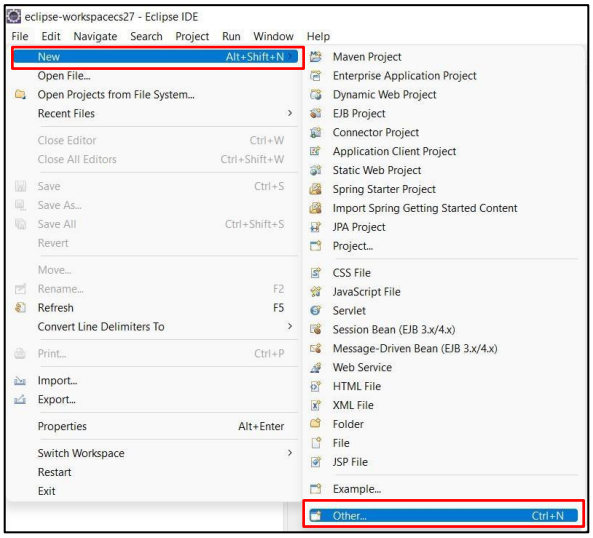
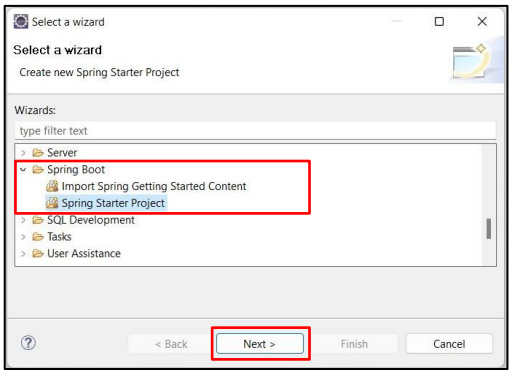
WEEK – 9

1. Build user authentication flow and authorization using SpringSecurity.

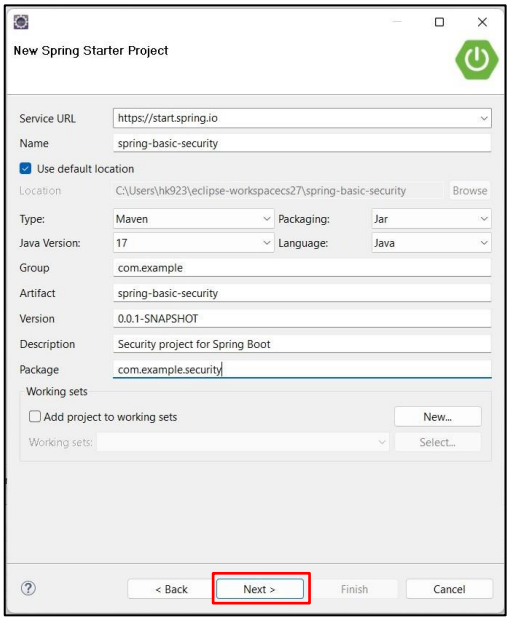
Step 1: Open ‘Eclipse IDE for Enterprise edition’ & go to → Help →Eclipse Marketplace as shown below. Now search for ‘Spring tool suite’ & click on ‘Install’ to install the latest version. After the installation is complete, go to → File → New → Other option as shown below.



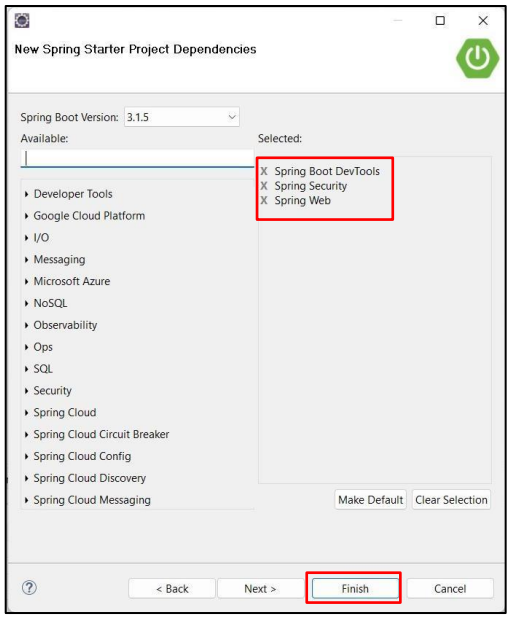
Step 2: Now select ‘Spring Starter Project’ & click on ‘Next’ option.



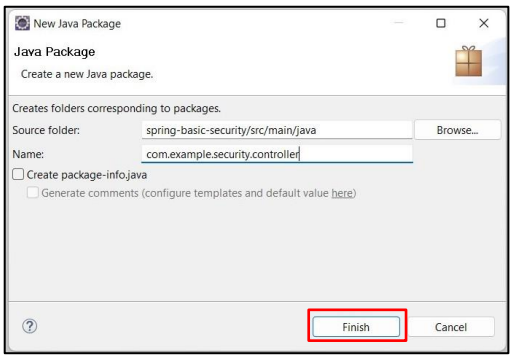
Step 3: Give the Name as ‘User’ → select ‘Maven’ → Description as User Registration Spring Boot Application → Package as ‘spring-basic-security’ & then click on ‘Next’.



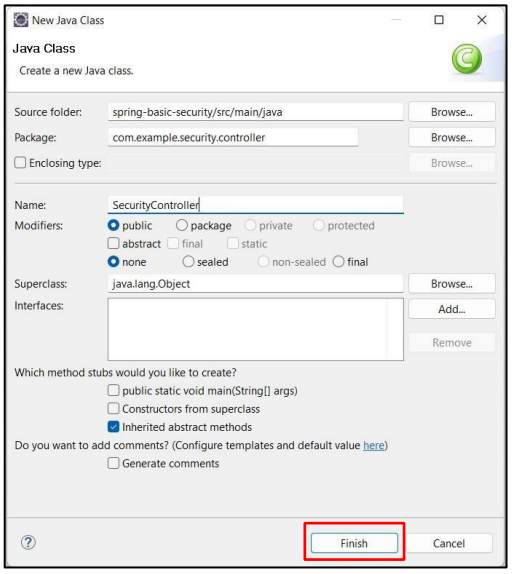
Step 4: Now add all the required dependencies as shown below & click on ‘Finish’.



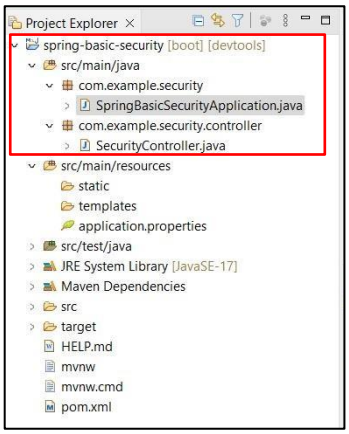
Step 5: Create a new package named ‘com.example.security.controller’ & click on ‘Finish’.



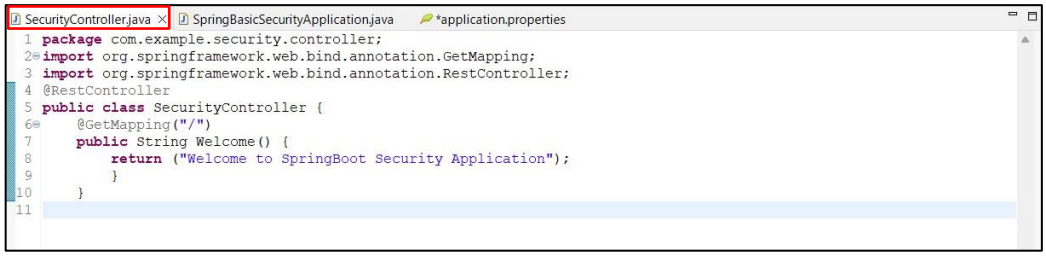
Step 6: Now create a new class named ‘SecurityController’ & click on ‘Finish’.



Step 7: All the created packages and classes are displayed here.



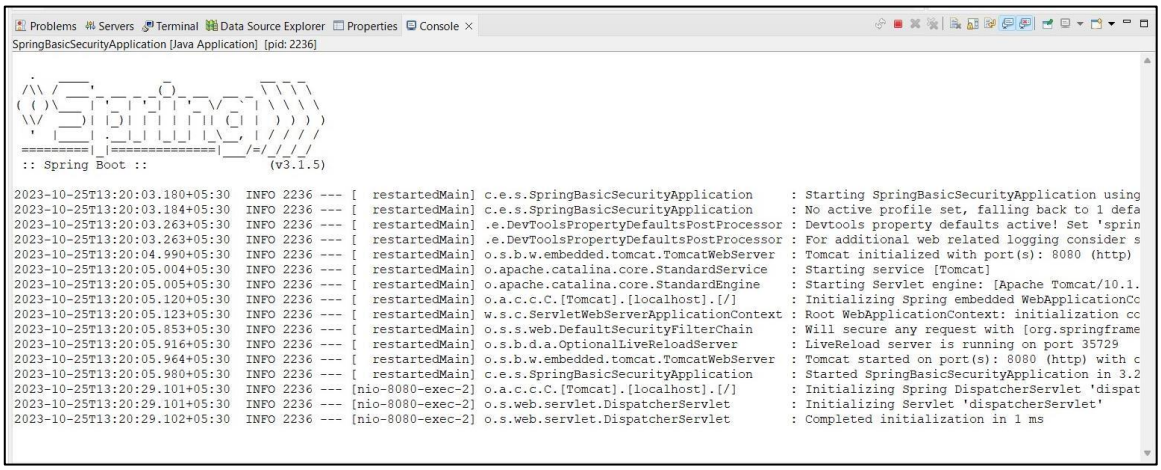
Step 8: Now add code to SecurityController.java file as shown below.



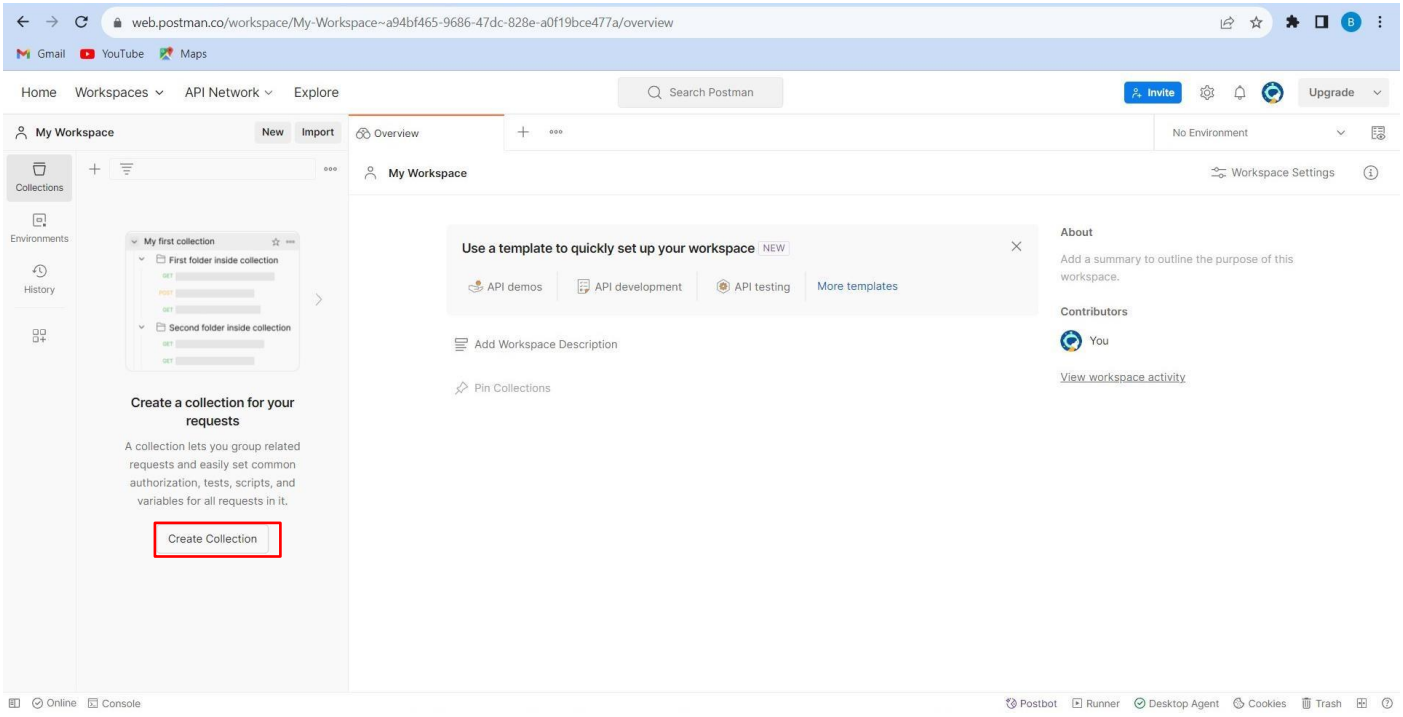
Step 9: Now add code to SpringBasicSecurityApplication.java file as shown below.

