

Name: Pratham Srivastava

Roll No: 2401330130192

Branch: IT

Section: C

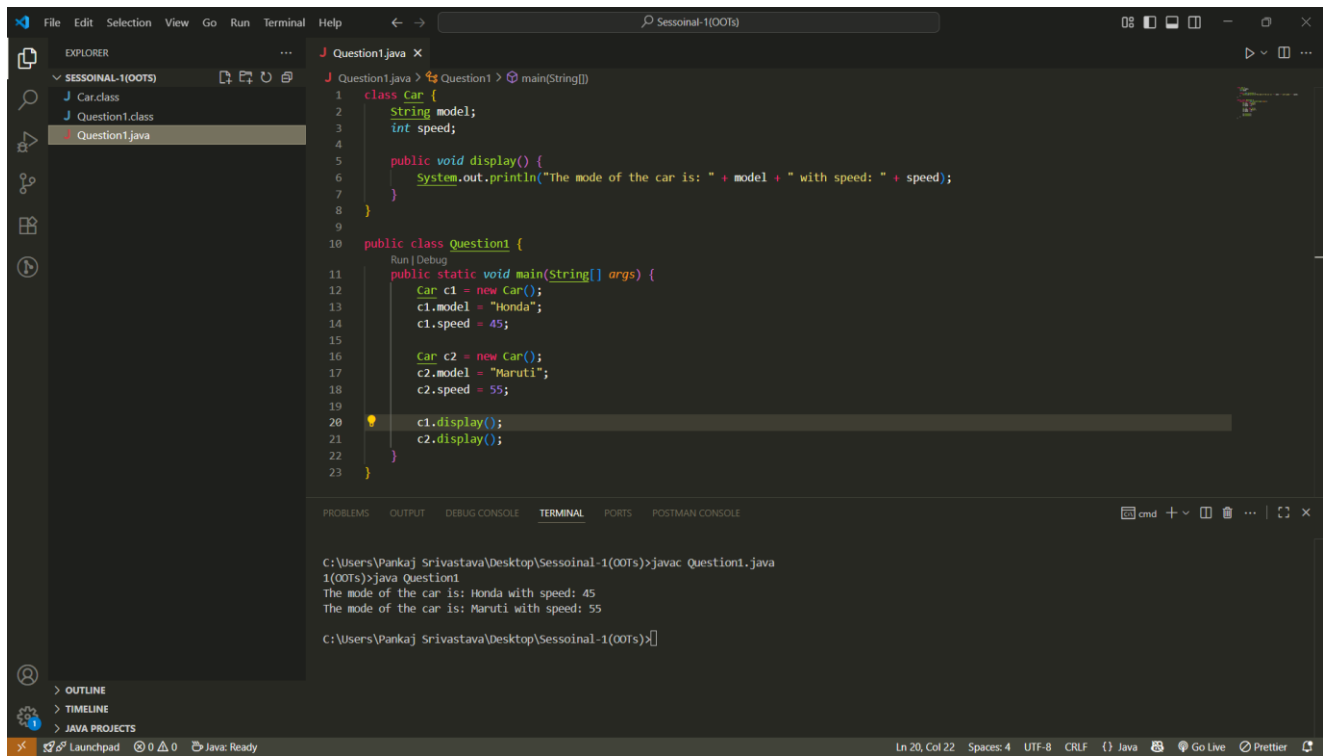
Workshop Lab Number: D-203

Q1.

```
class Car {  
    String model;  
    int speed;  
  
    public void display() {  
        System.out.println("The mode of the car is: " + model + " with speed: " +  
speed);  
    }  
}
```

```
public class Question1 {  
    public static void main(String[] args) {  
        Car c1 = new Car();  
        c1.model = "Honda";  
        c1.speed = 45;  
  
        Car c2 = new Car();  
        c2.model = "Maruti";  
        c2.speed = 55;  
  
        c1.display();  
        c2.display();  
    }  
}
```

