**Employee Name :** PravinKumar RajaSahayam
**Employee Id :** 11946

**Task :** Test your application.

1. **White Box Testing (Unit Testing):**

- Write the Unit Test cases by using JUnit for your Back-End [Spring Boot].

**Screenshots :**

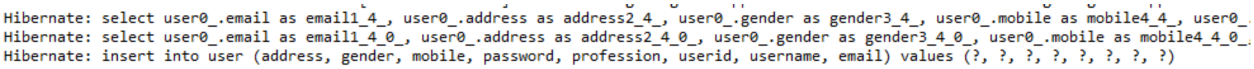
1. **For Registration :**

**Code :**

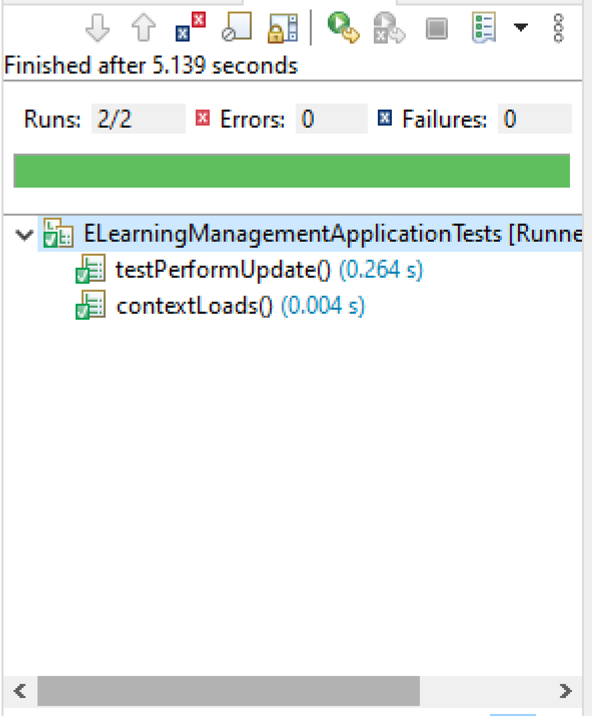
- **User(Student) Registration :**

```
@Test
          @Order(1)
          void contextLoads() {
          }
          @Test
  @Order(2)
  void testPerformUpdate() throws Exception {
    User obj = new User();
    obj.setUserid("101");
    obj.setUsername("PravinKumar");
    obj.setProfession("EEE");
    obj.setGender("Male");
    obj.setEmail("pravinkumar146@gmail.com");
    obj.setMobile("8095231476");
    obj.setAddress("Virudhunagar");
    obj.setPassword("PravinKumar@11946");
    registerUser = "Register Successfully";
    result = registrationcontroller.registerUser(obj);
    assertEquals(registerUser,result);
 }
```

**Output :**



```
Hibernate: select user0_.email as email1_4_, user0_.address as address2_4_, user0_.gender as gender3_4_, user0_.mobile as mobile4_4_, user0_
Hibernate: select user0_.email as email1_4_0_, user0_.address as address2_4_0_, user0_.gender as gender3_4_0_, user0_.mobile as mobile4_4_0_
Hibernate: insert into user (address, gender, mobile, password, profession, userid, username, email) values (?, ?, ?, ?, ?, ?, ?, ?)
```

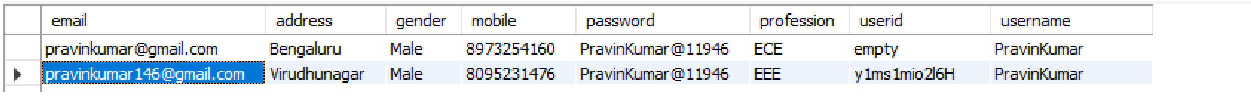


```
Finished after 5.139 seconds

Runs: 2/2    Errors: 0    Failures: 0

ELearningManagementApplicationTests [Runne
    testPerformUpdate() (0.264 s)
    contextLoads() (0.004 s)
```

## Database :

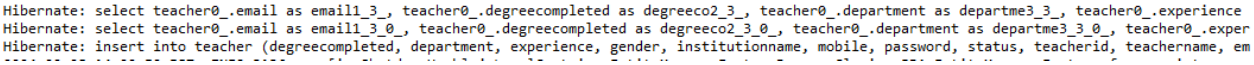


| | email | address | gender | mobile | password | profession | userid | username |
|---|---|---|---|---|---|---|---|---|
| | pravinkumar@gmail.com | Bengaluru | Male | 8973254160 | PravinKumar@11946 | ECE | empty | PravinKumar |
| ▶ | pravinkumar146@gmail.com | Virudhunagar | Male | 8095231476 | PravinKumar@11946 | EEE | y1ms1mio2l6H | PravinKumar |

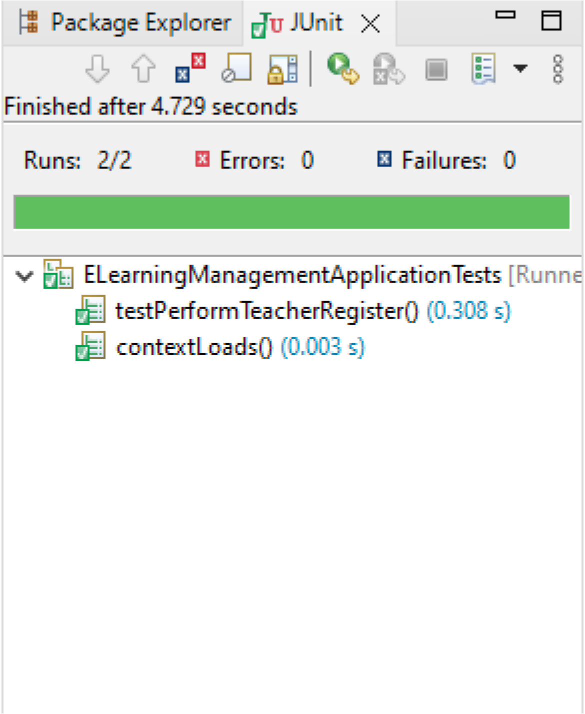
- **User(Teacher) Registration :**

```
@Test
  @Order(3)
  void testPerformTeacherRegister() throws Exception {
    Teacher obj = new Teacher();
    obj.setTeacherid("109");
    obj.setTeachername("Preethi");
    obj.setGender("Female");
    obj.setEmail("preethi12345@gmail.com");
    obj.setMobile("9875231460");
    obj.setExperience("3");
    obj.setDepartment("ECE");
    obj.setDegreecompleted("PHD");
    obj.setInstitutionname("VVE");
    obj.setPassword("Preethi@123456");
    obj.setStatus("accept");
    registerTeacher = "Registered Successfully";
    result = registrationcontroller.registerDoctor(obj);
    assertEquals(registerTeacher,result);
 }
```

## Output :



```
Hibernate: select teacher0_.email as email1_3_, teacher0_.degreecompleted as degreeco2_3_, teacher0_.department as departme3_3_, teacher0_.experience
Hibernate: select teacher0_.email as email1_3_0_, teacher0_.degreecompleted as degreeco2_3_0_, teacher0_.department as departme3_3_0_, teacher0_.exper
Hibernate: insert into teacher (degreecompleted, department, experience, gender, institutionname, mobile, password, status, teacherid, teachername, em
```

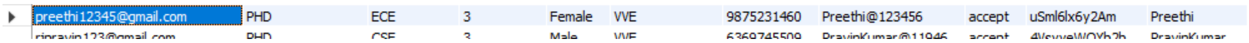


Finished after 4.729 seconds

Runs: 2/2 ☒ Errors: 0 ☒ Failures: 0

- ELearningManagementApplicationTests [Runne
  - testPerformTeacherRegister() (0.308 s)
  - contextLoads() (0.003 s)

## Database :



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| preethi12345@gmail.com | PHD | ECE | 3 | Female | VVE | 9875231460 | Preethi@123456 | accept | uSml6lx6y2Am | Preethi |
| ripravin123@gmail.com | PHD | CSE | 3 | Male | VVE | 6369745509 | PravinKumar@11946 | accept | 4VsyyeWOYh2b | PravinKumar |

---

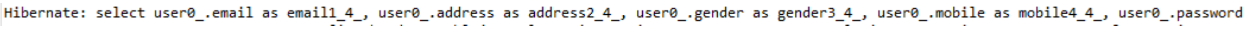
### 2. For Login :

## Code :

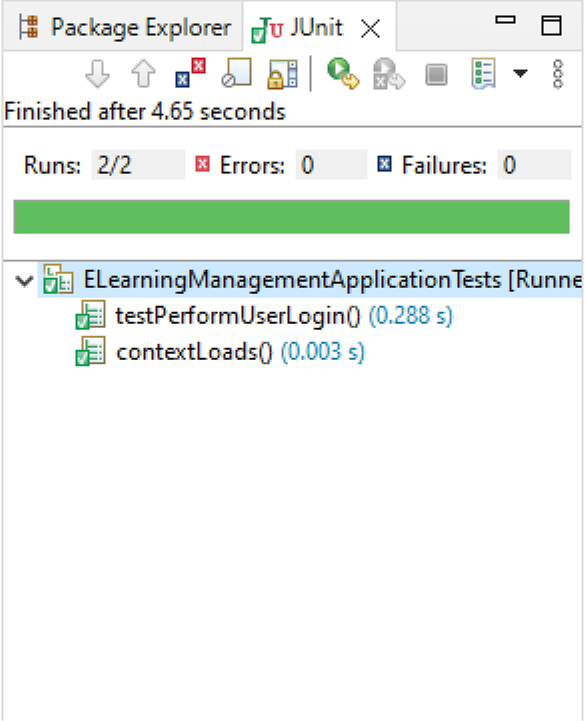
- **User(Student) Login :**

```
    @Test
@Order(4)
void testPerformUserLogin() throws Exception {
  User obj = new User();
  obj.setEmail("pravinkumar1468@gmail.com");
  obj.setPassword("PravinKumar@11946");
  loginUser = "Login Successfully";
  result1 = logincontroller.loginUser(obj);
  assertEquals(loginUser,result1);
}
```

