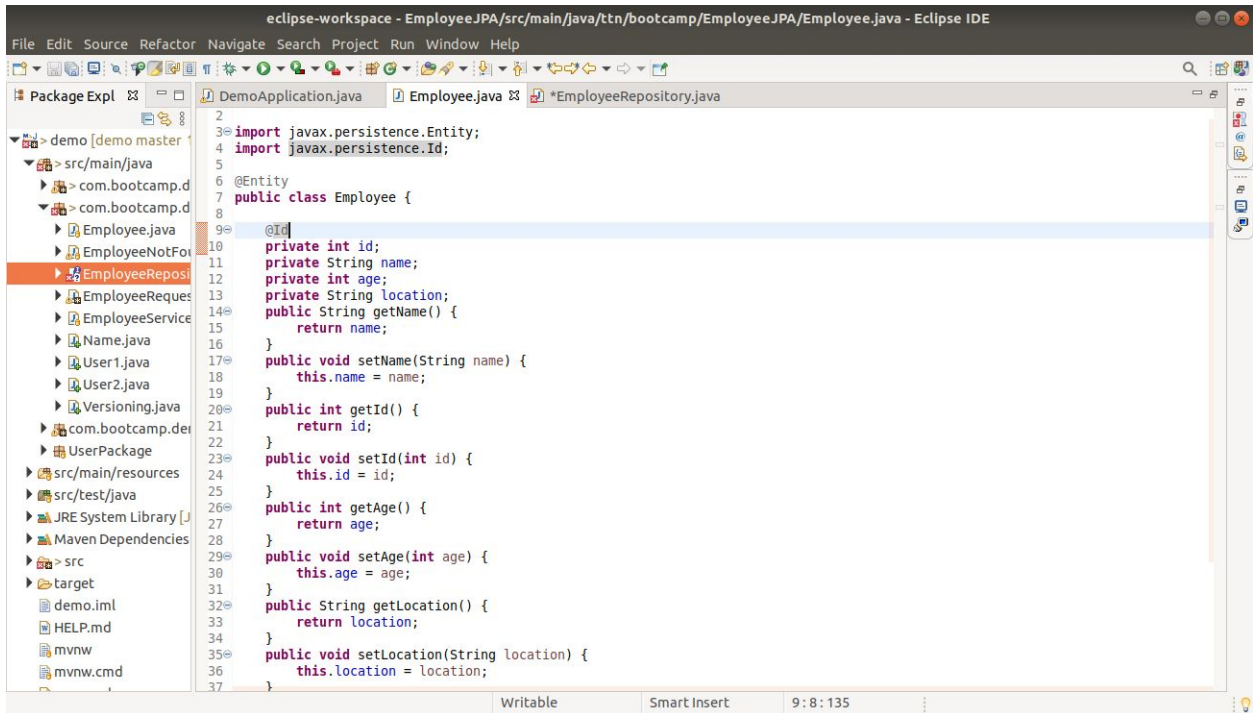
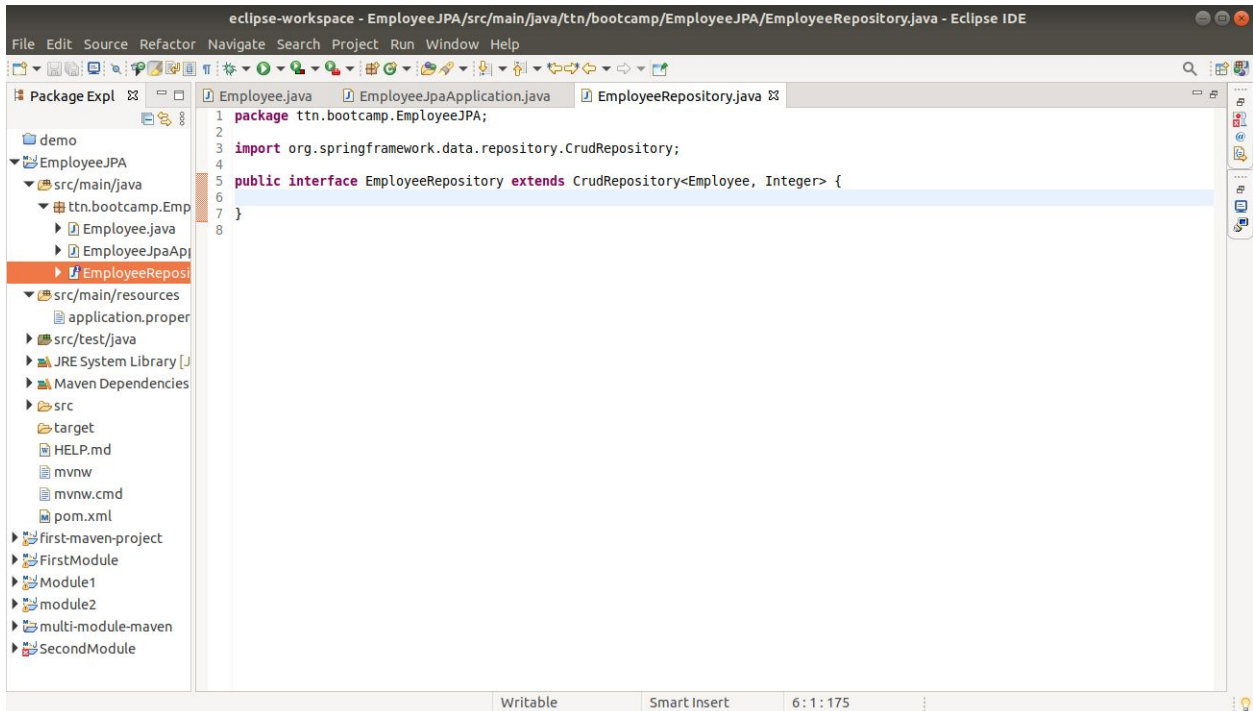


