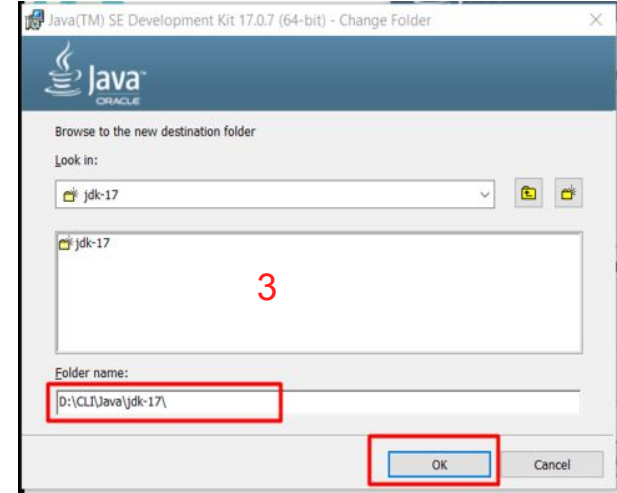
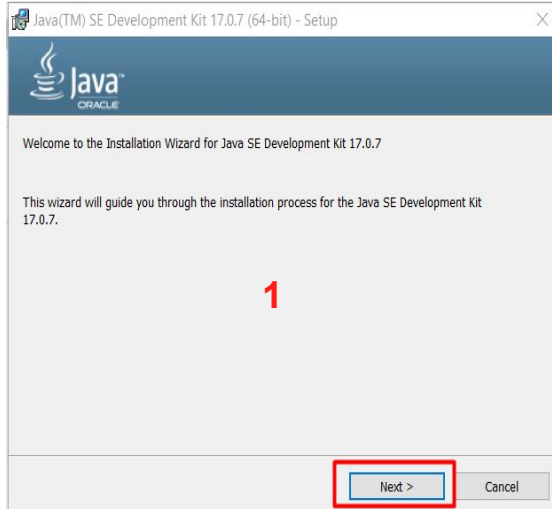


JAVA Spring Boot Configuration & Running Procedure

Prepared By
Zawadul Kawum
Senior Software Engineer
Adventure Dhaka Ltd.

