

Phase 3 Project: Handling User Authentication Project

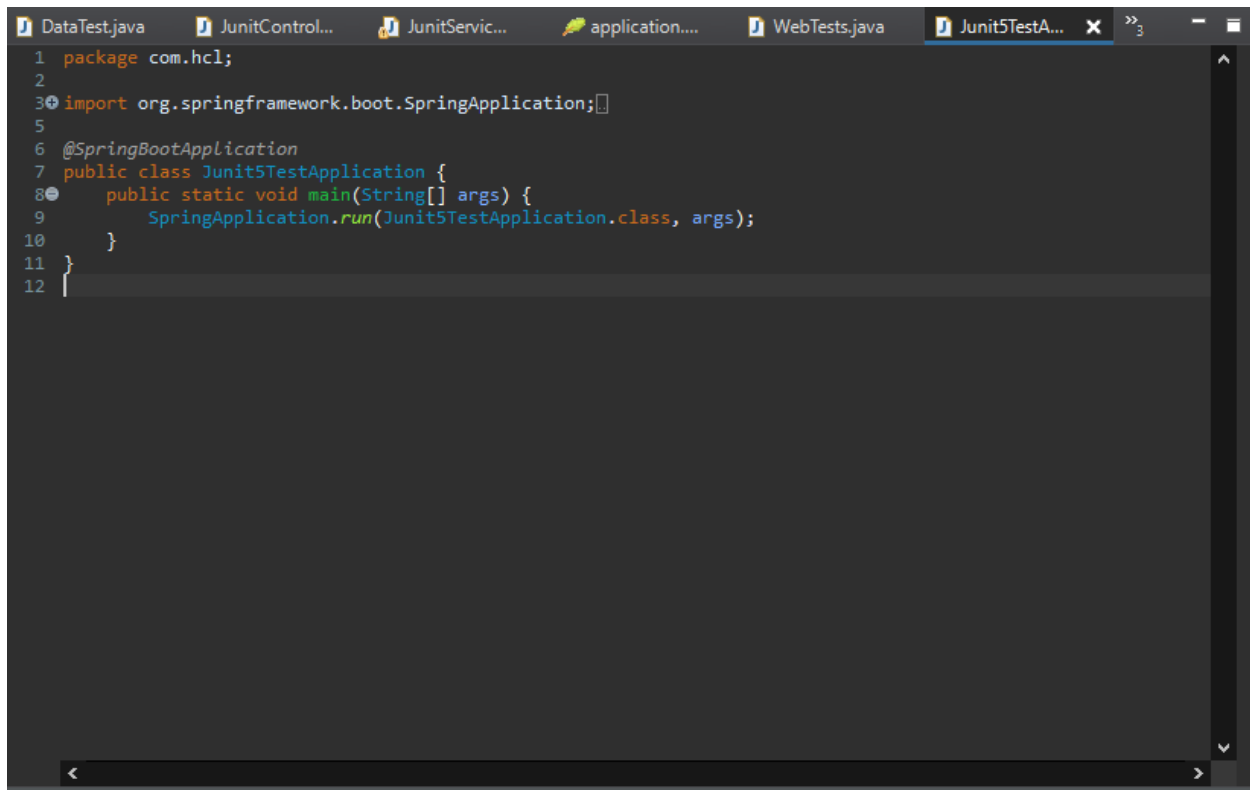
GitHub Link :

<https://github.com/Karthiga-web/JUnitAuthentication.git>

Algorithm:

1. A spring boot application is created with Controller, CrudRepository, Entity class, application properties file, services and JSP
2. The test class is created for testing business logic in ServiceImpl class and test class is created for web tests
3. The pom file has all the dependencies needed for the project
4. ServiceImpl class has all the business logics which have to be unit tested
5. Data Test class tests the business logic
6. Web tests tests the web connection logic for the spring boot project
7. Application test class test the working of the application
8. Data test class autowires the ServiceImpl class and creates Bean for the class with TestConfiguration static class
9. @Test, @BeforeEach, @BeforeAll, @Rollback annotations are used to define the test methods
10. Save test is done with saveTest method that checks the save method with different types of data
11. The authentication like checking of user name and password are done in existsByUsernameAndPassword method
12. Before save method, the existing data is checked by find by id and findby username methods
13. Thus all the methods in the repository are checked with various methods of DataTest class
14. Further, The parameterized tests will be created for better understanding of the Junit tests

