Role: Senior Java Developer,
Location: İstanbul
Team: Codespace, an OpenPayd company

You will develop a simple foreign exchange application which is one of the most common services used in financial applications. Requirements are as follows:

Functional Requirements:

1. Exchange Rate API
   - **input**: currency pair to retrieve the exchange rate
   - **output**: exchange rate
2. Conversion API:
   - **input:** source amount, source currency, target currency
   - **output:** amount in target currency, and transaction id.
3. Conversion List API
   - **input:** transaction id or transaction date (at least one of the inputs shall be provided for each call)
   - **output:** list of conversions filtered by the inputs and paging is required
4. The application shall use a service provider to retrieve exchange rates and optionally for calculating amounts. (see hints)
5. In the case of an error, a specific code to the error and a meaningful message shall be provided.

Technical Requirements:

1. Application shall be run without need of extra configuration.
2. APIs shall be developed as RESTful APIs with **_Spring Boot._**
3. Build & dependency management tools shall be used.
4. Design patterns shall be used when suitable.
5. Code structure (i.e. packages) shall reflect the separation of concerns.
6. Unit tests shall be provided.

Optional

1. Develop a central error logging mechanism for the application.
2. API Documentation shall be provided.

Hints

1. Example service providers for FX rates are listed below (any other one is also acceptable), limitations of service providers free services are acceptable (e.g. no real time rates, only one base currency support etc.)
   a. https://ratesapi.io/documentation/
   b. http://fixer.io/
2. For Spring Boot, their 15 minute tutorials can be used for guidance
   a. https://spring.io/guides/gs/spring-boot/
3. For quick setup and presentational convenience in memory DB can be used.