprise grade "natural language processing" (NLP) tools with a simple, yet powerful API.