## Output :



```
Hibernate: select user0_.email as email1_4_, user0_.address as address2_4_, user0_.gender as gender3_4_, user0_.mobile as mobile4_4_, user0_.password
```

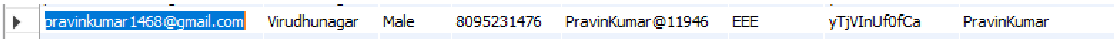


Finished after 4.65 seconds

Runs: 2/2 Errors: 0 Failures: 0

ELearningManagementApplicationTests [Runne
testPerformUserLogin() (0.288 s)
contextLoads() (0.003 s)

## Database :

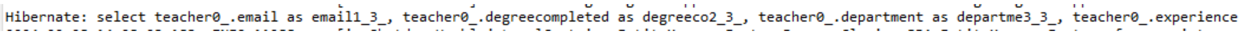


| | | | | | | | |
|---|---|---|---|---|---|---|---|
| pravinkumar1468@gmail.com | Virudhunagar | Male | 8095231476 | PravinKumar@11946 | EEE | yTjVInUf0fCa | PravinKumar |

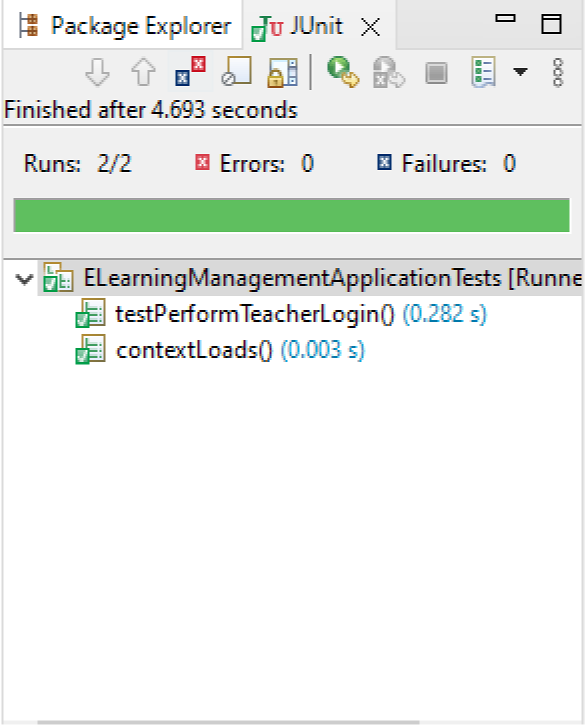
- ## User(Teacher) Login :

```
@Test
        @Order(5)
        void testPerformTeacherLogin() throws Exception {
          Teacher obj = new Teacher();
          obj.setEmail("preethi12345@gmail.com");
          obj.setPassword("Preethi123mail.com");
          loginTeacher = "Login Successfully";
          result1 = logincontroller.loginDoctor(obj);
          assertEquals(loginTeacher,result1);
        }
```

## Output :



```
Hibernate: select teacher0_.email as email1_3_, teacher0_.degreecompleted as degreeco2_3_, teacher0_.department as departme3_3_, teacher0_.experience
```

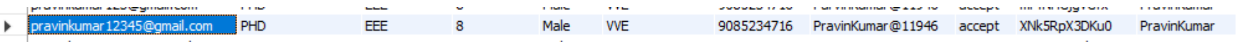


```
Finished after 4.693 seconds

Runs:  2/2        Errors:  0        Failures:  0

ELearningManagementApplicationTests [Runne
    testPerformTeacherLogin() (0.282 s)
    contextLoads() (0.003 s)
```

## Database :



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| pravinkumar12345@gmail.com | PHD | EEE | 8 | Male | VVE | 9085234716 | PravinKumar@11946 | accept | XNk5RpX3DKu0 | PravinKumar |

—-------------------------------------------------------------------------------------------------------------------

### 3. User(Student) :

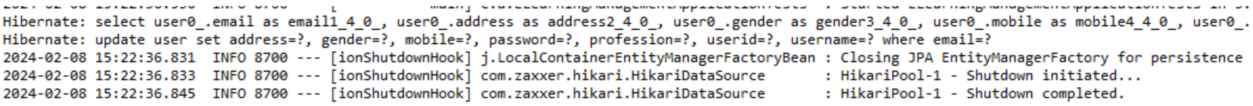
- **For update the user profile :**

## Code :

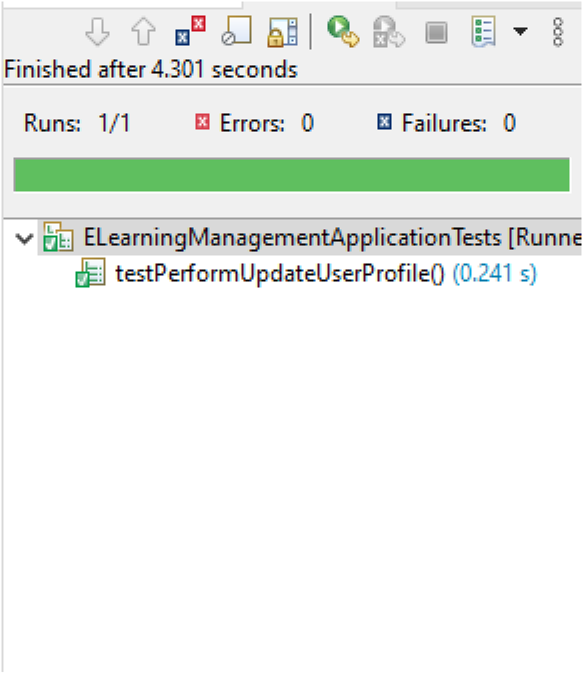
```
@Test
 @Order(7)
 void testPerformUpdateUserProfile() throws Exception {
   User obj = new User();
   obj.setUserid("1019");
   obj.setUsername("PravinKumar");
   obj.setProfession("EEE");
   obj.setGender("Male");
   obj.setEmail("pravinkumar1468@gmail.com");
```

```
    obj.setMobile("8095231746");
    obj.setAddress("Virudhunagar");
    obj.setPassword("PravinKumar@11946");
    updateUser = "User Updated Successfully";
    result = userservice.updateUserProfile(obj);
    assertEquals(updateUser,result);
}
```

**Output :**



```
Hibernate: select user0_.email as email1_4_0_, user0_.address as address2_4_0_, user0_.gender as gender3_4_0_, user0_.mobile as mobile4_4_0_, user0_.
Hibernate: update user set address=?, gender=?, mobile=?, password=?, profession=?, userid=?, username=? where email=?
2024-02-08 15:22:36.831  INFO 8700 --- [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence
2024-02-08 15:22:36.833  INFO 8700 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Shutdown initiated...
2024-02-08 15:22:36.845  INFO 8700 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Shutdown completed.
```

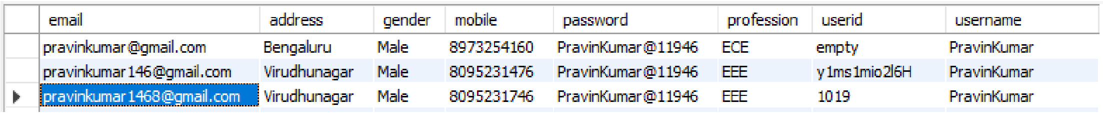


Finished after 4.301 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne
testPerformUpdateUserProfile() (0.241 s)

**Database :**



| | email | address | gender | mobile | password | profession | userid | username |
|---|---|---|---|---|---|---|---|---|
| | pravinkumar@gmail.com | Bengaluru | Male | 8973254160 | PravinKumar@11946 | ECE | empty | PravinKumar |
| | pravinkumar146@gmail.com | Virudhunagar | Male | 8095231476 | PravinKumar@11946 | EEE | y1ms1mio2l6H | PravinKumar |
| ▶ | pravinkumar1468@gmail.com | Virudhunagar | Male | 8095231746 | PravinKumar@11946 | EEE | 1019 | PravinKumar |

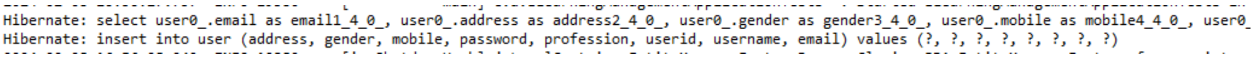
- **For save the new user :**

**Code :**

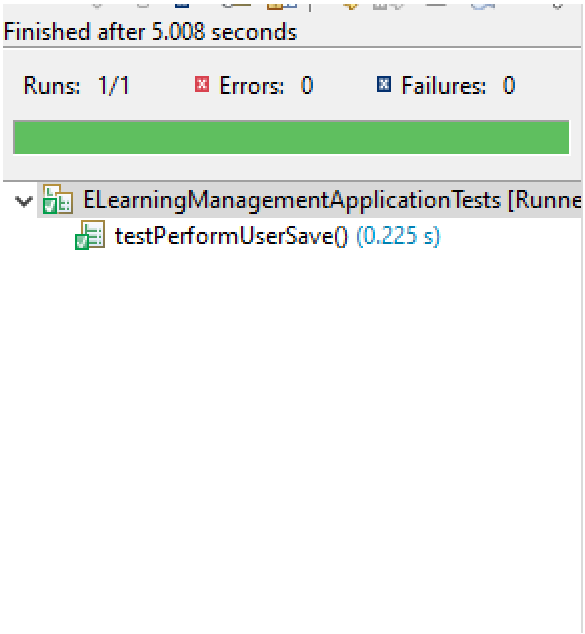
```
@Test
```

```
@Order(13)
void testPerformUserSave() throws Exception {
  User obj = new User();
  obj.setUserid("001");
  obj.setUsername("PravinKumar");
  obj.setProfession("EEE");
  obj.setGender("Male");
  obj.setEmail("pravinkumar1230@gmail.com");
  obj.setMobile("8095231476");
  obj.setAddress("Virudhunagar");
  obj.setPassword("PravinKumar@11946");
  userSave = "Register Successfully";
  result = userservice.saveUser(obj);
  assertEquals(userSave,result);
}
```