Step 10: Now add code to ‘application.properties’ file as shown below.

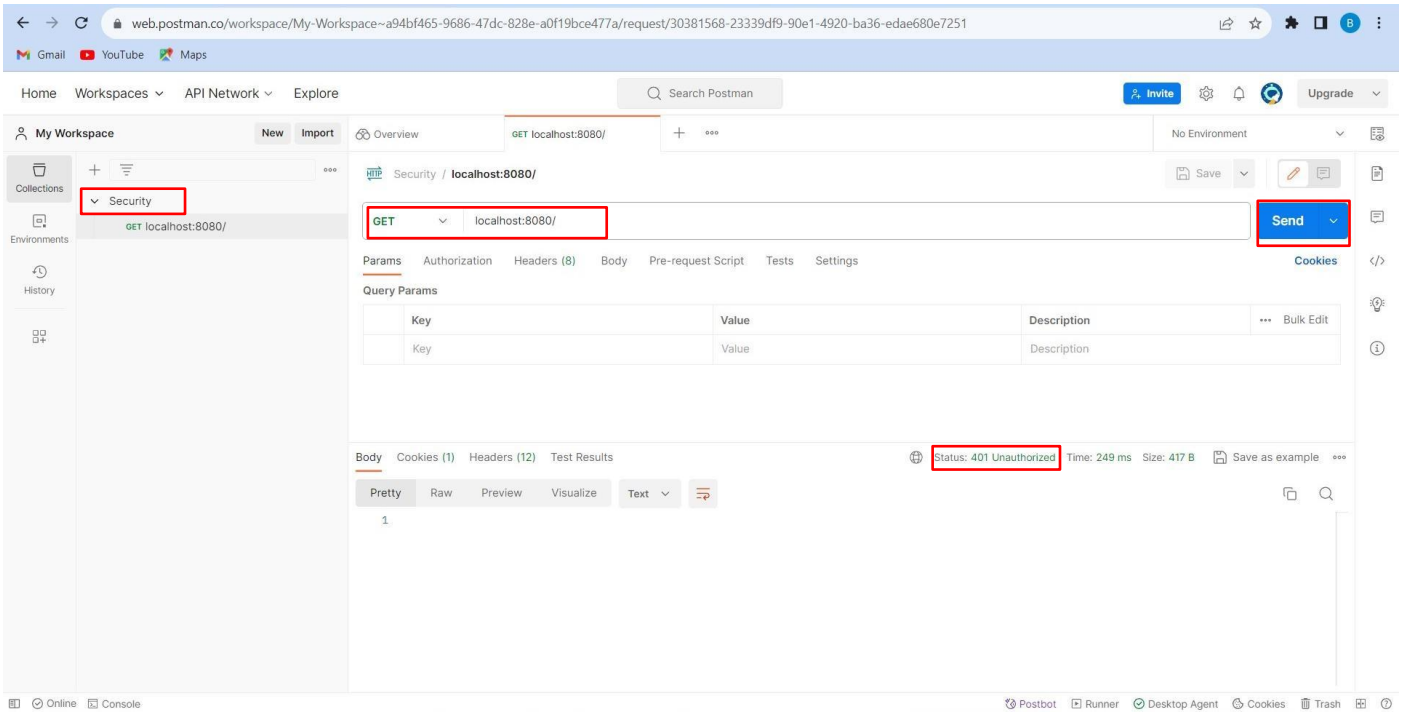
Step 11: Start the server as shown below.



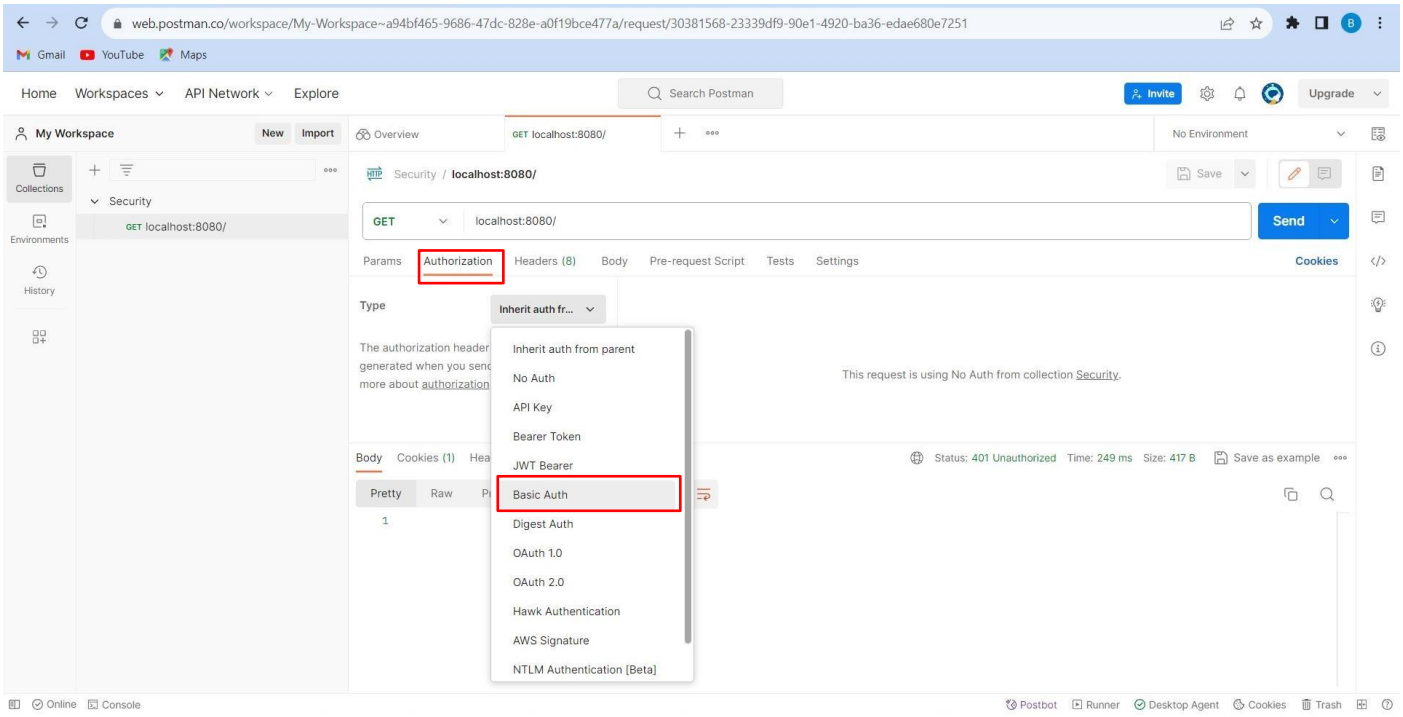
Step 12: Now login to the Postman & create a new collection as shown below.



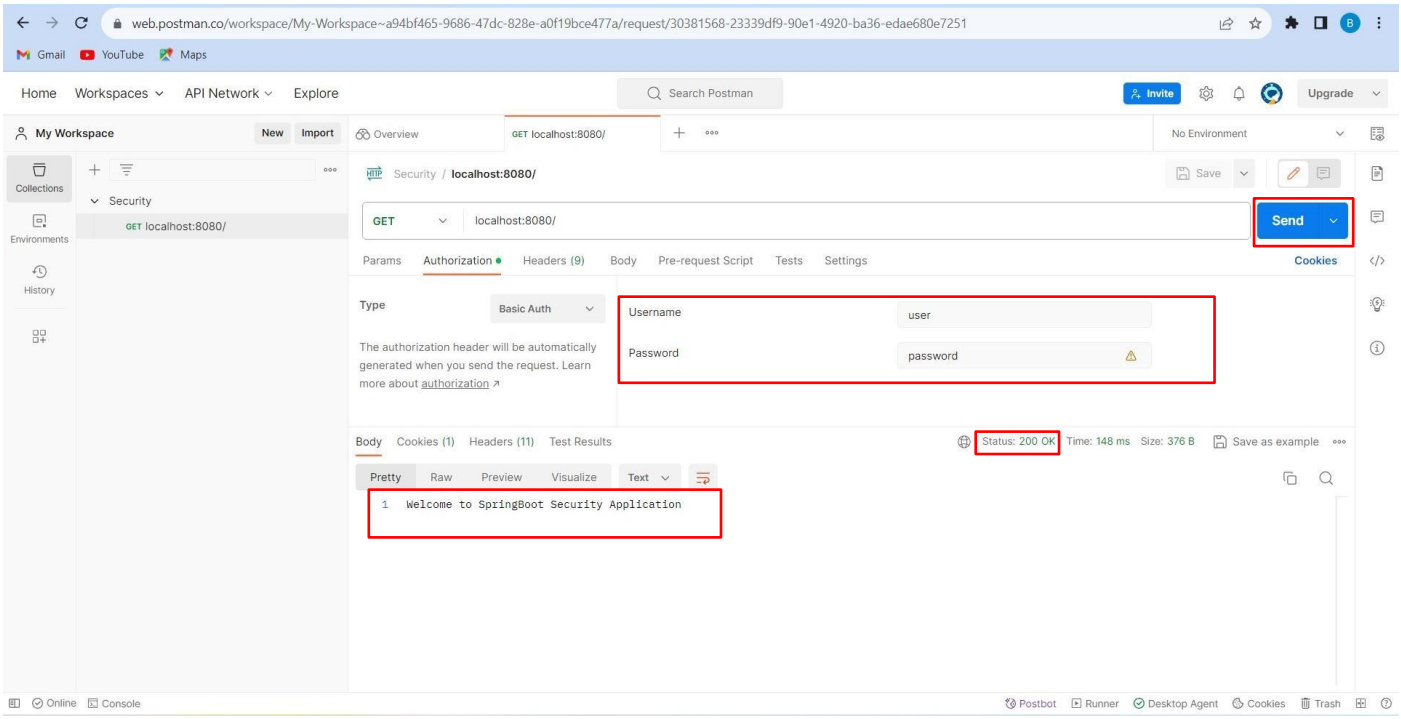
Step 13: Select GET method & enter ‘localhost:8080/’. Now Save it, then click on ‘Send’. The status displayed here is unauthorized so make sure to add the authorization to your code.



