

Name: Rupen Dhandhukiya

Company: MSG Global Solutions India Private Limited

Country: India

ASSIGNMENT

TOP 3 COUNTRIES HIGHEST STANDARD VAT RATE & TOP 3 COUNTRIES LOWEST REDUCED VAT RATES

25-JAN-2023

Objective

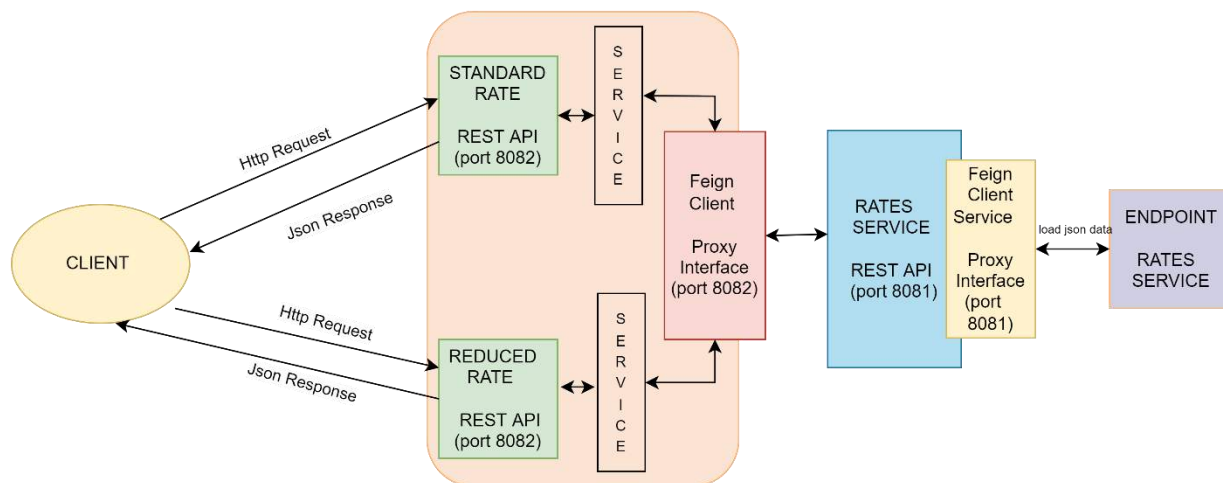
Build a microservice which exposes a REST API for:

- Return three EU countries with the highest standard VAT rate
- Return three EU countries with the lowest reduced VAT rates

Input: <https://euvatrates.com/rates.json>

Guidelines: Keep it clean and simple (with some design for future extensions). Implement as a Maven project using Spring Boot or Quarkus. Provide source code, a way to run it and some kind documentation for the API.

High Level Diagram



Frameworks, Tools & Languages

- Eclipse
- Spring Boot
- Java
- Swagger UI

API's Details

Service-1: rate-service

rate-service project is used to load json data from external end point service (<https://euvatrates.com//rates.json>). To communicate external service, it is used feignclient cloud API as proxy interface and the service runs on 8081 port as a separate service (http://localhost:8081/rates_microservice/vat/rates).

➤ **RateController Rest API**

The rest API is used to load the entire json data from external end point service and the loadRatesJsonData() method used to inject the service component to load the json data. Once the response is OK status then the countries and rates list sent as http ResponseEntity

➤ **RateService Service**

The service component is used to inject the rateServiceProxy interface to communicate and load the entire json data. Once data loaded then filtered the data as list of countries and rates by using Jackson API

➤ **RateServiceProxy Interface**

The proxy interface is used to register the cloud feign rest client by passing service name and URL details to communicate the services

Service-2: standard-reduced-rates-service

standard-reduced-rates-service project used to load the filtered json data from service1 (http://localhost:8081/rates_microservice/vat/rates). To communicate service1, it is used feignclient cloud API as proxy interface and the service2 runs on 8082 port as a separate services

(<http://localhost:8082/standard-reduced-rates-service//vat/rate/getStandardRates>) and (<http://localhost:8082/standard-reduced-rates-service//vat/rate/getReducedRates>)

➤ **StandardRateRestController Rest API**

The rest API is used to load the list of countries and rates json data from service1 and getHighestThreeCountriesStandardRates() method is used to inject the standardRateRestService service component to get the top highest three countries and standard vat rates data. Once the response is OK status then the countries and rates list sent as http ResponseEntity

➤ **ReducedRateRestController Rest API**

The rest API is used to load the list of countries and rates json data from service1 and getLowestThreeCountriesReducedRates() method is used to inject the

reducedRateRestService service component to get the top lowest three countries and reduced vat rates data. Once the response is OK status then the countries and rates list sent as http ResponseEntity

➤ **StandardRateRestService Service**

The service component is used to inject the countriesRatesServiceProxy interface to load the filtered json data and applied business logic to get the top highest countries and standard vat rates

➤ **ReducedRateRestService Service**

The service component is used to inject the countriesRatesServiceProxy interface to load the filtered json data and applied business logic to get the top lowest countries and reduced vat rates

➤ **CountriesRatesServiceProxy Interface**

The proxy interface is used to register the cloud feign rest client by passing service name and URL details to communicate the service1