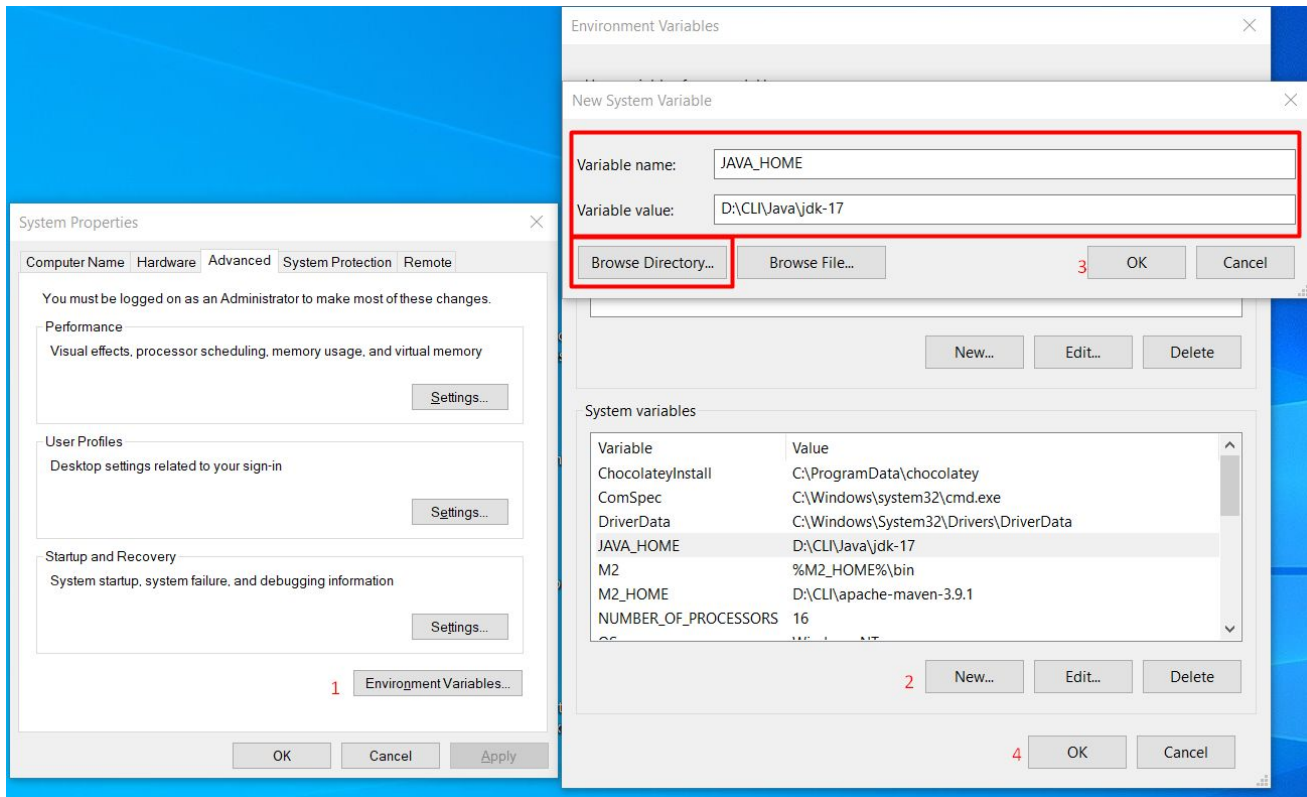
Install and Configure JDK

- **Step 01:** Download JDK from this link https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.exe
- **Step 02:** Install JDK



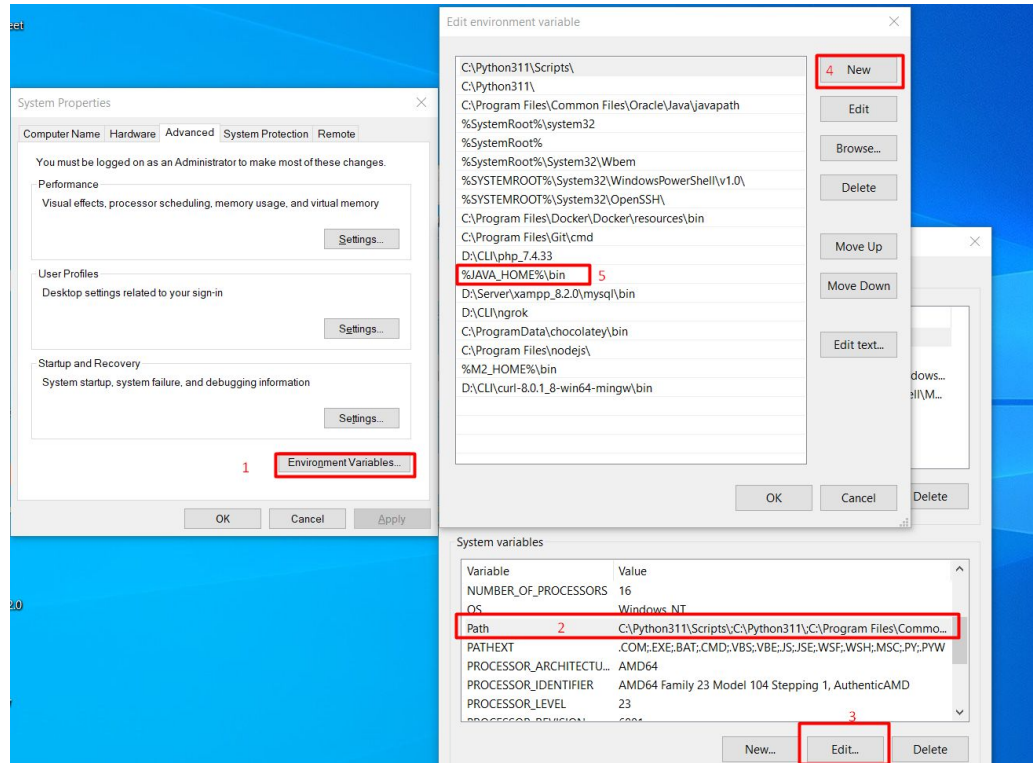
Install and Configure JAVA

- Step 03: Set **JAVA_HOME** in Environmental Variable for Java (Eg: Variable Name: **JAVA_HOME**, Value: **D:\CLI\Java\jdk-17**)