## Output :



```
Hibernate: select user0_.email as email1_4_0_, user0_.address as address2_4_0_, user0_.gender as gender3_4_0_, user0_.mobile as mobile4_4_0_, user0_
Hibernate: insert into user (address, gender, mobile, password, profession, userid, username, email) values (?, ?, ?, ?, ?, ?, ?, ?)
```

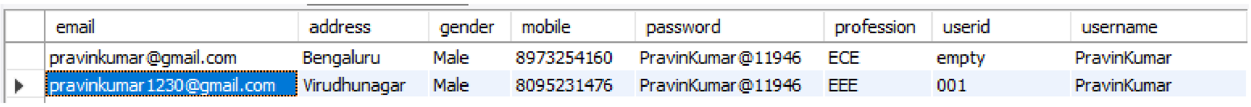


Finished after 5.008 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne
testPerformUserSave() (0.225 s)

## Database :



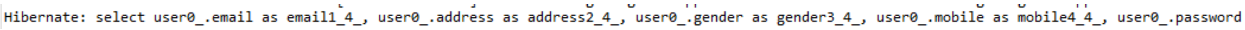
| | email | address | gender | mobile | password | profession | userid | username |
|---|---|---|---|---|---|---|---|---|
| | pravinkumar@gmail.com | Bengaluru | Male | 8973254160 | PravinKumar@11946 | ECE | empty | PravinKumar |
| ▶ | pravinkumar1230@gmail.com | Virudhunagar | Male | 8095231476 | PravinKumar@11946 | EEE | 001 | PravinKumar |

- **For get all the user details :**

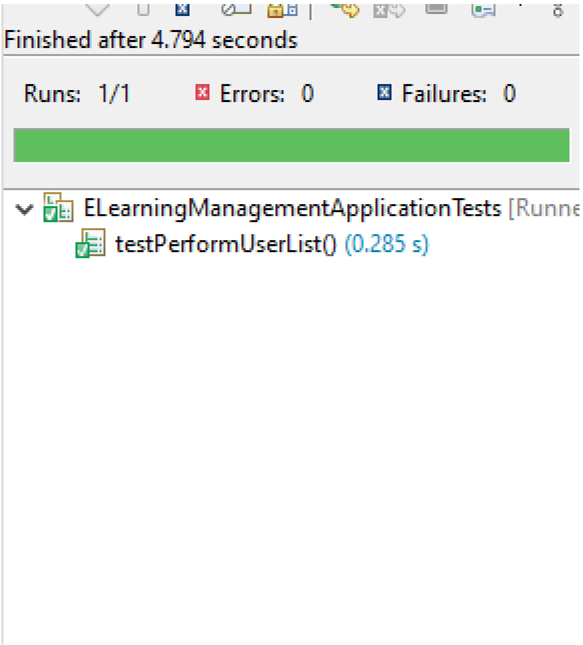
**Code :**

```
@Test
 @Order(6)
 void testPerformUserList() throws Exception {
     list1 = (List<User>) userservice.getAllUsers();
     user1 = new
User("1019","PravinKumar","EEE","Male","pravinkumar1468@gmail.com","8095231476","Virudhunagar
","PravinKumar@11946");
     user2 = new
User("uIC9r6oDV4XN","user","csa","Male","rjpravin11111@gmail.com","8976532140","east","PravinKu
mar@11946");
     user3 = new
User("cQvHIlylK9Cv","PravinKumarUserId","EEE","Female","rjpravin1236@gmail.com","8932541760","
East","PravinKumar@11946");
     list2.add(user1);
     list2.add(user2);
     list2.add(user3);
     list3.add(user1);
     list3.add(user2);
     list3.add(user3);
     list2.equals(list3);
     assertEquals(list2, list3);
 }
```

**Output :**



```
Hibernate: select user0_.email as email1_4_, user0_.address as address2_4_, user0_.gender as gender3_4_, user0_.mobile as mobile4_4_, user0_.password
```



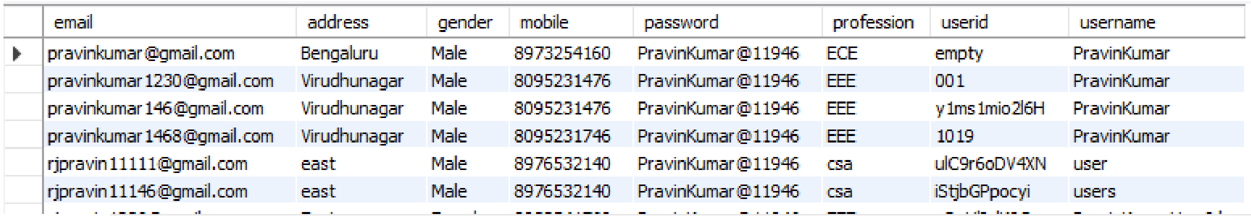
Finished after 4.794 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne

testPerformUserList() (0.285 s)

**Database :**



| | email | address | gender | mobile | password | profession | userid | username |
|---|---|---|---|---|---|---|---|---|
| ▶ | pravinkumar@gmail.com | Bengaluru | Male | 8973254160 | PravinKumar@11946 | ECE | empty | PravinKumar |
| | pravinkumar1230@gmail.com | Virudhunagar | Male | 8095231476 | PravinKumar@11946 | EEE | 001 | PravinKumar |
| | pravinkumar146@gmail.com | Virudhunagar | Male | 8095231476 | PravinKumar@11946 | EEE | y1ms1mio2l6H | PravinKumar |
| | pravinkumar1468@gmail.com | Virudhunagar | Male | 8095231746 | PravinKumar@11946 | EEE | 1019 | PravinKumar |
| | rjpravin11111@gmail.com | east | Male | 8976532140 | PravinKumar@11946 | csa | ulC9r6oDV4XN | user |
| | rjpravin11146@gmail.com | east | Male | 8976532140 | PravinKumar@11946 | csa | iStjbGPpocyi | users |

—------------------------------------------------------------------------------------------------------------------

### 4. User(Teacher) :

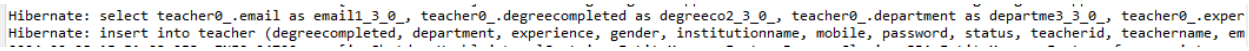
- **Add new teacher :**

**Code :**

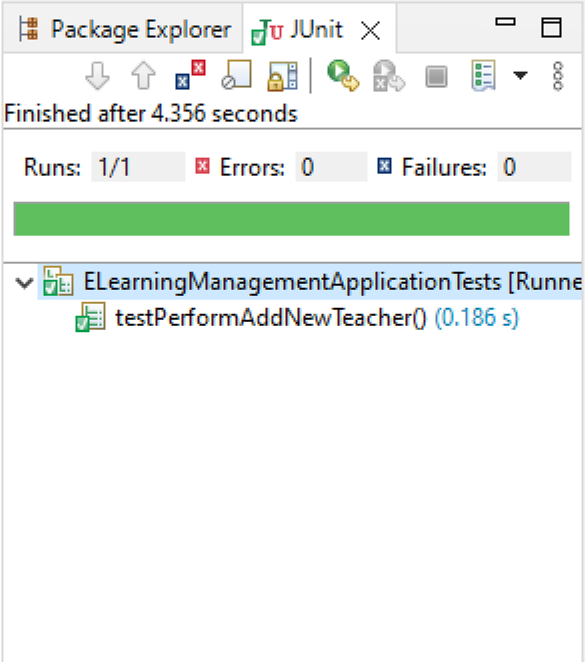
```
@Test
 @Order(9)
 void testPerformAddNewTeacher() throws Exception {
  Teacher obj = new Teacher();
  obj.setTeacherid("110");
  obj.setTeachername("PravinKumar");
  obj.setGender("Male");
  obj.setEmail("pravinkumar325@gmail.com");
  obj.setMobile("9875231460");
  obj.setExperience("3");
  obj.setDepartment("ECE");
  obj.setDegreecompleted("PHD");
```

```
   obj.setInstitutionname("VVE");
   obj.setPassword("Preethi@123456");
   obj.setStatus("accept");
   addteacher = "Add Successfully";
   result = teacherservice.addNewTeacher(obj);
   assertEquals(addteacher,result);
}
```

**Output :**



```
Hibernate: select teacher0_.email as email1_3_0_, teacher0_.degreecompleted as degreeco2_3_0_, teacher0_.department as departme3_3_0_, teacher0_.exper
Hibernate: insert into teacher (degreecompleted, department, experience, gender, institutionname, mobile, password, status, teacherid, teachername, em
```