OUTPUT FOR QUESTION-1



The screenshot shows an IDE with a Java project named 'Sessoinal-1(OOTs)'. The Explorer panel on the left lists 'Car.class', 'Question1.class', and 'Question1.java'. The main editor displays the source code for 'Question1.java'. The code defines a 'Car' class with 'model' and 'speed' attributes and a 'display()' method. The 'Question1' class has a 'main' method that creates two 'Car' objects, 'c1' (Honda) and 'c2' (Maruti), and calls their 'display()' methods. The Terminal panel at the bottom shows the command 'javac Question1.java' and the output of running the program, which prints the model and speed for both cars.

```
1 class Car {
2     String model;
3     int speed;
4
5     public void display() {
6         System.out.println("The mode of the car is: " + model + " with speed: " + speed);
7     }
8 }
9
10 public class Question1 {
11     public static void main(String[] args) {
12         Car c1 = new Car();
13         c1.model = "Honda";
14         c1.speed = 45;
15
16         Car c2 = new Car();
17         c2.model = "Maruti";
18         c2.speed = 55;
19
20         c1.display();
21         c2.display();
22     }
23 }
```

C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>javac Question1.java
1(OOTs)>java Question1
The mode of the car is: Honda with speed: 45
The mode of the car is: Maruti with speed: 55
C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>]

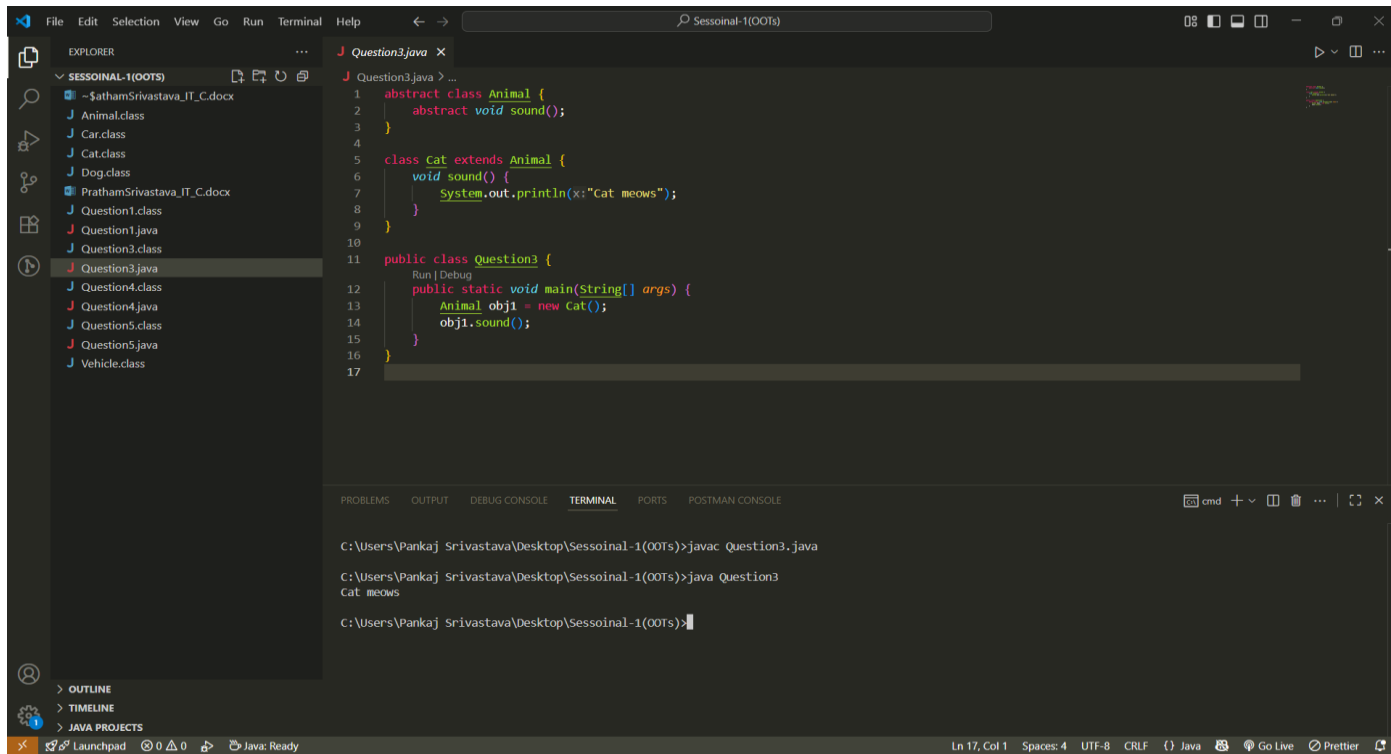
Q3.

```
abstract class Animal {  
    abstract void sound();  
}
```

```
class Cat extends Animal {  
    void sound() {  
        System.out.println("Cat meows");  
    }  
}
```

```
public class Question3 {  
    public static void main(String[] args) {  
        Animal obj1 = new Cat();  
        obj1.sound();  
    }  
}
```