Step 14: Now go to ‘Authorization’ tab select the type as ‘Basic Auth’.



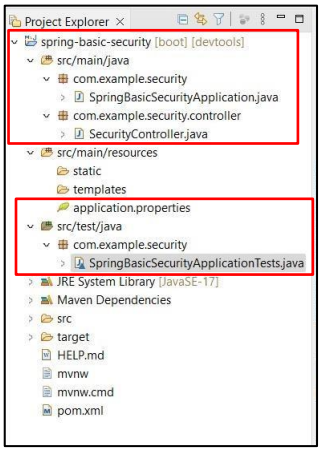
Step 15: Now enter Username & Password as mentioned below. Then Save it & click on Send to view the output.



- **Implementing Junit for the above Security API**

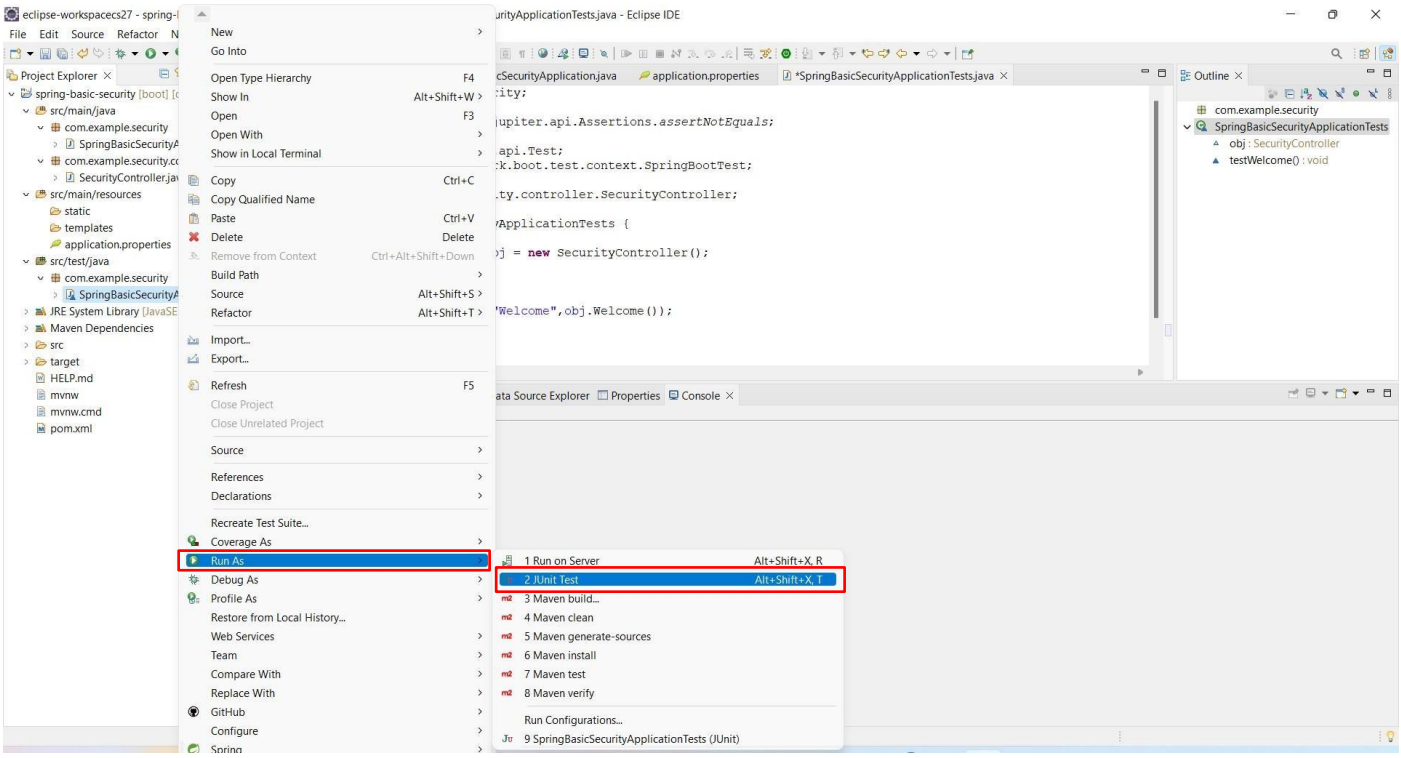
Step 16: Follow the same procedures as above from ‘step 1’.

Note: Create the packages & classes as shown below, then add code as mentioned from ‘step 8’.

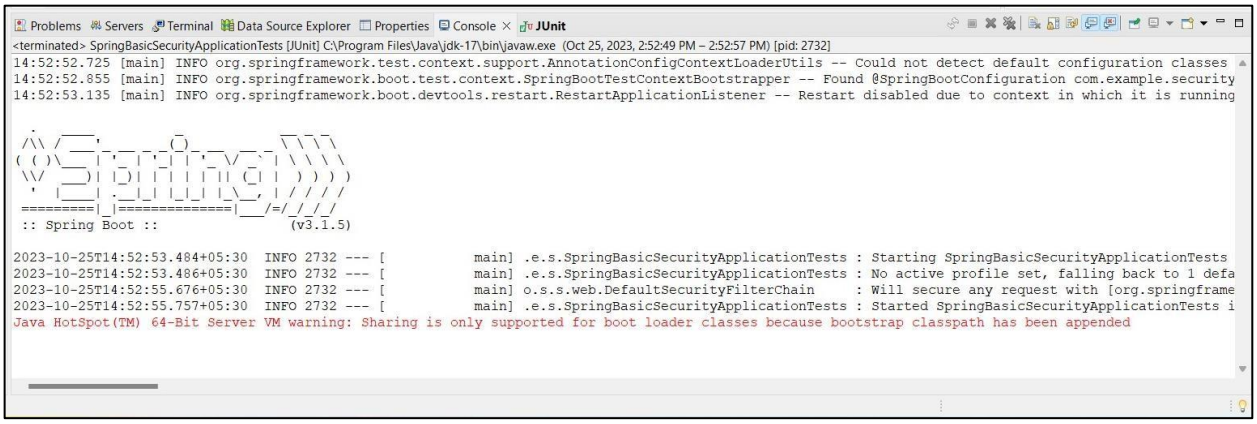


Step 17: Now add code to ‘SpringBasicSecurityApplicationTests.java’ as shown below.

Step 18: Right click on the ‘SpringBasicSecurityApplicationTests.java’ file Run As 2 Junit Test.

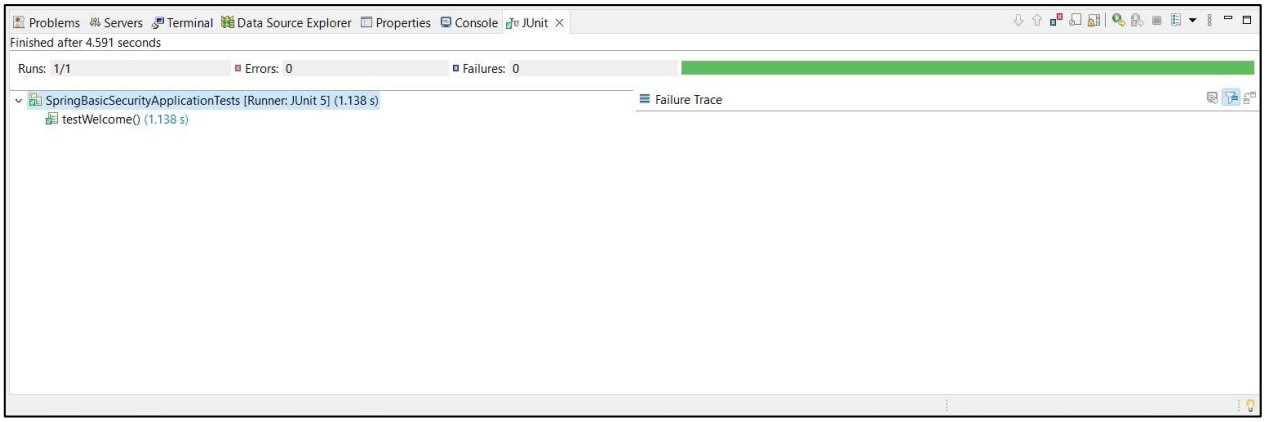


Step 19: View the output as shown below.



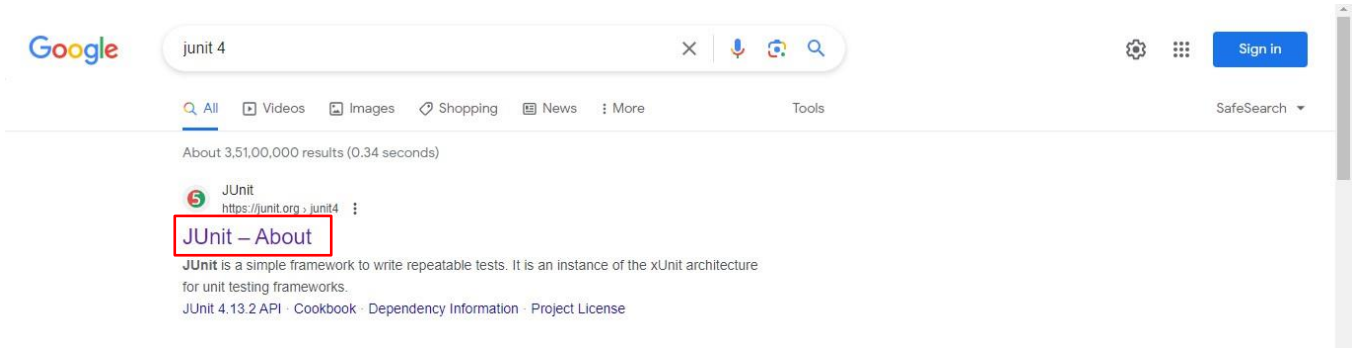
Step 20: The Junit has executed successfully.

Note: The green colour indicates the given test condition is true & passed. As well as red colour indicates the given test condition is false & failed.