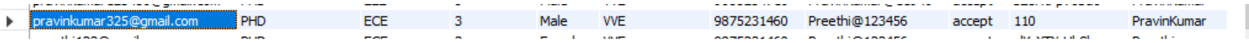


Finished after 4.356 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne
testPerformAddNewTeacher() (0.186 s)

**Database :**



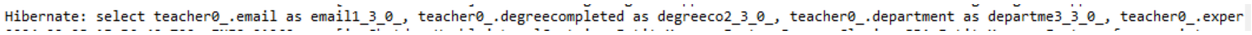
|  |  |  |  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|
| pravinkumar325@gmail.com | PHD | ECE | 3 | Male | VVE | 9875231460 | Preethi@123456 | accept | 110 | PravinKumar |

- **Update existing teacher :**

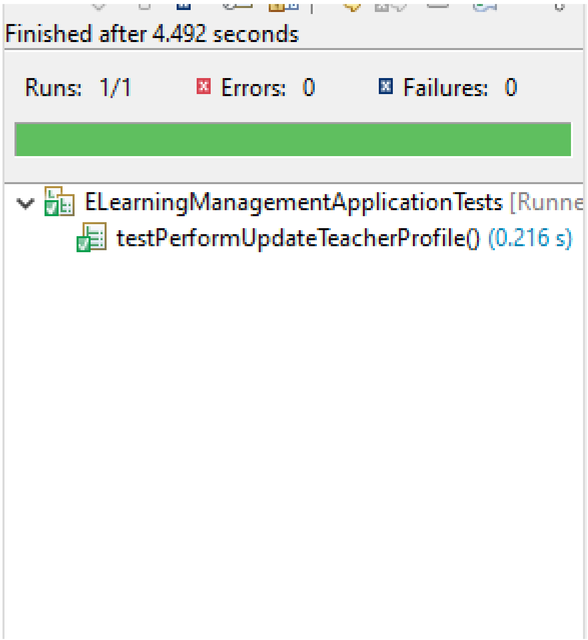
**Code :**

```
@Test
 @Order(10)
 void testPerformUpdateTeacherProfile() throws Exception {
   Teacher obj = new Teacher();
 obj.setTeacherid("110");
 obj.setTeachername("PravinKumar");
 obj.setGender("Male");
 obj.setEmail("pravinkumar3256@gmail.com");
 obj.setMobile("9875231460");
 obj.setExperience("3");
 obj.setDepartment("ECE");
 obj.setDegreecompleted("PHD");
 obj.setInstitutionname("VVE");
 obj.setPassword("Preethi@123456");
 obj.setStatus("accept");
   updateteacher = "Updated Successfully";
   result = teacherservice.updateteacherProfile(obj);
   assertEquals(updateteacher,result);
}
```

## Output :



```
Hibernate: select teacher0_.email as email1_3_0_, teacher0_.degreecompleted as degreeco2_3_0_, teacher0_.department as departme3_3_0_, teacher0_.exper
```

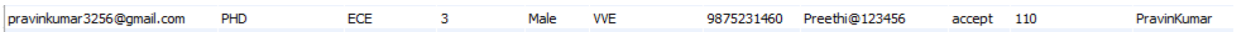


```
Finished after 4.492 seconds

Runs: 1/1      Errors: 0      Failures: 0

ELearningManagementApplicationTests [Runne
    testPerformUpdateTeacherProfile() (0.216 s)
```

## Database :



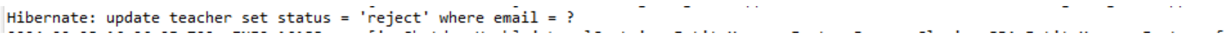
| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| pravinkumar3256@gmail.com | PHD | ECE | 3 | Male | VVE | 9875231460 | Preethi@123456 | accept | 110 | PravinKumar |

- **Update teacher status has reject :**

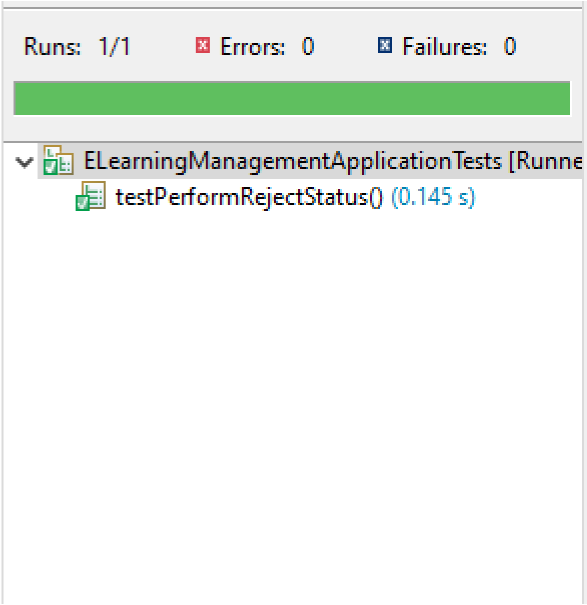
## Code :

```
@Test
        @Order(11)
        void testPerformRejectStatus() throws Exception {
          reject = "reject";
          result = teacherservice.rejectStatus(reject);
          assertEquals(reject,result);
        }
```

## Output :



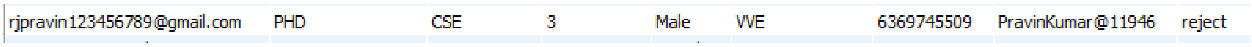
```
Hibernate: update teacher set status = 'reject' where email = ?
```



Runs: 1/1 Errors: 0 Failures: 0

```
ELearningManagementApplicationTests [Runne
    testPerformRejectStatus() (0.145 s)
```

## Database :



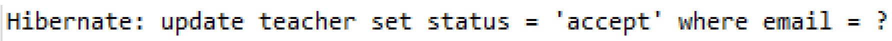
| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| rjpravin123456789@gmail.com | PHD | CSE | 3 | Male | WE | 6369745509 | PravinKumar@11946 | reject |

- **Update teacher status has accept :**

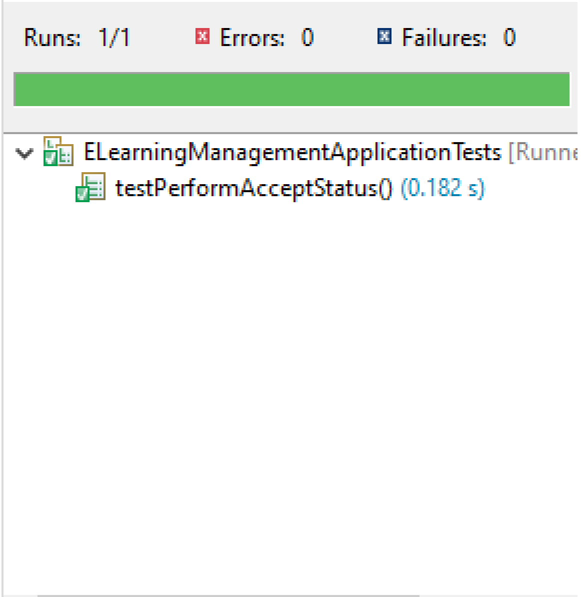
## Code :

```
@Test
        @Order(12)
        void testPerformAcceptStatus() throws Exception {
                accept = "accept";
                result = teacherservice.updateStatus(accept);
                assertEquals(accept,result);
        }
```

## Output :



```
Hibernate: update teacher set status = 'accept' where email = ?
```

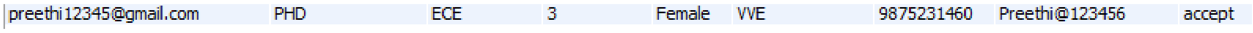


Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

🟩

ELearningManagementApplicationTests [Runne
- testPerformAcceptStatus() (0.182 s)

## Database :



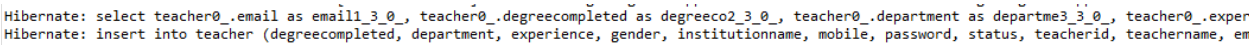
| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| preethi12345@gmail.com | PHD | ECE | 3 | Female | VVE | 9875231460 | Preethi@123456 | accept |

- **For save the new teacher :**

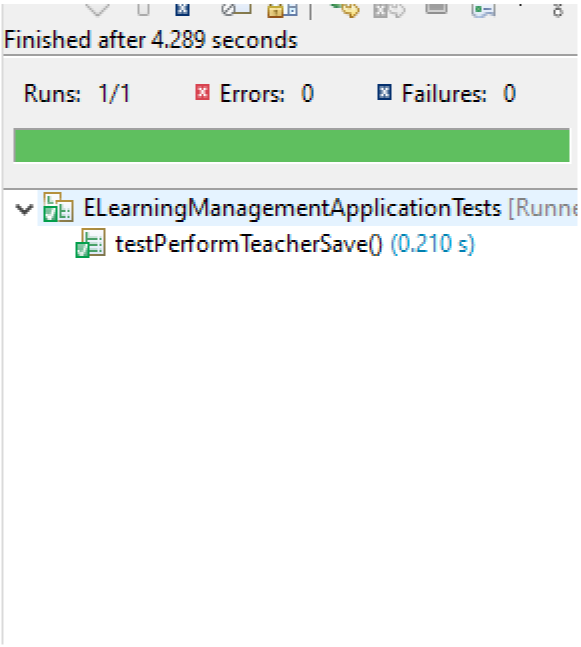
## Code :

```
@Test
 @Order(14)
 void testPerformTeacherSave() throws Exception {
   Teacher obj = new Teacher();
   obj.setTeacherid("6");
   obj.setTeachername("PravinKumar");
   obj.setGender("Male");
   obj.setEmail("pravinkumar321@gmail.com");
   obj.setMobile("9875231460");
   obj.setExperience("3");
   obj.setDepartment("ECE");
   obj.setDegreecompleted("PHD");
   obj.setInstitutionname("VVE");
   obj.setPassword("Preethi@123456");
   obj.setStatus("accept");
   teacherSave = "Registered Successfully";
   result = teacherservice.saveTeacher(obj);
   assertEquals(teacherSave,result);
}
```

