Name : Niraj Patill.

Q1)

Code:

Develop hibernate application to save and retrieve records from following database table (make use of

annotations , HQL queries and Criteria wherever possible) :

Question table:

Column Column_Type

id int (Primary Key : Auto Increment)

desc varchar(200)

postedDate date

answer varchar(400)

status Boolean

Code:

```
package pojos;

import java.time.LocalDate;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;
@Entity
@Table(name="myTable")
public class User {
        /*
         * id int (Primary Key : Auto Increment) desc varchar(200) postedDate date
         * answer varchar(400) status boolean
         */
@Id
        @GeneratedValue(strategy = GenerationType.IDENTITY)//auto_increment id..
@Column(name = "id")
private Integer Id;
@Column(length=200)
        private String desc;
@Column(name=" postedDate")
        private LocalDate regDate;
@Column(length=400)
        private String   answer;
@Column(name = "status")
        private boolean status;



        public User() {
                System.out.println("in thge default constructor..");
        }



        public User(String desc, LocalDate regDate, String answer, boolean status) {



                this.desc = desc;
                this.regDate = regDate;
                this.answer = answer;
                this.status = status;
```

```
}



public User( String desc, LocalDate regDate, String answer) {
	super();

	this.desc = desc;
	this.regDate = regDate;
	this.answer = answer;
}







public Integer getId() {
	return Id;
}
public void setId(Integer id) {
	Id = id;
}
public String getDesc() {
	return desc;
}
public void setDesc(String desc) {
	this.desc = desc;
}
public LocalDate getRegDate() {
	return regDate;
	}
	public void setRegDate(LocalDate regDate) {
		this.regDate = regDate;
	}
	public String getAnswer() {
		return answer;
	}
	public void setAnswer(String answer) {
		this.answer = answer;
	}
	public boolean isStatus() {
		return status;
	}
	public void setStatus(boolean status) {
		this.status = status;
	}
	@Override
	public String toString() {
		return "User [Id=" + Id + ", desc=" + desc + ", regDate=" + regDate + ", answer=" +
answer + ", status="
				+ status + "]";
	}



}
```

IUserDao:

```
package dao;

import pojos.User;

public interface IUserDao {
String addData(User user);
User getUserDetails(int userId);
String deleteUserDetails(int id);
}
```

UserDaoImpl:

```
package dao;

import pojos.User;
import static utils.HibernateUtils.getFactory;

import java.io.Serializable;

import org.hibernate.*;

public class UserDaoImpl implements IUserDao {


    @Override
    public User getUserDetails(int userId) {
        User user = null;
        Session session = getFactory().getCurrentSession();
        Transaction tx = session.beginTransaction();
        try {
            user = session.get(User.class, userId);
            tx.commit();
        } catch (RuntimeException e) {
            if (tx != null)
                tx.rollback();
            throw e;
        }

        return user;
}

@Override
public String addData(User user) {
        Session session=getFactory().getCurrentSession();
        Transaction tx = session.beginTransaction();
        String msg;
        try {
        Serializable id=session.save(user);
                tx.commit();
                msg="Success:-"+id;
        }catch (RuntimeException e) {
                if (tx != null)
                        tx.rollback();
                throw e;
        }
        return msg;
}

@Override
public String deleteUserDetails(int id) {
        Session session = getFactory().getCurrentSession();
        Transaction tx = session.beginTransaction();
        try {
        User u=session.get(User.class, id);
        session.delete(u);
                tx.commit();
        } catch (RuntimeException e) {
                if (tx != null)
                    tx.rollback();
                throw e;
            }

            return "Deleted";
        }




}
```

Testers:

```
package tester;

import static utils.HibernateUtils.getFactory;

import java.time.LocalDate;
import java.util.Scanner;

import org.hibernate.SessionFactory;

import dao.UserDaoImpl;
import pojos.User;


public class AddData {
    /*
     * id int (Primary Key : Auto Increment) desc varchar(200) postedDate date
```

```
         * answer varchar(400) status boolean
         */

        public static void main(String[] args) {
            try(SessionFactory sf=getFactory();Scanner sc=new Scanner(System.in);){
                System.out.println("Enter Details of  id, DESC, Date, Answer
,Status");


                System.out.println("Enter Desc: ");
                String desc=sc.next();
                System.out.println("Enter Date(year-month-date): ");
                String date=sc.next();
                System.out.println("Enter Answer: ");
                String ans=sc.next();
                System.out.println("Enter Status: ");
                boolean status=sc.nextBoolean();
                //User user=new User( desc, LocalDate.parse(date),ans, status);
                User user=new User( desc, LocalDate.parse(date),ans);
                  UserDaoImpl dao=new UserDaoImpl();
                  System.out.println(dao.addData(user));

            }catch (Exception e) {
                e.printStackTrace();
            }

        }

    }
```

