**Pom.xml**

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.cdac</groupId>

<artifactId>Hibernate-CURD-App</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<name>Hibernate CURD App</name>

<description>Hibernate CURD Project</description>

<properties>

<mysql.connector.version>8.0.13</mysql.connector.version>

<hibernate.version>4.3.0.Final</hibernate.version>

</properties>

<build>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-war-plugin</artifactId>

<version>3.3.1</version>

</plugin>

</plugins>

</build>

<dependencies>

<!-- https://mvnrepository.com/artifact/org.hibernate/hibernate-core -->

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-core</artifactId>

<version>5.6.9.Final</version>

</dependency>

<!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>${mysql.connector.version}</version>

</dependency>

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-entitymanager</artifactId>

<version>${hibernate.version}</version>

</dependency>

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-ehcache</artifactId>

<version>${hibernate.version}</version>

</dependency>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>javax.servlet-api</artifactId>

<version>3.1.0</version>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>javax.servlet-api</artifactId>

<version>3.0.1</version>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>javax.servlet.jsp</groupId>

<artifactId>javax.servlet.jsp-api</artifactId>

<version>2.2.1</version>

<scope>provided</scope>

</dependency>

</dependencies>

</project>

**Persistence.xml**

<persistence xmlns="http://xmlns.jcp.org/xml/ns/persistence"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2.2"

xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence http://xmlns.jcp.org/xml/ns/persistence/persistence\_2\_2.xsd">

<persistence-unit name="learning-hibernate">

<!-- Specifying which ORM we will be using -->

<provider>org.hibernate.jpa.HibernatePersistenceProvider</provider>

<properties>

<!-- Database settings -->

<property name="javax.persistence.jdbc.driver"

value="com.mysql.cj.jdbc.Driver" />

<property name="javax.persistence.jdbc.url"

value="jdbc:mysql://localhost:3306/pune?useSSL=false" />

<property name="javax.persistence.jdbc.user" value="root" />

<property name="javax.persistence.jdbc.password"

value="cdac" />

<!-- Additional settings -->

<property name="hibernate.hbm2ddl.auto" value="update" />

<!-- Auto creation of tables -->

<property name="hibernate.show\_sql" value="true" />

<!-- Log generated SQL -->

</properties>

</persistence-unit>

</persistence>

**Employee.java**

package com.cdac.model;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name = "table\_employee3")

public class Employee {

@Id

@Column(name = "id")

private int id;

@Column(name = "name")

private String name;

@Column(name = "department")

private String department;

@Column(name = "age")

private int age;

@Column(name = "salary")

private double salary;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getDepartment() {

return department;

}

public void setDepartment(String department) {

this.department = department;

}

public int getAge() {

return age;

}

public void setAge(int age) {

this.age = age;

}

public double getSalary() {

return salary;

}

public void setSalary(double salary) {

this.salary = salary;

}

**Employee Dao.java**

package com.cdac.dao;

import java.util.List;

import javax.persistence.EntityManager;

import javax.persistence.EntityManagerFactory;

import javax.persistence.EntityTransaction;

import javax.persistence.Persistence;

import javax.persistence.Query;

import com.cdac.model.Employee;

import com.cdac.model.Employee2;

//Data Access Object

//classes which contain database related code

//are commonly referred to as Dao classes

public class EmployeeDao {

// insert record

public static void createRecord(Employee emp) {

EntityManagerFactory emf = Persistence.createEntityManagerFactory("learning-hibernate");

EntityManager em = emf.createEntityManager();

EntityTransaction tx = em.getTransaction();

tx.begin();

em.persist(emp);

tx.commit();

emf.close();

}

// delete record

public static void deleteRecord(int id) {

EntityManagerFactory emf = Persistence.createEntityManagerFactory("learning-hibernate");

EntityManager em = emf.createEntityManager();

EntityTransaction tx = em.getTransaction();

tx.begin();

int key = Integer.valueOf(10);

Employee emp = em.find(Employee.class, key);

em.remove(emp);

tx.commit();

emf.close();

}

// get record

public static Employee getRecord(int id) {

EntityManagerFactory emf = Persistence.createEntityManagerFactory("learning-hibernate");

EntityManager em = emf.createEntityManager();