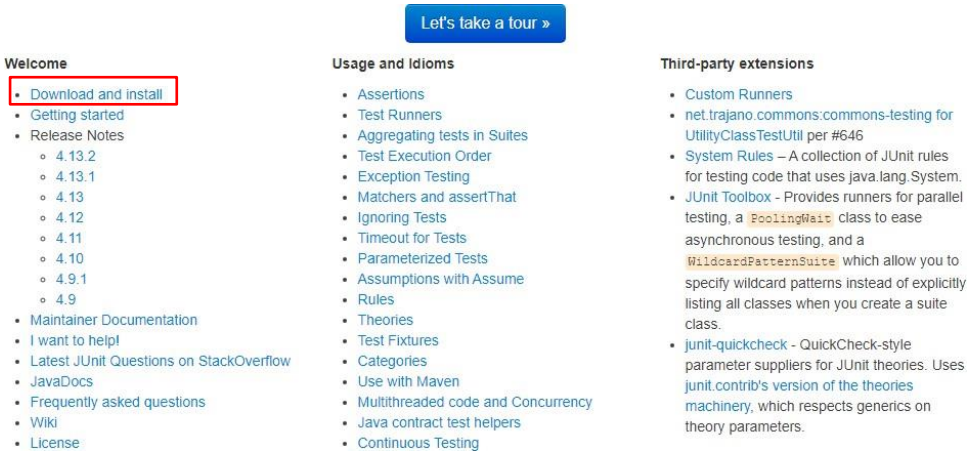


2. Writing Junit test cases for CRUD operations

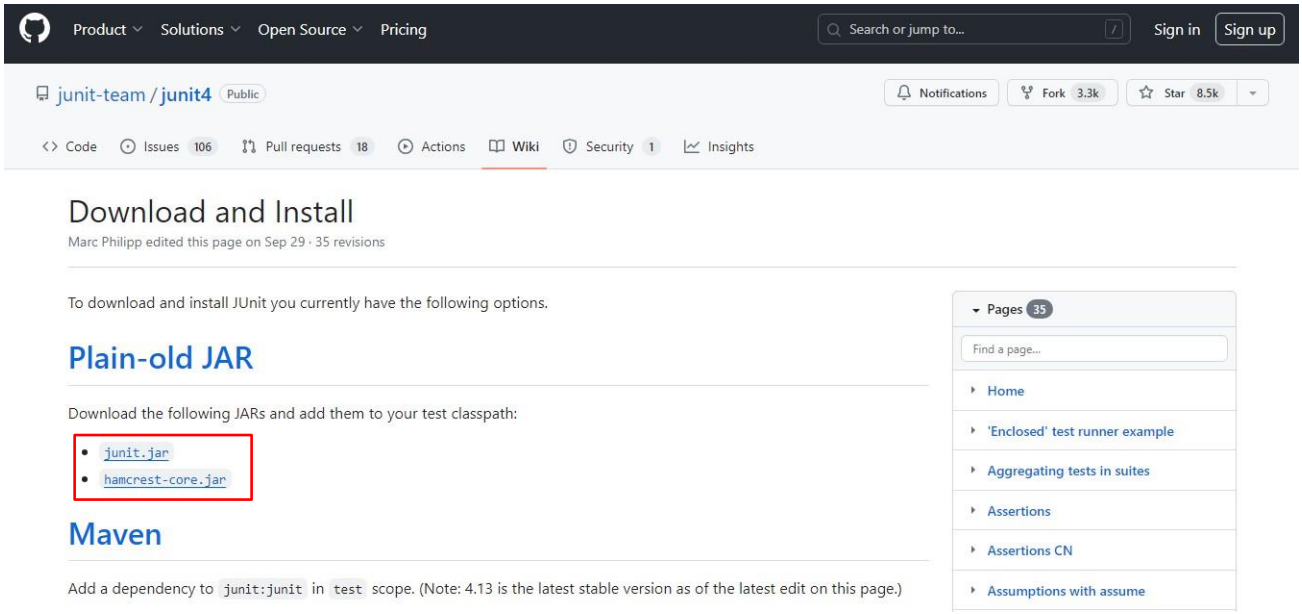
Step 1: Go web browser □ search for 'Junit 4' □ select the first link shown here.



Step 2: Click on 'Download & Install' option.



Step 3: Now click on 'junit.jar' file here.



Step 4: Then select the 'junit-4.13.2.jar' version & download it.

junit/junit/4.13.2

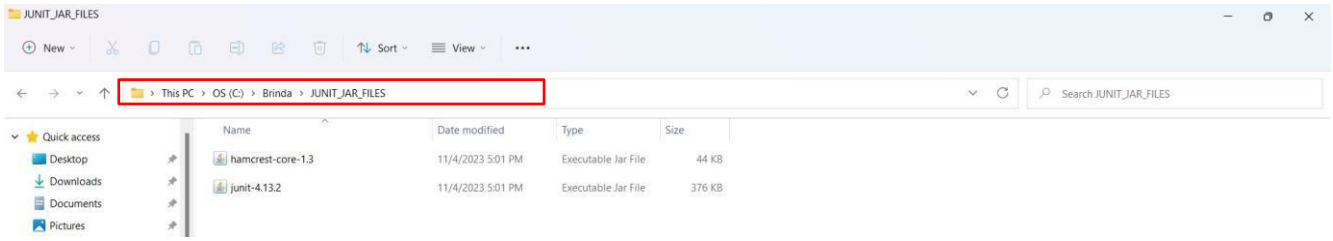
../		
junit-4.13.2-javadoc.jar	2021-02-13 16:31	1674580
junit-4.13.2-javadoc.jar.asc	2021-02-13 16:31	833
junit-4.13.2-javadoc.jar.asc.md5	2021-02-13 16:31	32
junit-4.13.2-javadoc.jar.asc.sha1	2021-02-13 16:31	40
junit-4.13.2-javadoc.jar.md5	2021-02-13 16:31	32
junit-4.13.2-javadoc.jar.sha1	2021-02-13 16:31	40
junit-4.13.2-sources.jar	2021-02-13 16:31	234540
junit-4.13.2-sources.jar.asc	2021-02-13 16:31	833
junit-4.13.2-sources.jar.asc.md5	2021-02-13 16:31	32
junit-4.13.2-sources.jar.asc.sha1	2021-02-13 16:31	40
junit-4.13.2-sources.jar.md5	2021-02-13 16:31	32
junit-4.13.2-sources.jar.sha1	2021-02-13 16:31	40
junit-4.13.2.jar	2021-02-13 16:31	384581
junit-4.13.2.jar.asc	2021-02-13 16:31	833
junit-4.13.2.jar.asc.md5	2021-02-13 16:31	32
junit-4.13.2.jar.asc.sha1	2021-02-13 16:31	40
junit-4.13.2.jar.md5	2021-02-13 16:31	32
junit-4.13.2.jar.sha1	2021-02-13 16:31	40
junit-4.13.2.pom	2021-02-13 16:31	27018
junit-4.13.2.pom.asc	2021-02-13 16:31	833
junit-4.13.2.pom.asc.md5	2021-02-13 16:31	32
junit-4.13.2.pom.asc.sha1	2021-02-13 16:31	40
junit-4.13.2.pom.md5	2021-02-13 16:31	32
junit-4.13.2.pom.sha1	2021-02-13 16:31	40

Step 5: Also select ‘hamcrest-core.jar’ file and download the ‘hamcrest-core-1.3.jar’ version here.

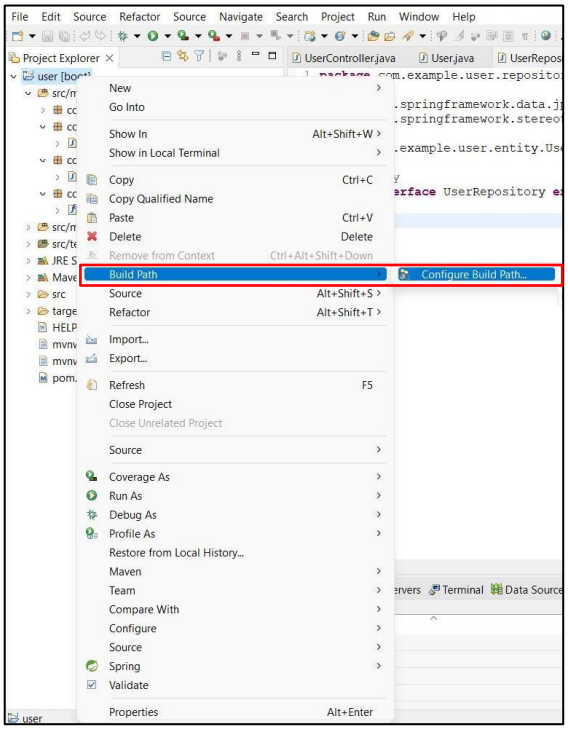
org/hamcrest/hamcrest-core/1.3

../		
hamcrest-core-1.3-javadoc.jar	2012-07-09 21:08	242519
hamcrest-core-1.3-javadoc.jar.asc	2012-07-09 21:08	499
hamcrest-core-1.3-javadoc.jar.asc.md5	2012-07-09 21:08	32
hamcrest-core-1.3-javadoc.jar.asc.sha1	2012-07-09 21:08	40
hamcrest-core-1.3-javadoc.jar.md5	2012-07-09 21:08	32
hamcrest-core-1.3-javadoc.jar.sha1	2012-07-09 21:08	40
hamcrest-core-1.3-sources.jar	2012-07-09 21:08	32624
hamcrest-core-1.3-sources.jar.asc	2012-07-09 21:08	499
hamcrest-core-1.3-sources.jar.asc.md5	2012-07-09 21:08	32
hamcrest-core-1.3-sources.jar.asc.sha1	2012-07-09 21:08	40
hamcrest-core-1.3-sources.jar.md5	2012-07-09 21:08	32
hamcrest-core-1.3-sources.jar.sha1	2012-07-09 21:08	40
hamcrest-core-1.3.jar	2012-07-09 21:08	45024
hamcrest-core-1.3.jar.asc	2012-07-09 21:08	499
hamcrest-core-1.3.jar.asc.md5	2012-07-09 21:08	32
hamcrest-core-1.3.jar.asc.sha1	2012-07-09 21:08	40
hamcrest-core-1.3.jar.md5	2012-07-09 21:08	32
hamcrest-core-1.3.jar.sha1	2012-07-09 21:08	40
hamcrest-core-1.3.pom	2012-07-09 21:08	766
hamcrest-core-1.3.pom.asc	2012-07-09 21:08	499
hamcrest-core-1.3.pom.asc.md5	2012-07-09 21:08	32
hamcrest-core-1.3.pom.asc.sha1	2012-07-09 21:08	40
hamcrest-core-1.3.pom.md5	2012-07-09 21:08	32
hamcrest-core-1.3.pom.sha1	2012-07-09 21:08	40

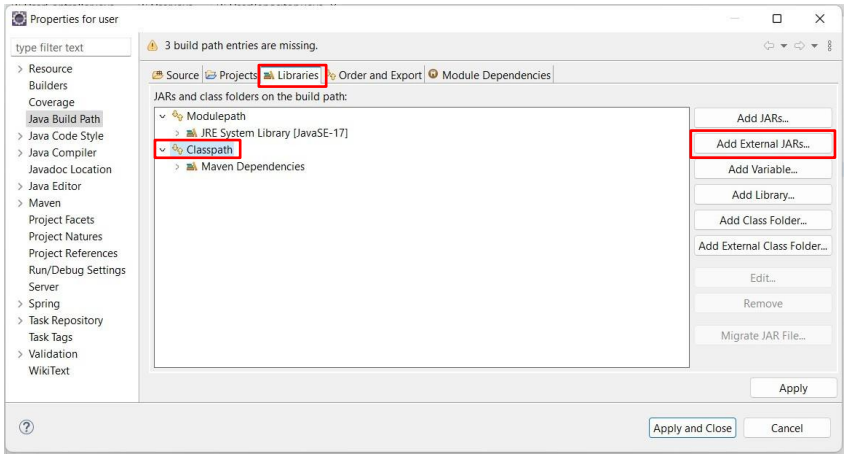
Step 6: Now select any drive & create a new folder within that folder, create a new folder named ‘JUNIT_JAR_FILES’ & copy the downloaded jar files & paste it here.



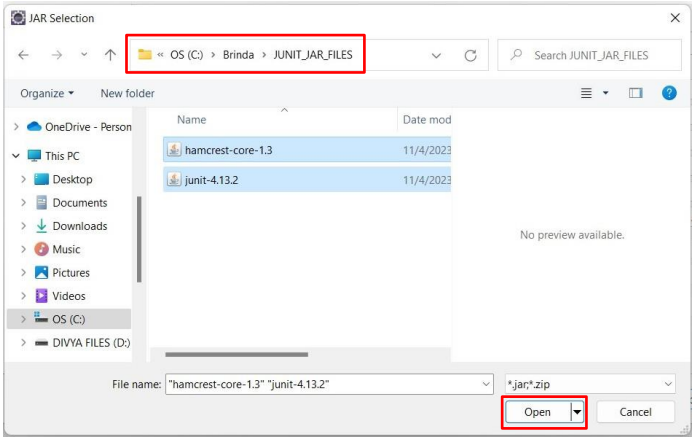
Step 7: Go to Eclipse □ user □ Build Path □ Configure Build Path here.