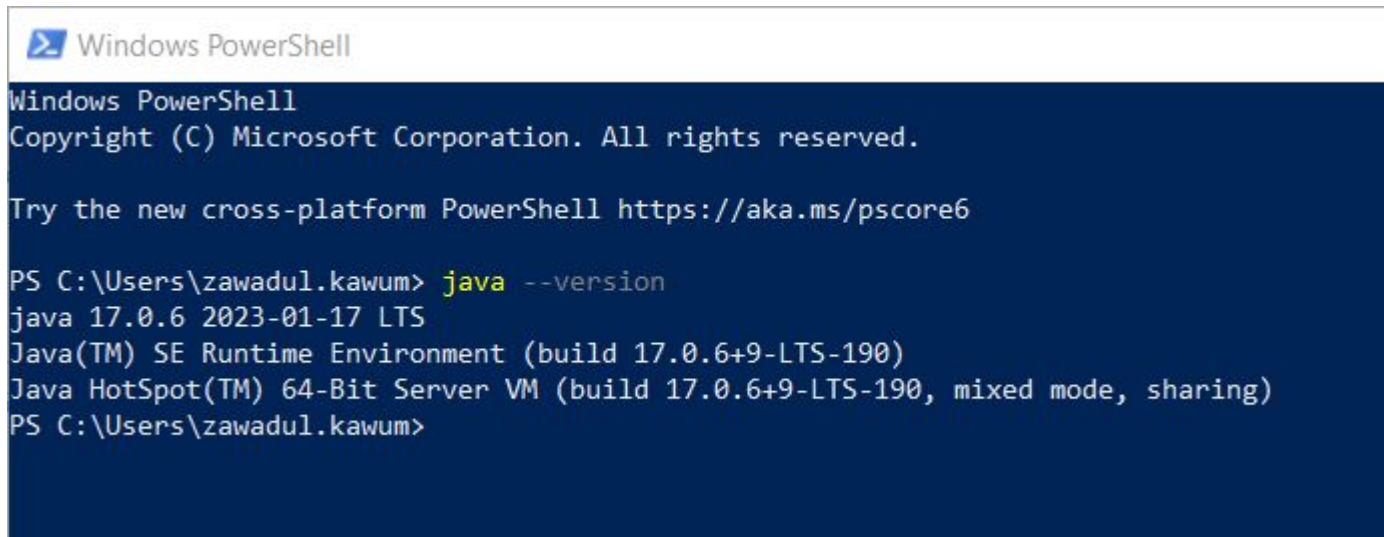
Install and Configure JAVA

- Step 04: Configure Environmental Variable for Java (eg: %JAVA_HOME%\bin)



Install and Configure JAVA

- **Step 05:** Now go to Powershell / CMD and write **java --version** command. If you got response like below screenshot then your JAVA installation have been done successfully.