EntityTransaction tx = em.getTransaction();

tx.begin();

int key = Integer.valueOf(10);

Employee emp = em.find(Employee.class, key);

System.out.println(emp.getName() + " " + emp.getSalary());

tx.commit();

emf.close();

return emp;

}

// update record

public static void updateRecord(int id) {

EntityManagerFactory emf = Persistence.createEntityManagerFactory("learning-hibernate");

EntityManager em = emf.createEntityManager();

EntityTransaction tx = em.getTransaction();

tx.begin();

int key = Integer.valueOf(50);

Employee2 emp = em.find(Employee2.class, key);

emp.setName("Suresh");

em.merge(emp);

tx.commit();

emf.close();

}

// get record

public static List<Employee> getAllRecord(int start, int total) {

EntityManagerFactory emf = Persistence.createEntityManagerFactory("learning-hibernate");

EntityManager em = emf.createEntityManager();

EntityTransaction tx = em.getTransaction();

tx.begin();

Query query = em.createQuery("from Employee limit" + (start - 1) + " , " + total);

List<Employee> empList = query.getResultList();

for (Employee emp : empList) {

System.out.println(emp.getId() + " " + emp.getName() + " " + emp.getDepartment() + " " + emp.getAge()

+ " " + emp.getSalary());

}

tx.commit();

emf.close();

return empList;

}

}

}

**Insert Employee.java**

package com.cdac.main;

import com.cdac.dao.EmployeeDao;

import com.cdac.model.Employee;

public class insertEmployee2 {

public static void main(String[] args) {

Employee emp = new Employee();

emp.setId(18);

emp.setName("Virat");

emp.setDepartment(".net Devloper");

emp.setAge(36);

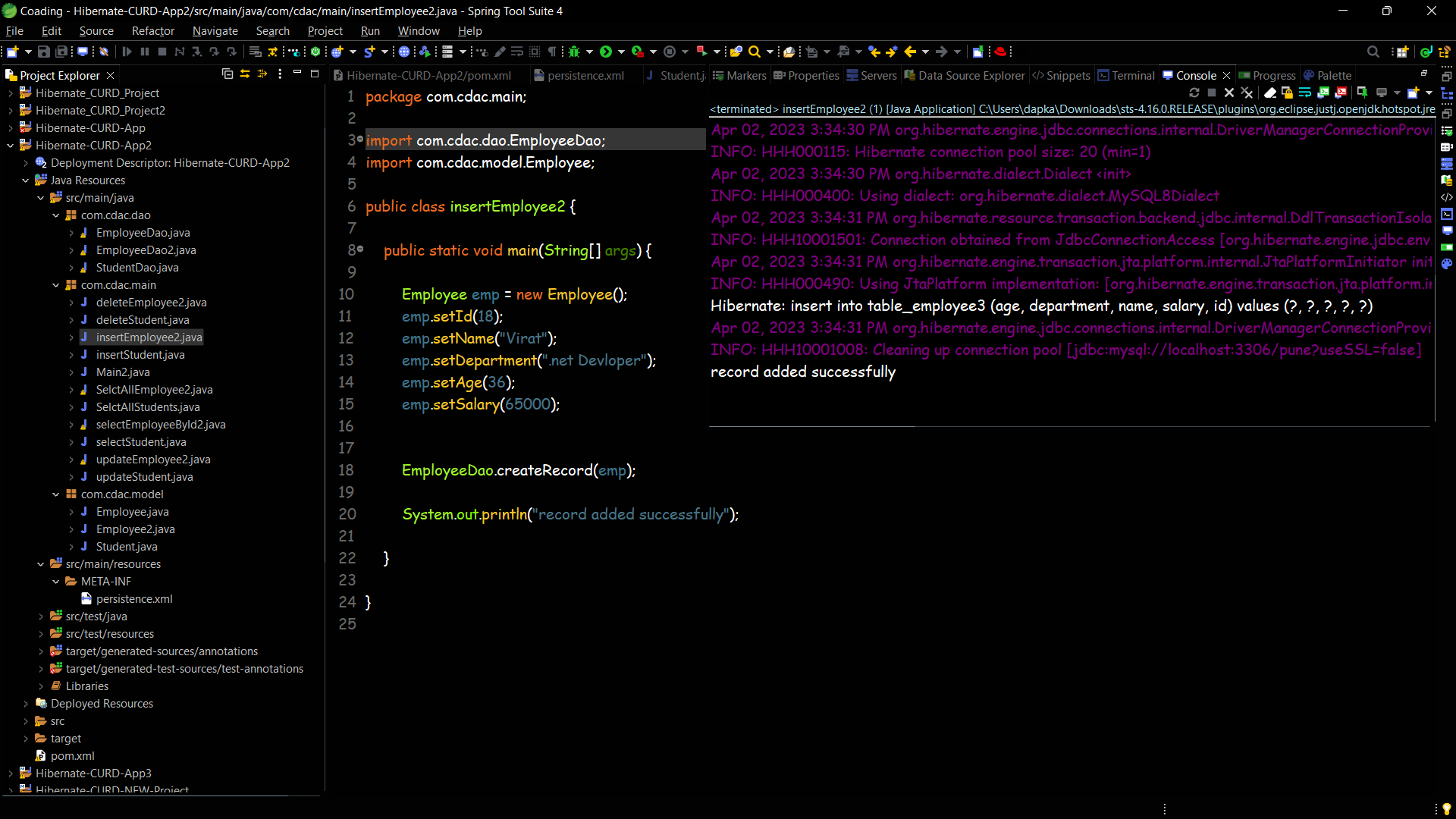
emp.setSalary(65000);