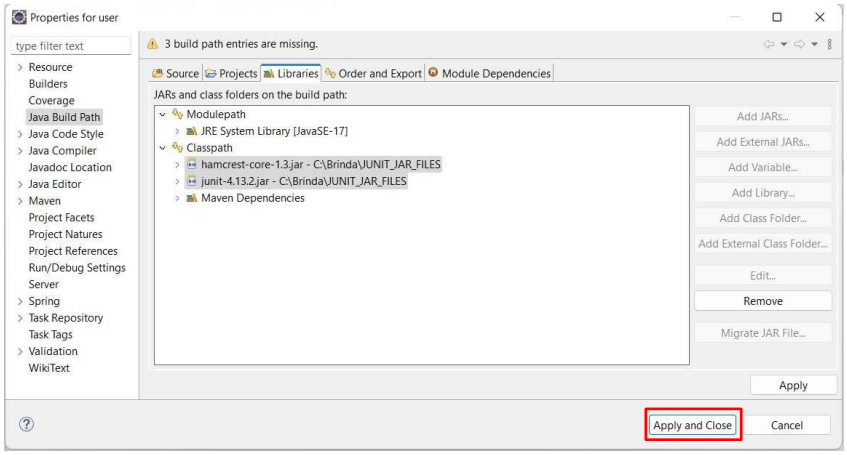
Step 8: Go to Libraries Classpath Add External JARs.. here.



Step 9: Select the two jar files downloaded here & click on ‘Open’.



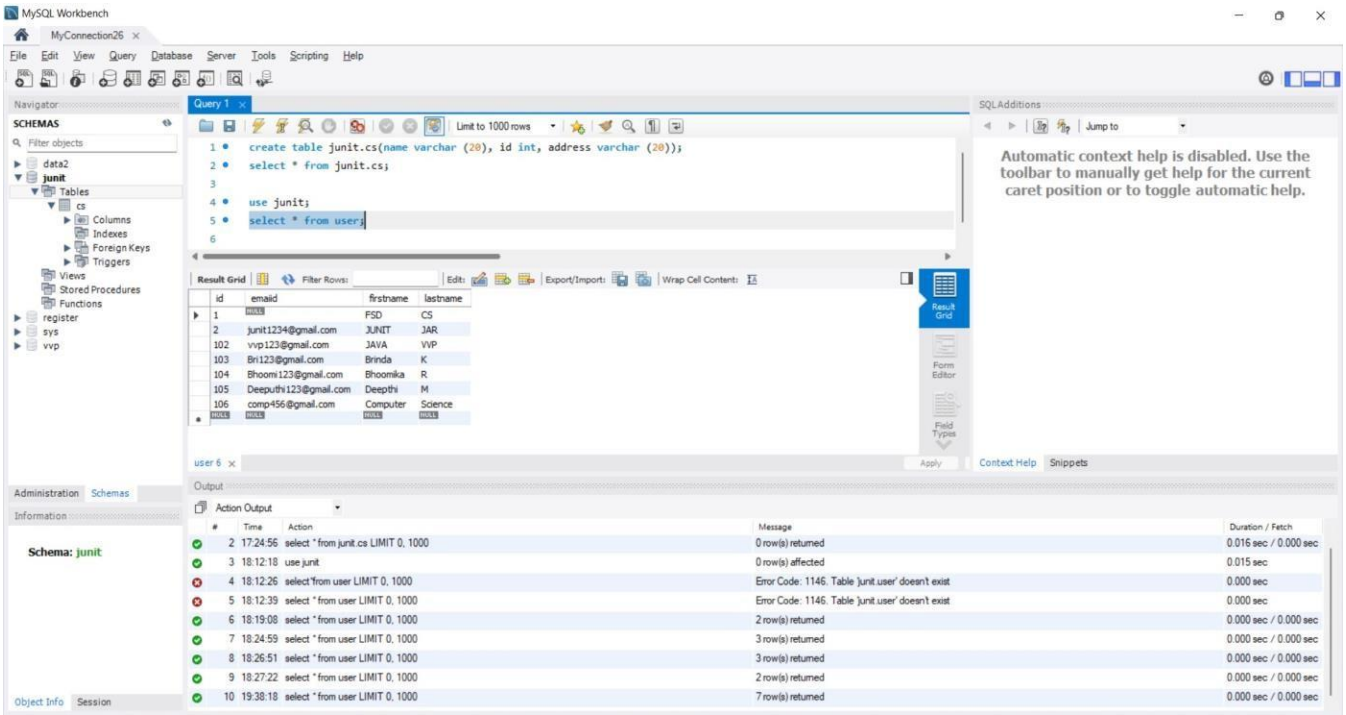
Step 10: Now click on ‘Apply and Close’ here.



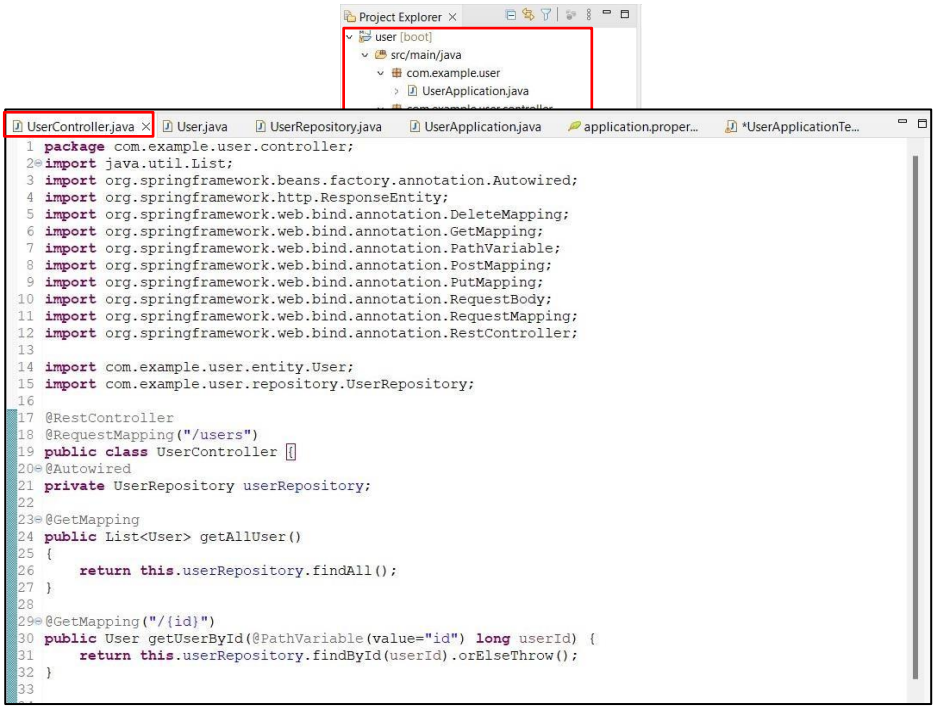
Step 11: Now create a new database in MySQL Command Line Client.

```
mysql> create database junit;
Query OK, 1 row affected (0.02 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| data2              |
| information_schema |
| junit              |
| mysql              |
| performance_schema |
| register           |
| sys                |
| vvp                |
+--------------------+
8 rows in set (0.00 sec)
```

Step 12: Go to MySQL Workbench and create few more tables using Postman as well as with CRUD options.



Step 13: Follow the same procedures followed in CRUD operations and add code to the classes created.



Step 14: Now add code to ‘UseController.java’ file here.

Step 15: Now add code to ‘User.java’ file here.

```java
package com.example.user.entity;

import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
import jakarta.persistence.Table;

@Entity
@Table(name="user")
public class User {
    @Id
    @GeneratedValue(strategy=GenerationType.AUTO)
    private Long id;
    private String firstname;
    private String lastname;
    private String emaiID;

    public User() {

    }
    public Long getId() {
        return id;
    }
    public void setId(Long id) {
        this.id = id;
    }
    public String getFirstname() {
        return firstname;
    }
    public void setFirstname(String firstname) {
        this.firstname = firstname;
    }
    public String getLastname() {
        return lastname;
    }
    public void setLastname(String lastname) {
        this.lastname = lastname;
    }
    public String getEmaiID() {
        return emaiID;
    }
    public void setEmaiID(String emaiID) {
        this.emaiID = emaiID;
    }
    public User(Long id, String firstname, String lasttname, String emaiID) {
        super();
        this.id = id;
        this.firstname = firstname;
        this.lastname = lasttname;
        this.emaiID = emaiID;
    }

}
```

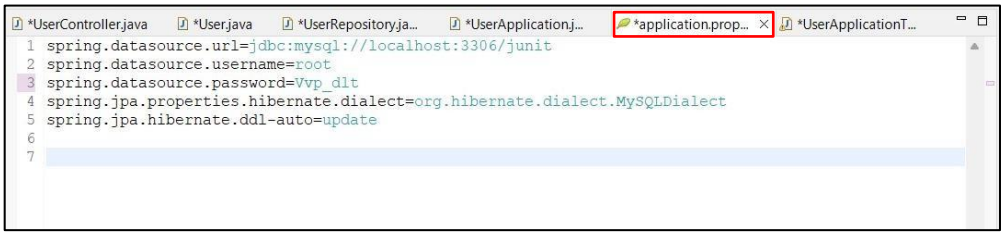
Step 16: Now add code to ‘UseRepository.java’ file here.

```java
package com.example.user.repository;

import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

import com.example.user.entity.User;

@Repository
public interface UserRepository extends JpaRepository<User,Long>
{
}
```

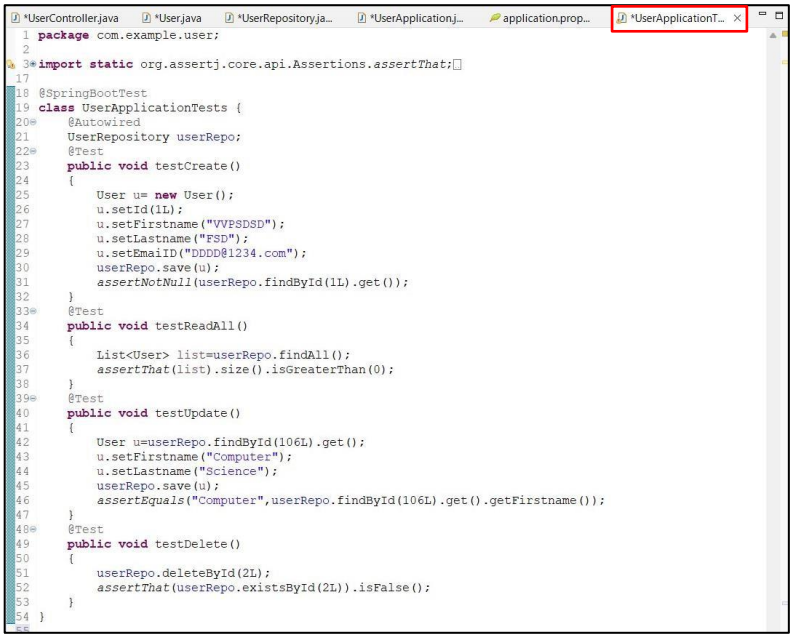
Step 17: Now add code to ‘UseApplication.java’ file where main method class present here.

```java
package com.example.user;

import org.springframework.boot.SpringApplication;

@SpringBootApplication
public class UserApplication {

    public static void main(String[] args) {
        SpringApplication.run(UserApplication.class, args);
    }

}
```

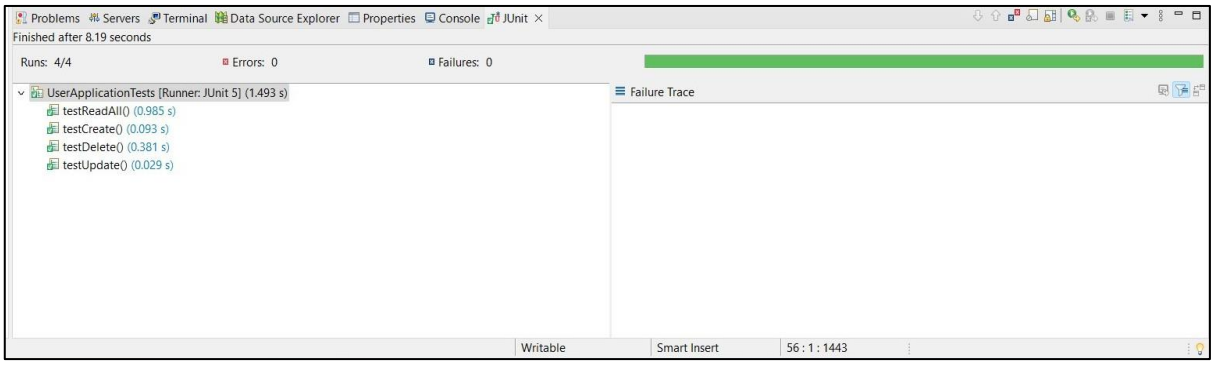
Step 18: Now add code to ‘application.properties’ file here.