OUTPUT FOR QUESTION-3



The screenshot displays an IDE with the following components:

- EXPLORER:** Shows a project named "SESSOINAL-1(OOTs)" containing files like `Animal.class`, `Car.class`, `Cat.class`, `Dog.class`, and `Question3.class`.
- Editor:** Displays the code for `Question3.java`. It defines an abstract class `Animal` with an abstract method `sound()`, a concrete class `Cat` that extends `Animal` and implements `sound()` to print "Cat meows", and a public class `Question3` with a `main` method that creates a `Cat` object and calls its `sound` method.
- TERMINAL:** Shows the execution of the program. The commands `javac Question3.java` and `java Question3` are entered, resulting in the output `Cat meows`.

```
1 abstract class Animal {
2     abstract void sound();
3 }
4
5 class Cat extends Animal {
6     void sound() {
7         System.out.println("Cat meows");
8     }
9 }
10
11 public class Question3 {
12     public static void main(String[] args) {
13         Animal obj1 = new Cat();
14         obj1.sound();
15     }
16 }
17
```

C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>javac Question3.java

C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>java Question3

Cat meows

C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>

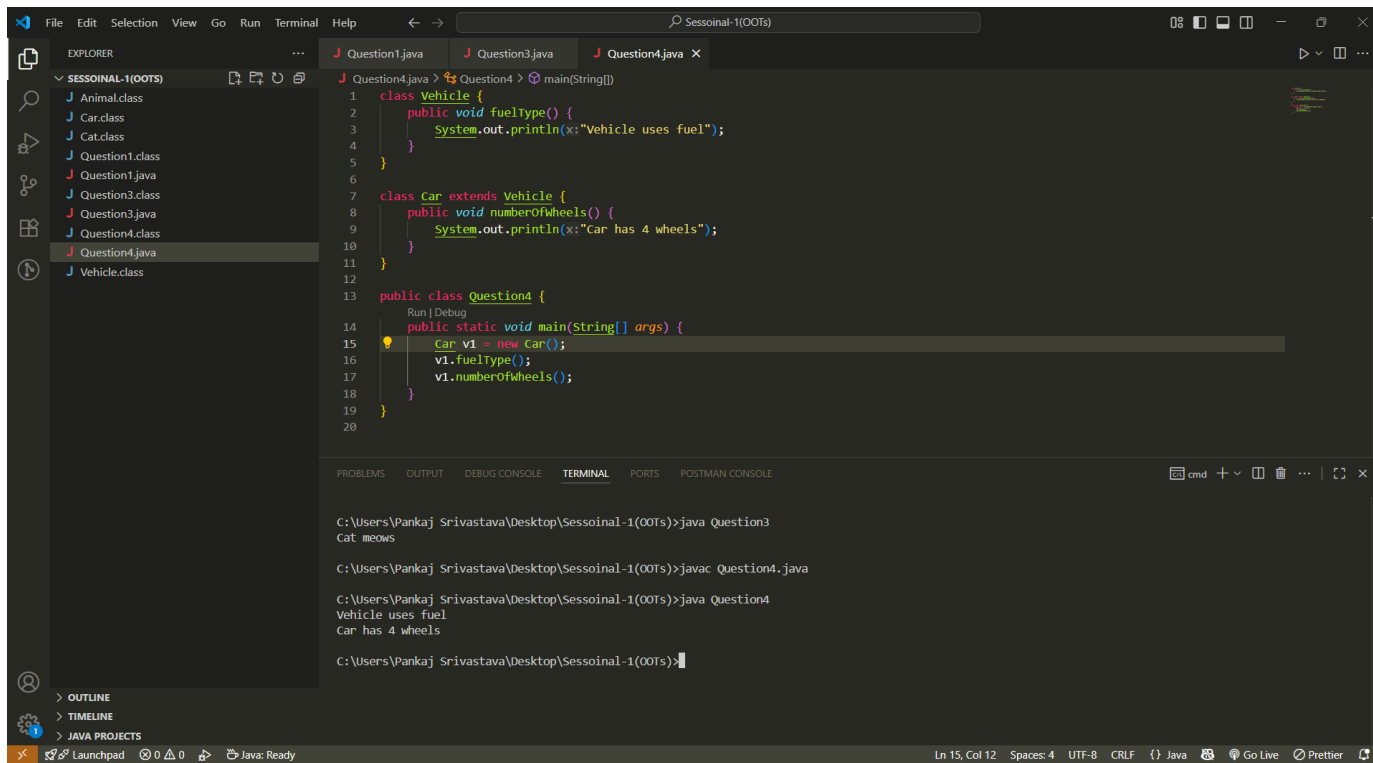
Q4.

```
class Vehicle {  
    public void fuelType() {  
        System.out.println("Vehicle uses fuel");  
    }  
}
```

```
class Car extends Vehicle {  
    public void numberOfWheels() {  
        System.out.println("Car has 4 wheels");  
    }  
}
```

```
public class Question4 {  
    public static void main(String[] args) {  
        Car v1 = new Car();  
        v1.fuelType();  
        v1.numberOfWheels();  
    }  
}
```