Code:

A screenshot of an IDE window with multiple tabs. The active tab is 'Junit5TestA...'. The code is in Java and defines a Spring Boot application. The code is as follows:

```
1 package com.hcl;  
2  
3 import org.springframework.boot.SpringApplication;  
4  
5  
6 @SpringBootApplication  
7 public class Junit5TestApplication {  
8     public static void main(String[] args) {  
9         SpringApplication.run(Junit5TestApplication.class, args);  
10    }  
11 }  
12
```

```
1 package com.hcl.controller;
2
3 import java.util.Optional;
4
5 @Controller
6 @RequestMapping("/junit5Test")
7 public class JunitController {
8
9     @Autowired
10     JunitService service;
11
12     Logger logger = LoggerFactory.getLogger(JunitController.class);
13
14     @GetMapping("/")
15     String hello() {
16         logger.info("Mapping to index");
17         return "/index";
18     }
19
20     @PostMapping("/find")
21     String find(@RequestParam(name = "username") String username, @RequestParam(name = "password") String password,
22               Model model) {
23         logger.info("post mapping /find");
24
25         Optional<JunitEntity> junit = service.findByUsername(username);
26         boolean valid = false;
27         valid = check1(username, password);
28         if (valid) {
29             model.addAttribute("username", junit.get().getUsername());
30             model.addAttribute("message", "Welcome to Page!");
31             return "/NewFile";
32         } else {
33             model.addAttribute("message", "User not Found");
34             return "/index";
35         }
36     }
37
38     public boolean check1(String username, String password) {
39         if (service.existsByUsernameAndPassword(username, password)) {
40             return true;
41         } else {
42             return false;
43         }
44     }
45
46     @PostMapping(path = "/update")
47     String update(@ModelAttribute("user") JunitEntity user, ModelMap model) {
48         logger.info("in /update");
49         Optional<JunitEntity> junit1 = service.findByUsername(user.getUsername());
50         if (junit1.isPresent()) {
51             model.addAttribute("message", "User Already Exists");
52             return "/welcome";
53         } else {
54
55         }
56     }
57 }
```

```
1 package com.hcl.entity;
2
3 import javax.persistence.Column;
4
5 @Data
6 @NoArgsConstructor
7 @Entity
8 @Table(name = "junit")
9 public class JunitEntity {
10
11     @Id
12     @Column(name = "id")
13     Integer id;
14     @Column(name = "username")
15     String username;
16     @Column(name = "password")
17     String password;
18     @Column(name = "firstname")
19     String firstname;
20     @Column(name = "lastname")
21     String lastname;
22     @Column(name = "email")
23     String email;
24 }
25
26
27
28
29
30
```

```
1 package com.hcl.repository;
2
3 import java.util.Optional;
4
5 @Repository
6 public interface JunitRepository extends CrudRepository<JunitEntity, Integer> {
7     Optional<JunitEntity> findByUsername(String username);
8
9     boolean existsByUsernameAndPassword(String username, String Password);
10 }
11
12
13
14
15
```

```
1 package com.hcl.service;
2
3 import java.util.Optional;
4
5 public interface JunitService {
6     public JunitEntity update(JunitEntity user);
7     public Optional<JunitEntity> findByUsername(String userid);
8     public Optional<JunitEntity> findById(int userid);
9     public boolean existsByUsernameAndPassword(String username, String password);
10 }