(1) Create an Employee Entity which contains following fields  
Name,Id,Age,Location



```
eclipse-workspace - EmployeeJPA/src/main/java/ttn/bootcamp/EmployeeJPA/Employee.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Expl DemoApplication.java Employee.java *EmployeeRepository.java
demo [demo master]
  src/main/java
    com.bootcamp.d
      com.bootcamp.d
        Employee.java
        EmployeeNotFor
        EmployeeReposi
        EmployeeReques
        EmployeeService
        Name.java
        User1.java
        User2.java
        Versioning.java
    com.bootcamp.der
    UserPackage
  src/main/resources
  src/test/java
  JRE System Library [J
  Maven Dependencies
  > src
  target
    demo.iml
    HELP.md
    mvnw
    mvnw.cmd
2
3 import javax.persistence.Entity;
4 import javax.persistence.Id;
5
6 @Entity
7 public class Employee {
8
9     @Id
10    private int id;
11    private String name;
12    private int age;
13    private String location;
14    public String getName() {
15        return name;
16    }
17    public void setName(String name) {
18        this.name = name;
19    }
20    public int getId() {
21        return id;
22    }
23    public void setId(int id) {
24        this.id = id;
25    }
26    public int getAge() {
27        return age;
28    }
29    public void setAge(int age) {
30        this.age = age;
31    }
32    public String getLocation() {
33        return location;
34    }
35    public void setLocation(String location) {
36        this.location = location;
37    }
38 }
```

(2) Set up EmployeeRepository with Spring Data JPA



```
eclipse-workspace - EmployeeJPA/src/main/java/ttn/bootcamp/EmployeeJPA/EmployeeRepository.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Expl Employee.java EmployeeJpaApplication.java EmployeeRepository.java
demo
  EmployeeJPA
    src/main/java
      ttn.bootcamp.Emp
        Employee.java
        EmployeeJpaAp
        EmployeeReposi
    src/main/resources
      application.proper
    src/test/java
    JRE System Library [J
    Maven Dependencies
    src
    target
      HELP.md
      mvnw
      mvnw.cmd
      pom.xml
  first-maven-project
  FirstModule
  Module1
  module2
  multi-module-maven
  SecondModule
1 package ttn.bootcamp.EmployeeJPA;
2
3 import org.springframework.data.repository.CrudRepository;
4
5 public interface EmployeeRepository extends CrudRepository<Employee, Integer> {
6
7 }
8
Writable Smart Insert 6:1:175
```

### (3) Perform Create Operation on Entity using Spring Data JPA

The screenshot shows the Eclipse IDE with the following components:

- Editor:** Displays `EmployeeJpaApplicationTests.java`. The code includes imports for `org.junit.jupiter.api.Test`, `org.springframework.beans.factory.annotation.Autowired`, and `org.springframework.boot.test.context.SpringBootTest`. The `@SpringBootTest` annotation is present. The `EmployeeJpaApplicationTests` class has an `@Autowired` `EmployeeRepository repo` and a `@Test` `create()` method. Inside `create()`, an `Employee` object is created with name "Mukul", age 23, and location "Delhi". It is then saved using `repo.save(emp)`, and a confirmation message is printed.
- Console:** Shows the output of the JUnit test. It includes logs for Hibernate dialect initialization, JPA platform initialization, and the successful execution of the `create()` test. The final output is `*****Employee Created*****`.

### (4) Perform Update Operation on Entity using Spring Data JPA

The screenshot shows the Eclipse IDE with the following components:

- Editor:** Displays `EmployeeJpaApplicationTests.java`. The `create()` method is now completed. A new `@Test` `update()` method is added. Inside `update()`, an `Employee` object is created with name "Rahul", age 23, and location "Delhi". It is then saved using `repo.save(emp)`, and a confirmation message is printed.
- Console:** Shows the output of the JUnit test. It includes logs for Hibernate dialect initialization, JPA platform initialization, and the successful execution of the `update()` test. The final output is `*****Employee Updated*****`.

eclipse-workspace - EmployeeJPA/src/test/java/ttn/bootcamp/EmployeeJPA/EmployeeJpaApplicationTests.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

EmployeeJpaApplication EmployeeRepository.java application.properties EmployeeJpaApplication EmployeeJPA/pom.xml

```
16 repo.save(emp);
17 System.out.println("*****Employee Created*****");
18
19 }
20 @Test
21 void update() {
22     Employee emp=new Employee("Rahul",2,23,"Delhi");
23     repo.save(emp);
24     System.out.println("*****Employee Updated*****");
25 }
26
27 @Test
28 void updateById() {
29     Employee emp=repo.findById(1).get();
30
31     emp.setName("Kanika");
32     repo.save(emp);
33     System.out.println("*****Employee Updated*****");
34 }
35
36 }
```

Console JUnit

```
<terminated> EmployeeJpaApplicationTests.updateById [JUnit] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (14-Mar-2021, 12:19:59 AM - 12:20:05 AM)
2021-03-14 00:20:04.422 INFO 12794 --- [main] o.n.e.t.j.p.i.JpaPlatformInitializer : Hibernate: using JpaPlatform implementation:
[org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2021-03-14 00:20:04.432 INFO 12794 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence uni
'default'
2021-03-14 00:20:04.985 INFO 12794 --- [main] t.b.E.EmployeeJpaApplicationTests : Started EmployeeJpaApplicationTests in 4.066 seconds (JV
running for 5.323)
Hibernate: select employee0_.id as id1_0_0_, employee0_.age as age2_0_0_, employee0_.location as location3_0_0_, employee0_.name as name4_0_0_ from employee
employee0_ where employee0_.id=?
Hibernate: select employee0_.id as id1_0_0_, employee0_.age as age2_0_0_, employee0_.location as location3_0_0_, employee0_.name as name4_0_0_ from employee
employee0_ where employee0_.id=?
*****Employee Updated*****
```

Writable Smart Insert 28:18:665

MySQL Workbench

unconnected Test

File Edit View Query Database Server Tools Scripting Help

Administration Schemas

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variable
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Query 1

```
1 use bootcamp;
2 show tables;
3 select * from employee;
4
5 ALTER TABLE Employee RENAME TO employee;
```

Result Grid

#	id	name	age	location
1	1	Kanika	23	Delhi
2	2	Rahul	23	Delhi
*	NULL	NULL	NULL	NULL

employee 8

History

#	Date	#	Time	SQL
1	2021-03-14	6	10:07:49 IST	use bootca...
2	2021-03-13	7	10:07:41 IST	use bootca... show tables
		8	10:07:21 IST	show tables
		9	10:07:06 IST	show datab...

