

Kishan Mishra

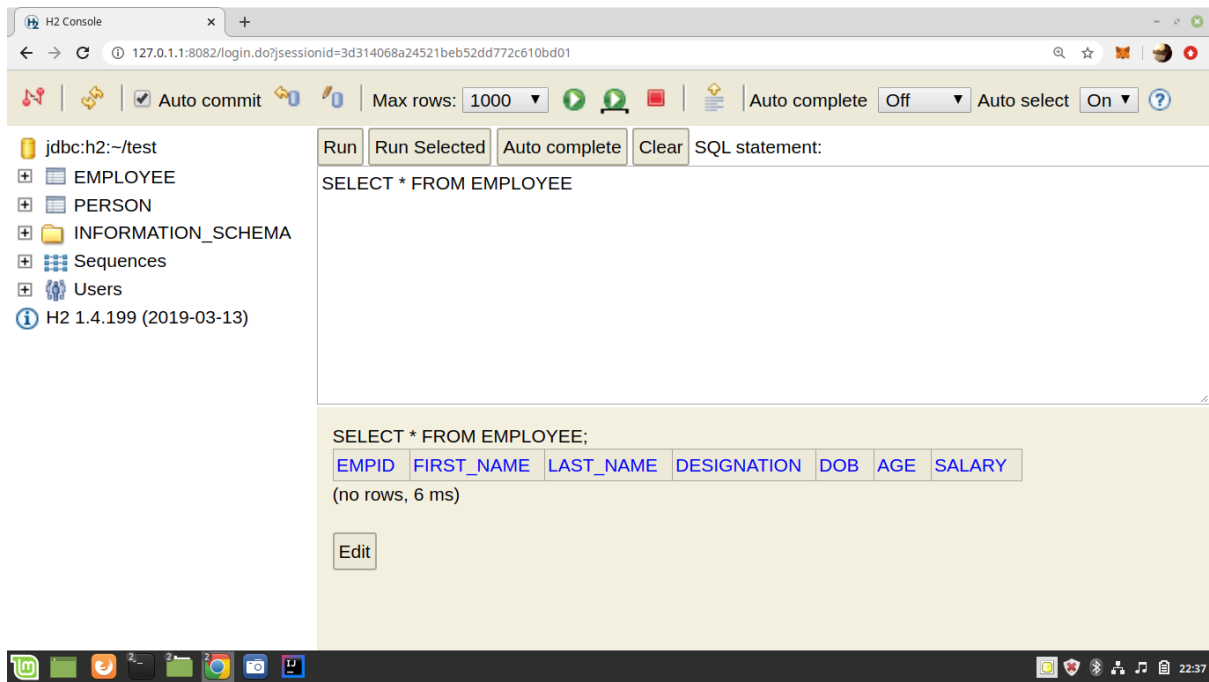
Email : [kishan.mishra@accolitedigital.com](mailto:kishan.mishra@accolitedigital.com)

## Hibernate 1 Assignment

### Employee management with CRUD in Hibernate

```
Employee {  
    id  
    fname  
    lname  
    age  
    salary  
    DOB  
    Designation  
}
```

- Menu:
1. Create employee
  2. Update all employee
  3. Update employee by id
  4. Delete employee - warning: this cannot be recovered
  5. Get all employee
  6. Get Employee by id
  7. exit Enter your choice (1-6)



Employee table empty (Schema)