11
12
13
```

```
1 package com.hcl.service;
2
3 import java.util.Optional;
4
5 @Service
6 public class JunitServiceImpl implements JunitService {
7     @Autowired
8     private JunitRepository repository;
9
10     @Override
11     public JunitEntity update(JunitEntity user) {
12         return repository.save(user);
13     }
14
15     @Override
16     public Optional<JunitEntity> findByUsername(String userid) {
17         return repository.findByUsername(userid);
18     }
19
20     @Override
21     public boolean existsByUsernameAndPassword(String username, String password) {
22         // 1000 Auto-generated method stub
23         return repository.existsByUsernameAndPassword(username, password);
24     }
25
26     @Override
27     public Optional<JunitEntity> findById(int userid) {
28         return repository.findById(userid);
29     }
30 }
31
32
33
34
35
36
37
38
39
```

```
DataTest.java x JUnitControl... welcome.jsp JUnitServic... application... x WebTests.java JUnit3TestA... JUnitEntity... JUnitReposi... JUnitServic... 32
1 spring.mvc.view.prefix = /WEB-INF/views/
2 spring.mvc.view.suffix = .jsp
3 spring.mvc.static-path-pattern=/resources/**
4 spring.jpa.hibernate.ddl-auto=update
5 spring.datasource.url= jdbc:mysql://localhost:3306/spring
6 spring.datasource.username=root
7 spring.datasource.password=root
8 spring.jpa.properties.hibernate.format_sql=true
9 spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
10 logging.level.org.springframework.web: DEBUG
11 server.port=8080
12 |
```

```
DataTest.java x JUnitControl... welcome.jsp JUnitServic... application... x WebTests.java JUnit3TestA... JUnitEntity... JUnitReposi... JUnitServic... 32
1 package com.hcl.service;
2
3 import static org.junit.jupiter.api.Assertions.assertFalse;
4 import static org.junit.jupiter.api.Assertions.assertEquals;
5 import static org.junit.jupiter.api.Assertions.assertTrue;
6
7 import java.util.NoSuchElementException;
8
9 import org.junit.jupiter.api.AfterAll;
10 import org.junit.jupiter.api.Assertions;
11 import org.junit.jupiter.api.BeforeAll;
12 import org.junit.jupiter.api.Test;
13 import org.springframework.beans.factory.annotation.Autowired;
14 import org.springframework.boot.test.autoconfigure.orm.jpa.DataJpaTest;
15 import org.springframework.boot.test.context.TestConfiguration;
16 import org.springframework.context.annotation.Bean;
17 import org.springframework.test.annotation.Rollback;
18
19 import com.hcl.entity.JunitEntity;
20
21 @DataJpaTest
22 public class DataTest {
23
24     static JunitEntity entity;
25
26     @TestConfiguration
27     static class JunitServiceImplTestContextConfiguration {
28
29         @Bean
30         public JunitService employeeService() {
31             return new JunitServiceImpl();
32         }
33     }
34
35     @Autowired
36     private JunitServiceImpl r;
37
38     @BeforeAll
39     static void beforeAll() {
40         entity = new JunitEntity();
41         System.out.println("Initialize connection to database");
42     }
43
44     @Test
45     @Rollback(false)
46     public void saveTest() {
47         entity.setEmail("dfb");
48         entity.setId(2);
49         entity.setFirstname("xfbx");
50         entity.setLastname("xfvbx");
51         entity.setPassword("xfvbxvfb");
52         entity.setUsername("xdfbx");
53         r.update(entity);
54     }
55 }
```

```
DataTest.java x JUnitControl... welcome.jsp JUnitServic... application... WebTests.java JUnit5TestA... JUnitEntity... JUnitReposi... JUnitServic...