EmployeeDao.createRecord(emp);

System.out.println("record added successfully");

}

}



**Update Employee.java**

package com.cdac.main;

import com.cdac.dao.EmployeeDao;

import com.cdac.model.Employee;

public class updateEmployee {

public static void main(String[] args) {

Employee emp = EmployeeDao.getRecord(18);

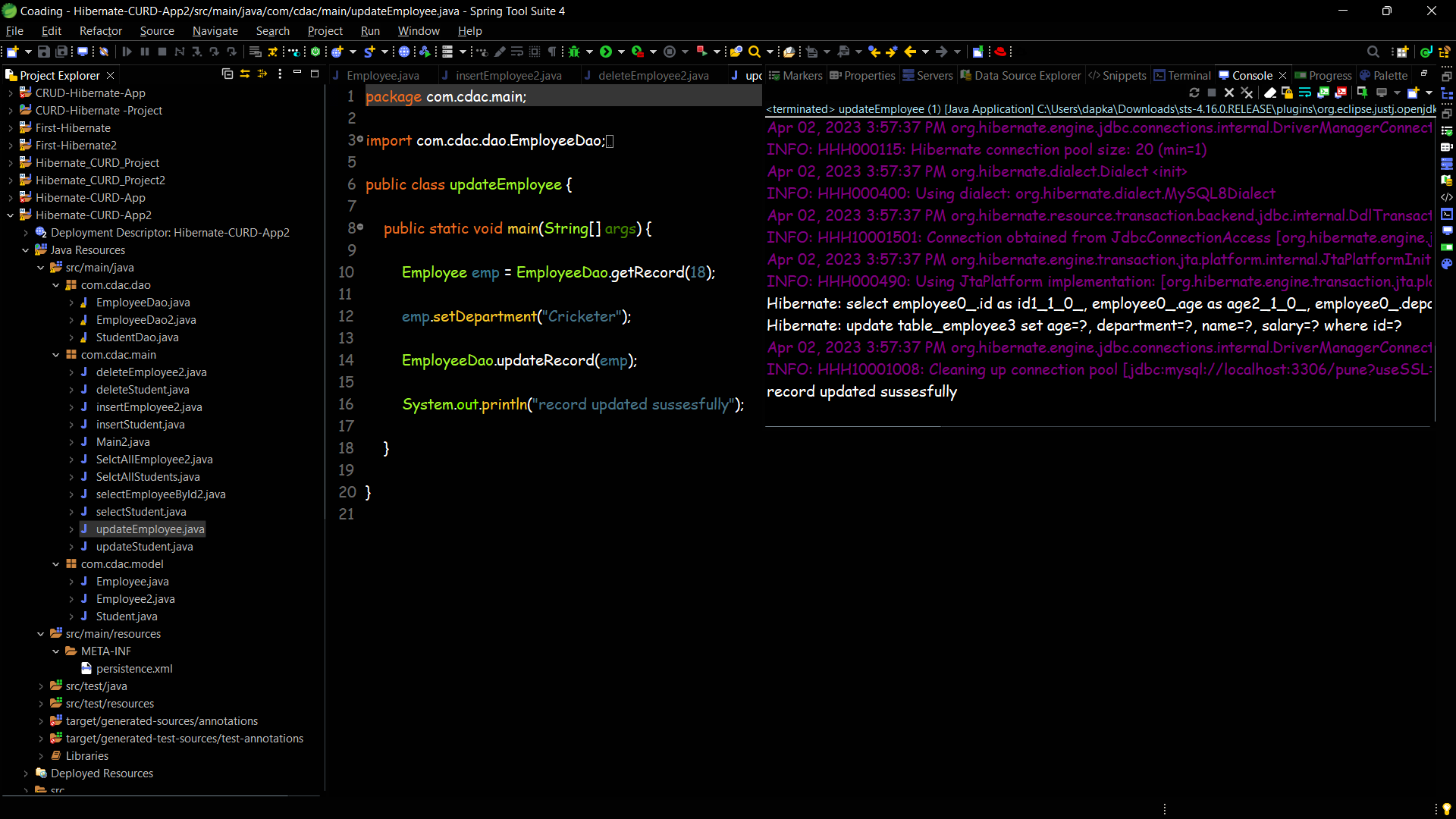
emp.setDepartment("Cricketer");