User Menu and creating 1 record in the table

The screenshot shows the console output of a Java application. It begins with Hibernate initialization logs, including the dialect (H2Dialect) and the creation of the Employee table. The application then displays a menu with seven options: 1. Create employee, 2. Update employee by id, 3. Update all employees record, 4. Delete employee, 5. Get all employee, 6. Get Employee by id, and 7. exit. The user has entered '1' for the first option.

```
Run: Manager1 x
Feb 28, 2021 9:01:29 PM org.hibernate.dialect.Dialect <init>
INFO: HHH000400: Using dialect: org.hibernate.dialect.H2Dialect
Hibernate: drop table if exists Employee CASCADE
Hibernate: create table Employee (EmpId integer generated by default as identity, First_name varchar(255), Last_name varchar(255), Designation varchar(255),
Feb 28, 2021 9:01:30 PM org.hibernate.resource.transaction.backend.jdbc.internal.DdlTransactionIsolatorNonJtaImpl getIsolatedConnection
INFO: HHH10001501: Connection obtained from JdbcConnectionAccess [org.hibernate.engine.jdbc.env.internal.JdbcEnvironmentInitiator$ConnectionProviderJdbcConn
Feb 28, 2021 9:01:30 PM org.hibernate.resource.transaction.backend.jdbc.internal.DdlTransactionIsolatorNonJtaImpl getIsolatedConnection
INFO: HHH10001501: Connection obtained from JdbcConnectionAccess [org.hibernate.engine.jdbc.env.internal.JdbcEnvironmentInitiator$ConnectionProviderJdbcConn
Feb 28, 2021 9:01:30 PM org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformInitiator initiateService
INFO: HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]

1. Create employee
2. Update employee by id
3. Update all employees record
4. Delete employee
5. Get all employee
6. Get Employee by id
7. exit
Enter your choice (1-7)
|
```