61 entity.setUsername("a");
62 r.update(entity);
63 entity.setEmail("b");
64 entity.setId(2);
65 entity.setFirstname("b");
66 entity.setLastname("b");
67 entity.setPassword("b");
68 entity.setUsername("b");
69 r.update(entity);
70 entity.setEmail("c");
71 entity.setId(3);
72 entity.setFirstname("c");
73 entity.setLastname("c");
74 entity.setPassword("c");
75 entity.setUsername("c");
76 r.update(entity);
77
78 @AfterAll
79 static void afterAll() {
80     System.out.println("Close connection to database");
81 }
82
83 @Test
84 public void findByUsernameTest() {
85     entity.setUsername("b");
86     JUnitEntity news = r.findByUsername(entity.getUsername()).get();
87     assertEquals(news.getUsername(), entity.getUsername());
88
89     entity.setUsername("dxfbd");
90
91     Assertions.assertThrows(NoSuchElementException.class, () -> {
92         r.findByUsername(entity.getUsername()).get();
93     });
94 }
95
96 @Test
97 public void existsByUsernameAndPasswordTest() {
98     entity.setUsername("c");
99     entity.setPassword("c");
100     assertTrue(r.existsByUsernameAndPassword(entity.getUsername(), entity.getPassword()));
101     entity.setUsername("2");
102     entity.setPassword("4");
103     assertFalse(r.existsByUsernameAndPassword(entity.getUsername(), entity.getPassword()));
104 }
105
106 @Test
107 public void findByIdTest() {
108     entity.setId(2);
109     JUnitEntity news = r.findById(entity.getId()).get();
110     assertEquals(news.getId(), entity.getId());
111 }
112
113 }
```

```
DataTest.java x JUnitControl... welcome.jsp JUnitServic... application... JUnit5TestA... JUnitEntity... JUnitReposi... JUnitServic... JUnit5TestA... x
1 package com.hcl.service;
2
3 import org.junit.jupiter.api.Test;
4
5 @SpringBootTest
6 class JUnit5TestApplicationTests {
7
8     @Test
9     void contextLoads() {
10     }
11 }
12
13 }
```

```
DataTest.java JUnitControl... welcome.jsp JUnitServic... application... WebTests.java x JUnitEntity... JUnitReposi... JUnitServic... JUnit5TestA...
1 package com.hcl.service;
2
3 import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.get;
4
5 @SpringBootTest(webEnvironment = SpringBootTest.WebEnvironment.RANDOM_PORT)
6 @AutoConfigureMockMvc
7 public class WebTests {
8
9     @LocalServerPort
10     private int port;
11
12     @Autowired
13     private MockMvc mockMvc;
14
15     @Test
16     public void shouldReturnDefaultMessage() throws Exception {
17         this.mockMvc.perform(get("/")).andDo(print()).andExpect(status().isOk());
18     }
19 }
```

```

1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta charset="ISO-8859-1">
7 <title>User Information</title>
8 </head>
9 <body>
10 <div align="center">
11 <form id="update"
12   action="${pageContext.request.contextPath}/junit5Test/update"
13   method="post" modelAttribute="user">
14 <table style="width: 50%;
15 <tr>
16 <td>Id</td>
17 <td><input type="number" name="id" required/></td>
18 </tr>
19 <tr>
20 <td>First Name</td>
21 <td><input type="text" name="firstname" required/></td>
22 </tr>
23 <tr>
24 <td>Last Name</td>
25 <td><input type="text" name="lastname" required/></td>
26 </tr>
27 <tr>
28 <td>Email Address</td>
29 <td><input type="text" name="email" required/></td>
30 </tr>
31 <tr>
32 <td>Username</td>
33 <td><input type="text" name="username" required/></td>
34 </tr>
35 <tr>
36 <td>Password</td>
37 <td><input type="text" name="password" required/></td>
38 </tr>
39 </table>
40 <input type="submit" value="Update" />
41 </form>
42 </div>
43 </body>
44 </html>