EmployeeDao.updateRecord(emp);

System.out.println("record updated sussesfully");

}

}



**GetEmployee.java**

package com.cdac.main;

import com.cdac.dao.EmployeeDao;

public class selectEmployeeById2 {

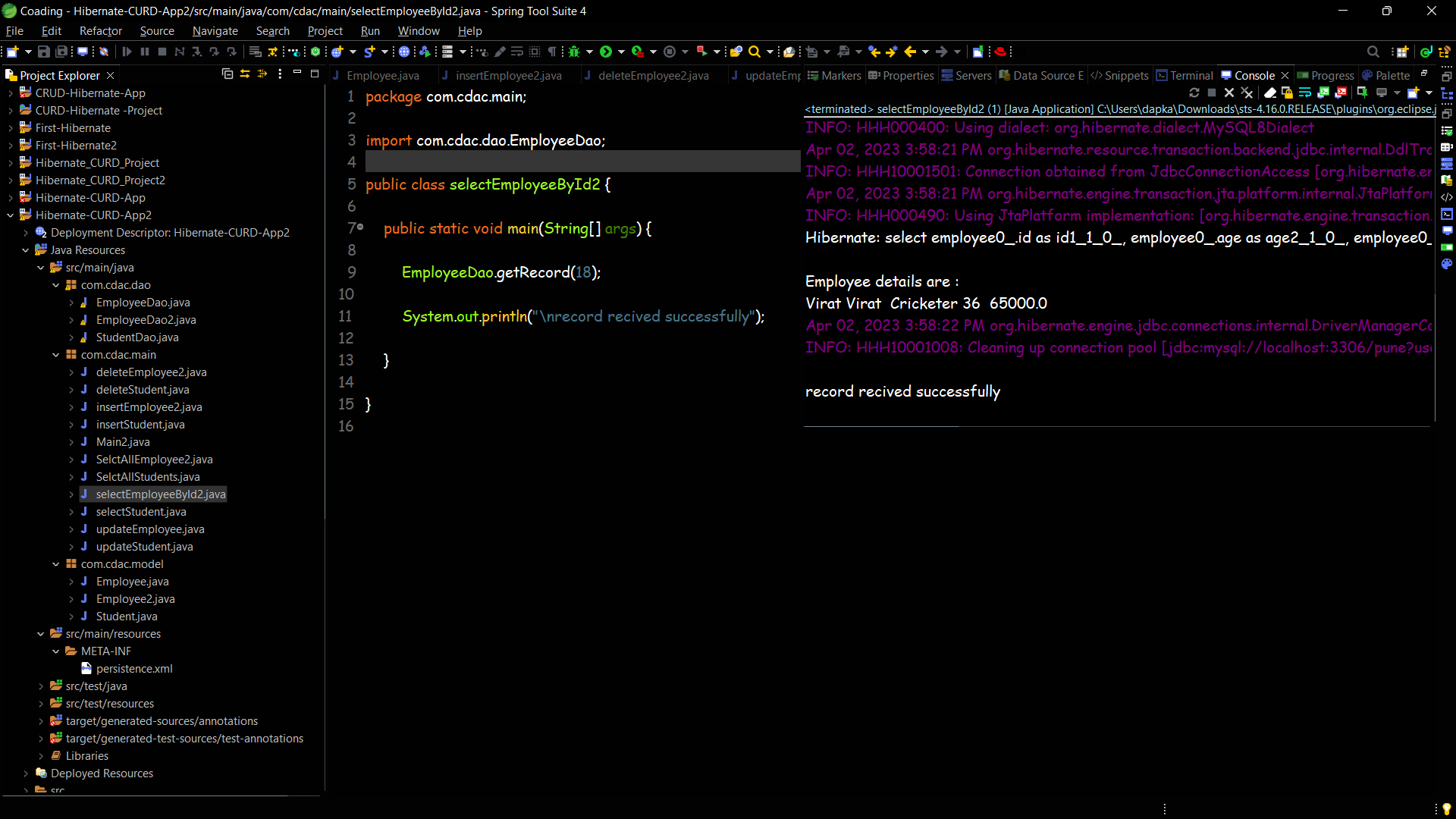
public static void main(String[] args) {