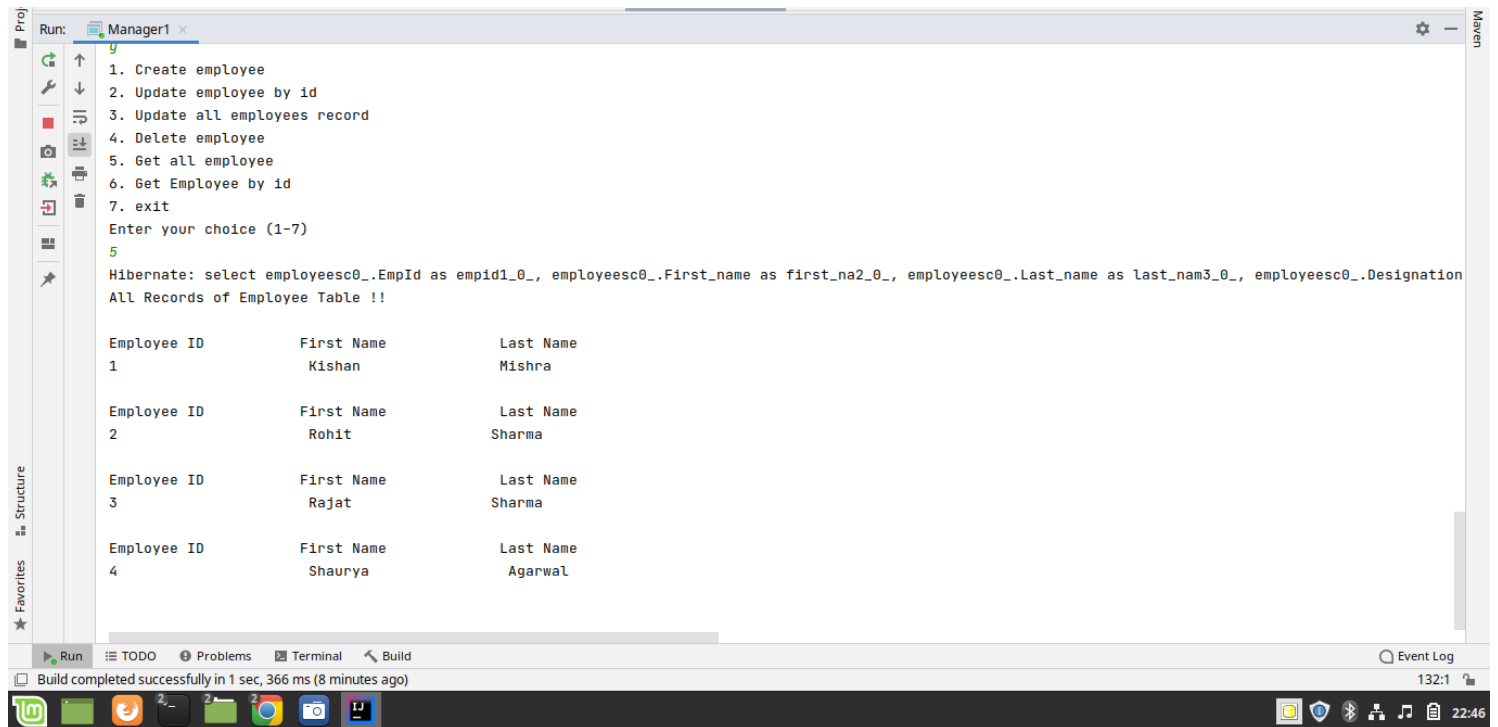
## User Menu

The screenshot shows the IDE with the 'hibernate1 - Manager1.java' file open. The code includes a menu and a case for option 6, 'Get Employee by id'. The console output shows the user selecting option 6, entering the ID '1', and the application displaying the details of the employee with ID 1: Kishan Mishra, born on 06/11/1999, age 22, designation Intern, and salary 20000. The application asks if the user wants to continue.

