Test-1

Q1.Write a Java program that demonstrates method overriding by creating a superclass called Animal and two subclasses called Dog and Cat. The Animal class should have a method called makeSound(), which simply prints "The animal makes a sound." The Dog and Cot classes should override this method to print "TheCat/The dog meows/barks" respectively. The program should allow the user to create and display objects of each class.

Answer:

```
class Animal
 {
   public void makeSound()
 {
       System.out.println("The animal makes a sound.");
   }
}

class Dog extends Animal
 {
   public void makeSound()
{
       System.out.println("The dog barks.");
   }
}

class Cat extends Animal
 {
   public void makeSound()
 {
       System.out.println("The cat meows.");
   }
}

public class Main
 {
   public static void main(String[] args)
{
       Animal animal = new Animal();
       Dog dog = new Dog();
       Cat cat = new Cat();

        animal.makeSound();
       dog.makeSound();
       cat.makeSound();
   }
}
```

Q2.Create abstract class vaccine.Create two variables age(int) nationality (String) create 2 concrete methods firstDose() and secondDose(). Scenario tuser can take the first dose if the user is indian and age is 18 After vaccination the user has to pay 250rs (which will be displayed on the console). Scenario 2: Users are eligible to take the second dose only after completing the first dose. Scenario 3 create abstract method booster Dese() in abstract class Vaccine Create one implementation class vaccinationSuccessful, where implement boosterDose() method. Create main class vaccination and invoice all methods accordingly. Hint: Create constructor to initialize variables age and nationality, Use flow control (ifelse) to check condition]

Answer:

```
    abstract class Vaccine {
    protected int age;
    protected String nationality;

    public Vaccine(int age, String nationality) {
        this.age = age;
        this.nationality = nationality;
    }

    abstract void firstDose();

    abstract void secondDose();

    abstract void boosterDose();
}

class VaccinationSuccessful extends Vaccine {

    public VaccinationSuccessful(int age, String nationality) {
        super(age, nationality);
    }
    void firstDose() {
        if (nationality.equalsIgnoreCase("Indian") && age >= 18) {
            System.out.println("First dose administered successfully.");
            System.out.println("Please pay 250 INR.");
        }
else {
            System.out.println("User is not eligible for the first dose.");
        }
    }
    void secondDose() {
        System.out.println("Second dose administered successfully.");
    }
    void boosterDose() {
        System.out.println("Booster dose administered successfully.");
```

```
    }
}

public class Vaccination {
    public static void main(String[] args)
 {
        Vaccine user1 = new VaccinationSuccessful(20, "Indian");
        user1.firstDose();

  Vaccine user2 = new VaccinationSuccessful(25, "Indian");
        user2.firstDose();
        user2.secondDose();

Vaccine user3 = new VaccinationSuccessful(30, "Indian");
        user3.boosterDose();
    }
}
```

Q3. Write code to determin rome.

Input String: Madam

Answer:

```
public class PalindromeChecker(

public static void main(String[] args)

String inputString "Madam":

boolean isPalindrome checkPalindrome(inputString):

System.out.printin("Input String:"imputString):

System.out.println(is/Palindrome? "Output:"inputString"is a Palindrome" "Output"
InputStringis not a Palindrome"):

public static boolean checkPalindrome(String input) [

String cleanlinput input.replaceAll("A", "").toLowerCase();

int length cleaninput.length();

for (int i=0<length/2; i++) {

if icleaninput.charAt(i) = cleaninput.charAt(length-1-(

return false;
```

```
1

return true;
}
}
```