EmployeeDao.getRecord(18);

System.out.println("\nrecord recived successfully");

}

}



**GetAllEmployee.java**

package com.cdac.main;

import com.cdac.dao.EmployeeDao;

public class SelctAllEmployee2 {

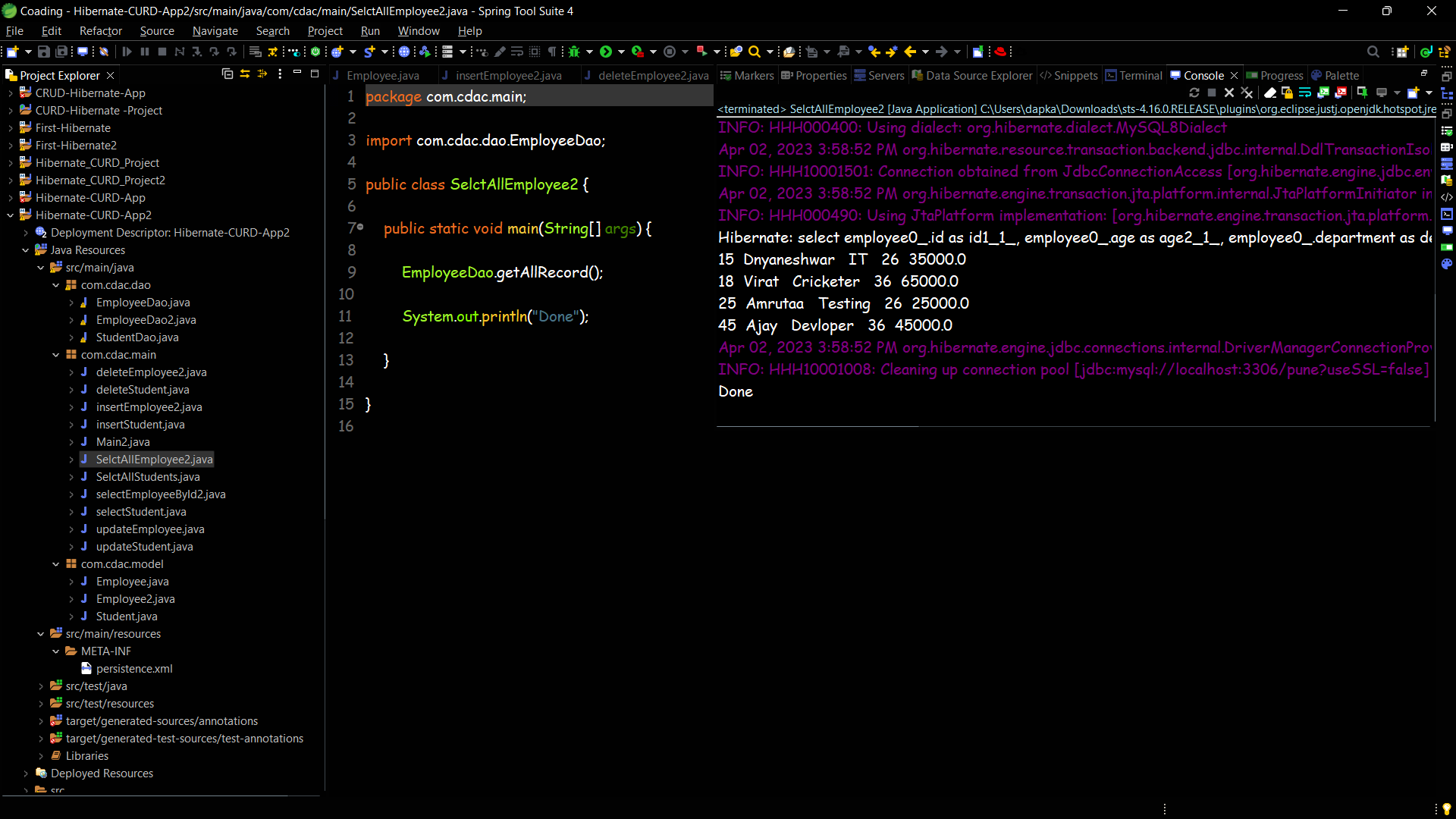
public static void main(String[] args) {