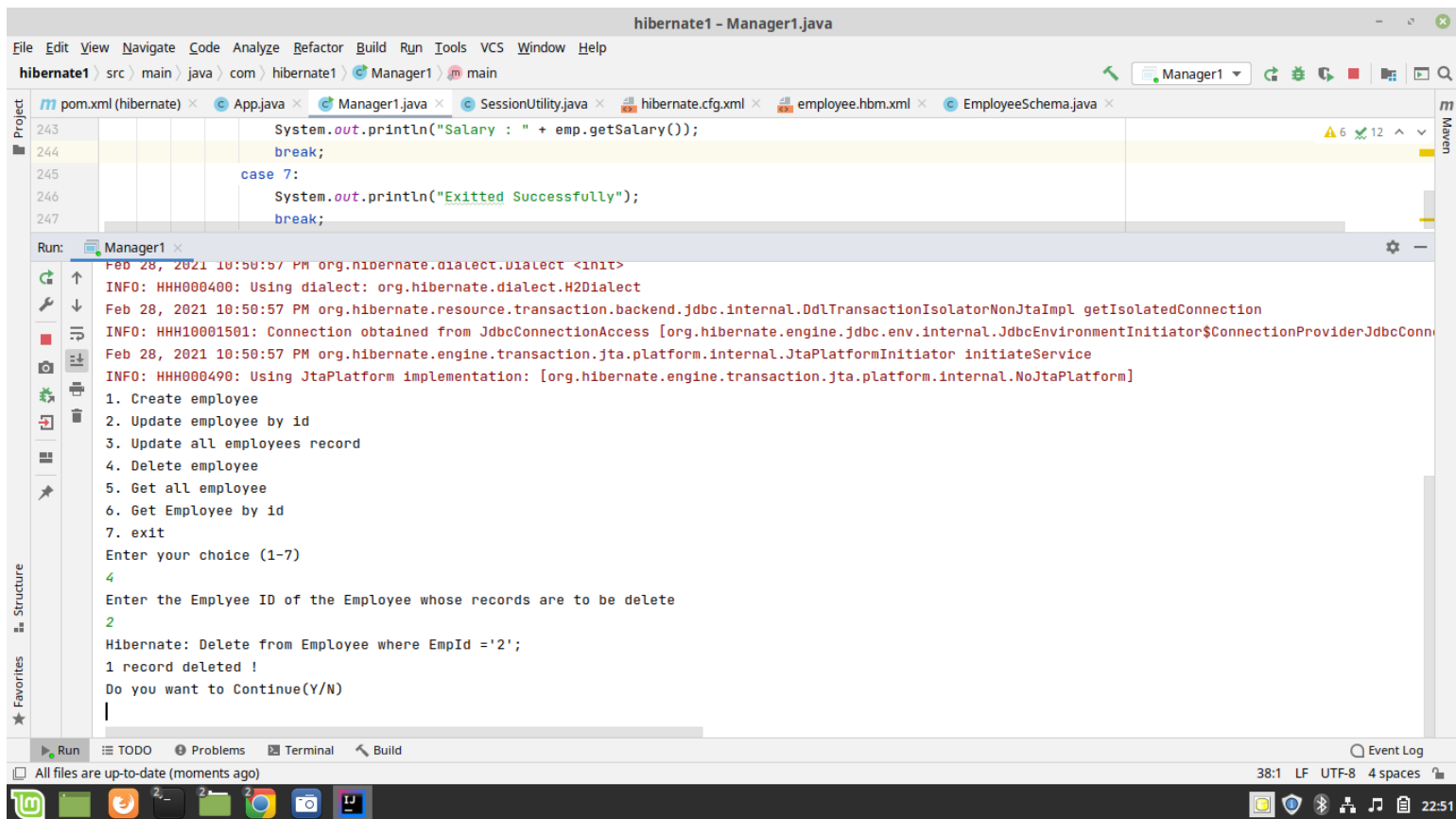
```
hibernate1 - Manager1.java
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
hibernate1 > src > main > java > com > hibernate1 > Manager1 > main
pom.xml (hibernate) x App.java x Manager1.java x SessionUtility.java x hibernate.cfg.xml x employee.hbm.xml x EmployeeSchema.java x
243 System.out.println("Salary : " + emp.getSalary());
244 break;
245 case 7:
246 System.out.println("Exited Successfully");
247 break;

Run: Manager1 x
1. Create employee
2. Update employee by id
3. Update all employees record
4. Delete employee
5. Get all employee
6. Get Employee by id
7. exit
Enter your choice (1-7)
6
Enter id of employee whose records are to be fetched
1
ID = 1
Hibernate: select employeeesc0_.EmpId as empid1_0_0_, employeeesc0_.First_name as first_na2_0_0_, employeeesc0_.Last_name as last_nam3_0_0_, employeeesc0_.Desig
First Name :Kishan
Last Name : Mishra
Date of Birth : 06/11/1999
Age : 22
Designation : Intern
Salary : 20000
Do you want to Continue(Y/N)
```

## Get Employee by Id



Get all Employees



Delete Employee by ID

H2 Console

127.0.1.1:8082/login.do?sessionId=3d314068a24521beb52dd772c610bd01

Auto commit ☒ Max rows: 1000 Auto complete Off Auto select On

jdbc:h2:~/test

- EMPLOYEE
- PERSON
- INFORMATION\_SCHEMA
- Sequences
- Users
- H2 1.4.199 (2019-03-13)

Run Run Selected Auto complete Clear SQL statement:

SELECT \* FROM EMPLOYEE

SELECT \* FROM EMPLOYEE;

EMPID	FIRST_NAME	LAST_NAME	DESIGNATION	DOB	AGE	SALARY
1	Kishan	Mishra	Intern	06/11/1999	22	20000
3	Rajat	Sharma	Intern	19/09/1998	23	240000
4	Shaurya	Agarwal	Intern	19/07/1998	23	250000

(3 rows, 4 ms)

Edit

Table before update all

H2 Console

127.0.1.1:8082/login.do?sessionId=3d314068a24521beb52dd772c610bd01

Auto commit ☒ Max rows: 1000 Auto complete Off Auto select On

jdbc:h2:~/test

- EMPLOYEE
- PERSON
- INFORMATION\_SCHEMA
- Sequences
- Users
- H2 1.4.199 (2019-03-13)