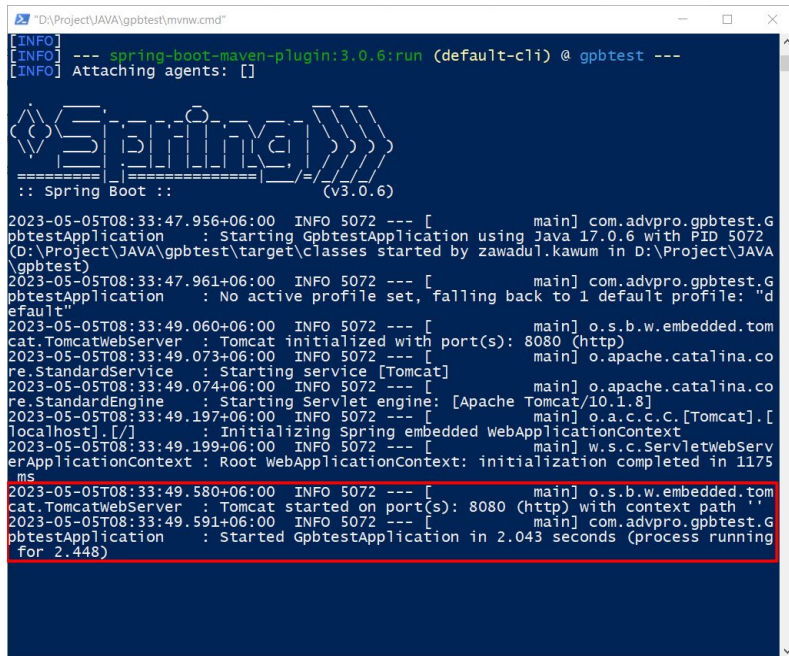
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\zawadul.kawum> java --version
java 17.0.6 2023-01-17 LTS
Java(TM) SE Runtime Environment (build 17.0.6+9-LTS-190)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.6+9-LTS-190, mixed mode, sharing)
PS C:\Users\zawadul.kawum>
```

Run Spring Boot Application

- Now unzip the spring boot application, go to project folder and open **PowerShell**. (You can run it by vscode terminal). Run following command **./mvnw spring-boot:run**. If your **PowerShell** or **VSCode** Terminal is showing like below screenshot then your spring boot application have been started successfully.



```
"D:\Project\JAVA\gpbtest\mvnw.cmd"
[INFO] --- spring-boot-maven-plugin:3.0.6:run (default-cli) @ gpbtest ---
[INFO] Attaching agents: []

:: Spring Boot :: (v3.0.6)

2023-05-05T08:33:47.956+06:00 INFO 5072 --- [main] com.advpro.gpbtest.G
pbtestApplication : Starting GpbtestApplication using Java 17.0.6 with PID 5072
(D:\Project\JAVA\gpbtest\target\classes started by zawadul.kawum in D:\Project\JAVA
\gpbtest)
2023-05-05T08:33:47.961+06:00 INFO 5072 --- [main] com.advpro.gpbtest.G
pbtestApplication : No active profile set, falling back to 1 default profile: "d
efault"
2023-05-05T08:33:49.060+06:00 INFO 5072 --- [main] o.s.b.w.embedded.tom
cat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2023-05-05T08:33:49.073+06:00 INFO 5072 --- [main] o.apache.catalina.co
re.StandardService : Starting service [Tomcat]
2023-05-05T08:33:49.074+06:00 INFO 5072 --- [main] o.apache.catalina.co
re.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.8]
2023-05-05T08:33:49.197+06:00 INFO 5072 --- [main] o.a.c.c.C.[Tomcat].[
localhost].[/] : Initializing Spring embedded webApplicationContext
2023-05-05T08:33:49.199+06:00 INFO 5072 --- [main] w.s.c.ServletWebServ
erApplicationContext : Root WebApplicationContext: initialization completed in 1175
ms
2023-05-05T08:33:49.580+06:00 INFO 5072 --- [main] o.s.b.w.embedded.tom
cat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2023-05-05T08:33:49.591+06:00 INFO 5072 --- [main] com.advpro.gpbtest.G
pbtestApplication : Started GpbtestApplication in 2.043 seconds (process running
for 2.448)
```

Install and Configure JAVA Spring Boot

- Go to <https://start.spring.io/> and put necessary information by below screenshot configuration and click **GENERATE** button for download spring boot file. Make sure you add **Dependencies**

Language

☒ Java ☐ Kotlin ☐ Groovy

Boot

☐ (SNAPSHOT) ☐ 3.1.0 (RC2) ☐ 3.1.0 (M2)

☐ (SNAPSHOT) ☒ 3.0.6 ☐ 2.7.12 (SNAPSHOT) ☐ 2.7.11

Metadata

Group

Artifact

Name

Description

Package name

Packaging

☒ Jar ☐ War

Dependencies

2 ADD DEPENDENCIES... CTRL + B

Spring Web

WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

Group: com.advpro

Artifact: gpbtest

Name: gpbtest

Description: Test Project for Global Payment Based

Package Name: com.advpro.gpbtest

Packaging: Jar

Java: 17

If you forget to add Dependencies

If you forget to add dependencies and something like below screenshot