## Output :



```
Hibernate: select teacher0_.email as email1_3_0_, teacher0_.degreecompleted as degreeco2_3_0_, teacher0_.department as departme3_3_0_, teacher0_.exper
Hibernate: insert into teacher (degreecompleted, department, experience, gender, institutionname, mobile, password, status, teacherid, teachername, em
```



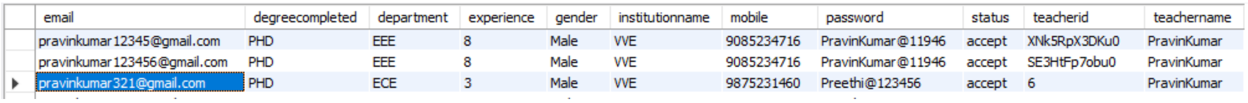
Finished after 4.289 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne

testPerformTeacherSave() (0.210 s)

## Database :



| | email | degreecompleted | department | experience | gender | institutionname | mobile | password | status | teacherid | teachername |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | pravinkumar12345@gmail.com | PHD | EEE | 8 | Male | VVE | 9085234716 | PravinKumar@11946 | accept | XNk5RpX3DKu0 | PravinKumar |
| | pravinkumar123456@gmail.com | PHD | EEE | 8 | Male | VVE | 9085234716 | PravinKumar@11946 | accept | SE3HtFp7obu0 | PravinKumar |
| ▶ | pravinkumar321@gmail.com | PHD | ECE | 3 | Male | VVE | 9875231460 | Preethi@123456 | accept | 6 | PravinKumar |

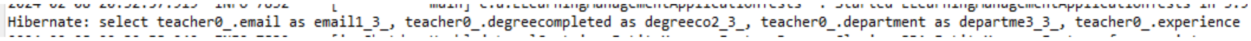
- **For get all the user(teacher) details :**

## Code :

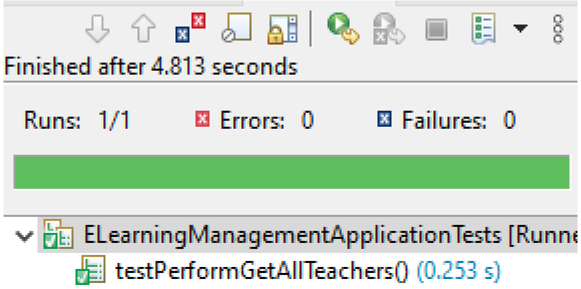
```
@Test
```

```
            @Order(22)
            void testPerformGetAllTeachers() throws Exception {
   list4 = teacherservice.getAllTeachers();
   teacher1 = new
Teacher("anish123@gmail.com","PHD","ECE","3","Male","VVS","0986432145","Anish@11946","accept"
,"yMf5dqrl0evc","Anish");
   teacher2 = new
Teacher("anitha1234@gmail.com","PHD","EEE","3","Female","VVS","0986432145","Anitha@11946","a
ccept","yMf5dqrl0oia3","Anitha");
   teacher3 = new
Teacher("pravinkumar123@gmail.com","PHD","CSE","3","Male","VVS","0986432145","PravinKumar@1
1946","accept","yMf5dqrl0ertys","PravinKumar");
   list5.add(teacher1);
   list5.add(teacher2);
   list5.add(teacher3);
   list6.add(teacher1);
   list6.add(teacher2);
   list6.add(teacher3);
   list5.equals(list6);
   assertEquals(list5, list6);
             }
```

**Output :**



```
Hibernate: select teacher0_.email as email1_3_, teacher0_.degreecompleted as degreeco2_3_, teacher0_.department as departme3_3_, teacher0_.experience
```



Finished after 4.813 seconds

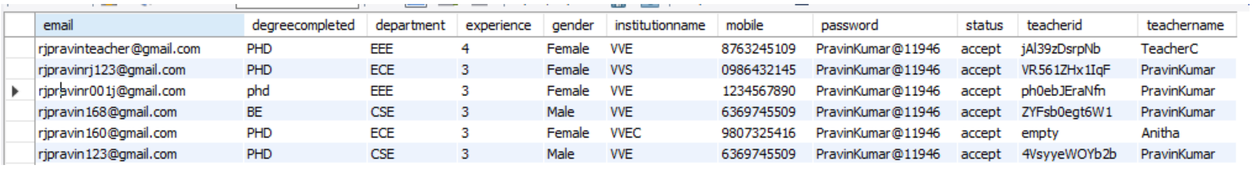
Runs: 1/1    ☒ Errors: 0    ☒ Failures: 0

🟩

ELearningManagementApplicationTests [Runne

testPerformGetAllTeachers() (0.253 s)

**Database :**



| | email | degreecompleted | department | experience | gender | institutionname | mobile | password | status | teacherid | teachername |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | rjpravinteacher@gmail.com | PHD | EEE | 4 | Female | VVE | 8763245109 | PravinKumar@11946 | accept | jAl39zDsrpNb | TeacherC |
| | rjpravinrj123@gmail.com | PHD | ECE | 3 | Female | VVS | 0986432145 | PravinKumar@11946 | accept | VR561ZHx1IqF | PravinKumar |
| ▶ | rjpravinr001j@gmail.com | phd | EEE | 3 | Female | VVE | 1234567890 | PravinKumar@11946 | accept | ph0ebJEraNfn | PravinKumar |
| | rjpravin168@gmail.com | BE | CSE | 3 | Male | VVE | 6369745509 | PravinKumar@11946 | accept | ZYFsb0egt6W1 | PravinKumar |
| | rjpravin160@gmail.com | PHD | ECE | 3 | Female | VVEC | 9807325416 | PravinKumar@11946 | accept | empty | Anitha |
| | rjpravin123@gmail.com | PHD | CSE | 3 | Male | VVE | 6369745509 | PravinKumar@11946 | accept | 4VsyyeWOYb2b | PravinKumar |

—-------------------------------------------------------------------------------------------------------------------------------

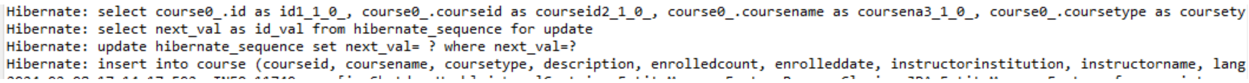
## 5. Course :

- **Save Course :**

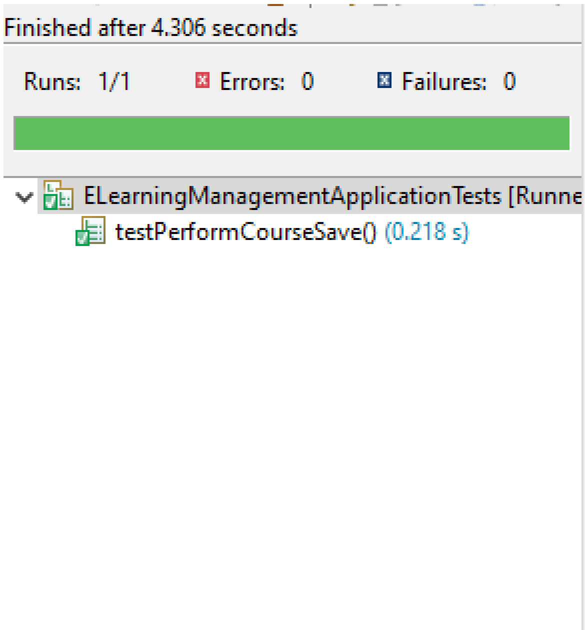
**Code :**

```
@Test
 @Order(15)
 void testPerformCourseSave() throws Exception {
   Course obj = new Course();
   obj.setCourseid("81");
   obj.setCoursename("Spring");
   obj.setCoursetype("Website");
   obj.setDescription("Sprin Boot for the freshers");
   obj.setEnrolledcount("0");
   obj.setEnrolleddate("03/09/2023");
   obj.setId(81);
   obj.setInstructorinstitution("VVE");
   obj.setInstructorname("PravinKumar");
   obj.setLanguage("English");
   obj.setSkilllevel("Basic");
   obj.setWebsiteurl("www.javatutorial.com");
   CourseSave = "Sucessfully Saved";
   result = courseservice.saveCourse(obj);
   assertEquals(CourseSave,result);
}
```

**Output :**



```
Hibernate: select course0_.id as id1_1_0_, course0_.courseid as courseid2_1_0_, course0_.coursename as coursena3_1_0_, course0_.coursetype as coursety
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: insert into course (courseid, coursename, coursetype, description, enrolledcount, enrolleddate, instructorinstitution, instructorname, lang
```



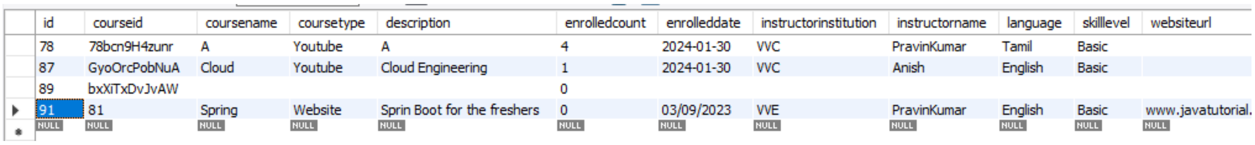
Finished after 4.306 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne

testPerformCourseSave() (0.218 s)