Object Info Session

No object selected

Query Completed

Context Help Snippets

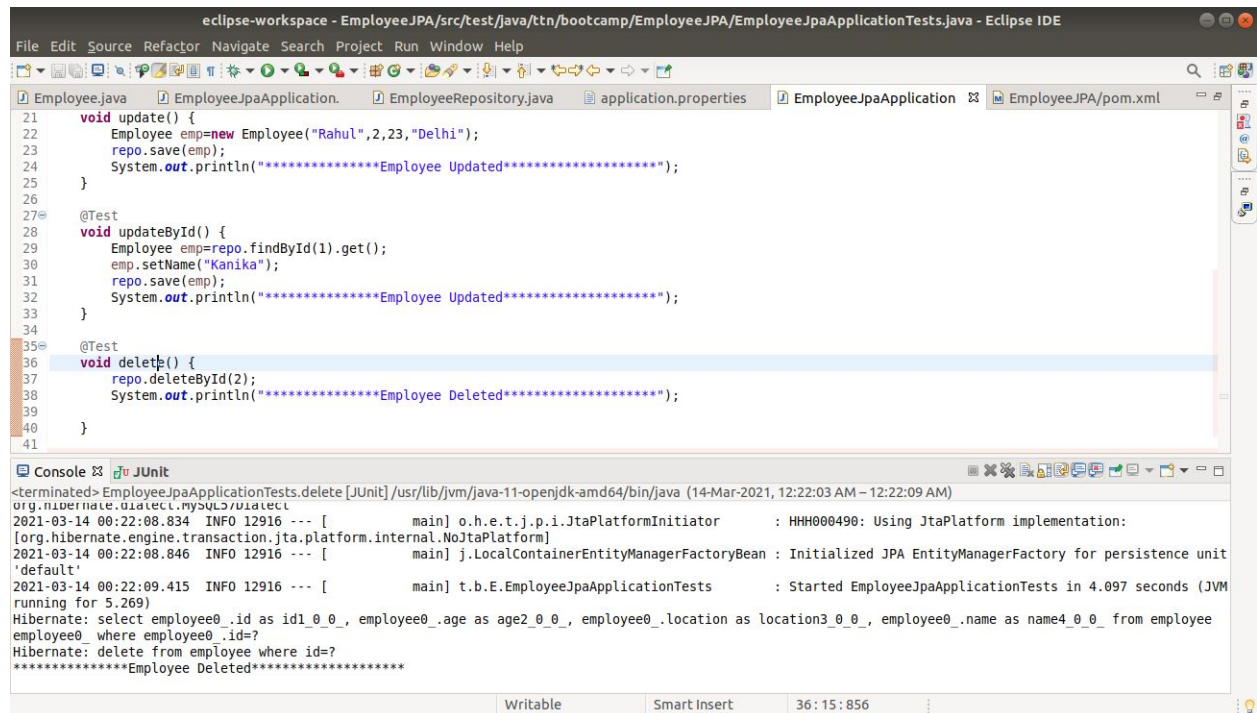
Automatic context help is disabled. The toolbar to manually get help at the current caret position or to toggle automatic help.

Result Grid Form Editor

Apply Revert



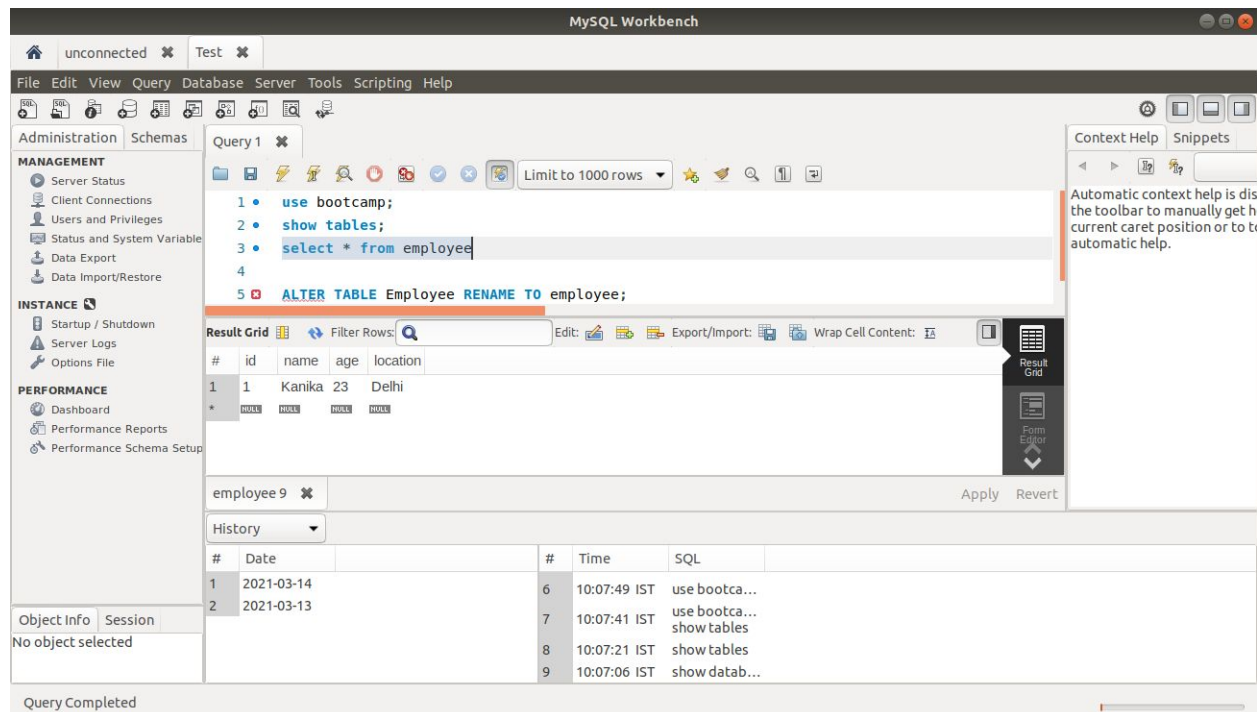
## (5) Perform Delete Operation on Entity using Spring Data JPA