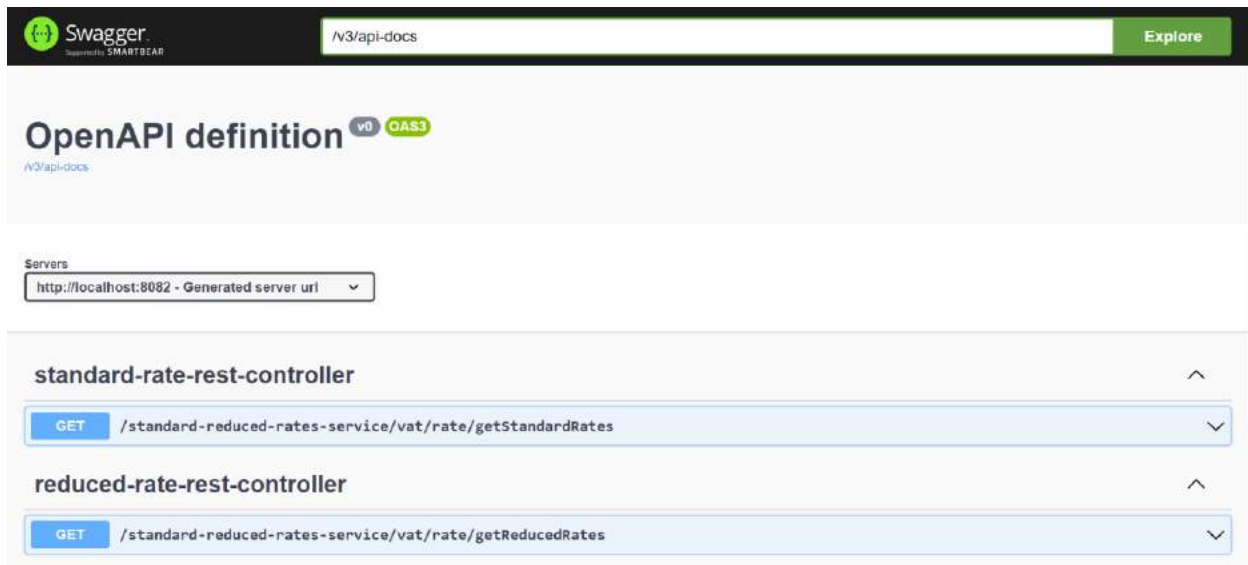
Steps to Test the Application

- Return three EU countries with the highest standard VAT rate
- Return three EU countries with the lowest reduced VAT rates

1. Execute the URL :

<http://localhost:8082/swagger-ui/index.html#>

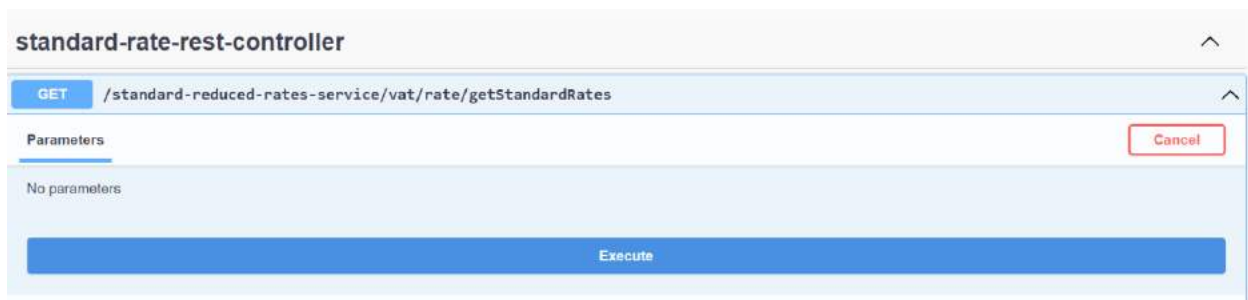
2. click "**GET**" button to test as per functional requirements



3. Click "**Try it out**" button



4. Click "**Execute**" button



5. Results

Return three EU countries with the highest standard VAT rate

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8082/standard-reduced-rates-service/vat/rate/getStandardRates' \
  -H 'accept: */*'
```

Request URL

```
http://localhost:8082/standard-reduced-rates-service/vat/rate/getStandardRates
```

Server response

| Code | Details |
|------|--|
| 200 | <p>Response body</p> <pre>[{ "country": "Hungary", "rate": 27 }, { "country": "Denmark", "rate": 25 }, { "country": "Croatia", "rate": 25 }]</pre> <p>Download</p> |

Return three EU countries with the lowest reduced VAT rates

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8082/standard-reduced-rates-service/vat/rate/getReducedRates' \
  -H 'accept: */*'
```

Request URL

```
http://localhost:8082/standard-reduced-rates-service/vat/rate/getReducedRates
```

Server response

| Code | Details |
|------|---|
| 200 | <p>Response body</p> <pre>[{ "country": "Denmark", "rate": 0 }, { "country": "United Kingdom", "rate": 5 }, { "country": "United Kingdom", "rate": 5 }]</pre> <p>Download</p> |

Steps to download and execute (in eclipse ide)

1. Go to command prompt and navigate to workspace folder and execute below said git command

git clone <https://github.com/rupen-dhandhukiya/assessment>

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.2251]
(c) Microsoft Corporation. All rights reserved.

