

Currency Exchange Service API Documentation

Link to interactive API documentation: <http://localhost:8080/swagger-ui/index.html>

Overview

This API provides endpoints for foreign exchange functionalities, including retrieving exchange rates, currency conversion, and accessing conversion history.

Base URL

The base URL for accessing the API is <http://localhost:8080>.

Endpoints

1. Exchange Rate Endpoint

Description: Retrieve the current exchange rate between two currencies.

URL: [/exchange-rate ?sourceCurrency={ sourceCurrency }&targetCurrency={ targetCurrency }](#)

Method: GET

Parameters:

sourceCurrency: The source currency code (e.g., EUR).

targetCurrency: The target currency code (e.g., BGN).

Response:

HTTP Status Code: 200 OK

Body:

```
{  
  "from": "EUR",
```

```
"to": "BGN",  
"rate": 1.957338  
}
```

Error Response:

HTTP Status Code: 400 Bad Request

Body:

```
{  
  "from": "EUR",  
  "to": "BGN2",  
  "errorMessage": "Not found rate by given currencies! Please ensure that both the  
source and target currencies are valid and supported for conversion."  
}
```

2. Currency Conversion Endpoint

Description: Convert an amount from one currency to another.

URL: */convert?sourceCurrency={ sourceCurrency } &targetCurrency={
targetCurrency } &sourceAmount={ sourceAmount }*

Method: **GET**

sourceCurrency: The source currency code (e.g., EUR).

targetCurrency: The target currency code (e.g., BGN).

sourceAmount: The amount of source currency (e.g., 1 EUR).

Response:

HTTP Status Code: 200 OK

Body:

```
{
```

```
"transactionId": 4,  
"convertedAmount": 1.95734,  
"currency": "BGN"  
}
```

Error Response:

HTTP Status Code: 400 Bad Request

Body:

```
{  
  "currency": "B2GN",  
  "error": "Currency Conversion Error: Unable to convert between the specified  
currencies. Please ensure that both the source and target currencies are valid and  
supported for conversion."  
}
```

3. Conversion History Endpoint

Description: Retrieve a paginated list of currency conversions.

URL: /conversion-history? transactionId={ transactionId } ?transactionDate={ transactionDate } &page=1&size=2

Method: **GET**

Parameters:

transactionId (optional): The unique transaction identifier.

transactionDate (optional): The date of the transaction.

page (optional): The number of page.

size(optional): the size of page.

At least one of the parameters *transactionId* or *transactionDate* should be passed!

Response with passed *transactionId*:

HTTP Status Code: 200 OK

Body:

```
[
  {
    "transactionId": 1,
    "convertedAmount": 1.96,
    "currency": "BGN"
  }
]
```

Response with NOT passed *transactionId*, but passed *transactionDate* :

HTTP Status Code: 200 OK

Body:

```
[
  {
    "transactionId": 1,
    "convertedAmount": 51.09,
    "currency": "EUR"
  },
  {
    "transactionId": 2,
    "convertedAmount": 51.09,
    "currency": "EUR"
  }
]
```

```
    },  
    {  
      "transactionId": 3,  
      "convertedAmount": 51.09,  
      "currency": "EUR"  
    },  
    {  
      "transactionId": 4,  
      "convertedAmount": 1.96,  
      "currency": "BGN"  
    }  
  ]
```

Error Response in case of not passed any parameters:

HTTP Status Code: 400 Bad Request

Body:

```
[  
  {  
    "error": "At least one of transactionId or transactionDate must be provided."  
  }  
]
```

Error Response in case of not found based on the provided criteria:

HTTP Status Code: 400 Bad Request

Body:

```
[  
  {  
    "error": "Information Not Found: No information was found based on the  
provided criteria. Please check your input and try again."  
  }  
]
```