```

```

1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta charset="ISO-8859-1">
7 <title>Page</title>
8 </head>
9 <body>
10 <div align="center">
11 <div style="color: red">${username}</div>
12 <br>
13 <div style="color: red">${message}</div>
14 </div>
15 </body>
16 </html>

```

```

1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <title>Login Application</title>
7 </head>
8 <body>
9 <div align="center">
10 <h2>Enter your Login Credentials!</h2>
11 <div style="color: red">${message}</div>
12 <form id="find"
13   action="${pageContext.request.contextPath}/junit5Test/find"
14   method="post">
15 <table>
16 <tr>
17 <td>Username</td>
18 <td><input type="text" name="username" required/></td>
19 </tr>
20 <tr>
21 <td>Password</td>
22 <td><input type="text" name="password" required/></td>
23 </tr>
24 </table>
25 <input type="submit" value="Submit" />
26 </form>
27 </div>
28 </body>
29 </html>

```

junit5Test/pom.xml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
5   <modelVersion>4.0.0</modelVersion>
6   <parent>
7     <groupId>org.springframework.boot</groupId>
8     <artifactId>spring-boot-starter-parent</artifactId>
9     <version>2.4.2</version>
10    <relativePath /> <!-- lookup parent from repository -->
11  </parent>
12  <groupId>com.hcl</groupId>
13  <artifactId>junit5Test</artifactId>
14  <version>0.0.1-SNAPSHOT</version>
15  <packaging>war</packaging>
16  <name>junit5Test</name>
17  <description>Test project for Spring Boot</description>
18  <properties>
19    <java.version>1.8</java.version>
20  </properties>
21  <dependencies>
22    <dependency>
23      <groupId>org.springframework.boot</groupId>
24      <artifactId>spring-boot-starter-data-jpa</artifactId>
25    </dependency>
26    <dependency>
27      <groupId>org.springframework.boot</groupId>
28      <artifactId>spring-boot-starter-web</artifactId>
29    </dependency>
30    <dependency>
31      <groupId>com.h2database</groupId>
32      <artifactId>h2</artifactId>
33      <scope>test</scope>
34    </dependency>
35    <dependency>
36      <groupId>org.apache.tomcat.embed</groupId>
37      <artifactId>tomcat-embed-jasper</artifactId>
38      <scope>provided</scope>
39    </dependency>
40    <dependency>
41      <groupId>mysql</groupId>
42      <artifactId>mysql-connector-java</artifactId>
43      <scope>runtime</scope>
44    </dependency>
45    <dependency>
46      <groupId>org.projectlombok</groupId>
47      <artifactId>lombok</artifactId>
48      <optional>true</optional>
49    </dependency>
50    <dependency>
51      <groupId>org.junit.jupiter</groupId>
52      <artifactId>junit-jupiter</artifactId>
```


junit5Test/pom.xml X

```
41         <groupId>mysql</groupId>
42         <artifactId>mysql-connector-java</artifactId>
43         <scope>runtime</scope>
44     </dependency>
45     <dependency>
46         <groupId>org.projectlombok</groupId>
47         <artifactId>lombok</artifactId>
48         <optional>true</optional>
49     </dependency>
50     <dependency>
51         <groupId>org.junit.jupiter</groupId>
52         <artifactId>junit-jupiter-api</artifactId>
53         <scope>test</scope>
54     </dependency>
55     <dependency>
56         <groupId>javax.servlet</groupId>
57         <artifactId>jstl</artifactId>
58     </dependency>
59     <dependency>
60         <groupId>org.springframework.boot</groupId>
61         <artifactId>spring-boot-starter-tomcat</artifactId>
62         <scope>provided</scope>
63     </dependency>
64     <dependency>
65         <groupId>org.springframework.boot</groupId>
66         <artifactId>spring-boot-starter-data-jpa</artifactId>
67     </dependency>
68     <dependency>
69         <groupId>org.springframework.boot</groupId>
70         <artifactId>spring-boot-starter-test</artifactId>
71         <scope>test</scope>
72     </dependency>
73 </dependencies>
74
75 <build>
76     <plugins>
77         <plugin>
78             <groupId>org.springframework.boot</groupId>
79             <artifactId>spring-boot-maven-plugin</artifactId>
80         </plugin>
81         <plugin>
82             <groupId>org.apache.maven.plugins</groupId>
83             <artifactId>maven-surefire-plugin</artifactId>
84             <configuration>
85                 <testFailureIgnore>true</testFailureIgnore>
86             </configuration>
87         </plugin>
88     </plugins>
89 </build>
90
91 </project>
92
```

Outputs:

```
1 package com.hcl.service;
2
3 import org.junit.jupiter.api.Test;
4
5 @SpringBootTest
6 class JUnit5TestApplicationTests {
7
8     @Test
9     void contextLoads() {
10     }
11 }
12
13
14
```

```
<terminated> JUnit5TestApplicationTests (1) [Junit] C:\Program Files\Java\jdk1.8.0_281\bin\java.exe (Feb 9, 2021, 3:17:30 PM - 3:17:37 PM)
15:17:32.253 [main] DEBUG org.springframework.boot.test.context.SpringBootTestContextBootstrapper - @TestExecutionListeners is not present for class [com.hcl.service.JUnit5TestApplicationTests]
15:17:32.253 [main] INFO org.springframework.boot.test.context.SpringBootTestContextBootstrapper - Loaded default TestExecutionListener class names from location [META-INF/spring.factories]:
15:17:32.283 [main] INFO org.springframework.boot.test.context.SpringBootTestContextBootstrapper - Using TestExecutionListeners: [org.springframework.test.context.web.ServletTestExecutionListener, org.springframework.test.context.support.AbstractDirtiesContextTestExecutionListener, org.springframework.test.context.support.AbstractTransactionalTestExecutionListener, org.springframework.test.context.support.DefaultTestExecutionListener, org.springframework.test.context.support.DependencyInjectionTestExecutionListener, org.springframework.test.context.support.DefaultTestExecutionListener, org.springframework.test.context.support.MethodInvocationTestExecutionListener]
15:17:32.293 [main] DEBUG org.springframework.test.context.support.AbstractDirtiesContextTestExecutionListener - Before test class: context [DefaultTestContext@569f390 testClass = JUnit5TestApplicationTests, testMethod = contextLoads(), testInstance = null, testArguments = [], testName = null]
15:17:32.344 [main] DEBUG org.springframework.test.context.support.TestPropertySourceUtils - Adding inlined properties to environment: {spring.jmx.enabled=false, org.springframework.boot.test.context.SpringBootTestContextBootstrapper=true}

=====
:: Spring Boot ::
(v2.4.2)