```
2023-05-05T11:47:56.200+06:00 INFO 4212 --- [          main] com.advpro.gbptest.GbptestApplication : Starting Gbpte
stApplication using Java 17.0.7 with PID 4212 (D:\Work-Project\gbptest\target\classes started by hasibul.hasan in D:\Wor
k-Project\gbptest)
2023-05-05T11:47:56.204+06:00 INFO 4212 --- [          main] com.advpro.gbptest.GbptestApplication : No active prof
ile set, falling back to 1 default profile: "default"
2023-05-05T11:47:56.609+06:00 INFO 4212 --- [          main] com.advpro.gbptest.GbptestApplication : Started Gbptes
tApplication in 0.702 seconds (process running for 1.003)
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2.481 s
[INFO] Finished at: 2023-05-05T11:47:56+06:00
[INFO] -----
```

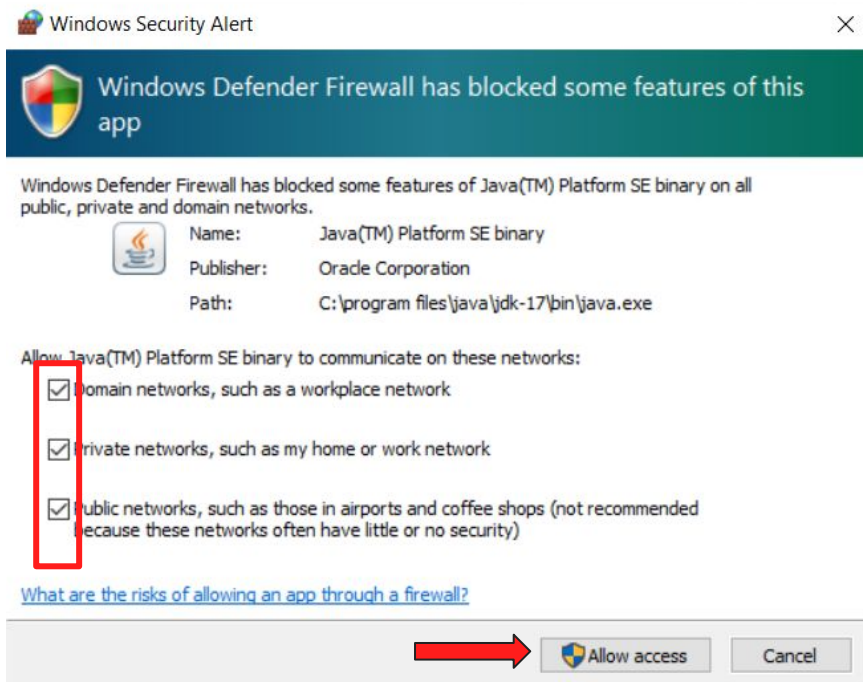
1. Open pom.xml
2. Change `spring-boot-starter` to `spring-boot-starter-web`
3. Then run `.\mvnw spring-boot:run` again

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>

  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>
    <scope>test</scope>
  </dependency>
</dependencies>
```

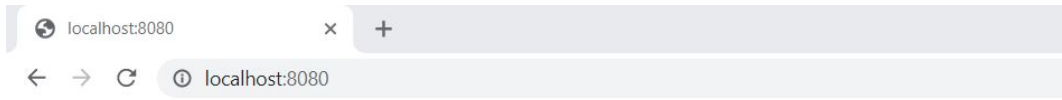

Run Spring Boot Application

- You may get **Windows Security Alert** dialogue for first time when you run spring boot command. Make sure to check all network and Click Allow Access



Check Spring Boot Running Application

- Now Go to browser and type <http://localhost:8080/>. If you see your browser response like below screenshot then your Spring Boot Application is running successfully.