Run Run Selected Auto complete Clear SQL statement:

SELECT \* FROM EMPLOYEE

SELECT \* FROM EMPLOYEE;

EMPID	FIRST_NAME	LAST_NAME	DESIGNATION	DOB	AGE	SALARY
1	Kishan	Mishra	Software Engineer	06/11/1999	22	20000
3	Rajat	Sharma	Software Engineer	19/09/1998	23	240000
4	Shaurya	Agarwal	Software Engineer	19/07/1998	23	250000

(3 rows, 5 ms)

Edit

Table after update all

```
Run: Manager1 x
Feb 28, 2021 10:53:08 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl$PooledConnections <init>
INFO: HHH000115: Hibernate connection pool size: 20 (min=1)
Feb 28, 2021 10:53:08 PM org.hibernate.dialect.Dialect <init>
INFO: HHH000400: Using dialect: org.hibernate.dialect.H2Dialect
Feb 28, 2021 10:53:09 PM org.hibernate.resource.transaction.backend.jdbc.internal.DdlTransactionIsolatorNonJtaImpl getIsolatedConnection
INFO: HHH10001501: Connection obtained from JdbcConnectionAccess [org.hibernate.engine.jdbc.env.internal.JdbcEnvironmentInitiator$ConnectionProviderJdbcConn
Feb 28, 2021 10:53:09 PM org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformInitiator initiateService
INFO: HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]

1. Create employee
2. Update employee by id
3. Update all employees record
4. Delete employee
5. Get all employee
6. Get Employee by id
7. exit
Enter your choice (1-7)
3
Updating the designation of all employees
Hibernate: Update Employee SET Designation = 'Software Engineer' where Designation = 'Intern'
All records updated
Do you want to Continue(Y/N)
|
```

Build completed successfully in 2 sec, 499 ms (moments ago)

Update all execution

```
Run: Manager1 x
1. Create employee
2. Update employee by id
3. Update all employees record
4. Delete employee
5. Get all employee
6. Get Employee by id
7. exit
Enter your choice (1-7)
2
Enter Employee Id whose details are to be updated
1
Enter the attribute to update:
1- Employee ID
2- First Name
3- Last Name
4- Age
5- Designation
6- Date of Birth
7- Salary
7
Enter the updated Salary of Employee
25000
Hibernate: Update Employee SET Salary = '25000' where EmpId = '1';
Salary of employee with ID = 1Updated
Updating the designation of all employees
Hibernate: Update Employee SET Designation = 'Software Engineer' where Designation = 'Intern'
All records updated
```

Build completed successfully in 3 sec, 840 ms (3 minutes ago)

## Update Employee by ID (Updating Salary)



```
Run: Manager1 x
All records updated
Do you want to Continue(Y/N)
y
1. Create employee
2. Update employee by id
3. Update all employees record
4. Delete employee
5. Get all employee
6. Get Employee by id
7. exit
Enter your choice (1-7)
2
Enter Employee Id whose details are to be updated
1
Enter the attribute to update:
1- Employee ID
2- First Name
3- Last Name
4- Age
5- Designation
6- Date of Birth
7- Salary
1
Enter New ID
3195
Hibernate: Update Employee SET EmpId ='3195' where EmpId ='1';
Employee ID successfully updated !
Updating the designation of all employees
```

Build completed successfully in 3 sec, 840 ms (5 minutes ago)

## Update Employee By ID (Updating Employee ID)

Run Run Selected Auto complete Clear SQL statement:

SELECT \* FROM EMPLOYEE|

SELECT \* FROM EMPLOYEE;

EMPID	FIRST_NAME	LAST_NAME	DESIGNATION	DOB	AGE	SALARY
3195	Kishan	Mishra	Software Engineer	06/11/1999	22	25000

(1 row, 3 ms)

Edit

Output after updating employee by ID