EmployeeDao.getAllRecord();

System.out.println("Done");

}

}



**DeleteEmployee.java**

package com.cdac.main;

import com.cdac.dao.EmployeeDao;

public class deleteEmployee2 {

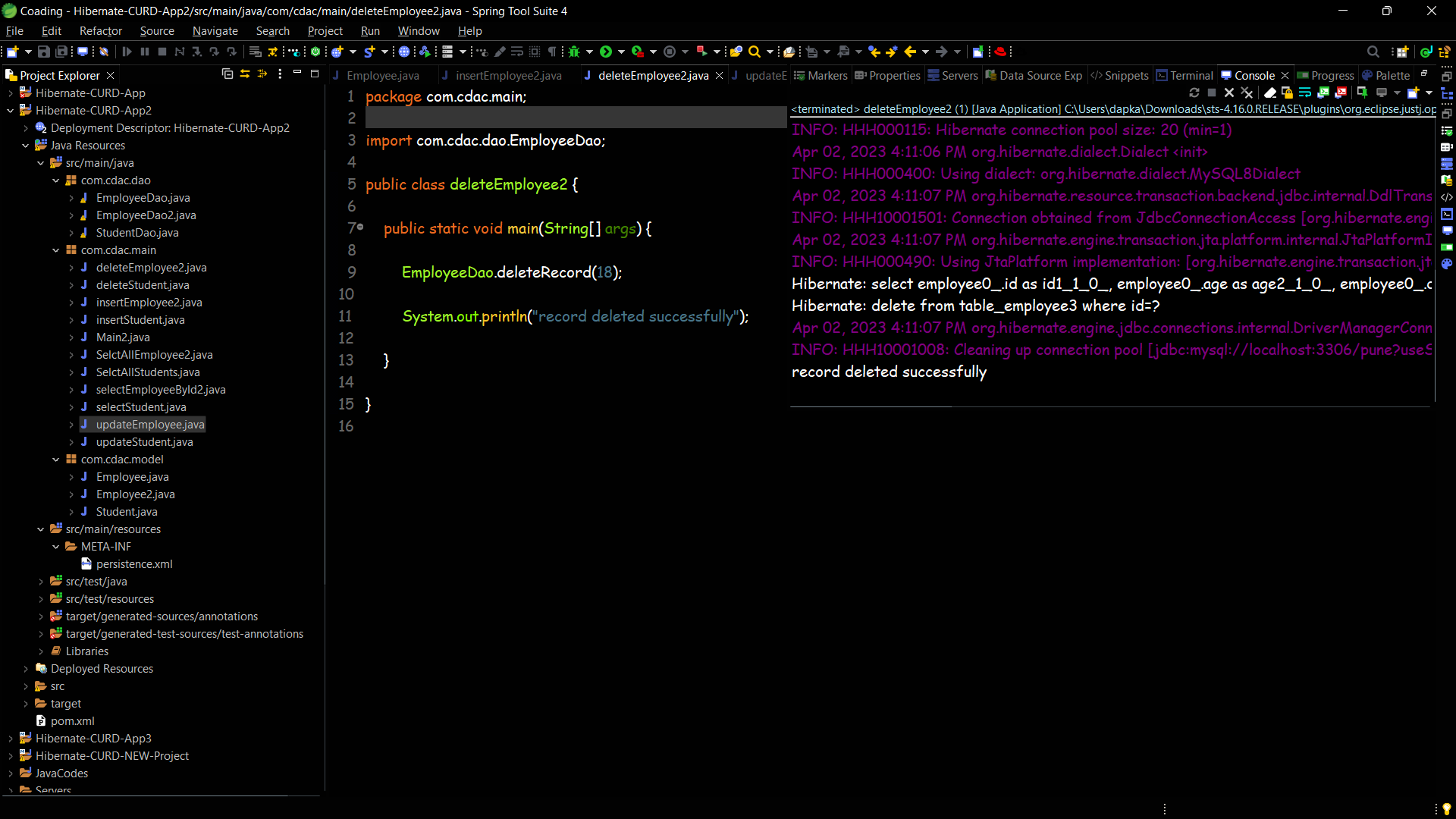
public static void main(String[] args) {