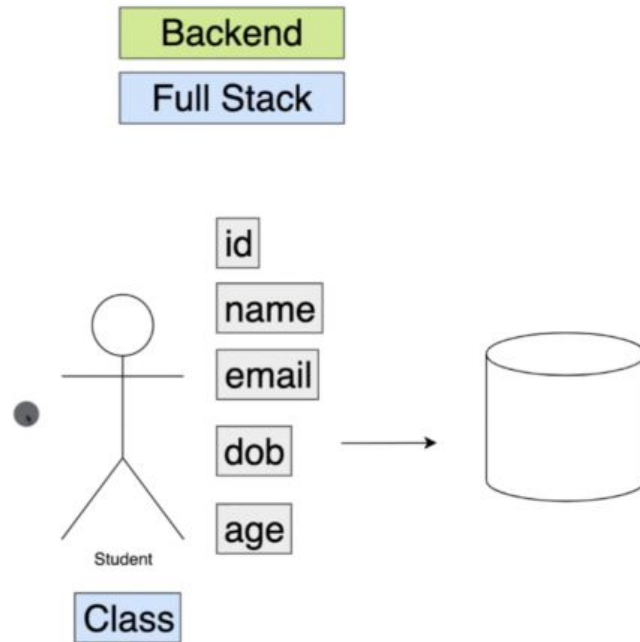
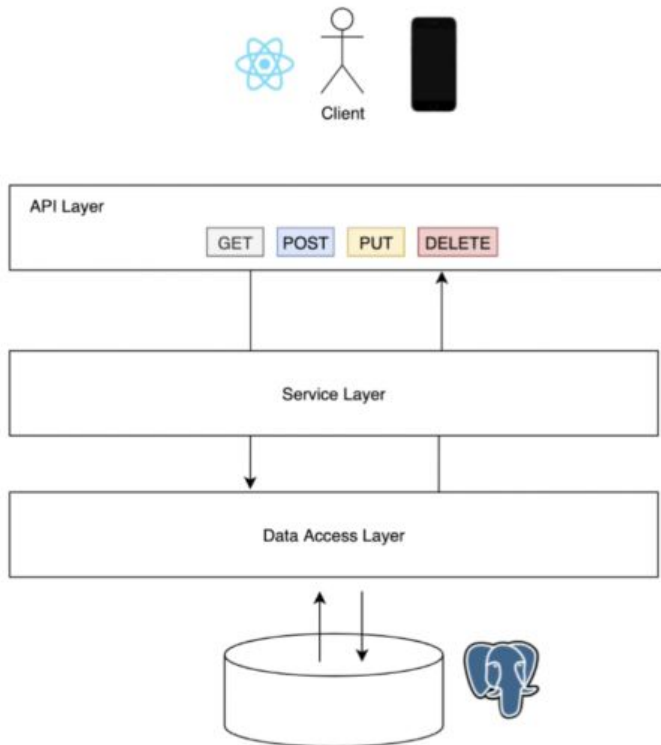
Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Fri May 05 08:39:24 BDT 2023

There was an unexpected error (type=Not Found, status=404).

Spring Boot Architecture



Database Creation & VS Code Configuration

- Install and start XAMPP serve
- Changing the port number for apache is optional
- Go to phpmyadmin and create database named **gpb**
- Download necessary extension pack for Java in VS Code
 - Extension Pack for Java (vscjava.vscode-java-pack)
<https://marketplace.visualstudio.com/items?vscjava.vscode-java-pack>
 - Spring Boot Extension Pack (vmware.vscode-boot-dev-pack)
<https://marketplace.visualstudio.com/items?itemName=vmware.vscode-boot-dev-pack>
 - Maven for Java (vscjava.vscode-maven)
<https://marketplace.visualstudio.com/items?vscjava.vscode-maven>