The screenshot shows the Eclipse IDE with the file `EmployeeJpaApplicationTests.java` open. The `delete()` method is highlighted, which calls `repo.deleteById(2)` and prints "Employee Deleted". The console output shows the application running successfully, with logs for JUnit, Hibernate, and the application's main method. The console also shows the output of the `delete()` method.

```
21 void update() {
22     Employee emp=new Employee("Rahul",2,23,"Delhi");
23     repo.save(emp);
24     System.out.println("*****Employee Updated*****");
25 }
26
27 @Test
28 void updateById() {
29     Employee emp=repo.findById(1).get();
30     emp.setName("Kanika");
31     repo.save(emp);
32     System.out.println("*****Employee Updated*****");
33 }
34
35 @Test
36 void delete() {
37     repo.deleteById(2);
38     System.out.println("*****Employee Deleted*****");
39 }
40
41 }
```

```
<terminated> EmployeeJpaApplicationTests.delete [JUnit] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (14-Mar-2021, 12:22:09 AM)
org.hibernate.dialect.MySQL5Dialect
2021-03-14 00:22:08.834 INFO 12916 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation:
[org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2021-03-14 00:22:08.846 INFO 12916 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit
'default'
2021-03-14 00:22:09.415 INFO 12916 --- [main] t.b.E.EmployeeJpaApplicationTests : Started EmployeeJpaApplicationTests in 4.097 seconds (JVM
running for 5.269)
Hibernate: select employee0.id as id_0_0_, employee0.age as age2_0_0_, employee0.location as location3_0_0_, employee0.name as name4_0_0_ from employee
employee0 where employee0.id=?
Hibernate: delete from employee where id=?
*****Employee Deleted*****
```



The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
1 use bootcamp;
2 show tables;
3 select * from employee;
4
5 ALTER TABLE Employee RENAME TO employee;
```

The result grid shows the output of the `select * from employee;` query:

#	id	name	age	location
1	1	Kanika	23	Delhi

The console shows the execution history of the queries:

#	Date	#	Time	SQL
1	2021-03-14	6	10:07:49 IST	use bootca...
2	2021-03-13	7	10:07:41 IST	use bootca...
		8	10:07:21 IST	show tables
		9	10:07:06 IST	show datab...

## (5) Perform Read Operation on Entity using Spring Data JPA

The screenshot shows the MySQL Workbench interface. The 'Query 1' tab is active, displaying the following SQL script:

```
1 • use bootcamp;
2 • show tables;
3 • select * from employee;
4
5 ALTER TABLE Employee RENAME TO employee;
```

The 'Result Grid' shows the output of the 'select \* from employee;' query:

#	id	name	age	location
1	1	Kanika	23	Delhi
2	2	Gaurav	23	Delhi
3	3	Abhishek	33	UP

The 'History' tab shows the sequence of queries executed:

#	Date	Time	SQL
1	2021-03-14	10:07:49 IST	use bootca...
2	2021-03-13	10:07:41 IST	use bootca...
6		10:07:21 IST	show tables
7		10:07:06 IST	show datab...

The status bar at the bottom indicates 'Query Completed'.

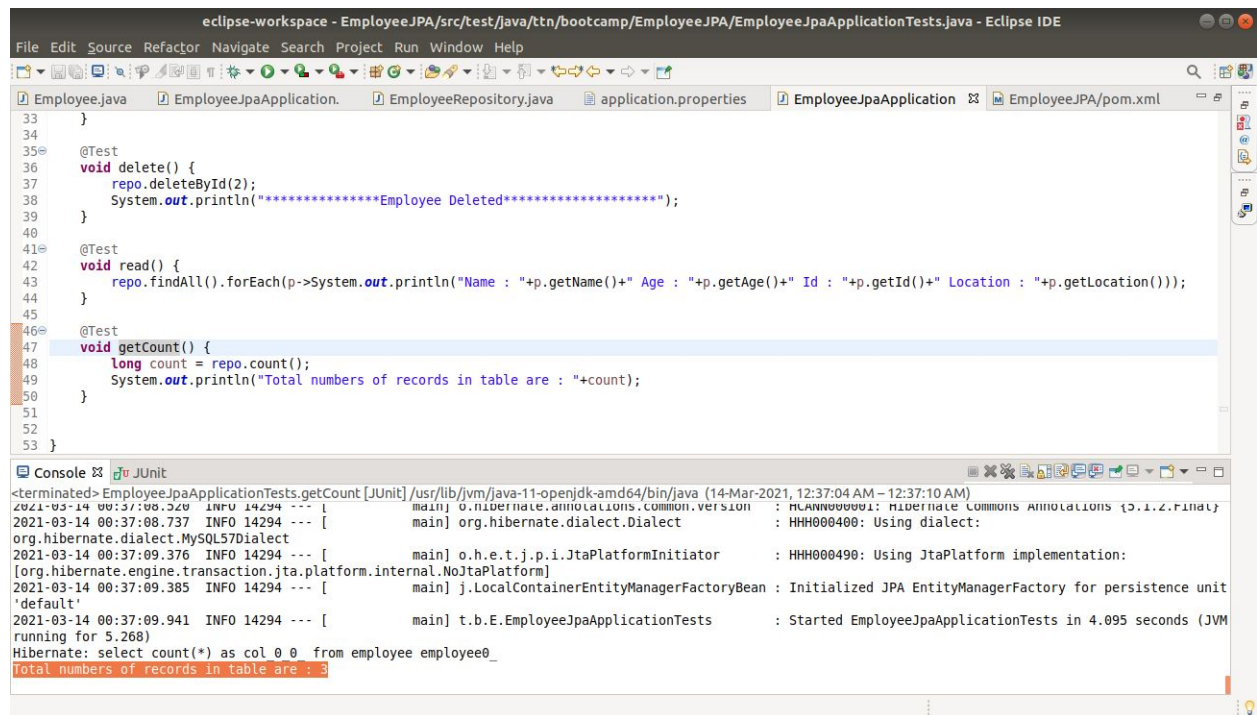
The screenshot shows the Eclipse IDE with the 'EmployeeJpaApplicationTests.java' file open. The code includes the following methods:

```
33 }
34
35 @Test
36 void delete() {
37     repo.deleteByid(2);
38     System.out.println("*****Employee Deleted*****");
39 }
40
41 @Test
42 void read() {
43     repo.findAll().forEach(p->System.out.println("Name : "+p.getName()+" Age : "+p.getAge()+" Id : "+p.getId()+" Location : "+p.getLocation()));
44 }
45
46 @Test
47 void getCount() {
48     long count = repo.count();
49     System.out.println("Total numbers of records in table are : "+count);
50 }
51
52
53 }
```

The 'Console' view shows the output of the 'read()' method:

```
<terminated> EmployeeJpaApplicationTests.read [JUnit] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (14-Mar-2021, 12:35:51 AM - 12:35:58 AM)
2021-03-14 00:35:57.512 INFO 14072 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490: using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2021-03-14 00:35:57.524 INFO 14072 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2021-03-14 00:35:58.128 INFO 14072 --- [main] t.b.E.EmployeeJpaApplicationTests : Started EmployeeJpaApplicationTests in 4.483 seconds (JVM running for 5.889)
Hibernate: select employee0_.id as id1_0_, employee0_.age as age2_0_, employee0_.location as location3_0_, employee0_.name as name4_0_ from employee employee0_
Name : Kanika Age : 23 Id : 1 Location : Delhi
Name : Gaurav Age : 23 Id : 2 Location : Delhi
Name : Abhishek Age : 33 Id : 3 Location : UP
```

## (6) Get the total count of the number of Employees

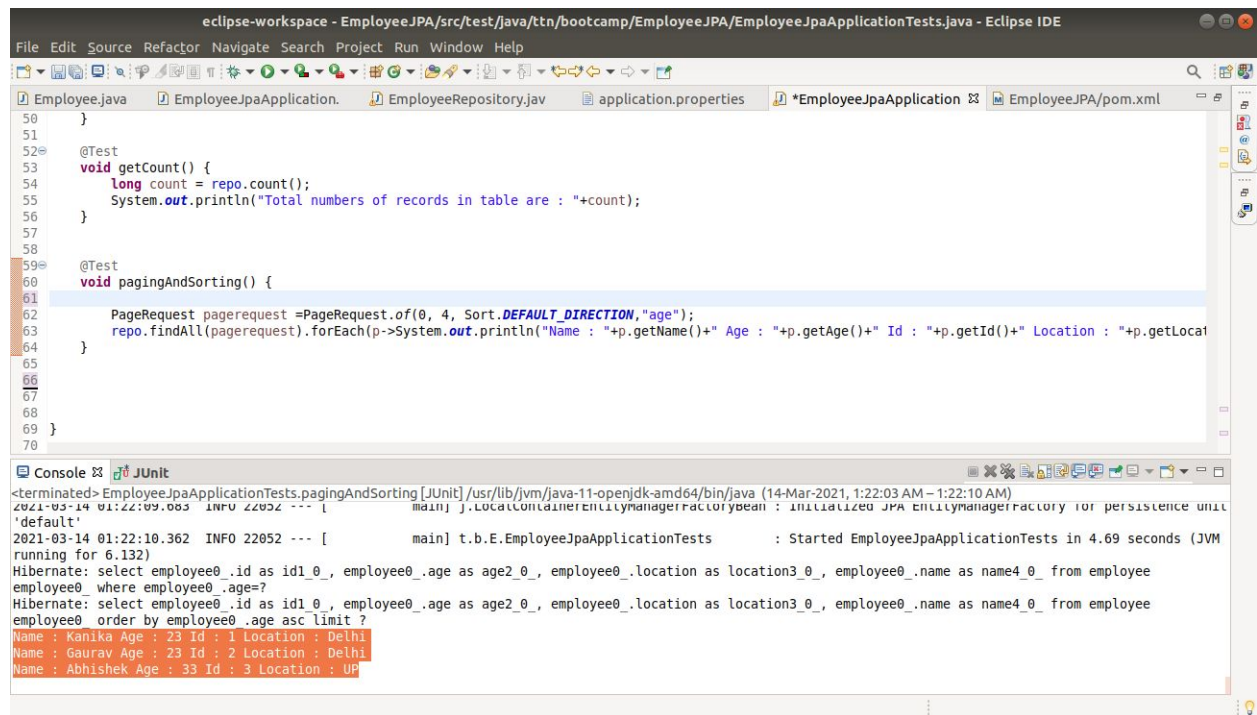


The screenshot shows the Eclipse IDE with the `EmployeeJpaApplicationTests.java` file open. The code includes three test methods: `delete()`, `read()`, and `getCount()`. The `getCount()` method is highlighted, showing it calls `repo.count()` and prints the total number of records. The console output at the bottom shows the execution of the `getCount` test, including Hibernate logs and the final output: `Total numbers of records in table are : 3`.

