Name: SAKSHAM

UID: 22BCS10991

Section:616-A

Experiment: 9

Easy level

```java
public class Course {

    private String courseName;

    private int duration;


    public Course(String courseName, int duration) {

        this.courseName = courseName;

        this.duration = duration;

    }


    public String getCourseName() {

        return courseName;

    }


    public int getDuration() {

        return duration;

    }


    public String toString() {

        return "Course: " + courseName + ", Duration: " + duration + " months";

    }
}

public class Student {

    private String name;
```

```java
    private Course course;

    public Student(String name, Course course) {
        this.name = name;
        this.course = course;
    }

    public void showDetails() {
        System.out.println("Student Name: " + name);
        System.out.println(course.toString());
    }
}
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;

@Configuration
public class AppConfig {

    @Bean
    public Course course() {
        return new Course("Java Spring Boot", 3);
    }

    @Bean
    public Student student() {
        return new Student("John Doe", course());
    }
}
import org.springframework.context.ApplicationContext;
import org.springframework.context.annotation.AnnotationConfigApplicationContext;
```

```java
public class App {

    public static void main(String[] args) {

        ApplicationContext context = new
AnnotationConfigApplicationContext(AppConfig.class);

        Student student = context.getBean(Student.class);

        student.showDetails();

    }
}
```

Medium level

```java
import jakarta.persistence.*;

@Entity
@Table(name = "student")
public class Student {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;
    private String name;
    private int age;

    // Getters and setters
    // toString() method
}
```
```xml
<hibernate-configuration>
 <session-factory>
    <property
name="hibernate.connection.driver_class">com.mysql.cj.jdbc.Driver</property>
    <property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/yourdb</property>
    <property name="hibernate.connection.username">root</property>
```

```xml
            <property name="hibernate.connection.password">password</property>
            <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
            <property name="show_sql">true</property>
            <mapping class="Student"/>
    </session-factory>
</hibernate-configuration>
```

```java
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;


public class MainCrudApp {
    public static void main(String[] args) {
        SessionFactory factory = new Configuration().configure("hibernate.cfg.xml")
                                .addAnnotatedClass(Student.class)
                                .buildSessionFactory();
        Session session = factory.getCurrentSession();


        try {
            Student s1 = new Student();
            s1.setName("Bob");
            s1.setAge(22);


            session.beginTransaction();
            session.save(s1);  // Create
            session.getTransaction().commit();


            // Read
            session = factory.getCurrentSession();
            session.beginTransaction();
            Student readStudent = session.get(Student.class, s1.getId());
            System.out.println("Read: " + readStudent);
```

```java
        // Update
        readStudent.setAge(23);
        session.getTransaction().commit();


        // Delete
        session = factory.getCurrentSession();
        session.beginTransaction();
        session.delete(readStudent);
        session.getTransaction().commit();
    } finally {
        factory.close();
    }
  }
}
```

Hard Level:

```java
import jakarta.persistence.*;


@Entity
public class Account {
    @Id
    private int accountId;
    private String owner;
    private double balance;


    // Getters and setters
}
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.springframework.transaction.annotation.Transactional;
```

```java
public class TransactionService {

    private SessionFactory sessionFactory;

    public TransactionService(SessionFactory factory) {

        this.sessionFactory = factory;

    }

    @Transactional
    public void transferFunds(int fromId, int toId, double amount) {

        Session session = sessionFactory.getCurrentSession();

        session.beginTransaction();


        Account from = session.get(Account.class, fromId);

        Account to = session.get(Account.class, toId);


        if (from.getBalance() < amount) {

            throw new RuntimeException("Insufficient balance");

        }


        from.setBalance(from.getBalance() - amount);

        to.setBalance(to.getBalance() + amount);


        session.getTransaction().commit();

    }
}
import org.hibernate.SessionFactory;

import org.hibernate.cfg.Configuration;


public class BankApp {

    public static void main(String[] args) {
```

```java
        SessionFactory factory = new Configuration().configure()
                            .addAnnotatedClass(Account.class)
                            .buildSessionFactory();

        TransactionService service = new TransactionService(factory);

        try {
            service.transferFunds(1, 2, 1000);
            System.out.println("Transfer successful.");
        } catch (Exception e) {
            System.out.println("Transfer failed: " + e.getMessage());
        } finally {
            factory.close();
        }
    }
}
```