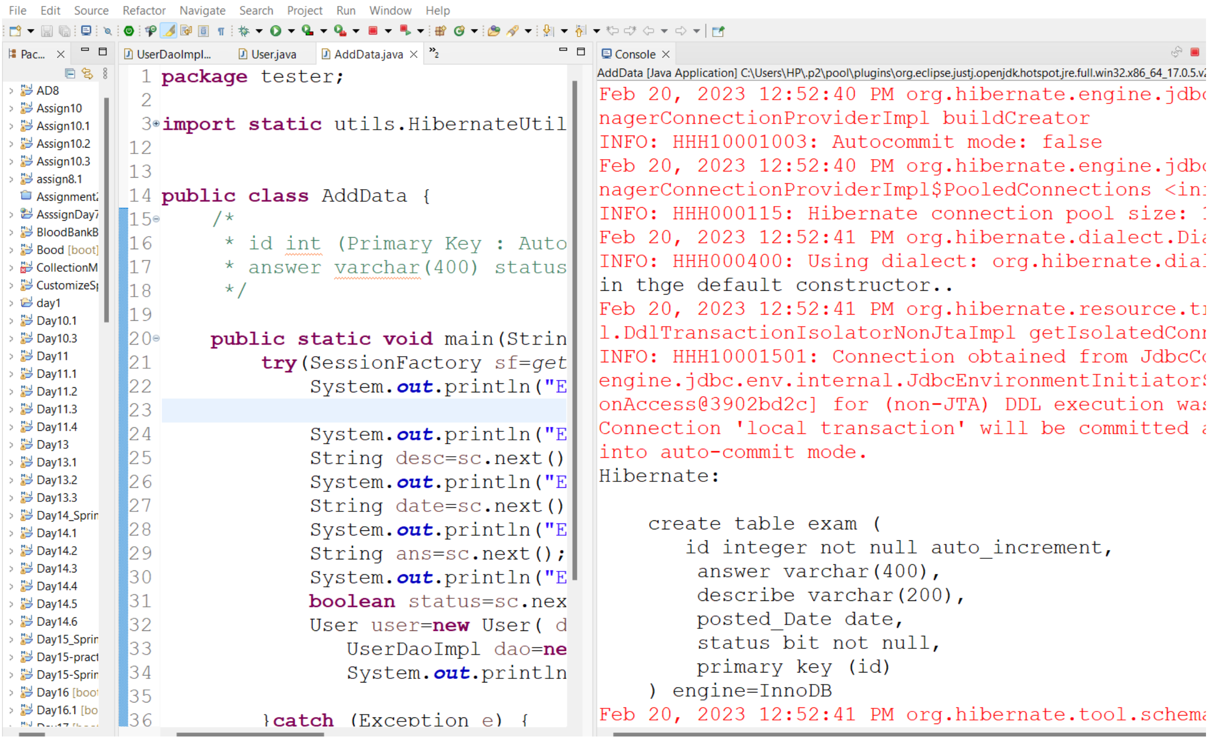
Deletion//

```
package tester;

import java.util.Scanner;

import dao.UserDaoImpl;

public class DeleteData {

    public static void main(String[] args) {
        try(Scanner sc=new
Scanner(System.in);){
            System.out.println("Enter UserId");

                UserDaoImpl dao=new
UserDaoImpl();

System.out.println(dao.deleteUserDetails(sc.nex
tInt()));

        }catch (Exception e) {
            e.printStackTrace();
        }

    }
}
```

GetData:

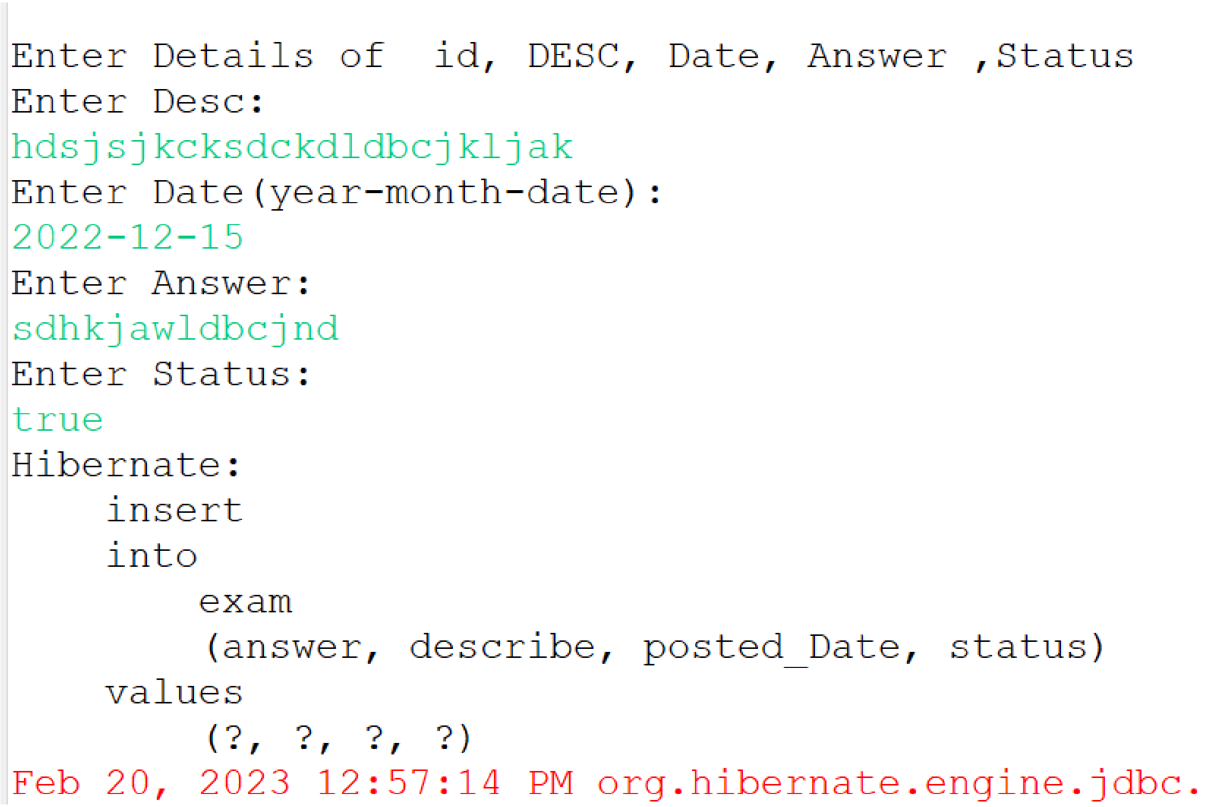
```
package tester;


import java.util.Scanner;


import dao.UserDaoImpl;


public class GetUserDetails {
    public static void main(String[] args) {


        try(Scanner sc=new Scanner(System.in);){
            System.out.println("Enter UserId");


              UserDaoImpl dao=new UserDaoImpl();
              System.out.println(dao.getUserDetails(sc.nextInt()));


        }catch (Exception e) {
            e.printStackTrace();
        }


    }


}
```

Output:

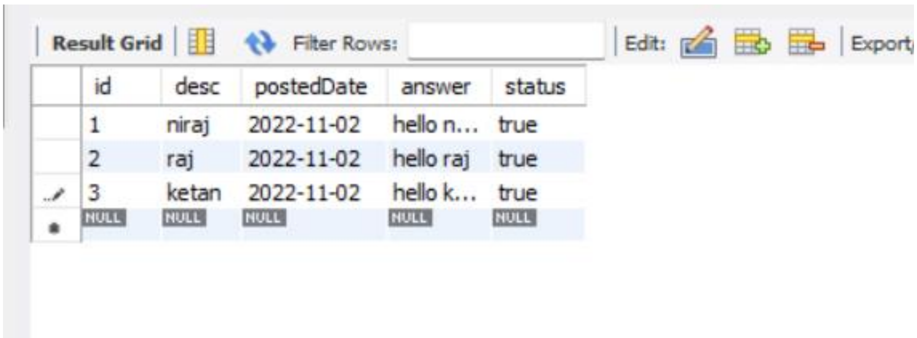


```java
package tester;

import static utils.HibernateUtil


public class AddData {
    /*
     * id int (Primary Key : Auto
     * answer varchar(400) status
     */

    public static void main(Strin
        try(SessionFactory sf=get
            System.out.println("E

            System.out.println("E
            String desc=sc.next()
            System.out.println("E
            String date=sc.next()
            System.out.println("E
            String ans=sc.next();
            System.out.println("E
            boolean status=sc.nex
            User user=new User( d
                UserDaoImpl dao=ne
                System.out.println

        }catch (Exception e) {
```

```
AddData [Java Application] C:\Users\HP\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.5.v
Feb 20, 2023 12:52:40 PM org.hibernate.engine.jdb
nagerConnectionProviderImpl buildCreator
INFO: HHH10001003: Autocommit mode: false
Feb 20, 2023 12:52:40 PM org.hibernate.engine.jdb
nagerConnectionProviderImpl$PooledConnections <in
INFO: HHH000115: Hibernate connection pool size: 
Feb 20, 2023 12:52:41 PM org.hibernate.dialect.Di
INFO: HHH000400: Using dialect: org.hibernate.dia
in thge default constructor..
Feb 20, 2023 12:52:41 PM org.hibernate.resource.t
l.DdlTransactionIsolatorNonJtaImpl getIsolatedCon
INFO: HHH10001501: Connection obtained from JdbcC
engine.jdbc.env.internal.JdbcEnvironmentInitiator
onAccess@3902bd2c] for (non-JTA) DDL execution wa
Connection 'local transaction' will be committed 
into auto-commit mode.
Hibernate:

    create table exam (
       id integer not null auto_increment,
        answer varchar(400),
        describe varchar(200),
        posted_Date date,
        status bit not null,
        primary key (id)
    ) engine=InnoDB
Feb 20, 2023 12:52:41 PM org.hibernate.tool.schem
```