Add Maven Dependency

- Open Project in vscode and edit pom.xml file. Go to <dependencies> section. Add the dependencies by following.

```
pom.xml
11 <groupId>com.advpoc</groupId>
12 <artifactId>gbptest</artifactId>
13 <version>0.0.1-SNAPSHOT</version>
14 <name>gbptest</name>
15 <description>Test Project for Global Payment Base</description>
16 <properties>
17   <java.version>17</java.version>
18 </properties>
19 <dependencies>
20   <dependency>
21     <groupId>org.springframework.boot</groupId>
22     <artifactId>spring-boot-starter-web</artifactId>
23   </dependency>
24   <dependency>
25     <groupId>org.springframework.boot</groupId>
26     <artifactId>spring-boot-starter-test</artifactId>
27     <scope>test</scope>
28   </dependency>
29   <dependency>
30     <groupId>org.springframework.boot</groupId>
31     <artifactId>spring-boot-devtools</artifactId>
32     <optional>true</optional>
33   </dependency>
34   <dependency>
35     <groupId>org.springframework.boot</groupId>
36     <artifactId>spring-boot-starter-data-jpa</artifactId>
37   </dependency>
38   <dependency>
39     <groupId>com.mysql</groupId>
40     <artifactId>mysql-connector-j</artifactId>
41     <version>8.0.33</version>
42   </dependency>
43 </dependencies>
```

Change Web service to Build Service

- Inside pom.xml file. Go to <dependencies> section. change the spring-boot-starter-web artifactid

```
<artifactId>spring-boot-starter-web</artifactId>
```

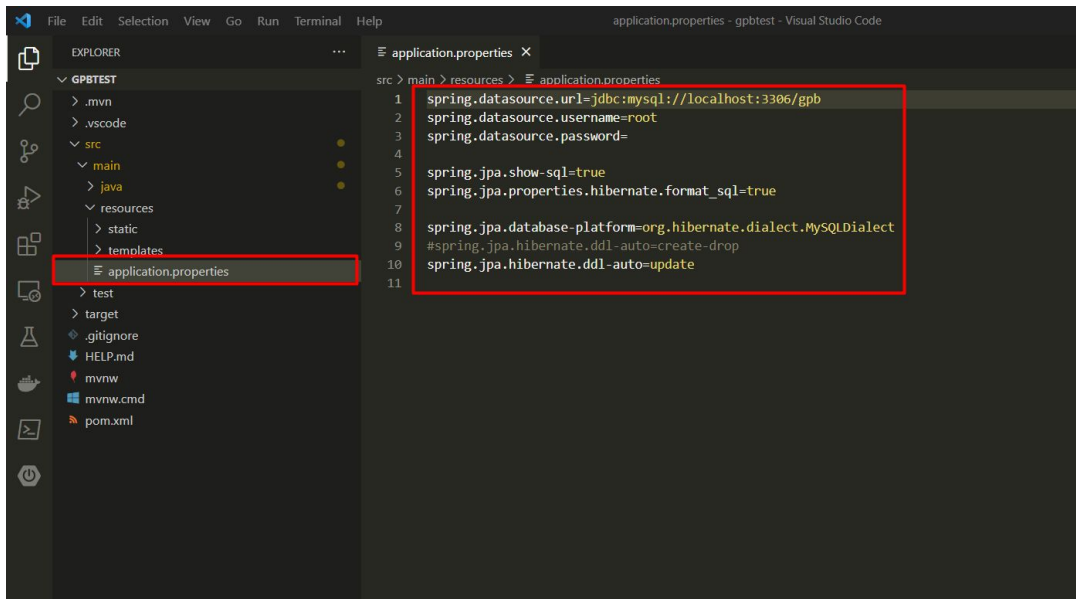
to

```
<artifactId>spring-boot-starter</artifactId>
```

- This changes will only build the application. Not running it. We will change is later