```
33 }
34
35 @Test
36 void delete() {
37     repo.deleteById(2);
38     System.out.println("*****Employee Deleted*****");
39 }
40
41 @Test
42 void read() {
43     repo.findAll().forEach(p->System.out.println("Name : "+p.getName()+" Age : "+p.getAge()+" Id : "+p.getId()+" Location : "+p.getLocation()));
44 }
45
46 @Test
47 void getCount() {
48     long count = repo.count();
49     System.out.println("Total numbers of records in table are : "+count);
50 }
51
52
53 }
```

```
<terminated> EmployeeJpaApplicationTests.getCount [JUnit] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (14-Mar-2021, 12:37:04 AM - 12:37:10 AM)
2021-03-14 00:37:00.320 INFO 14294 --- [main] o.hibernate.annotations.common.Version : HHH0000001: Hibernate Commons Annotations {5.1.2.Final}
2021-03-14 00:37:08.737 INFO 14294 --- [main] org.hibernate.dialect.Dialect : HHH000400: Using dialect: org.hibernate.dialect.MySQL57Dialect
2021-03-14 00:37:09.376 INFO 14294 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2021-03-14 00:37:09.385 INFO 14294 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2021-03-14 00:37:09.941 INFO 14294 --- [main] t.b.E.EmployeeJpaApplicationTests : Started EmployeeJpaApplicationTests in 4.095 seconds (JVM running for 5.268)
Hibernate: select count(*) as col_0_0_ from employee employee0_
Total numbers of records in table are : 3
```

## (7) Implement Pagination and Sorting on the bases of Employee Age



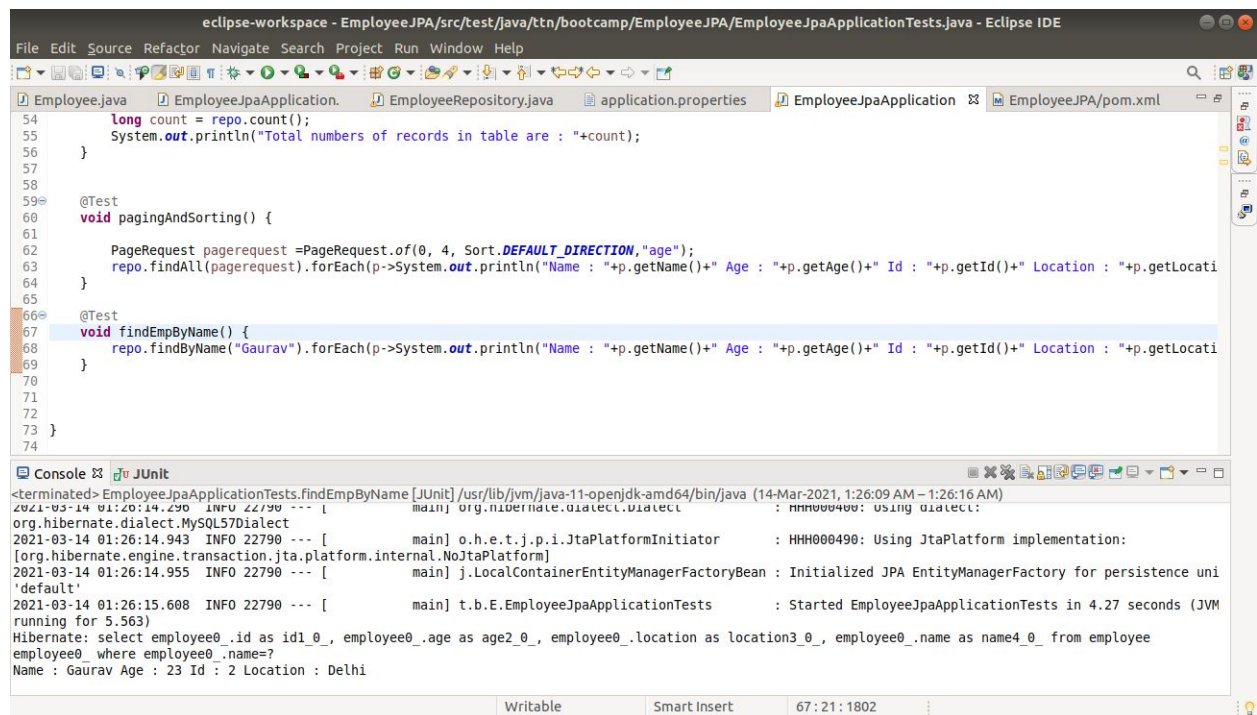
```
eclipse-workspace - EmployeeJPA/src/test/java/ttn/bootcamp/EmployeeJPA/EmployeeJpaApplicationTests.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Employee.java EmployeeJpaApplication EmployeeRepository.java application.properties *EmployeeJpaApplication EmployeeJPA/pom.xml

50 }
51
52 @Test
53 void getCount() {
54     long count = repo.count();
55     System.out.println("Total numbers of records in table are : "+count);
56 }
57
58
59 @Test
60 void pagingAndSorting() {
61
62     PageRequest pagerequest =PageRequest.of(0, 4, Sort.DEFAULT_DIRECTION,"age");
63     repo.findAll(pagerequest).forEach(p->System.out.println("Name : "+p.getName()+" Age : "+p.getAge()+" Id : "+p.getId()+" Location : "+p.getLocation()));
64 }
65
66
67
68 }
69
70