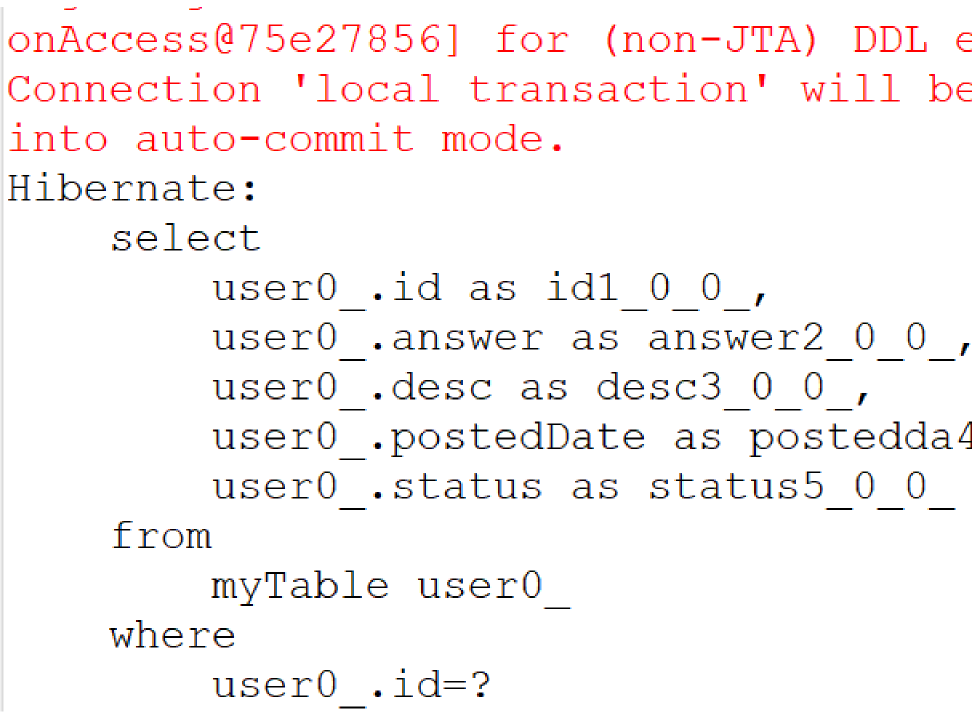


```
Enter Details of  id, DESC, Date, Answer ,Status
Enter Desc:
hdsjsjkcksdckdldbcjkljak
Enter Date(year-month-date):
2022-12-15
Enter Answer:
sdhkjawldbcjnd
Enter Status:
true
Hibernate:
    insert
    into
        exam
        (answer, describe, posted_Date, status)
    values
        (?, ?, ?, ?)
Feb 20, 2023 12:57:14 PM org.hibernate.engine.jdbc.
```



id	desc	postedDate	answer	status
1	niraj	2022-11-02	hello n...	true
2	raj	2022-11-02	hello raj	true
3	ketan	2022-11-02	hello k...	true
NULL	NULL	NULL	NULL	NULL

2)delete operation:



```
onAccess@75e27856] for (non-JTA) DDL e
Connection 'local transaction' will be
into auto-commit mode.
Hibernate:
    select
        user0_.id as id1_0_0_,
        user0_.answer as answer2_0_0_,
        user0_.desc as desc3_0_0_,
        user0_.postedDate as postedda4
        user0_.status as status5_0_0_
    from
        myTable user0_
    where
        user0_.id=?
```