OUTPUT FOR QUESTION-4



The screenshot shows an IDE with the following components:

- EXPLORER:** A file tree for 'Sessoinal-1(OOTs)' containing files: Animal.class, Car.class, Cat.class, Question1.class, Question1.java, Question3.class, Question3.java, Question4.class, Question4.java, and Vehicle.class. 'Question4.java' is selected.
- EDITOR:** Displays the code for 'Question4.java'. The code defines a 'Vehicle' class with a 'fuelType()' method, a 'Car' class that extends 'Vehicle' with a 'numberOfWheels()' method, and a 'Question4' class with a 'main()' method. The 'main()' method creates a 'Car' object 'v1' and calls both 'fuelType()' and 'numberOfWheels()'.
- TERMINAL:** Shows the execution output. It displays the command 'java Question4' being run from the directory 'C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)'. The output is 'Vehicle uses fuel' followed by 'Car has 4 wheels' on the next line.

```
1 class Vehicle {
2     public void fuelType() {
3         System.out.println(x:"Vehicle uses fuel");
4     }
5 }
6
7 class Car extends Vehicle {
8     public void numberOfWheels() {
9         System.out.println(x:"Car has 4 wheels");
10    }
11 }
12
13 public class Question4 {
14     Run | Debug
15     public static void main(String[] args) {
16         Car v1 = new Car();
17         v1.fuelType();
18         v1.numberOfWheels();
19     }
20 }
```

C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>java Question3
Cat meows

C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>javac Question4.java

C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>java Question4
Vehicle uses fuel
Car has 4 wheels

C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>

Q5.

