1. **Import project into eclipse**

File -> Import -> Maven -> Existing Maven Projects -> Browse Project from cloned location (task)

1. **Right click on project in eclipse and then Maven -> Update Projects**
2. **Update database credential and other configuration into application.properties available in src/main/java/resources**

spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver

spring.datasource.url=jdbc:mysql://localhost:3306/algodomain

spring.datasource.username=root

spring.datasource.password=

spring.jpa.show-sql=true

spring.jpa.hibernate.ddl-auto=update

spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.MySQL5Dialect

server.port=9090

* Create a table in your database – algodomain
* Change mysql username and password as per your installation

open src/main/resources/application.properties

change spring.datasource.username and spring.datasource.password

as per your mysql installation

* Build and run the app using maven

mvn package-

java -jar target/spring-boot-rest-api-tutorial-0.0.1-SNAPSHOT.jar

Alternatively, you can run the app without packaging it using –

mvn spring-boot:run

* The app will start running at http://localhost:9090.

1. **Right click on Application.java file and run as Java Application**

**Once Sprint Boot Application will be started successfully then we**

* can call following Endpoints by using POSTMAN

1. **To get Product call following endpoint with** GET **Request type in postman**

* To get List of product- <http://localhost:9090/products>
* To get by id- <http://localhost:9090/productById>/{Id}
* To get by name - [http://localhost:9090/product/{name}](http://localhost:9090/product/%7bname%7d)

1. **To Create New Product use following url with** POST **Request type in postman**

* To add a product- <http://localhost:9090/addProduct>
* **set content type as in header as application/json**
* **set request body as raw with JSON payload**

{

    "productId": 1002,

    "name": "LG Washing Machine ",

    "productType": "Washing Machine",

    "category": "Home Appliances",

    "basePrice": 26000

}

1. **To update Product in database, use following url with**PUT**Request type in postman**

* Set request body as raw with JSON payload

{

    "productId": 1002,

    "name": "LG Washing Machine ",

    "productType": "Washing Machine",

    "category": "Home Appliances",

    "basePrice": 26000

}

* To Update - <http://localhost:9090/update>
* set content type as in header as application/json

1. **To delete a particular Book from database, use following url with**DELETE**Request type in postman**

* To Delete -[http://localhost:9090/](http://localhost:9090/update)delete/{id}

**Note - Replace with actual id**