C:\Rupen\assessment_workspace>git clone https://github.com/rupen-dhandhukiya/assessment
Cloning into 'assessment'...
remote: Enumerating objects: 179, done.
remote: Counting objects: 100% (81/81), done.
remote: Compressing objects: 100% (60/60), done.
Receiving objects: 74% (133/179)remote: Total 179 (delta 38), reused 7 (delta 7), pack-reused 98
Receiving objects: 100% (179/179), 34.85 KiB | 2.18 MiB/s, done.
Resolving deltas: 100% (54/54), done.

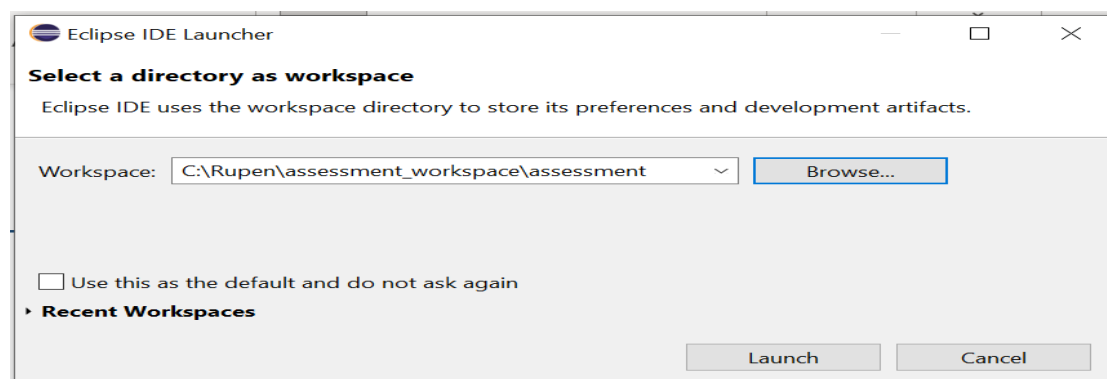
C:\Rupen\assessment_workspace>
```

2. Go to windows workspace folder path and make sure all files downloaded from git repo

Windows (C:) > Rupen > assessment_workspace > assessment

| <input type="checkbox"/> Name | Date modified | Type | Size |
|--------------------------------|-------------------|---------------|------|
| .git | 1/25/2023 6:58 PM | File folder | |
| rate-service | 1/25/2023 6:58 PM | File folder | |
| standard-reduced-rates-service | 1/25/2023 6:58 PM | File folder | |
| .gitignore | 1/25/2023 6:58 PM | Text Document | 1 KB |