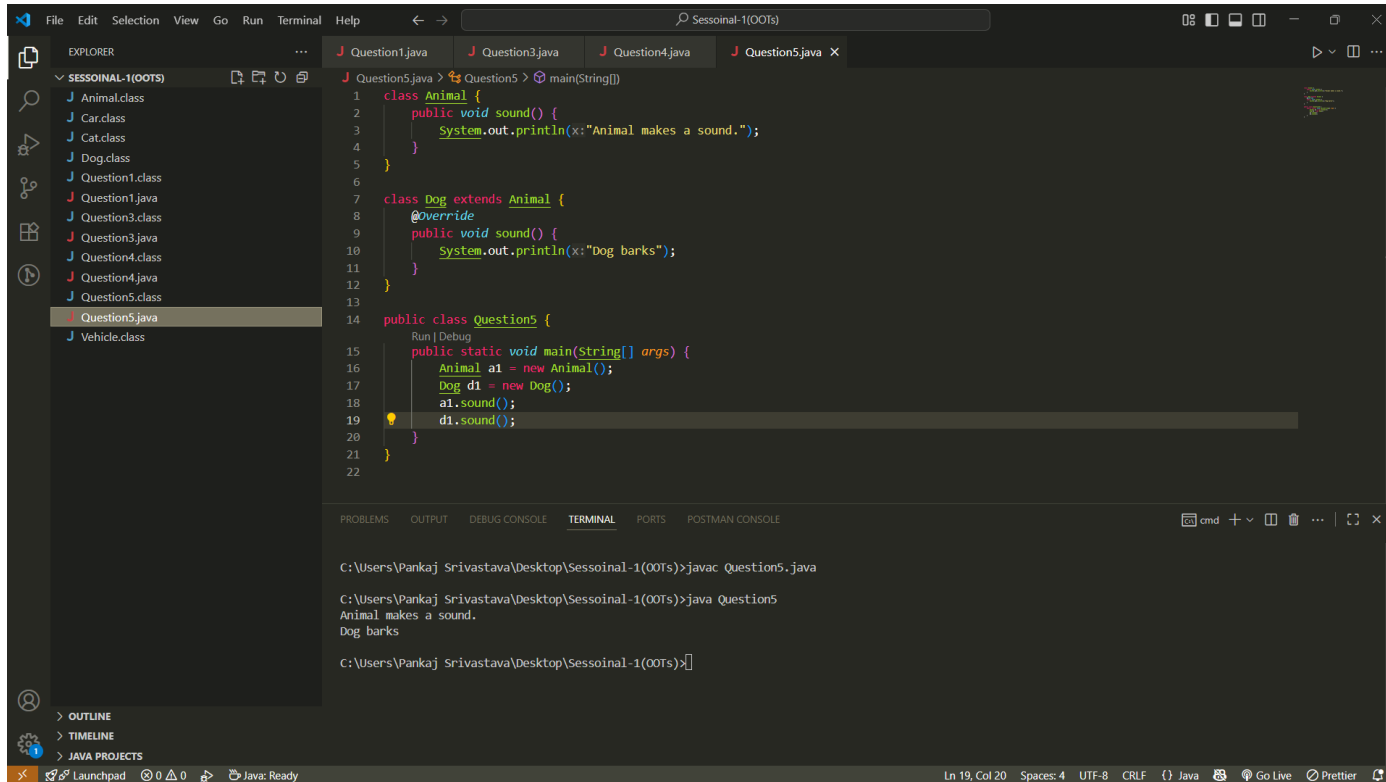
```
class Animal {  
    public void sound() {  
        System.out.println("Animal makes a sound.");  
    }  
}
```

```
class Dog extends Animal {  
    @Override  
    public void sound() {  
        System.out.println("Dog barks");  
    }  
}
```

```
public class Question5 {  
    public static void main(String[] args) {  
        Animal a1 = new Animal();  
        Dog d1 = new Dog();  
        a1.sound();  
        d1.sound();  
    }  
}
```


}

OUTPUT FOR QUESTION-5



The screenshot shows an IDE with the following components:

- EXPLORER:** A project named 'SESSOINAL-1(OOTs)' containing several Java files. 'Question5.java' is selected.
- EDITOR:** Displays the code for 'Question5.java'. It includes an 'Animal' class with a 'sound()' method, a 'Dog' class that extends 'Animal' and overrides 'sound()', and a 'Question5' class with a 'main' method that creates instances of 'Animal' and 'Dog' and calls their 'sound()' methods.
- TERMINAL:** Shows the execution of the program. The commands and output are:

```
C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>javac Question5.java
C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>java Question5
Animal makes a sound.
Dog barks
C:\Users\Pankaj Srivastava\Desktop\Sessoinal-1(OOTs)>]
```