Console JUnit
<terminated> EmployeeJpaApplicationTests.pagingAndSorting [JUnit] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (14-Mar-2021, 1:22:03 AM - 1:22:10 AM)
2021-03-14 01:22:09.005 INFO 22052 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2021-03-14 01:22:10.362 INFO 22052 --- [main] t.b.E.EmployeeJpaApplicationTests : Started EmployeeJpaApplicationTests in 4.69 seconds (JVM running for 6.132)
Hibernate: select employee0_.id as id1_0_, employee0_.age as age2_0_, employee0_.location as location3_0_, employee0_.name as name4_0_ from employee employee0_ where employee0_.age=?
Hibernate: select employee0_.id as id1_0_, employee0_.age as age2_0_, employee0_.location as location3_0_, employee0_.name as name4_0_ from employee employee0_ order by employee0_.age asc limit ?
Name : Kahika Age : 23 Id : 1 Location : Delhi
Name : Gaurav Age : 23 Id : 2 Location : Delhi
Name : Abhishek Age : 33 Id : 3 Location : UP
```

## (8) Create and use finder to find Employee by Name



```
eclipse-workspace - EmployeeJPA/src/test/java/ttn/bootcamp/EmployeeJPA/EmployeeJpaApplicationTests.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Employee.java EmployeeJpaApplication EmployeeRepository.java application.properties EmployeeJpaApplication EmployeeJPA/pom.xml

54     long count = repo.count();
55     System.out.println("Total numbers of records in table are : "+count);
56 }
57
58
59 @Test
60 void pagingAndSorting() {
61
62     PageRequest pagerequest =PageRequest.of(0, 4, Sort.DEFAULT_DIRECTION,"age");
63     repo.findAll(pagerequest).forEach(p->System.out.println("Name : "+p.getName()+" Age : "+p.getAge()+" Id : "+p.getId()+" Location : "+p.getLocation()));
64 }
65
66 @Test
67 void findEmpByName() {
68     repo.findByName("Gaurav").forEach(p->System.out.println("Name : "+p.getName()+" Age : "+p.getAge()+" Id : "+p.getId()+" Location : "+p.getLocation()));
69 }
70
71
72
73 }
74

