

AllRatesToday

API Documentation

Developer Guide

<https://allratetoday.com>

- Real-time Exchange Rates
- Historical Data
- Time-series Analysis
- JSON API

• Version 1.0 • January 2026 •

<https://allratetoday.com/developers>

Contents

1	Introduction	2
1.1	What You Can Build	2
1.2	Key Features	2
2	Getting Started	2
2.1	Base URL	2
2.2	Authentication	2
2.3	Rate Limits	2
3	API Endpoint	3
3.1	GET /api/v1/rates	3
3.2	Query Parameters	3
3.3	Response Format	3
3.4	Response Fields	3
4	Example Requests	4
4.1	Get Latest Rate for Currency Pair	4
4.2	Get Rate at Specific Time	4
4.3	Get Historical Data Range	4
4.4	Get Grouped Historical Data	4
5	Code Examples	5
5.1	JavaScript / Node.js	5
5.2	Python	5
5.3	PHP	5
5.4	cURL	6
6	Error Handling	7
6.1	HTTP Status Codes	7
6.2	Handling Rate Limits	7
6.3	Best Practices	7
7	Support	7

1 Introduction

The AllRatesToday API provides developers with access to real-time and historical currency exchange rate data. Our RESTful API returns JSON responses and supports filtering by currency pairs, date ranges, and time intervals.

1.1 What You Can Build

- Currency converters for websites and mobile apps
- Financial dashboards with live exchange rates
- E-commerce platforms with multi-currency pricing
- Analytics tools for tracking currency trends
- Invoicing systems with automatic currency conversion
- Travel applications with current exchange rates

1.2 Key Features

- Real-time exchange rates updated continuously
- Historical rate data with flexible date ranges
- Time-series data grouped by day, hour, or minute
- Simple RESTful JSON API
- Rate limiting headers for efficient usage tracking

2 Getting Started

2.1 Base URL

All API requests should be made to:

```
https://allratetoday.com/api/v1
```

2.2 Authentication

Include your API key in the Authorization header:

```
Authorization: Bearer YOUR_API_KEY
```

2.3 Rate Limits

The API allows **100 requests per minute** per IP address. Rate limit information is included in response headers:

Header	Description
X-RateLimit-Limit	Maximum requests per minute
X-RateLimit-Remaining	Requests remaining in current window
X-RateLimit-Reset	Unix timestamp when limit resets

Table 1: Rate Limit Headers

3 API Endpoint

3.1 GET /api/v1/rates

Retrieves exchange rate data with optional filtering by currency pair, specific timestamp, or date range.

3.2 Query Parameters

Parameter	Type	Required	Description
source	string	No	Source currency code (e.g., USD, EUR)
target	string	No	Target currency code (e.g., INR, GBP)
time	ISO 8601	No	Get rate at specific timestamp
from	YYYY-MM-DD	No	Historical range start date
to	YYYY-MM-DD	No	Historical range end date
group	string	No	Group results: day, hour, or minute

Table 2: Available Query Parameters

3.3 Response Format

The API returns an array of rate objects in JSON format:

```
[
  {
    "rate": 1.0856,
    "source": "USD",
    "target": "EUR",
    "time": "2026-01-14T12:00:00Z"
  }
]
```

3.4 Response Fields

Field	Type	Description
rate	number	Exchange rate value
source	string	Source currency code
target	string	Target currency code
time	string	ISO 8601 timestamp

Table 3: Response Object Fields

4 Example Requests

4.1 Get Latest Rate for Currency Pair

```
GET /api/v1/rates?source=USD&target=EUR
```

Response:

```
[
  {
    "rate": 0.9234,
    "source": "USD",
    "target": "EUR",
    "time": "2026-01-14T15:30:00Z"
  }
]
```

4.2 Get Rate at Specific Time

```
GET /api/v1/rates?source=GBP&target=INR&time=2026-01-10T09:00:00Z
```

4.3 Get Historical Data Range

```
GET /api/v1/rates?source=USD&target=GBP&from=2026-01-01&to=2026-01-14
```

4.4 Get Grouped Historical Data

```
GET /api/v1/rates?source=EUR&target=JPY&from=2026-01-01&to=2026-01-07&group=day
```

Response:

```
[
  {"rate": 156.23, "source": "EUR", "target": "JPY", "time": "2026-01-01"},
  {"rate": 156.45, "source": "EUR", "target": "JPY", "time": "2026-01-02"},
  {"rate": 155.98, "source": "EUR", "target": "JPY", "time": "2026-01-03"},
  {"rate": 156.12, "source": "EUR", "target": "JPY", "time": "2026-01-04"},
  {"rate": 156.67, "source": "EUR", "target": "JPY", "time": "2026-01-05"},
  {"rate": 156.34, "source": "EUR", "target": "JPY", "time": "2026-01-06"},
  {"rate": 156.89, "source": "EUR", "target": "JPY", "time": "2026-01-07"}
]
```

5 Code Examples

5.1 JavaScript / Node.js

```
const response = await fetch(
  'https://allratetoday.com/api/v1/rates?source=USD&target=EUR',
  {
    headers: {
      'Authorization': 'Bearer YOUR_API_KEY'
    }
  }
);

const data = await response.json();
console.log('USD to EUR: ${data[0].rate}');
```

5.2 Python

```
import requests

response = requests.get(
  'https://allratetoday.com/api/v1/rates',
  params={
    'source': 'GBP',
    'target': 'INR',
    'from': '2026-01-01',
    'to': '2026-01-14'
  },
  headers={
    'Authorization': 'Bearer YOUR_API_KEY'
  }
)

data = response.json()
for rate in data:
  print(f"{rate['time']}: {rate['rate']}")
```

5.3 PHP

```
<?php
$curl = curl_init();

curl_setopt_array($curl, [
  CURLOPT_URL => 'https://allratetoday.com/api/v1/rates?source=EUR
    &target=USD',
  CURLOPT_RETURNTRANSFER => true,
  CURLOPT_HTTPHEADER => [
    'Authorization: Bearer YOUR_API_KEY'
  ]
]);

$response = curl_exec($curl);
$data = json_decode($response, true);

echo "EUR to USD: " . $data[0]['rate'];
```

```
curl_close($curl);  
?>
```

5.4 cURL

```
curl -X GET \  
  'https://allratetoday.com/api/v1/rates?source=USD&target=GBP' \  
  -H 'Authorization: Bearer YOUR_API_KEY'
```

6 Error Handling

6.1 HTTP Status Codes

Code	Status	Description
200	OK	Request successful
401	Unauthorized	Missing or invalid API key
429	Too Many Requests	Rate limit exceeded
500	Internal Server Error	Server-side error

Table 4: HTTP Status Codes

6.2 Handling Rate Limits

When you receive a 429 status code, check the `Retry-After` header:

```
if (response.status === 429) {  
  const retryAfter = response.headers.get('Retry-After');  
  console.log('Rate limited. Retry after ${retryAfter} seconds');  
}
```

6.3 Best Practices

- Cache responses to reduce API calls
- Monitor rate limit headers to avoid hitting limits
- Implement exponential backoff for retries
- Use date ranges efficiently to minimize requests
- Store your API key securely using environment variables

7 Support

- Documentation: <https://allratetoday.com/developers>
- Website: <https://allratetoday.com>

Build powerful currency applications with AllRatesToday API