| README | 1/25/2023 6:58 PM | MD File | 1 KB |

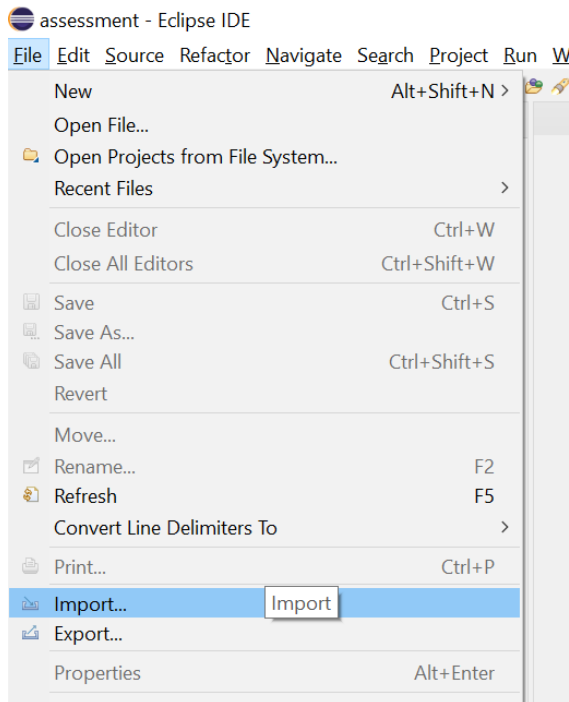
3. Open eclipse tool and select the downloaded project workspace folder and launch



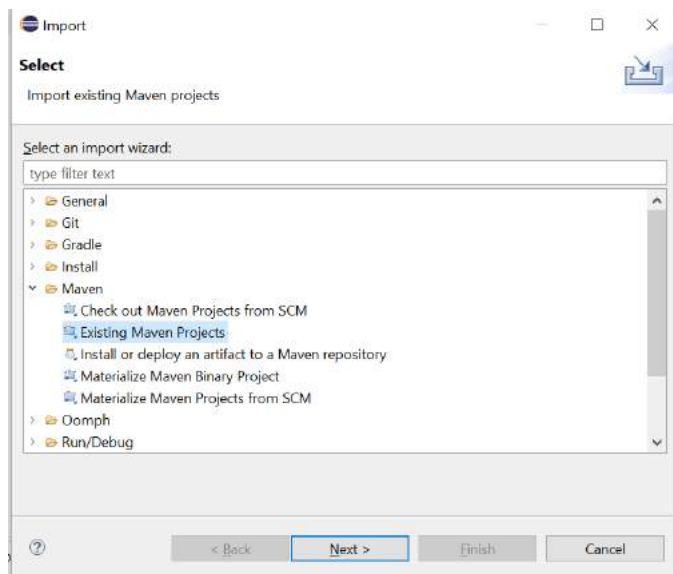
Below steps to be followed for both projects (**rate-service** and **standard-reduced-rates-service**)

rate-service

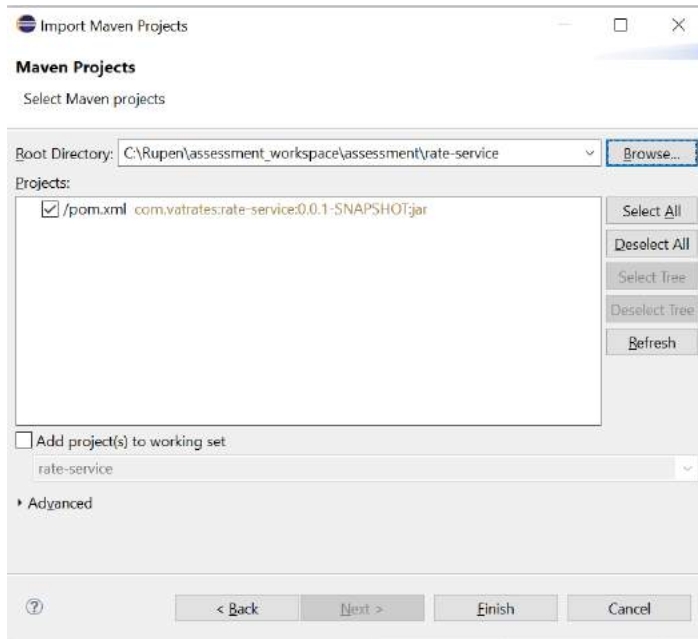
4. Select **File** and **Import**



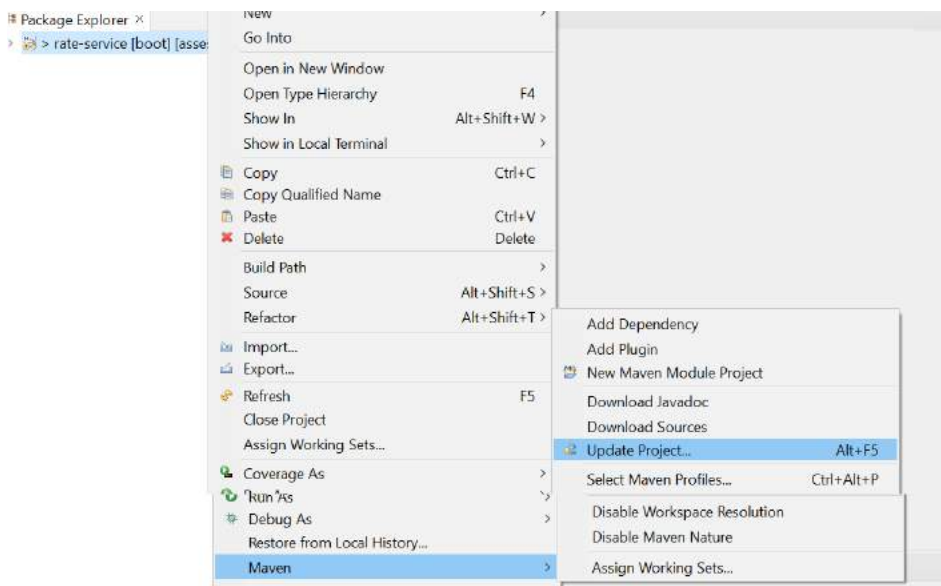
5. Select **"Existing Maven Projects"**



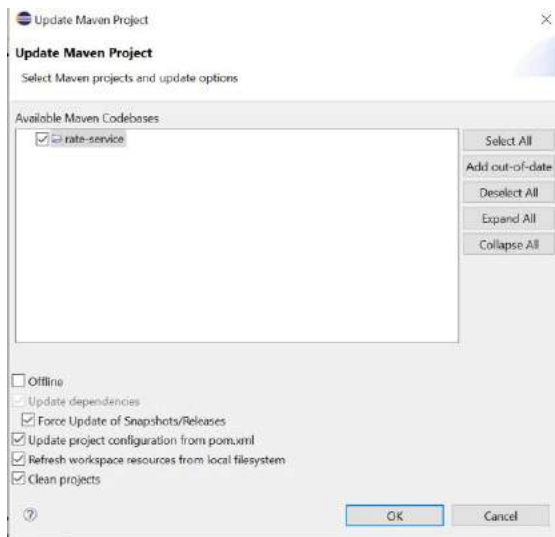
- Click **Browse** to select the directory and select **pom.xml** checkbox and Click **Finish**



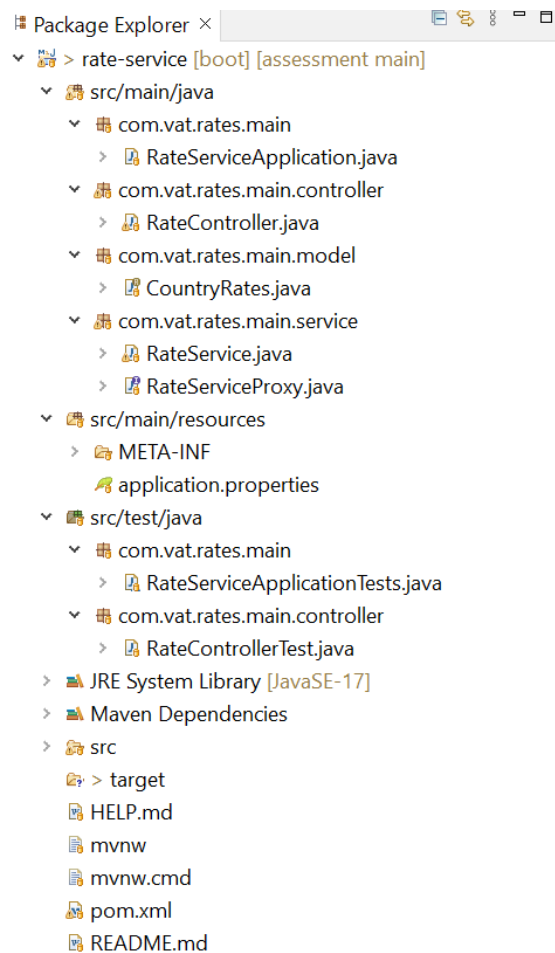
- Once project downloaded into eclipse then select the project and **right click -> Maven -> Update Project**



8. Select project and Select "**Force Update of Snapshots/Releases** checkbox and Click **OK** button



9. Project Structure



10. Go to Command Prompt and navigate to project workspace and execute below command

mvn spring-boot:run