Console JUnit
<terminated> EmployeeJpaApplicationTests.findEmpByName [JUnit] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (14-Mar-2021, 1:26:09 AM - 1:26:16 AM)
2021-03-14 01:26:14.290 INFO 22790 --- [main] org.hibernate.dialect.MySQL57Dialect : hibernate.dialect: using dialect:
2021-03-14 01:26:14.943 INFO 22790 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation:
[org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2021-03-14 01:26:14.955 INFO 22790 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2021-03-14 01:26:15.608 INFO 22790 --- [main] t.b.E.EmployeeJpaApplicationTests : Started EmployeeJpaApplicationTests in 4.27 seconds (JVM running for 5.563)
Hibernate: select employee0_.id as id1_0_, employee0_.age as age2_0_, employee0_.location as location3_0_, employee0_.name as name4_0_ from employee employee0_ where employee0_.name=?
Name : Gaurav Age : 23 Id : 2 Location : Delhi
Writable Smart Insert 67:21:1802
```



(9) Create and use finder to find Employees starting with A character

The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying a SQL query: `1 use bootcamp;`, `2 show tables;`, `3 select * from employee;`, `4`, and `5 ALTER TABLE Employee RENAME TO employee;`. The 'Result Grid' shows the output of the query, listing columns: #, id, name, age, location. The results are: 1 Kanika 23 Delhi, 2 Gaurav 23 Delhi, 3 Aman 31 Mumbai, 4 Amit 30 UP. The 'History' tab shows the sequence of queries executed, including the same query as above. The 'Object Info' tab is also visible, showing 'No object selected'.

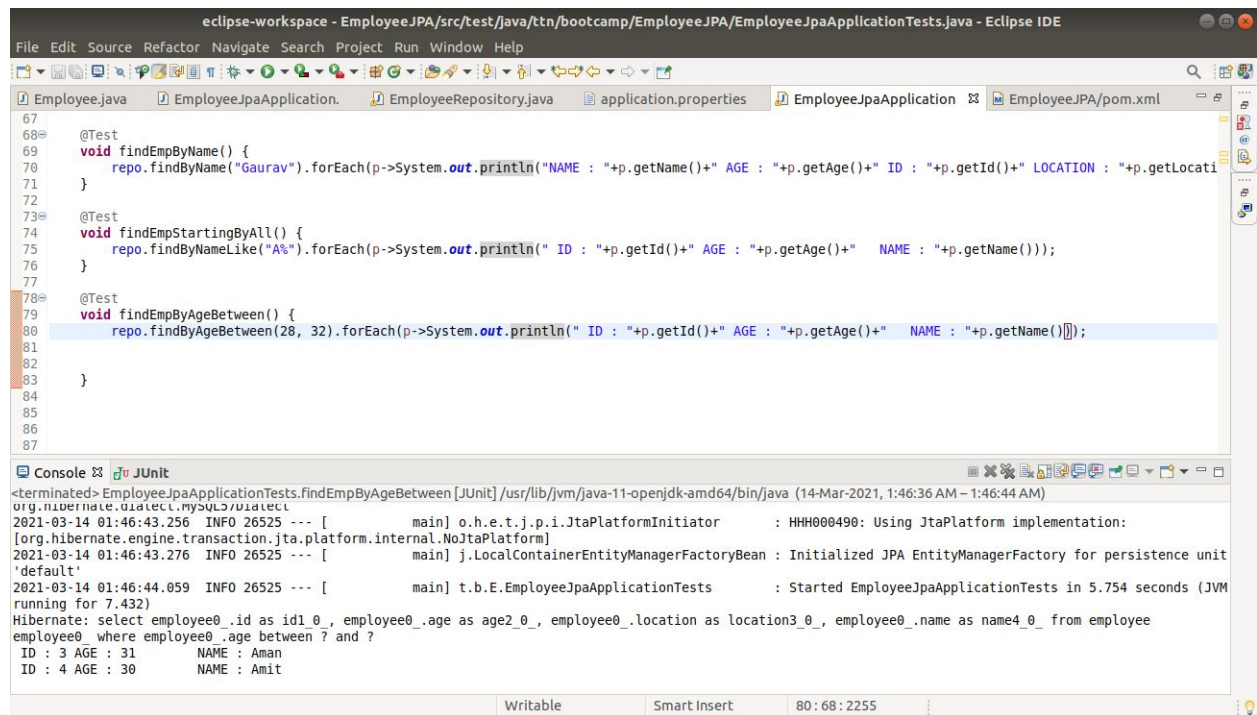
#	id	name	age	location
1	1	Kanika	23	Delhi
2	2	Gaurav	23	Delhi
3	3	Aman	31	Mumbai
4	4	Amit	30	UP

#	Date	#	Time	SQL
1	2021-03-14	6	10:07:49 IST	use bootca...
2	2021-03-13	7	10:07:41 IST	use bootca...
		8	10:07:21 IST	show tables
		9	10:07:06 IST	show datab...

The screenshot shows the Eclipse IDE interface. The 'EmployeeJpaApplicationTests.java' file is open, showing the following code: `57`, `58`, `59 @Test`, `60 void pagingAndSorting() {`, `61`, `62 PageRequest pagerequest =PageRequest.of(0, 4, Sort.DEFAULT_DIRECTION,"age");`, `63 repo.findAll(pagerequest).forEach(p->System.out.println("Name : "+p.getName()+" Age : "+p.getAge()+" Id : "+p.getId()+" Location : "+p.getLocati`, `64 }`, `65`, `66 @Test`, `67 void findEmpByName() {`, `68 repo.findByName("Gaurav").forEach(p->System.out.println("NAME : "+p.getName()+" AGE : "+p.getAge()+" ID : "+p.getId()+" LOCATION : "+p.getLocati`, `69 }`, `70`, `71 @Test`, `72 void findEmpStartingByAll() {`, `73 repo.findByNameLike("A%").forEach(p->System.out.println(" ID : "+p.getId()+" AGE : "+p.getAge()+" NAME : "+p.getName()));`, `74 }`, `75`, `76`, `77`. The 'Console' tab shows the output of the test, including the following log messages: `<terminated> EmployeeJpaApplicationTests.findEmpStartingByAll [JUnit] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (14-Mar-2021, 1:38:15 AM - 1:38:21 AM)`, `org.hibernate.dialect.MySQL5Dialect`, `2021-03-14 01:38:20.615 INFO 25059 --- [ main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation:`, `[org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]`, `2021-03-14 01:38:20.627 INFO 25059 --- [ main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence uni`, `'default'`, `2021-03-14 01:38:21.283 INFO 25059 --- [ main] t.b.E.EmployeeJpaApplicationTests : Started EmployeeJpaApplicationTests in 4.419 seconds (JV`, `running for 5.821)`, `Hibernate: select employee0_.id as id1_0_, employee0_.age as age2_0_, employee0_.location as location3_0_, employee0_.name as name4_0_ from employee`, `employee0_ where employee0_.name like ? escape ?`, `ID : 3 AGE : 31 NAME : Aman`, `ID : 4 AGE : 30 NAME : Amit`.



(10) Create and use finder to find Employees Between the age of 28 to 32



The screenshot shows the Eclipse IDE interface. The main editor displays the `EmployeeJpaApplicationTests.java` file. The code includes three test methods: `findEmpByName()`, `findEmpStartingByAll()`, and `findEmpByAgeBetween()`. The `findEmpByAgeBetween()` method is currently selected and highlighted. It uses `repo.findByAgeBetween(28, 32)` to find employees. The console output at the bottom shows the execution of the tests, including log messages from JUnit and Hibernate. The final output shows two employees: ID 3, AGE 31, NAME Aman; and ID 4, AGE 30, NAME Amit.

```
67
68 @Test
69 void findEmpByName() {
70     repo.findByName("Gaurav").forEach(p->System.out.println("NAME : "+p.getName()+" AGE : "+p.getAge()+" ID : "+p.getId()+" LOCATION : "+p.getLocation()));
71 }
72
73 @Test
74 void findEmpStartingByAll() {
75     repo.findByNameLike("A%").forEach(p->System.out.println(" ID : "+p.getId()+" AGE : "+p.getAge()+" NAME : "+p.getName()));
76 }
77
78 @Test
79 void findEmpByAgeBetween() {
80     repo.findByAgeBetween(28, 32).forEach(p->System.out.println(" ID : "+p.getId()+" AGE : "+p.getAge()+" NAME : "+p.getName()));
81 }
82
83
84
85
86
87
```

Console Output:

```
<terminated> EmployeeJpaApplicationTests.findEmpByAgeBetween [JUnit] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (14-Mar-2021, 1:46:36 AM - 1:46:44 AM)
org.hibernate.dialect.MySQL5Dialect
2021-03-14 01:46:43.256 INFO 26525 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation:
[org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2021-03-14 01:46:43.276 INFO 26525 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2021-03-14 01:46:44.059 INFO 26525 --- [main] t.b.E.EmployeeJpaApplicationTests : Started EmployeeJpaApplicationTests in 5.754 seconds (JVM running for 7.432)
Hibernate: select employee0_.id as id1_0_, employee0_.age as age2_0_, employee0_.location as location3_0_, employee0_.name as name4_0_ from employee
employee0_ where employee0_.age between ? and ?
ID : 3 AGE : 31 NAME : Aman
ID : 4 AGE : 30 NAME : Amit
```