Configure Database

- Open application.properties and configure like below screenshot



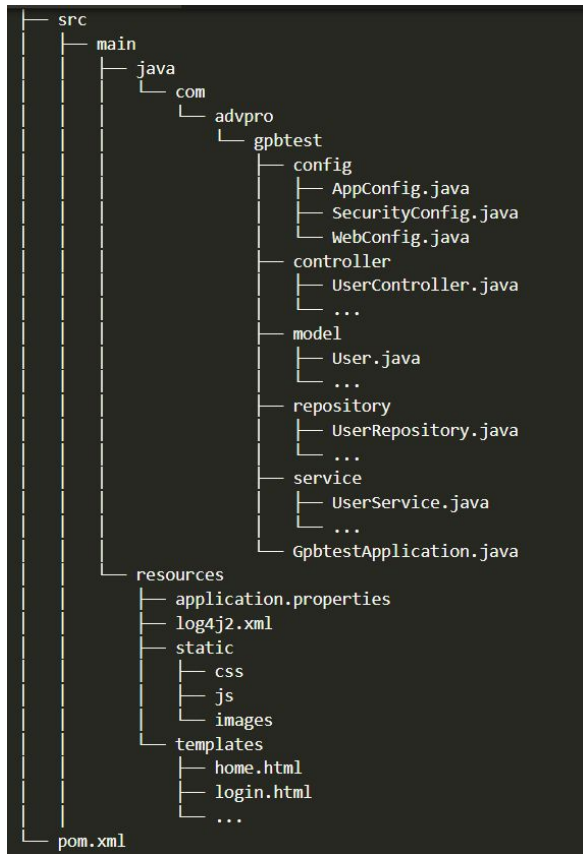
spring.datasource.url=jdbc:mysql://localhost:3306/gpb
spring.datasource.username=root
spring.datasource.password=

spring.jpa.show-sql=true
spring.jpa.properties.hibernate.format_sql=true

spring.jpa.database-platform=org.hibernate.dialect.MySQLDialect
#spring.jpa.hibernate.ddl-auto=create-drop
spring.jpa.hibernate.ddl-auto=update

- Run `./mvnw spring-boot:run` again to build the project for download the dependencies.

Create Codebase



1. Create File and Folder on `src\main\java\com\advpro\gpbtest`

- **config**
 - **Appconfig.java**
 - **SecurityConfig.java**
 - **WebConfig.java**
- **controller**
 - **UserController.java**
- **model**
 - **User.java**
- **repository**
 - **UserRepository.java**
- **service**
 - **UserService.java**

Configure User Model

- Open src → main → java → com → advpro → gpbtest → model → **User.java**
- Set **@Entity**, **@Table(name="users")** annotation in **User** model
 - The **@Entity** annotation specifies that the class is an entity and is mapped to a database table.
 - The **@Table** annotation specifies the name of the database table to be used for mapping
- Declare Private Object for Id, Name, DOB etc
- Set **@transient** annotation for Age Calculation
 - **@Transient** annotation is used to indicate that a field is not to be persisted in the database, i.e. their semantics are different.
- Create 3 constructor
 - without param
 - all params
 - all params except id

Configure User Model (Cont.)

- Generate getter, setter for User Model
- Edit age getter function in **`return Period.between(this.dob, LocalDate.now()).getYears();`**
- Generate toString function
- Now build the Project and check table creation
- Insert Data into users table in phpmyadmin. We will insert it later by API

Configure UserRepository

- Open src → main → java → com → advpro → gpbtest → repository→ **UserRepository.java**
- Define **UserRepository** as Interface
- Include **@Repository** annotation
 - **@Repository** annotation is used to indicate that the class provides the mechanism for storage, retrieval, search, update and delete operation on objects
- Import Student model

Configure UserService

- Open src → main → java → com → advpro → gpbtest → service → **UserService.java**
- Include **@Service** annotation
 - **@Service** annotation is it can be applied only to classes. It is used to mark the class as a service provider
- Declare private **UserRepository** in **UserService** Class
- Define constructor with **userRepository** parameter
- Create getUsers() function to fetch data
- Create **BaseController** in controller folder (we will use it for future purpose)

Configure UserController

- Open src → main → java → com → advpro → gpbtest → controller → **UserController.java**
- Declare private UserService
- Create constructor class with **UserService** parameter
- Include annotation of **@RestController**, **@RequestMapping** **@Override**
- Create getUsers function and call **userservice.getUser()**

Change Web service to Build Service

- Inside pom.xml file. Go to <dependencies> section. change the spring-boot-starter artifactid

```
<artifactId>spring-boot-starter</artifactId>
```

to

```
<artifactId>spring-boot-starter-web</artifactId>
```

- Run `./mvnw spring-boot:run` again browse <http://localhost:8080/api/users/list> to see the result.
- Download the project from github <https://github.com/zawadul-00032/gpbtest> . You need permission to download this project. DM me your email id and I'll provide you the access.