EmployeeDao.deleteRecord(10);

System.out.println("record deleted successfully");

}

}

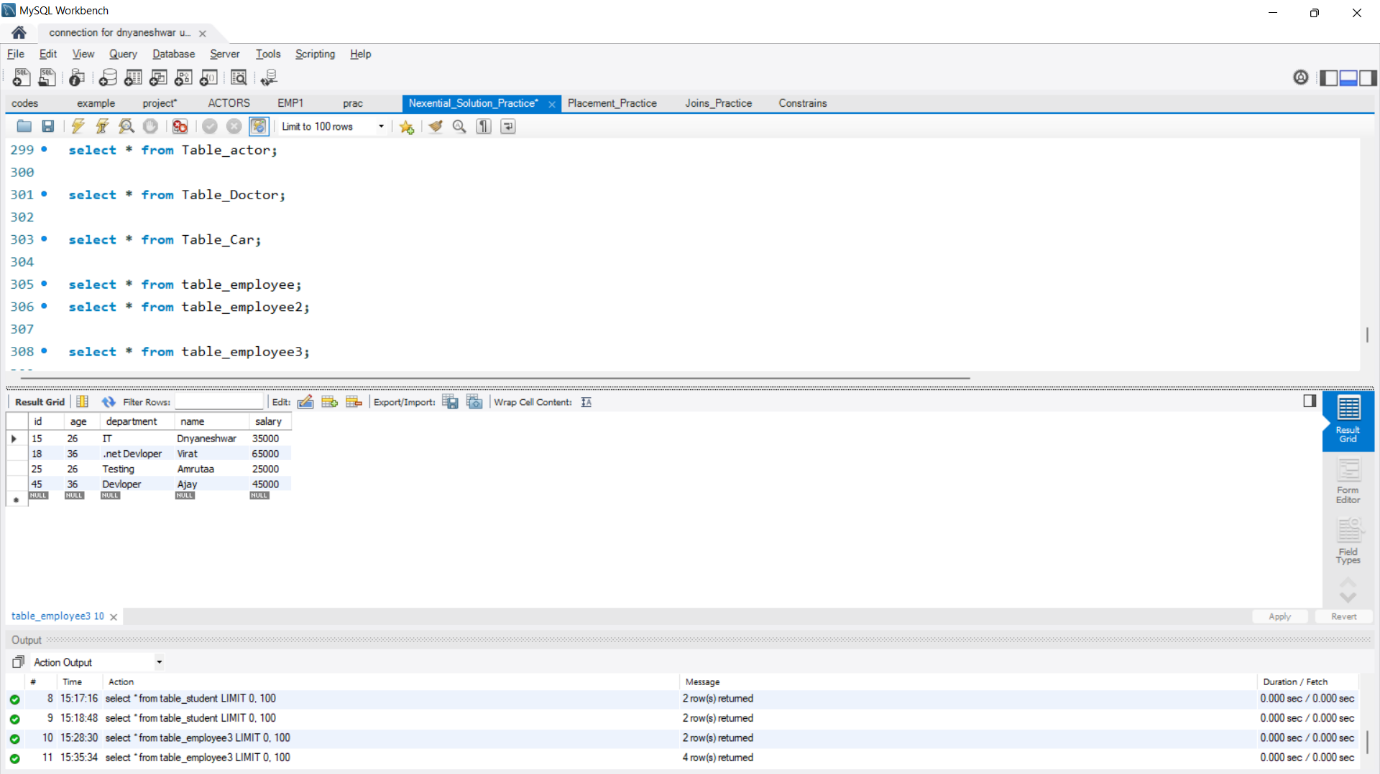