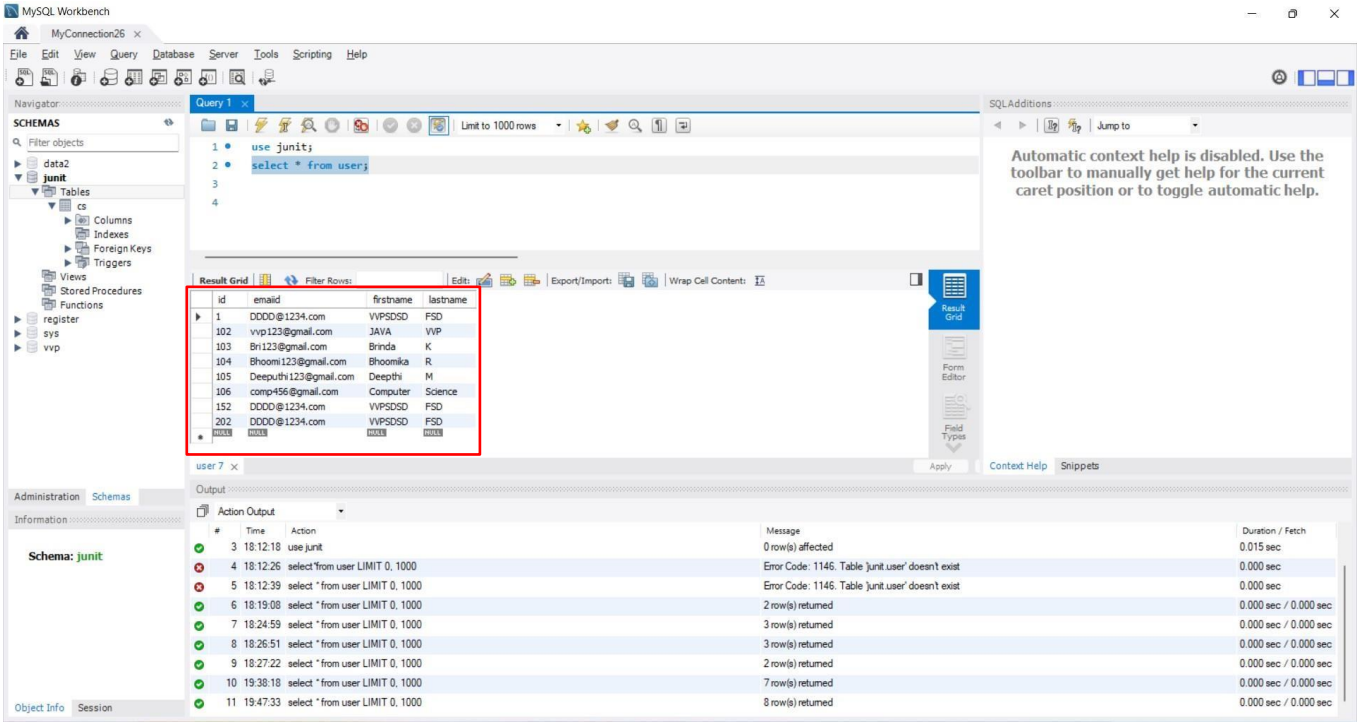
Step 19: Go to ‘src/test/java’ folder in the Project explorer section & add code junit test code to ‘UserApplicationTest.java’ class file as shown here.



Step 20: Now right click on the ‘UserApplicationTest.java’ file □ Run As □ 2Junit Test.

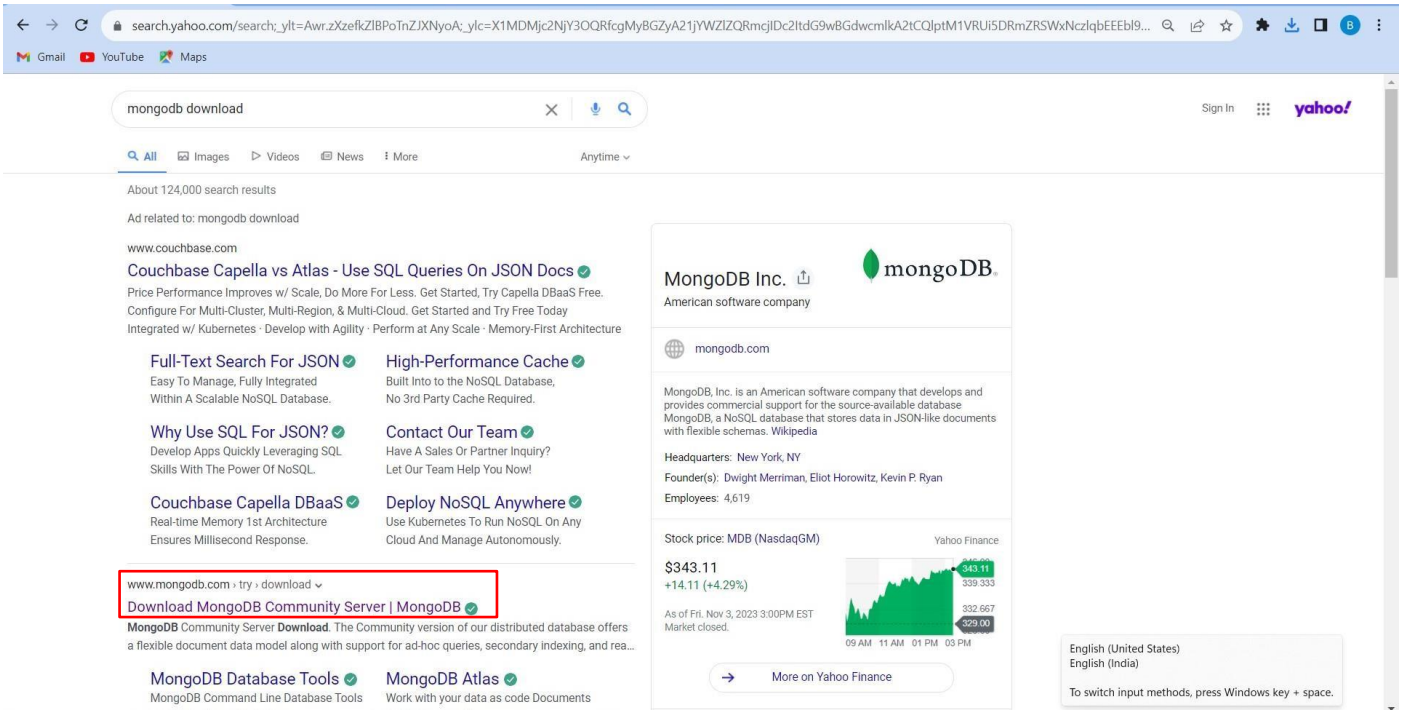


Step 21: The above test condition is successfully working in MySQL Workbench as shown below.

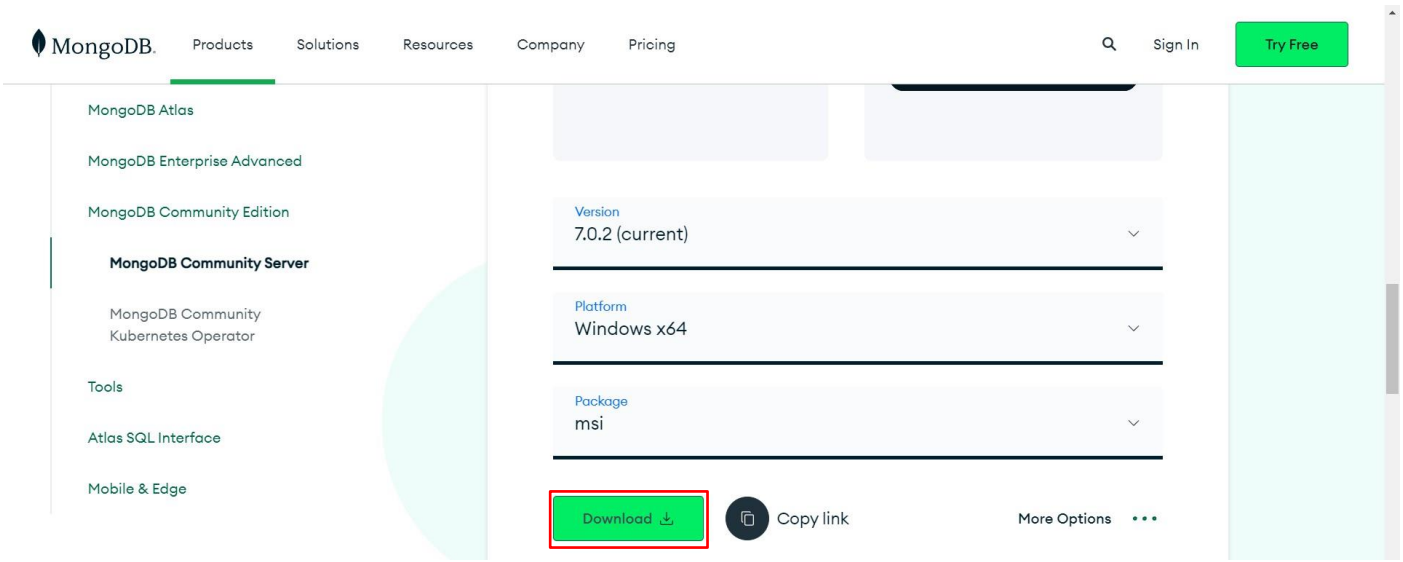


3. **Installation of Mongodb, mongosh, compass.**

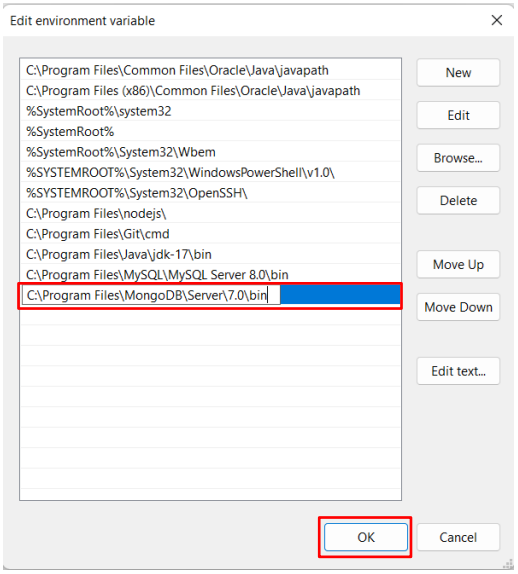
Step 1: Go to web browser Search for 'mongodb download' click on the link as shown below.