```
C:\Rupen\assessment_workspace\assessment\rate-service>mvn spring-boot:run
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.vatrates:rate-service >-----
[INFO] Building rate-service 0.0.1-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] >>> spring-boot-maven-plugin:2.6.6:run (default-cli) > test-compile @ rate-service >>>
[INFO]
[INFO] --- maven-resources-plugin:3.2.0:resources (default-resources) @ rate-service ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Using 'UTF-8' encoding to copy filtered properties files.
[INFO] Copying 1 resource
[INFO] Copying 1 resource
[INFO]
[INFO] --- maven-compiler-plugin:3.8.1:compile (default-compile) @ rate-service ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 5 source files to C:\Rupen\assessment_workspace\assessment\rate-service\target\classes
[INFO]
[INFO] --- maven-resources-plugin:3.2.0:testResources (default-testResources) @ rate-service ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Using 'UTF-8' encoding to copy filtered properties files.
[INFO] skip non existing resourceDirectory C:\Rupen\assessment_workspace\assessment\rate-service\src\test\resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.1:testCompile (default-testCompile) @ rate-service ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 2 source files to C:\Rupen\assessment_workspace\assessment\rate-service\target\test-classes
[INFO]
[INFO] <<< spring-boot-maven-plugin:2.6.6:run (default-cli) < test-compile @ rate-service <<<
[INFO]
[INFO] --- spring-boot-maven-plugin:2.6.6:run (default-cli) @ rate-service ---
[INFO] Attaching agents: []

Spring
=====
```

11. Once Rate-Service (8081) project server started then go to browser run the swagger UI in browser to confirm the application service is UP

```
Spring
=====
:: Spring Boot :: (v2.6.6)

2023-01-25 19:24:54.885 INFO 17072 --- [main] c.c.c.ConfigServicePropertySourceLocator : Fetching config from server at : http://localhost:8888
2023-01-25 19:24:54.856 INFO 17072 --- [main] c.c.c.ConfigServicePropertySourceLocator : Connect Timeout Exception on Url - http://localhost:8888. Will be trying the next url if available
2023-01-25 19:24:54.857 WARN 17072 --- [main] c.c.c.ConfigServicePropertySourceLocator : Could not locate PropertySource: I/O error on GET request for "http://localhost:8888/rate-service/default": Connection refused: no further information; nested exception is java.net.ConnectException: Connection refused: no further information
2023-01-25 19:24:54.861 INFO 17072 --- [main] c.vat.rates.main.RateServiceApplication : No active profile set, falling back to 1 default profile: "default"
2023-01-25 19:24:55.162 INFO 17072 --- [main] o.s.cloud.context.scope.GenericScope : BeanFactory id=a4a96b27-fd2f-36fd-8091-0d52d44e2986
2023-01-25 19:24:55.343 INFO 17072 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8081 (http)
2023-01-25 19:24:55.350 INFO 17072 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2023-01-25 19:24:55.350 INFO 17072 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.60]
2023-01-25 19:24:55.437 INFO 17072 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2023-01-25 19:24:55.437 INFO 17072 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 566 ms
2023-01-25 19:24:56.058 INFO 17072 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8081 (http) with context path ''
2023-01-25 19:24:56.123 INFO 17072 --- [main] c.vat.rates.main.RateServiceApplication : Started RateServiceApplication in 1.906 seconds (JVM running for 2.115)
2023-01-25 19:25:03.256 INFO 17072 --- [nio-8081-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'
2023-01-25 19:25:03.256 INFO 17072 --- [nio-8081-exec-1] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'
2023-01-25 19:25:03.264 INFO 17072 --- [nio-8081-exec-1] o.s.web.servlet.DispatcherServlet : Completed initialization in 1 ms
```

<http://localhost:8081/swagger-ui/index.html>

rate-controller

GET /rates_microservice/vat/rates

Parameters

Cancel

No parameters

Execute

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8081/rates_microservice/vat/rates' \
  -H 'accept: */*'
```