**Project Structure**

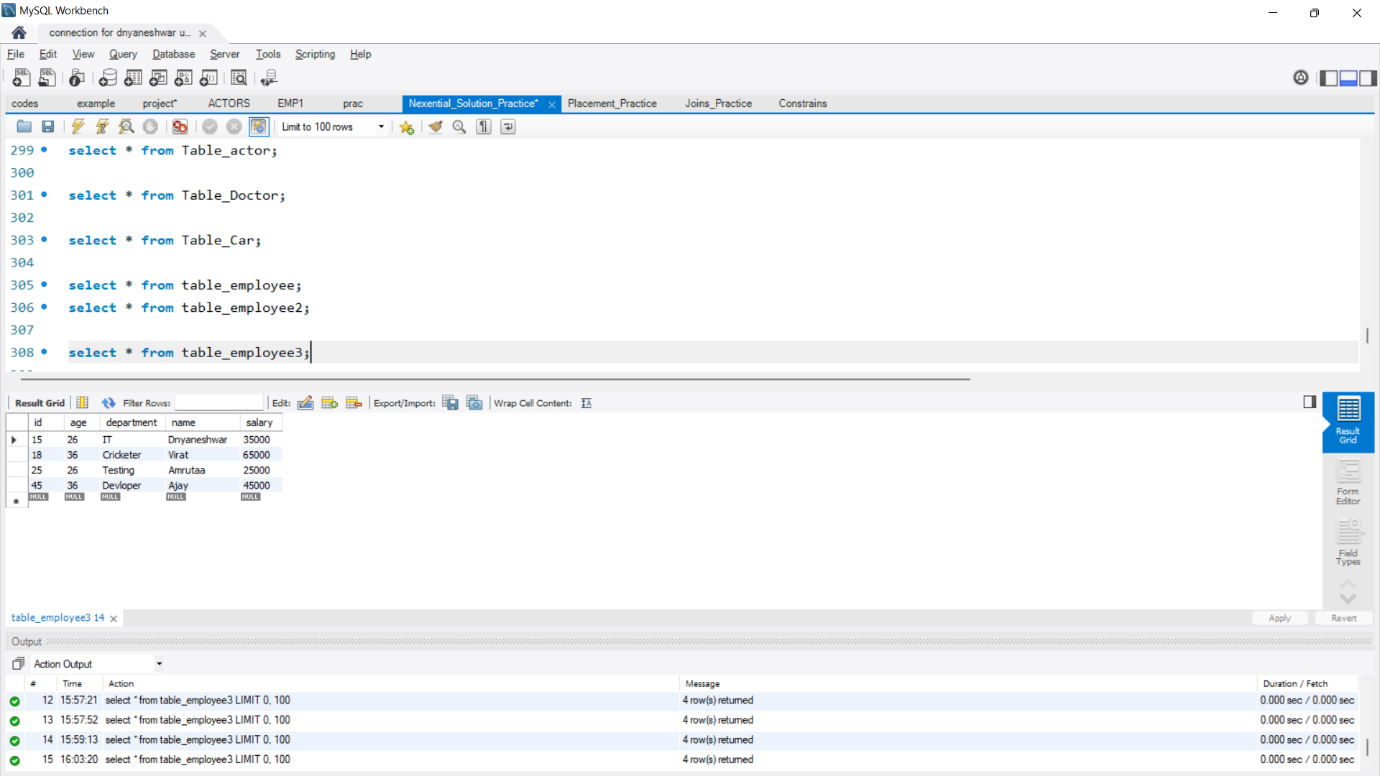


**Database Screenshots**

Records added :



Records updated :



Records deleted :