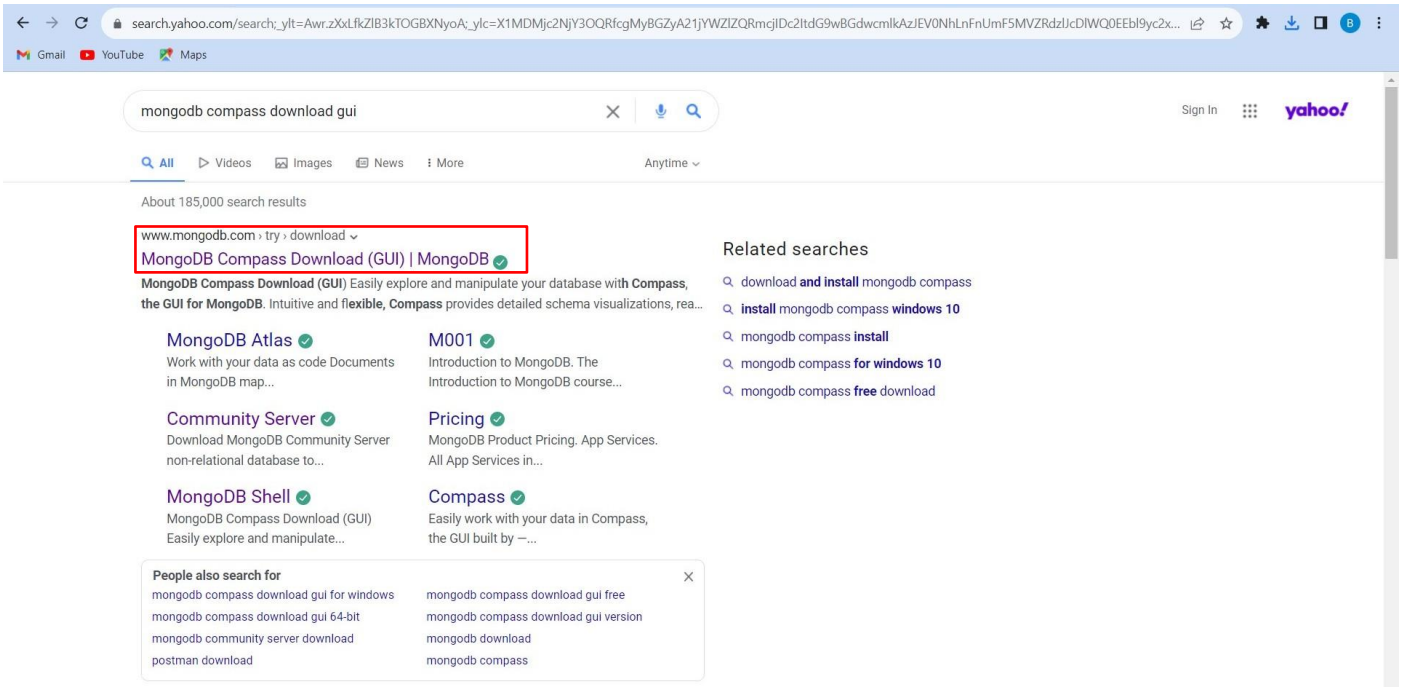
Step 2: Click on 'Download' here.



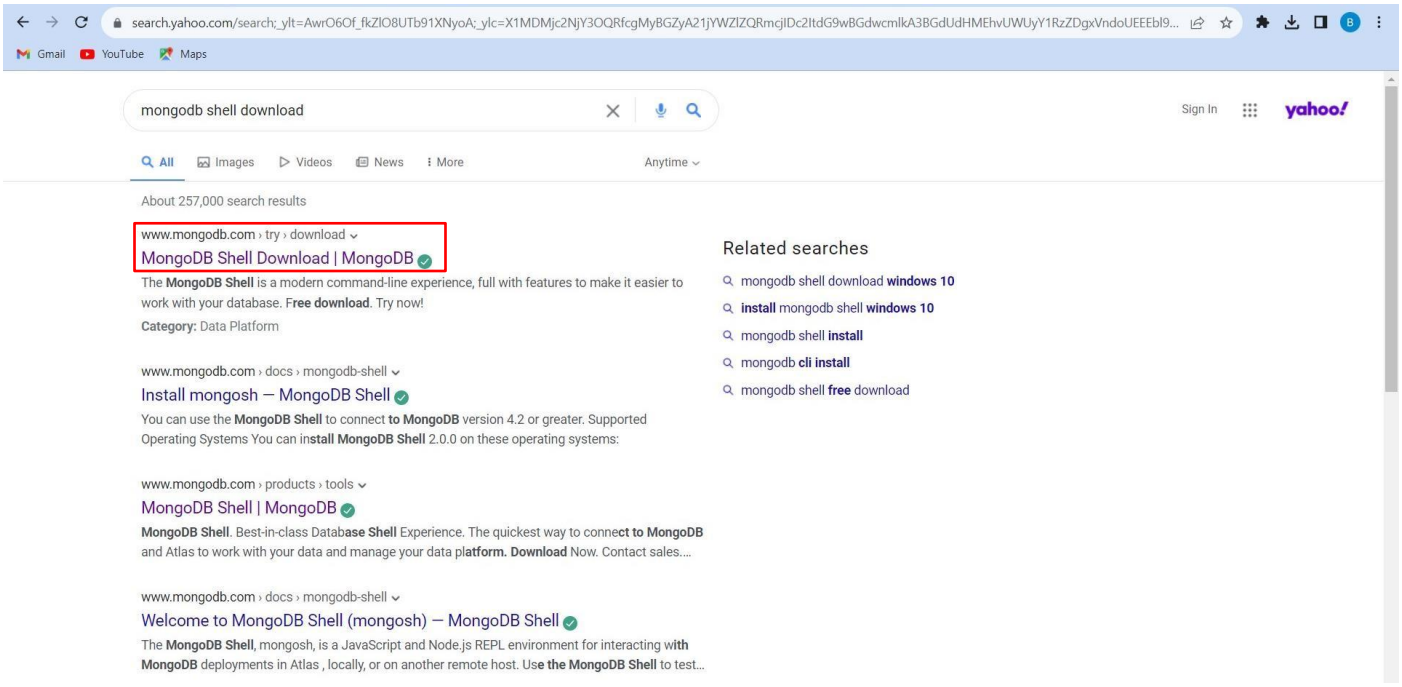
Step 2: Now add ‘Environment variables’ for the downloaded file.



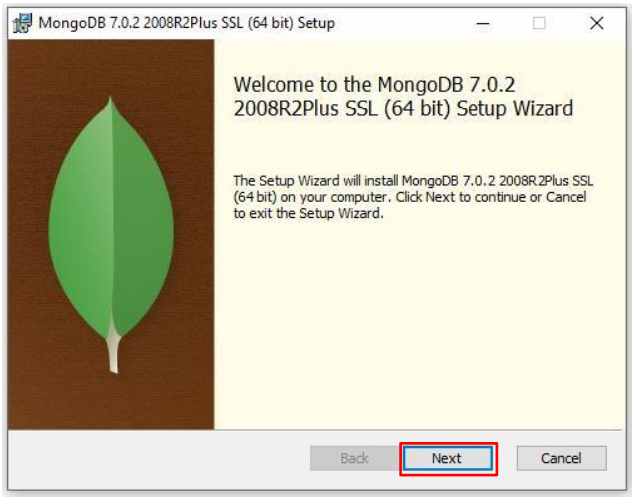
Step 3: Go to web browser search for ‘mongodb compass download (GUI)’ click on the first link here & download the MongoDB Compass.



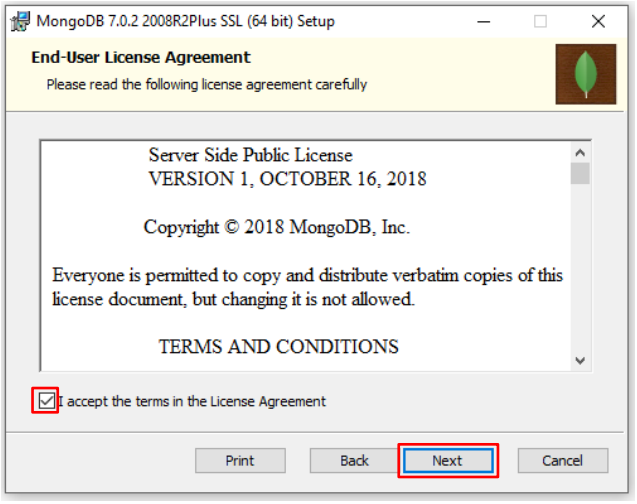
Step 4: Go to web browser search for ‘mongodb shell download’ click on the first link here & download the MongoDB Shell.



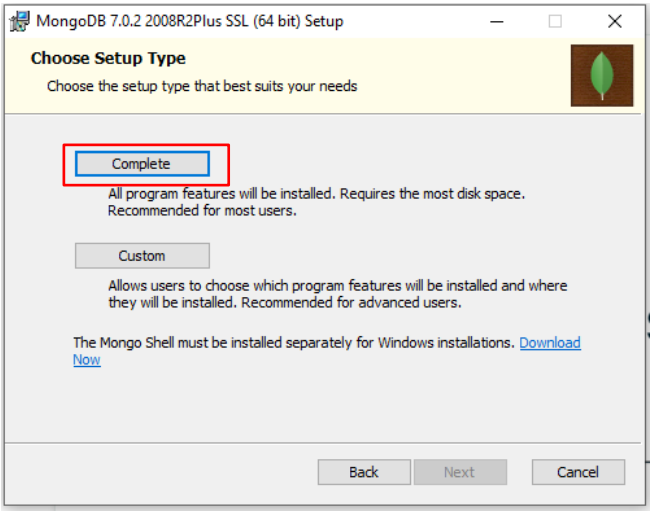
Step 5: Follow the ongoing steps to install mongodb compass, click on ‘Next’.



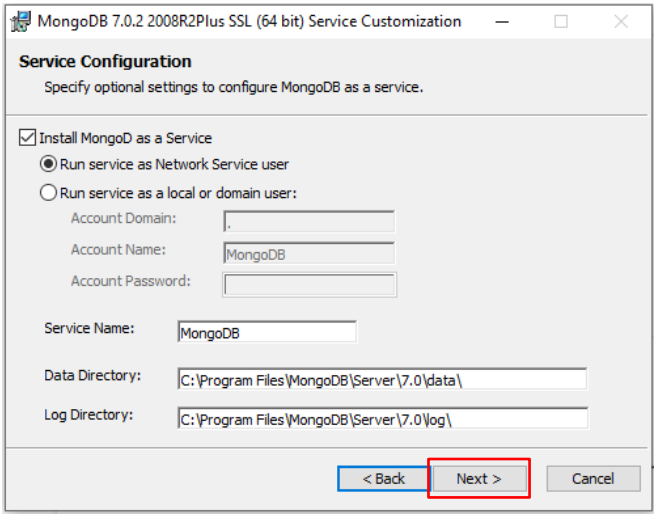
Step 6: Enable the checkbox & click on ‘Next’.