Request URL

http://localhost:8081/rates_microservice/vat/rates

Server response

Code Details

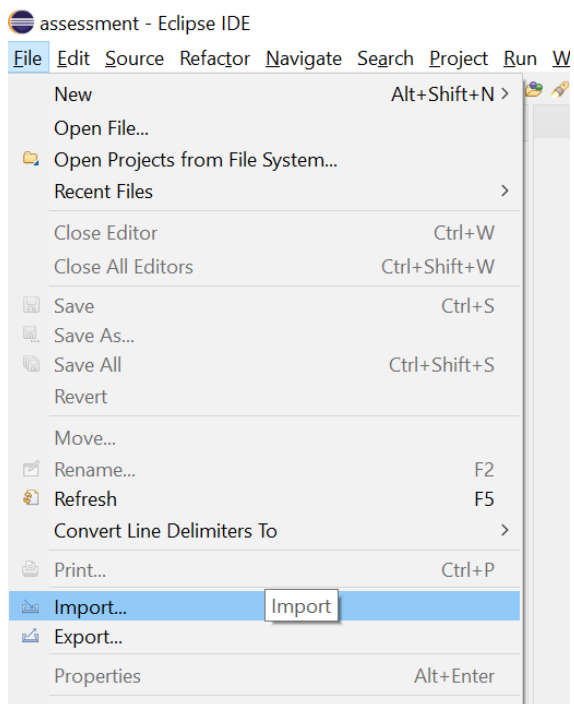
200

Response body

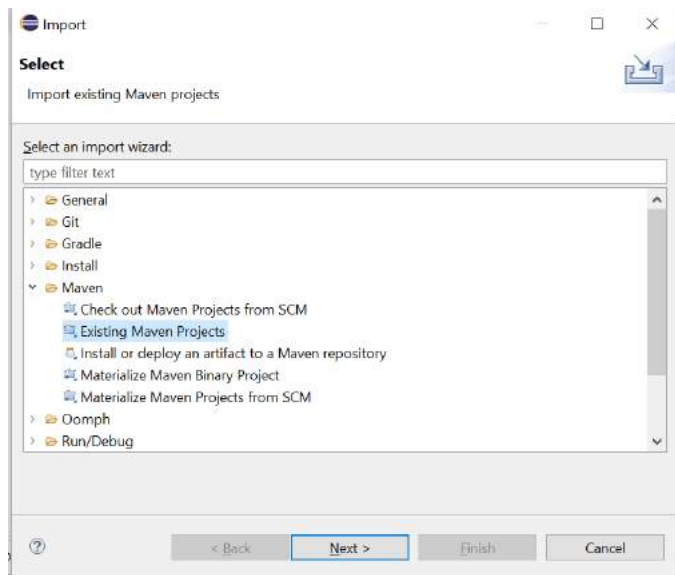
```
[
  {
    "country": "Austria",
    "standard_rate": 20,
    "reduced_rate": 10
  },
  {
    "country": "Belgium",
    "standard_rate": 21,
    "reduced_rate": 12
  },
  {
    "country": "Bulgaria",
    "standard_rate": 20,
    "reduced_rate": 9
  },
  {
    "country": "Czechia"
```