## Database :



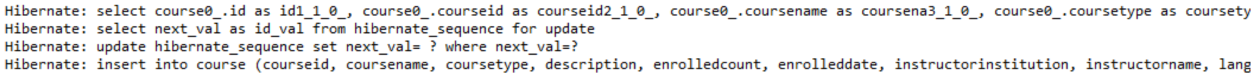
| id | courseid | coursename | coursetype | description | enrolledcount | enrolleddate | instructorinstitution | instructorname | language | skilllevel | websiteurl |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 78 | 78bcn9H4zunr | A | Youtube | A | 4 | 2024-01-30 | VVC | PravinKumar | Tamil | Basic | |
| 87 | GyoOrcPobNuA | Cloud | Youtube | Cloud Engineering | 1 | 2024-01-30 | VVC | Anish | English | Basic | |
| 89 | bxXiTxDvJvAW | | | | 0 | | | | | | |
| 91 | 81 | Spring | Website | Sprin Boot for the freshers | 0 | 03/09/2023 | VVE | PravinKumar | English | Basic | www.javatutorial. |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

- **Add new course :**

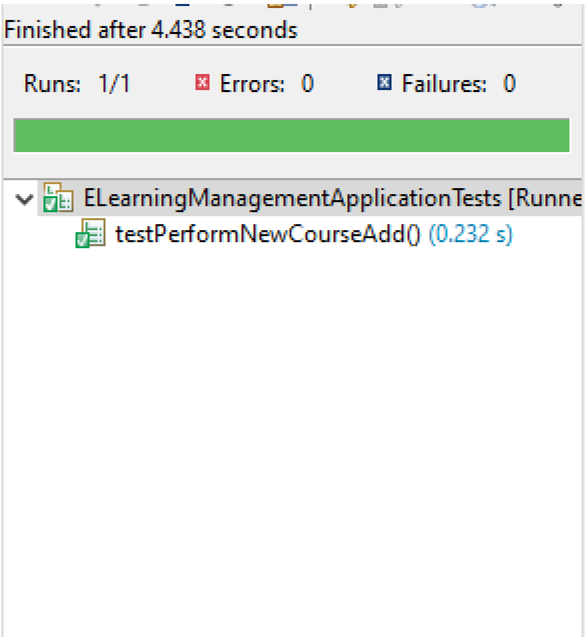
## Code :

```
@Test
 @Order(16)
 void testPerformNewCourseAdd() throws Exception {
   Course obj = new Course();
   obj.setCourseid("84");
   obj.setCoursename("Spring Boot");
   obj.setCoursetype("Website");
   obj.setDescription("Spring Boot for the freshers");
   obj.setEnrolledcount("0");
   obj.setEnrolleddate("06/09/2023");
   obj.setId(81);
   obj.setInstructorinstitution("VVE");
   obj.setInstructorname("PravinKumar");
   obj.setLanguage("English");
   obj.setSkilllevel("Basic");
   obj.setWebsiteurl("www.javatutorial.com");
   Courseadd = "Successfully added";
   result = courseservice.addNewCourse(obj);
   assertEquals(Courseadd,result);
}
```

## Output :



```
Hibernate: select course0_.id as id1_1_0_, course0_.courseid as courseid2_1_0_, course0_.coursename as coursena3_1_0_, course0_.coursetype as coursety
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: insert into course (courseid, coursename, coursetype, description, enrolledcount, enrolleddate, instructorinstitution, instructorname, lang
```



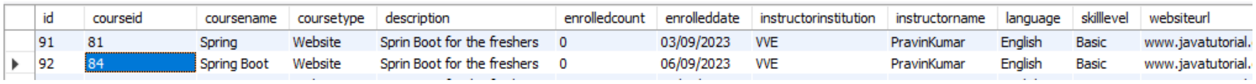
Finished after 4.438 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne

testPerformNewCourseAdd() (0.232 s)

## Database :



| | id | courseid | coursename | coursetype | description | enrolledcount | enrolleddate | instructorinstitution | instructorname | language | skilllevel | websiteurl |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 91 | 81 | Spring | Website | Sprin Boot for the freshers | 0 | 03/09/2023 | VVE | PravinKumar | English | Basic | www.javatutorial. |
| ▶ | 92 | 84 | Spring Boot | Website | Sprin Boot for the freshers | 0 | 06/09/2023 | VVE | PravinKumar | English | Basic | www.javatutorial. |

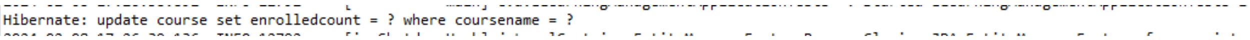
- **Update enrolled count for the particular course :**

## Code :

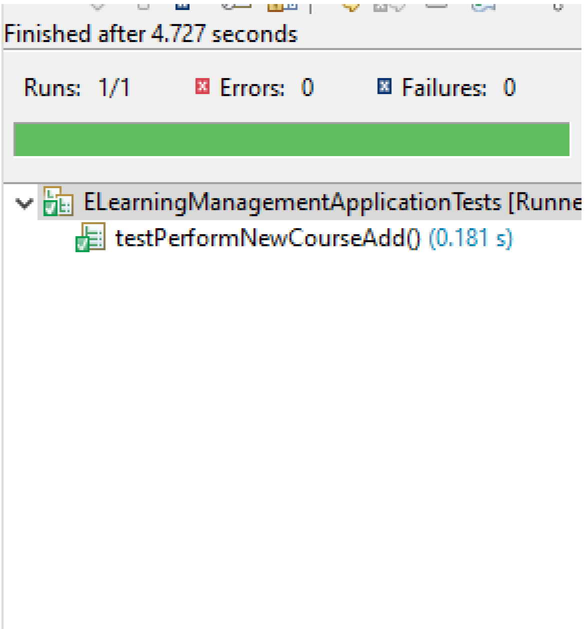
```
@Test
 @Order(17)
 void testPerformNewCourseAdd() throws Exception {
   Course obj = new Course();
   EnrolledUpdate = "Enrolled Updated";
   result = courseservice.updateEnrolledcount(Courseadd, 1);
```

```
  assertEquals(EnrolledUpdate,result);
}
```

## Output :



```
Hibernate: update course set enrolledcount = ? where coursename = ?
```

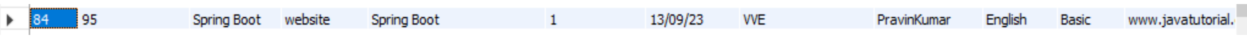


Finished after 4.727 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne
testPerformNewCourseAdd() (0.181 s)

## Database :



| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ▶ | 84 | 95 | Spring Boot | website | Spring Boot | 1 | 13/09/23 | VVE | PravinKumar | English | Basic | www.javatutorial. |

- **For get all the course details :**

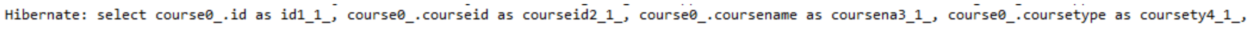
## Code :

```
@Test
        @Order(6)
        void testPerformGetAllCourse() throws Exception {
          list7 = courseservice.getAllCourses();
          course1 = new Course(84,"95","Spring Boot","website","Spring
Boot","1","13/09/23","VVE","PravinKumar","English","Basic","www.javatutorial.com",
"www.javatutorial.com");
          course2 = new
Course(85,"96","Spring","website","Spring","1","13/09/23","VVE","PravinKumar","English","Basic","www
.javatutorial.com", "www.javatutorial.com");
          course3 = new
Course(86,"98","Java","website","Java","1","13/09/23","VVE","PravinKumar","English","Basic","www.jav
atutorial.com", "www.javatutorial.com");
```

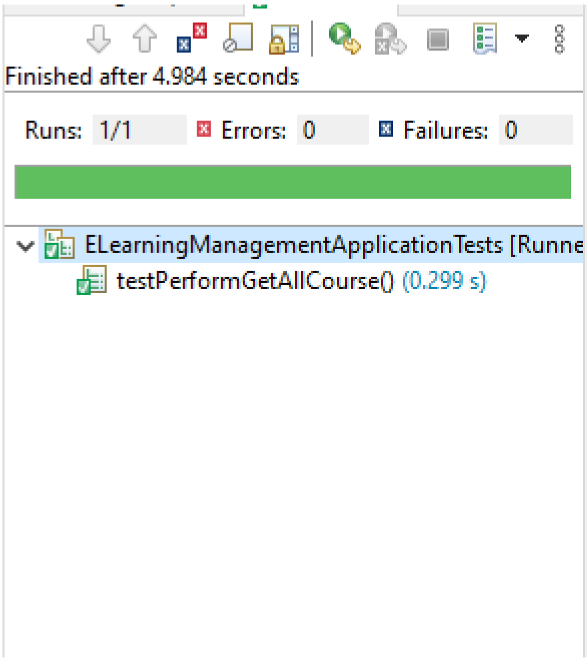
```
            list8.add(course1);
            list8.add(course2);
            list8.add(course3);
            list9.add(course1);
            list9.add(course2);
            list9.add(course3);
            list8.equals(list9);
            assertEquals(list8, list9);
          }

    list5.add(teacher1);
    list5.add(teacher2);
    list5.add(teacher3);
    list6.add(teacher1);
    list6.add(teacher2);
    list6.add(teacher3);
    list5.equals(list6);
    assertEquals(list5, list6);
          }
```