2021-02-09 15:17:32.707 INFO 12544 --- [main] c.h.service.JUnit5TestApplicationTests : Starting JUnit5TestApplicationTests using Java 1.8.0_281 on LAPTOP-HL50N9WP with PID 12544
2021-02-09 15:17:32.711 INFO 12544 --- [main] c.h.service.JUnit5TestApplicationTests : No active profile set, falling back to default profiles: default
2021-02-09 15:17:32.788 DEBUG 12544 --- [main] o.s.w.c.s.GenericWebApplicationContext : Refreshing org.springframework.web.context.support.GenericWebApplicationContext@25bfcfd
2021-02-09 15:17:33.349 INFO 12544 --- [main] s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2021-02-09 15:17:33.403 INFO 12544 --- [main] s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 54 ms. Found 1 JPA repository interfaces.
2021-02-09 15:17:34.180 INFO 12544 --- [main] o.hibernate.jpa.internal.util.LogHelper : HH0000284: Processing PersistenceUnitInfo [name: default]
2021-02-09 15:17:34.245 INFO 12544 --- [main] org.hibernate.Version : HH0000412: Hibernate ORM core version 5.4.27.Final
2021-02-09 15:17:34.481 INFO 12544 --- [main] o.hibernate.annotations.common.Version : HCANN000001: Hibernate Commons Annotations (5.1.2.Final)
2021-02-09 15:17:34.685 INFO 12544 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2021-02-09 15:17:35.123 INFO 12544 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2021-02-09 15:17:35.157 INFO 12544 --- [main] org.hibernate.dialect.Dialect : HH0000400: Using dialect: org.hibernate.dialect.MySQL5Dialect
2021-02-09 15:17:35.836 INFO 12544 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HH0000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2021-02-09 15:17:36.000 DEBUG 12544 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2021-02-09 15:17:36.512 WARN 12544 --- [main] jpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may perform more slowly during development.
2021-02-09 15:17:36.782 INFO 12544 --- [main] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService 'applicationTaskExecutor'
2021-02-09 15:17:36.800 DEBUG 12544 --- [main] s.w.s.m.m.a.RequestMappingHandlerAdapter : ControllerAdvice beans: 0 @ModelAttribute, 0 @InitBinder, 1 RequestBodyAdvice, 1 ResponseBodyAdvice
2021-02-09 15:17:36.906 DEBUG 12544 --- [main] s.w.s.m.m.a.RequestMappingHandlerMapping : 5 mappings in 'requestMappingHandlerMapping'
2021-02-09 15:17:36.912 INFO 12544 --- [main] o.s.b.a.w.s.WelcomePageHandlerMapping : Adding welcome page template: index
2021-02-09 15:17:37.055 DEBUG 12544 --- [main] o.s.w.s.handler.SimpleHandlerMapping : Patterns [/webjars/**, /resources/**] in 'resourceHandlerMapping'
2021-02-09 15:17:37.525 INFO 12544 --- [main] m.m.a.ExceptionHandlerExceptionResolver : ControllerAdvice beans: 0 @ExceptionHandler, 1 ResponseBodyAdvice
2021-02-09 15:17:37.415 INFO 12544 --- [main] c.h.service.JUnit5TestApplicationTests : Started JUnit5TestApplicationTests in 5.863 seconds (JVM running for 6.218)
2021-02-09 15:17:37.522 DEBUG 12544 --- [main] o.s.w.c.s.GenericWebApplicationContext : Closing org.springframework.web.context.support.GenericWebApplicationContext@25bfcfd, started at 15:17:32.707
2021-02-09 15:17:37.525 INFO 12544 --- [extShutdownHook] o.s.s.concurrent.ThreadPoolTaskExecutor : Shutting down ExecutorService 'applicationTaskExecutor'
2021-02-09 15:17:37.526 INFO 12544 --- [extShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'default'
2021-02-09 15:17:37.528 INFO 12544 --- [extShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated...
2021-02-09 15:17:37.535 INFO 12544 --- [extShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.
```

```
1 package com.hcl.service;
2
3 import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.get;
4
5 @SpringBootTest(webEnvironment = SpringBootTest.WebEnvironment.RANDOM_PORT)
6 class WebTests {
7
8     @LocalServerPort
9     private int port;
10
11     @Autowired
12     private MockMvc mockMvc;
13
14     @Test
15     void shouldReturnDefaultMessage() {
16     }
17 }
18
19
```

```
<terminated> WebTests (2) [Junit] C:\Program Files\Java\jdk1.8.0_281\bin\java.exe (Feb 9, 2021, 3:18:26 PM - 3:18:33 PM)
MockMvcServletRequest:
HTTP Method = GET
Request URI = /
Parameters = {}
Headers = {}
Body = null
Session Attrs = {}

Handler:
Type = org.springframework.web.servlet.mvc.ParameterizableViewController

Async:
Async started = false
Async result = null

Resolved Exception:
Type = null

ModelAndView:
View name = index
View = null
Model = null

FlashMap:
Attributes = null

MockMvcServletResponse:
Status = 200
Error message = null
Headers = [Content-Language:en"]
Content type = null
Body =
Forwarded URL = /WEB-INF/views/index.jsp
Redirected URL = null
Cookies = {}