standard-reduced-rates-service

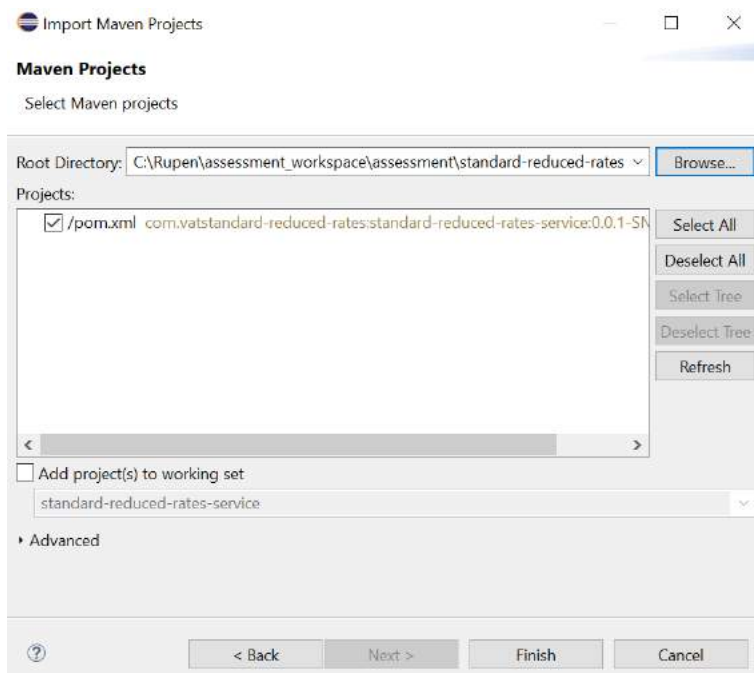
1. Select **File** and **Import**



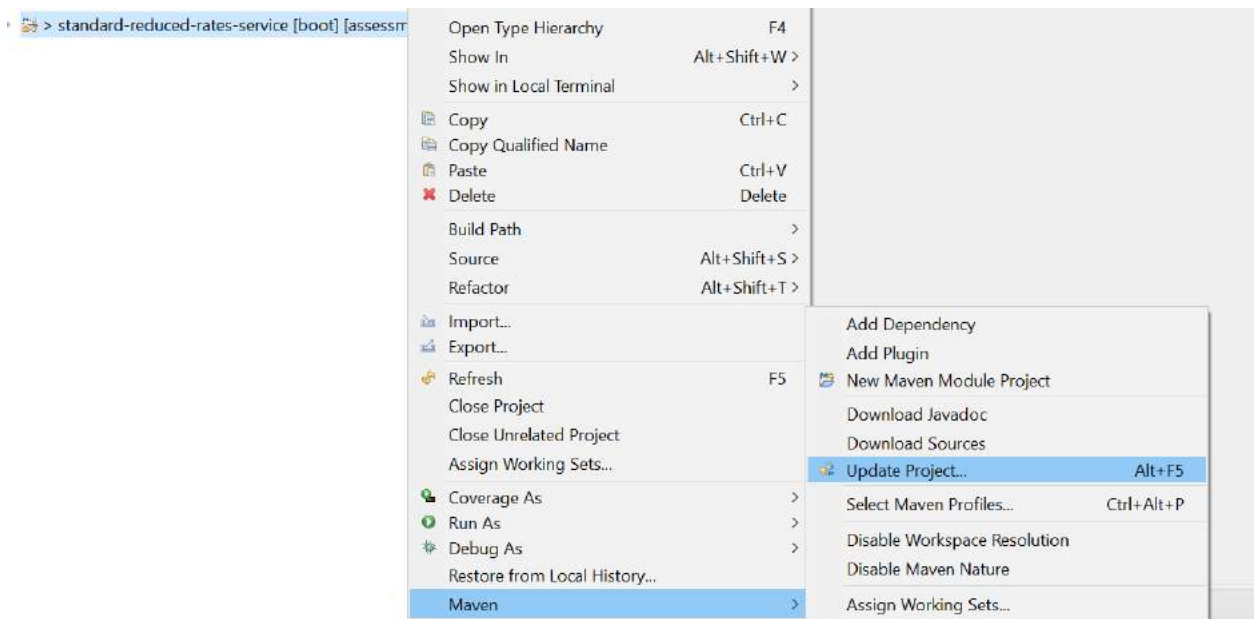
2. Select "Existing Maven Projects"



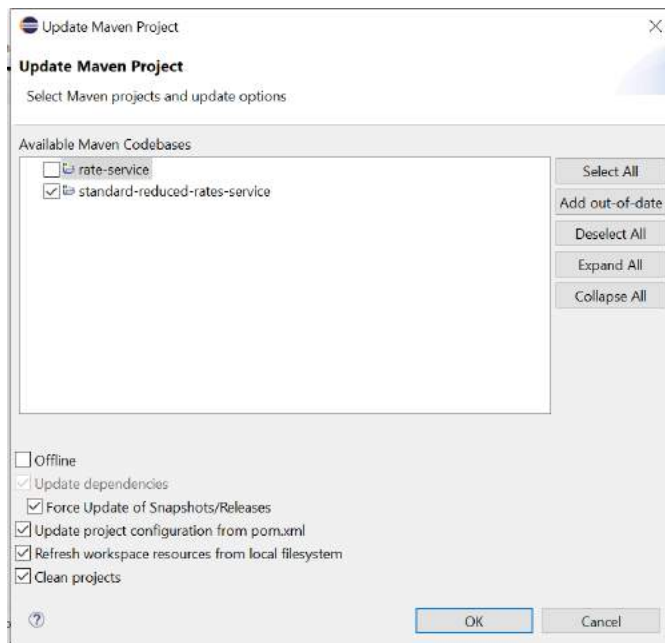
3. Click **Browse** to select the directory and select **pom.xml** checkbox and Click **Finish**