**Output :**



```
Hibernate: select course0_.id as id1_1_, course0_.courseid as courseid2_1_, course0_.coursename as coursena3_1_, course0_.coursetype as coursety4_1_,
```



Finished after 4.984 seconds

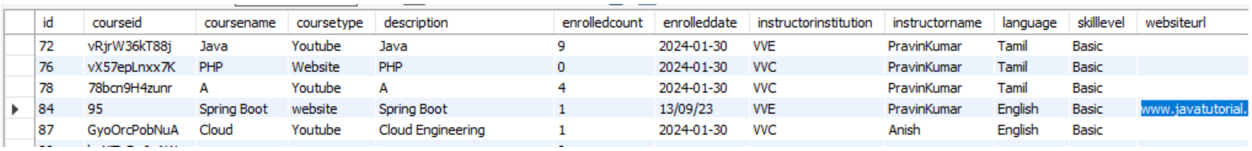
Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

🟩

ELearningManagementApplicationTests [Runne

testPerformGetAllCourse() (0.299 s)

## Database :



| | id | courseid | coursename | coursetype | description | enrolledcount | enrolleddate | instructorinstitution | instructorname | language | skilllevel | websiteurl |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 72 | vRjrW36kT88j | Java | Youtube | Java | 9 | 2024-01-30 | VVE | PravinKumar | Tamil | Basic | |
| | 76 | vX57epLnxx7K | PHP | Website | PHP | 0 | 2024-01-30 | VVC | PravinKumar | Tamil | Basic | |
| | 78 | 78bcn9H4zunr | A | Youtube | A | 4 | 2024-01-30 | VVC | PravinKumar | Tamil | Basic | |
| ▶ | 84 | 95 | Spring Boot | website | Spring Boot | 1 | 13/09/23 | VVE | PravinKumar | English | Basic | www.javatutorial. |
| | 87 | GyoOrcPobNuA | Cloud | Youtube | Cloud Engineering | 1 | 2024-01-30 | VVC | Anish | English | Basic | |

—------------------------------------------------------------------------------------------------------------------------

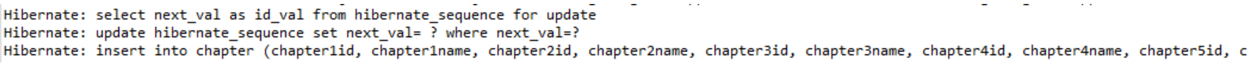
### 6. Chapter :

- **Save Chapter :**

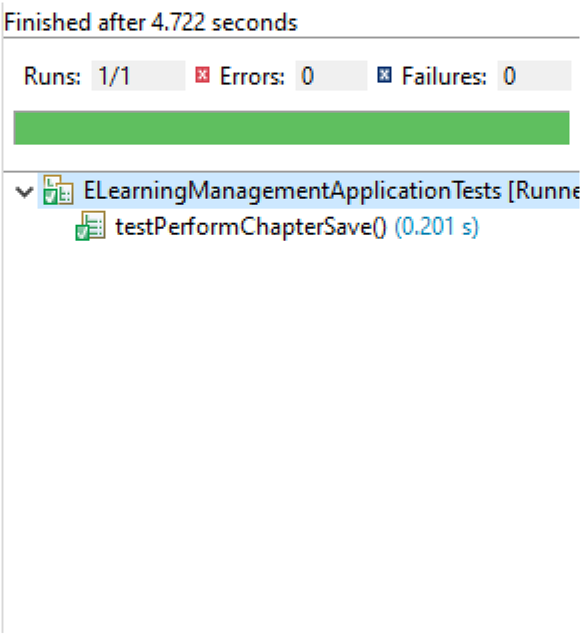
## Code :

```
@Test
 @Order(18)
 void testPerformChapterSave() throws Exception {
   Chapter obj = new Chapter();
   obj.setChapter1id("009");
   obj.setChapter1name("Spring Boot");
   ChapterSave = "Chapter saved";
   result = chapterservice.saveChapter(obj);
   assertEquals(ChapterSave,result);
}
```

## Output :



```
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: insert into chapter (chapter1id, chapter1name, chapter2id, chapter2name, chapter3id, chapter3name, chapter4id, chapter4name, chapter5id, c
```



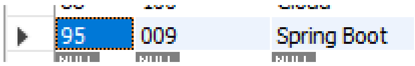
Finished after 4.722 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne

testPerformChapterSave() (0.201 s)

## Database :



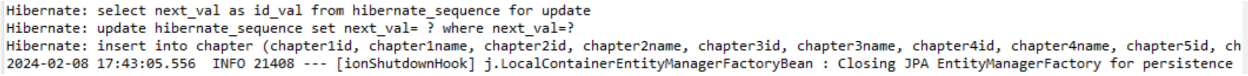
| | | |
|---|---|---|
| 95 | 009 | Spring Boot |

- **Add new chapter :**

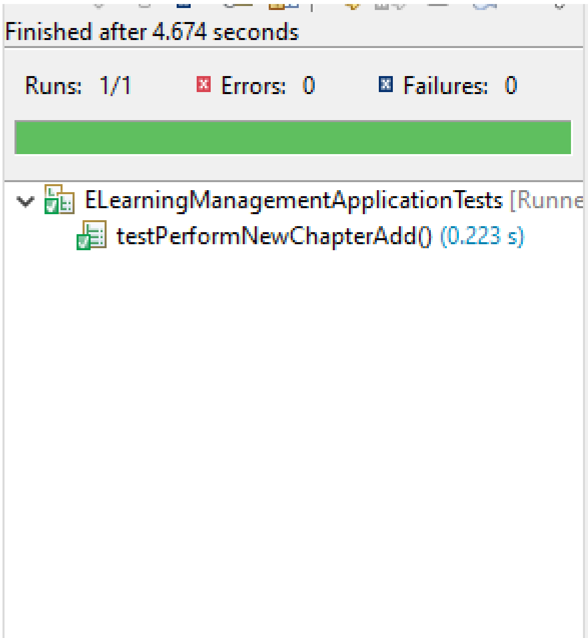
## Code :

```
        @Test
@Order(19)
void testPerformNewChapterAdd() throws Exception {
  Chapter obj = new Chapter();
  obj.setChapter1id("008");
  obj.setChapter1name("Angular");
  Chapteradd = "Chapter Added";
  result = chapterservice.addNewChapter(obj);
  assertEquals(Chapteradd,result);
}
```

## Output :



```
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: insert into chapter (chapter1id, chapter1name, chapter2id, chapter2name, chapter3id, chapter3name, chapter4id, chapter4name, chapter5id, ch
2024-02-08 17:43:05.556  INFO 21408 --- [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence
```

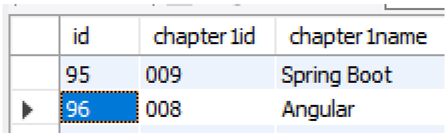


Finished after 4.674 seconds

Runs: 1/1    ☒ Errors: 0    ☒ Failures: 0

ELearningManagementApplicationTests [Runne
testPerformNewChapterAdd() (0.223 s)

**Database :**



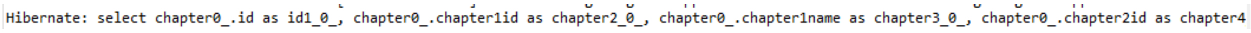
| | id | chapter1id | chapter1name |
|---|---|---|---|
| | 95 | 009 | Spring Boot |
| ▶ | 96 | 008 | Angular |

- **For get all the chapter details :**

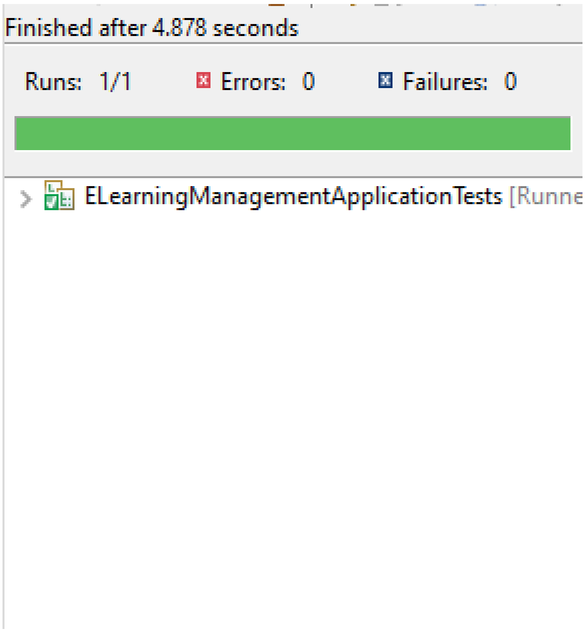
**Code :**

```
@Test
            @Order(24)
            void testPerformGetAllChapter() throws Exception {
               list10 = chapterservice.getAllChapters();
               chapter1 = new Chapter(73,"100","Java", null, null, null, null, null, null, null, null, null,
null, null, null, null, null, null);
               chapter2 = new Chapter(74,"111","Spring", null, null, null, null, null, null, null, null, null,
null, null, null, null, null, null);
               chapter3 = new Chapter(75,"98","Spring Boot", null, null, null, null, null, null, null, null,
null, null, null, null, null, null, null);            list11.add(chapter1);
               list11.add(chapter2);
               list11.add(chapter3);
               list14.add(chapter1);
               list14.add(chapter2);
               list14.add(chapter3);
               list11.equals(list14);
               assertEquals(list11, list14);
            }
```