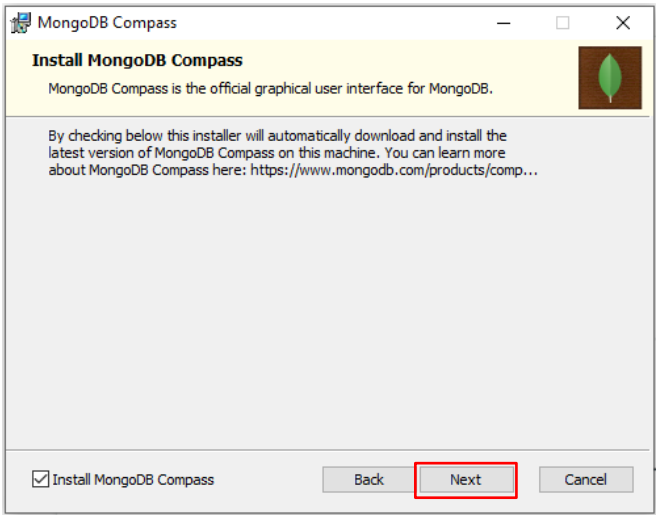
Step 7: Click on ‘complete’ here.



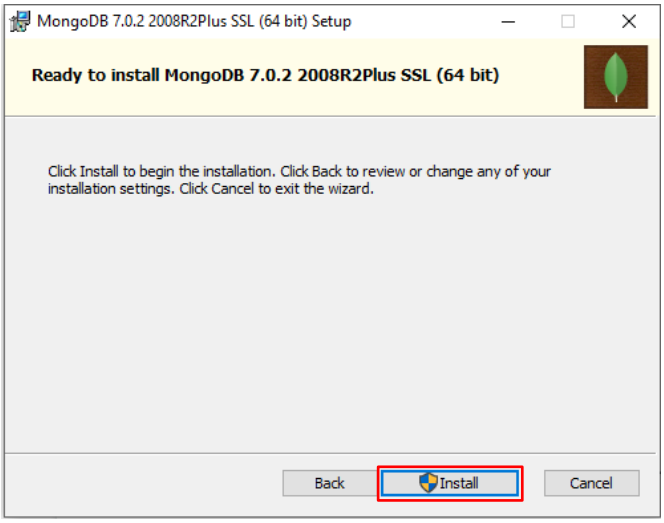
Step 8: Just click on ‘Next’ here.



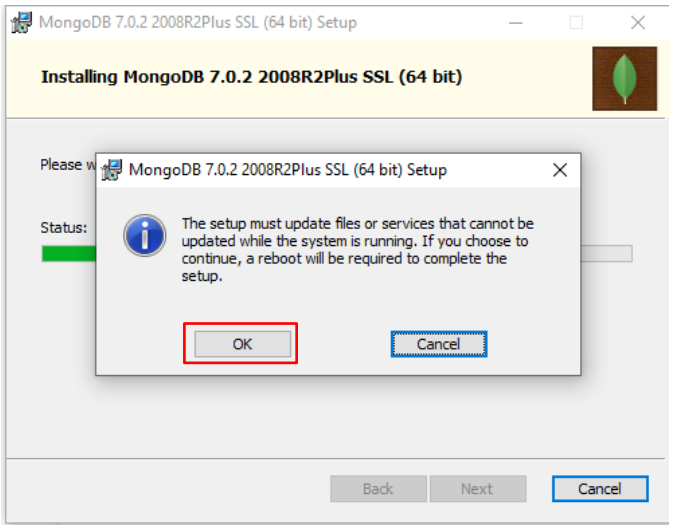
Step 9: Click on ‘Next’.



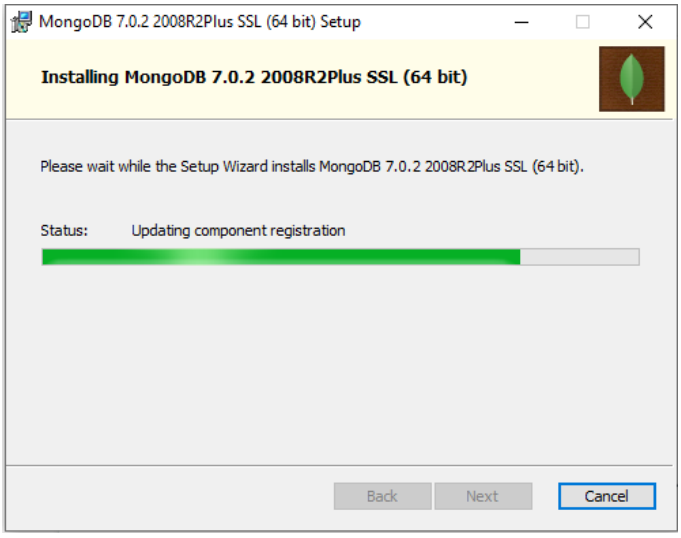
Step 10: Now click on ‘Install’.



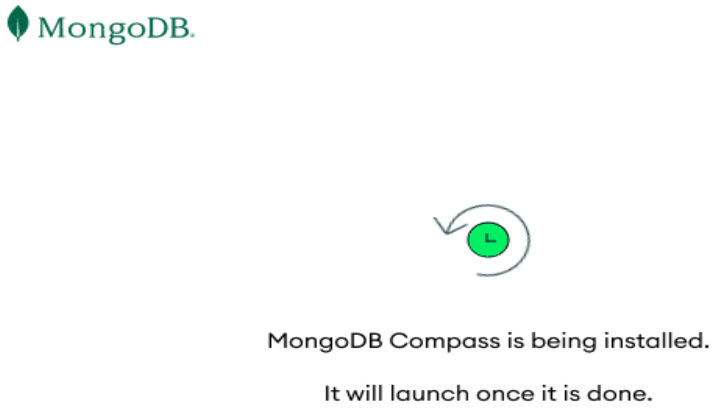
Step 11: Click on ‘OK’ here.



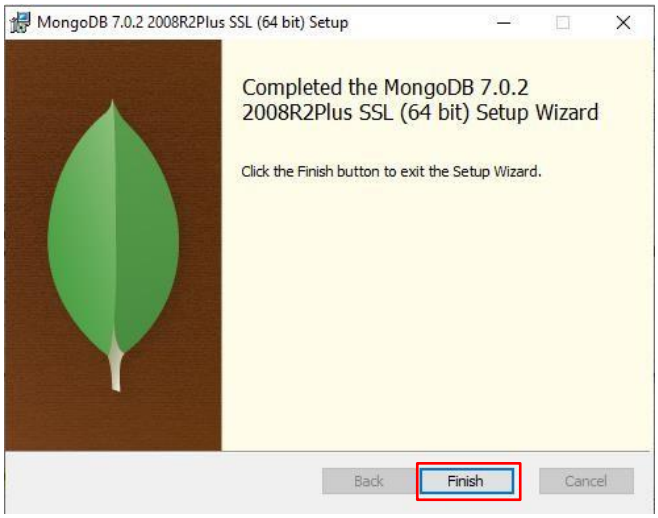
Step 12: Wait until the installation is complete.



Step 13: MongoDB Compass is ready to launch.



Step 14: Then click on ‘Finish’ here.



Step 15: Once all installation done, To verify mongod ,Open cmd prompt –execute (mongod --version and mongosh --version)

- For mongodb server –msi installer version 7 and above
- For mongosh-extract zip file in a secure folder
- For Mongodb compass run the application(.exe) file.

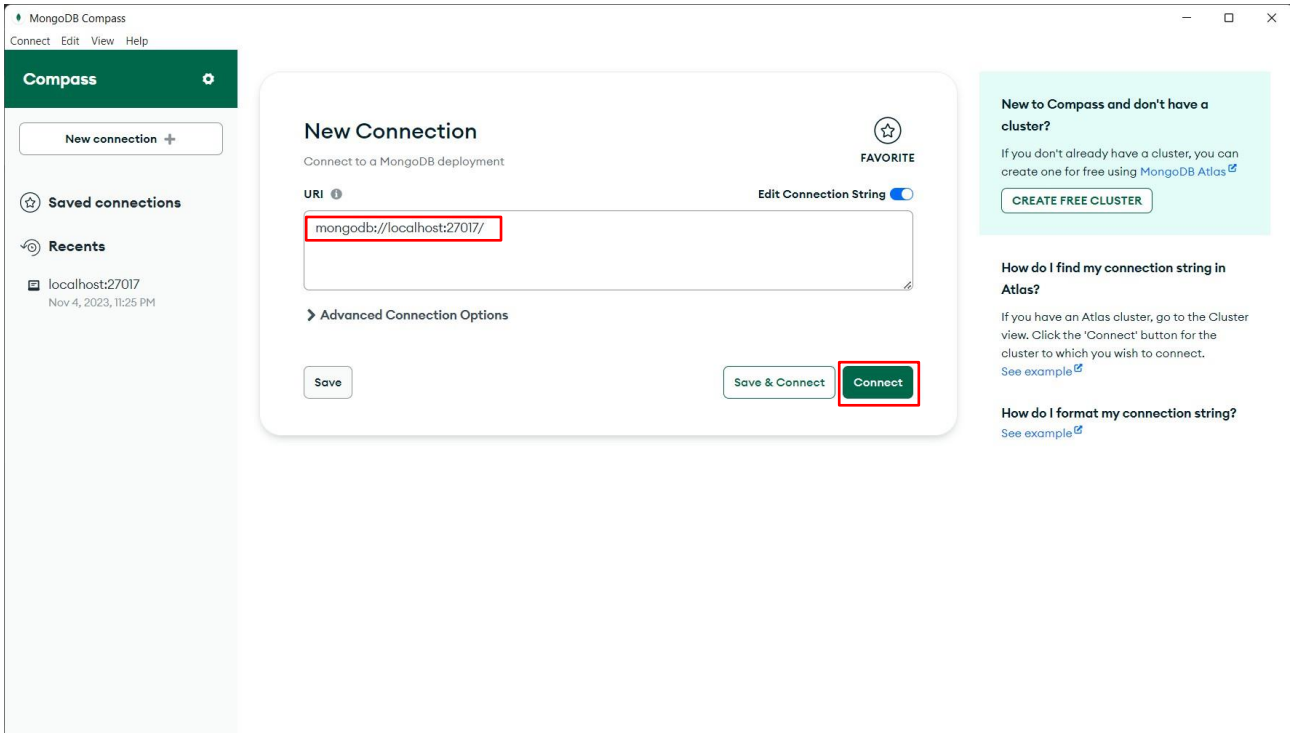
Step 16: Next in search bar type ‘mongoshell’ ->click and open and press enter

Next to create new database ‘use fsd’ as shown below and ‘show dbs’ shows all the databases created.

Remember: In MongoDB, a database is not actually created until it gets content!

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
Please enter a MongoDB connection string (Default: mongodb://localhost/):

Current Mongosh Log ID: 654684cfa96eef8170f34808
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.0.2
Using MongoDB:          7.0.2
Using Mongosh:          2.0.2

For mongosh info see: https://docs.mongodb.com/mongodb-shell/


To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

------
   The server generated these startup warnings when booting
   2023-11-04T23:16:27.845+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
------

test> show dbs
admin   40.00 KiB
config  12.00 KiB
local   40.00 KiB
test> use fsd
switched to db fsd
fsd> db.createCollection("Student")
{ ok: 1 }
fsd> show dbs
admin   40.00 KiB
config  12.00 KiB
fsd      8.00 KiB
local   40.00 KiB
fsd> use test
switched to db test
test> show dbs
admin   40.00 KiB
config  12.00 KiB
fsd      8.00 KiB
local   40.00 KiB
test>
```

Step 17: Now search for ‘mongoDB’ compass in start menu and open it. Now click on ‘Connect’ to connect with the localhost here.



Step 18: Select the database created & click on ‘ADD DATA’ to insert data.