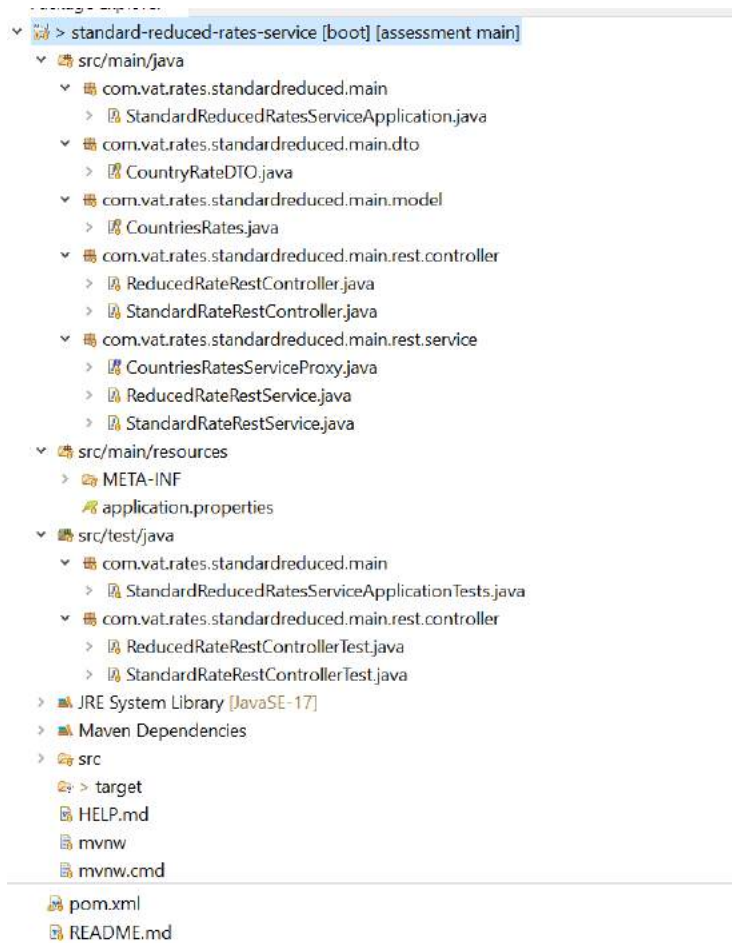
4. Once project downloaded into eclipse then select the project and **right click -> Maven -> Update Project**



5. Select project and Select "**Force Update of Snapshots/Releases**" checkbox and Click **OK** button



6. Project Structure



7. Go to Command Prompt and navigate to project workspace and execute below command

mvn spring-boot:run

```
C:\Windows\System32\cmd.exe - mvn spring-boot:run
Microsoft Windows [Version 10.0.19044.2251]
(c) Microsoft Corporation. All rights reserved.

C:\Rupen\assessment_workspace\assessment\standard-reduced-rates-service>mvn spring-boot:run
[INFO] Scanning for projects...
[INFO]
```



```
Spring Boot (v2.6.6)
:: Spring Boot ::

023-01-25 19:42:53.788 INFO 10464 --- [main] c.e.c.ConfigServicePropertySourceLocator : Fetching config from server at : http://localhost:8888
023-01-25 19:42:53.833 INFO 10464 --- [main] c.e.c.ConfigServicePropertySourceLocator : Connect Timeout Exception on Url - http://localhost:8888. Will be trying the next url if available
023-01-25 19:42:53.834 WARN 10464 --- [main] c.e.c.ConfigServicePropertySourceLocator : Could not locate PropertySource: I/O error on GET request for "http://localhost:8888/standard-reduced-rate-service/default": Connection refused: no further information; nested exception is java.net.ConnectException: Connection refused: no further information
023-01-25 19:42:53.836 INFO 10464 --- [main] m.StandardReducedRatesServiceApplication : No active profile set, falling back to 1 default profile: "default"
023-01-25 19:42:54.139 INFO 10464 --- [main] o.s.cloud.context.scope.GenericScope : BeanFactory id=655a12be-4e49-3d46-afe7-34f284a28a8c
023-01-25 19:42:54.321 INFO 10464 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8082 (http)
023-01-25 19:42:54.328 INFO 10464 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
023-01-25 19:42:54.329 INFO 10464 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.60]
023-01-25 19:42:54.410 INFO 10464 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
023-01-25 19:42:54.411 INFO 10464 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 561 ms
023-01-25 19:42:55.026 INFO 10464 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8082 (http) with context path ''
023-01-25 19:42:55.095 INFO 10464 --- [main] m.StandardReducedRatesServiceApplication : Started StandardReducedRatesServiceApplication in 1.873 seconds (JVM running for 2.086s)
```

8. Once **standard-reduced-rates-service** (8082) project server started then go to browser and run the swagger UI to test the requirements

Return three EU countries with the highest standard VAT rate

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8082/standard-reduced-rates-service/vat/rate/getStandardRates' \
  -H 'accept: */*'
```

Request URL

```
http://localhost:8082/standard-reduced-rates-service/vat/rate/getStandardRates
```

Server response

| Code | Details |
|------|--|
| 200 | <p>Response body</p> <pre>{ { "country": "Hungary", "rate": 27 }, { "country": "Denmark", "rate": 25 }, { "country": "Croatia", "rate": 25 } }</pre> <p>Download</p> |

Return three EU countries with the lowest reduced VAT rates

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8082/standard-reduced-rates-service/vat/rate/getReducedRates' \
  -H 'accept: */*'
```

Request URL

```
http://localhost:8082/standard-reduced-rates-service/vat/rate/getReducedRates
```

Server response

| Code | Details |
|------|---|
| 200 | <p>Response body</p> <pre>{ { "country": "Denmark", "rate": 0 }, { "country": "United Kingdom", "rate": 5 }, { "country": "United Kingdom", "rate": 5 } }</pre> <p>Download</p> |