**Output :**



```
Hibernate: select chapter0_.id as id1_0_, chapter0_.chapter1id as chapter2_0_, chapter0_.chapter1name as chapter3_0_, chapter0_.chapter2id as chapter4
```

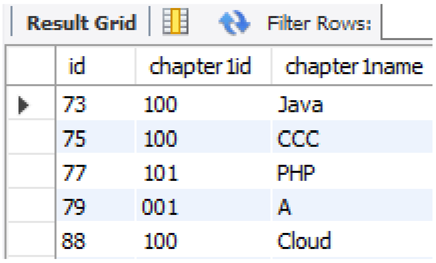


```
Finished after 4.878 seconds

Runs:  1/1      Errors:  0      Failures:  0

> ELearningManagementApplicationTests [Runne
```

**Database :**



| id | chapter1id | chapter1name |
|---|---|---|
| 73 | 100 | Java |
| 75 | 100 | CCC |
| 77 | 101 | PHP |
| 79 | 001 | A |
| 88 | 100 | Cloud |

---

## 7. Enrollment :

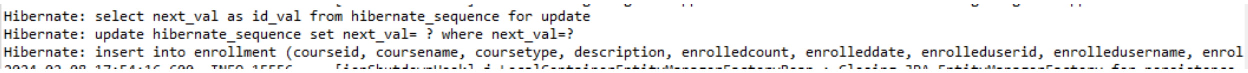
- **Save Enrollment :**

**Code :**

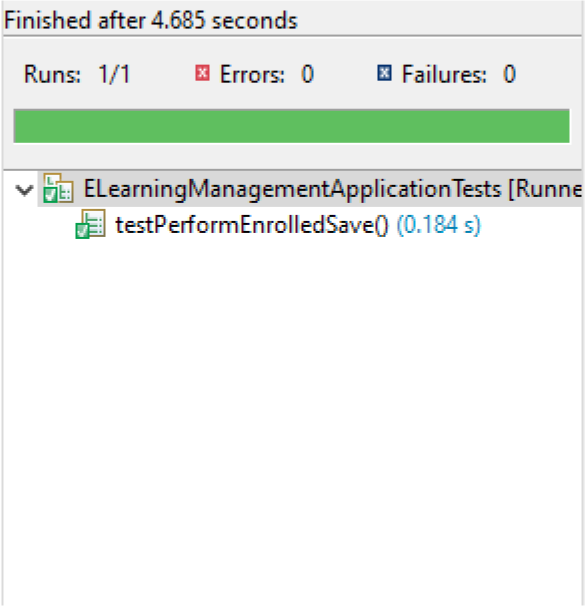
```
@Test
```

```
@Order(20)
void testPerformEnrolledSave() throws Exception {
  Enrollment obj = new Enrollment();
  obj.setCourseid("81");
  obj.setCoursename("Spring");
  obj.setCoursetype("website");
  obj.setDescription("Spring");
  obj.setEnrolledcount("1");
  obj.setEnrolleddate("03/09/23");
  obj.setEnrolleduserid("y1ms1mio2l6H");
  obj.setEnrolledusername("PravinKumar");
  obj.setEnrolledusertype("user");
  EnrollSave = "Enrolled saved";
  result = enrollementservice.saveEnrollment(obj);
  assertEquals(EnrollSave,result);
}
```

**Output :**



```
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: insert into enrollment (courseid, coursename, coursetype, description, enrolledcount, enrolleddate, enrolleduserid, enrolledusername, enrol
```



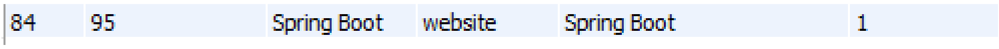
Finished after 4.685 seconds

Runs: 1/1  ☒ Errors: 0  ☒ Failures: 0

ELearningManagementApplicationTests [Runne

testPerformEnrolledSave() (0.184 s)

**Database :**



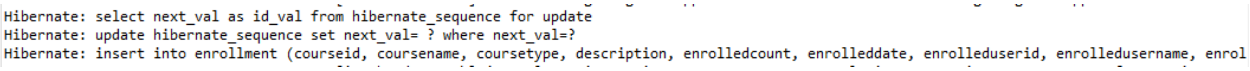
| | | | | | |
|---|---|---|---|---|---|
| 84 | 95 | Spring Boot | website | Spring Boot | 1 |

- **Add new enrolled :**

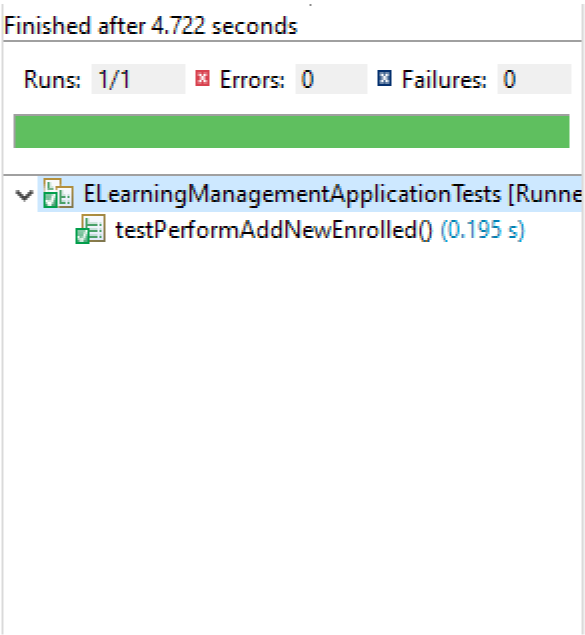
## Code :

```
@Test
 @Order(21)
 void testPerformAddNewEnrolled() throws Exception {
   Enrollment obj = new Enrollment();
   obj.setCourseid("89");
   obj.setCoursename("Java");
   obj.setCoursetype("website");
   obj.setDescription("Java");
   obj.setEnrolledcount("1");
   obj.setEnrolleddate("03/09/23");
   obj.setEnrolleduserid("y1ms1mio2l6H");
   obj.setEnrolledusername("PravinKumar");
   obj.setEnrolledusertype("user");
   Enrolladd = "Enrolled added";
   result = enrollementservice.addNewEnrollment(obj);
   assertEquals(Enrolladd,result);
}
```

## Output :



```
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: insert into enrollment (courseid, coursename, coursetype, description, enrolledcount, enrolleddate, enrolleduserid, enrolledusername, enrol
```



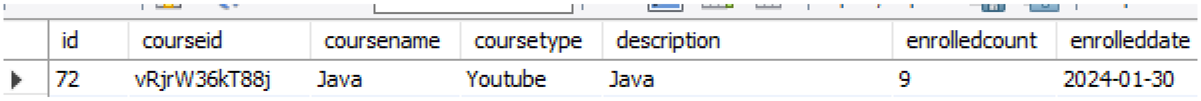
Finished after 4.722 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne

testPerformAddNewEnrolled() (0.195 s)

## Database :



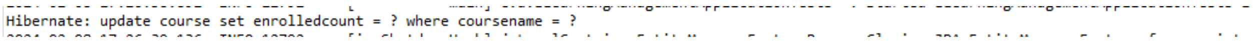
| | id | courseid | coursename | coursetype | description | enrolledcount | enrolleddate |
|---|---|---|---|---|---|---|---|
| ▶ | 72 | vRjrW36kT88j | Java | Youtube | Java | 9 | 2024-01-30 |

- **Update enrolled count for the particular course :**

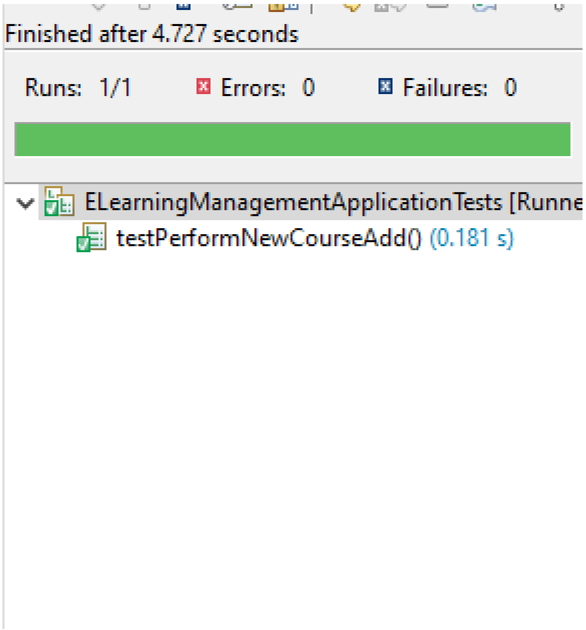
## Code :

```
@Test
 @Order(17)
 void testPerformNewCourseAdd() throws Exception {
   Course obj = new Course();
   EnrolledUpdate = "Enrolled Updated";
   result = courseservice.updateEnrolledcount(Courseadd, 1);
   assertEquals(EnrolledUpdate,result);
}
```

## Output :



```
Hibernate: update course set enrolledcount = ? where coursename = ?
```



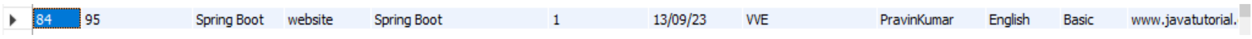
Finished after 4.727 seconds

Runs: 1/1 ☒ Errors: 0 ☒ Failures: 0

ELearningManagementApplicationTests [Runne

testPerformNewCourseAdd() (0.181 s)

## Database :



| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ▶ | 84 | 95 | Spring Boot | website | Spring Boot | 1 | 13/09/23 | VVE | PravinKumar | English | Basic | www.javatutorial. |

—-------------------------------------------------------------------------------------------------------------------------------