2021-02-09 15:18:33.889 INFO 4828 --- [extShutdownHook] o.s.s.concurrent.ThreadPoolTaskExecutor : Shutting down ExecutorService 'applicationTaskExecutor'
2021-02-09 15:18:33.882 INFO 4828 --- [extShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'default'
2021-02-09 15:18:33.884 INFO 4828 --- [extShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated...
2021-02-09 15:18:33.893 INFO 4828 --- [extShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.
```

Debug Project Explorer JUnit WebTests.java

finished after 3.789 seconds

Runs: 4/4 Errors: 0 Failures: 0

JUnit [Runner: JUnit 5] (0.306 s)

- saveTest() (0.152 s)
- findByUsernameTest() (0.024 s)
- existsByHibernateAndRecordTest() (0.013 s)
- findByIdTest() (0.017 s)

Failure Trace

Console Problems Debug Shell Progress Search Git Repositories Git Staging

<terminated> DataTest [JUnit] C:\Program Files\Java\jdk1.8.0_281\bin\java.exe (Feb 9, 2021, 3:19:13 PM - 3:19:18 PM)

```

from
  junit junitentit0_
where
  junitentit0_username=?
2021-02-09 15:19:18.158 INFO 16416 --- [main] o.s.t.c.transaction.TransactionContext : Rolled back transaction for test: [DefaultTestContext@2e3967ea testClass = DataTest, test
2021-02-09 15:19:18.152 INFO 16416 --- [main] o.s.t.c.transaction.TransactionContext : Began transaction (1) for test context [DefaultTestContext@2e3967ea testClass = DataTest,
Hibernate:
select
  junitentit0_id as col_0_0_
from
  junit junitentit0_
where
  junitentit0_username=?
  and junitentit0_password=? limit ?
Hibernate:
select
  junitentit0_id as col_0_0_
from
  junit junitentit0_
where
  junitentit0_username=?
  and junitentit0_password=? limit ?
2021-02-09 15:19:18.163 INFO 16416 --- [main] o.s.t.c.transaction.TransactionContext : Rolled back transaction for test: [DefaultTestContext@2e3967ea testClass = DataTest, test
2021-02-09 15:19:18.165 INFO 16416 --- [main] o.s.t.c.transaction.TransactionContext : Began transaction (1) for test context [DefaultTestContext@2e3967ea testClass = DataTest,
Hibernate:
select
  junitentit0_id as id1_0_0_,
  junitentit0_email as email2_0_0_,
  junitentit0_firstname as firstnam3_0_0_,
  junitentit0_lastname as lastname4_0_0_,
  junitentit0_password as passwords_0_0_,
  junitentit0_username as username5_0_0_
from
  junit junitentit0_
where
  junitentit0_id=?
2021-02-09 15:19:18.176 INFO 16416 --- [main] o.s.t.c.transaction.TransactionContext : Rolled back transaction for test: [DefaultTestContext@2e3967ea testClass = DataTest, test
Close connection to database
2021-02-09 15:19:18.183 INFO 16416 --- [extShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'default'

```

Mail - X | Safari - X | Testin - X | Integri - X | N - X | Step - X | JUnit - X | Karth - X | Self - X | test - X | JUnit - X | slack - X | Get SI - X | Slack - X | assert - X | JUnit - X | User - X | + | - | X

localhost:8080

Not syncing

Id	<input type="text"/>
First Name	<input type="text"/>
Last Name	<input type="text"/>
Email Address	<input type="text"/>
Username	<input type="text"/>
Password	<input type="text"/>
	<input type="button" value="Update"/>

3:20 PM 2/9/2021

localhost:8080

Not syncing

Id	<input type="text" value="3"/>
First Name	<input type="text" value="k"/>
Last Name	<input type="text" value="r"/>
Email Address	<input type="text" value="gfgthbbr@gmail.com"/>
Username	<input type="text" value="dfr"/>
Password	<input type="text" value="dgdgr"/>
	<input type="button" value="Update"/>

Enter your Login Credentials!

User Information Updated

Username

Password

Submit

Enter your Login Credentials!

User Information Updated

Username

dfh

Password

dgdgj

Submit

dfh

Welcome to Page!