**BCSE103E -**

**Computer Programming: Java**

Digital Assignment – 3

**Name:** Dhruv Rajeshkumar Shah

**Registration No –** 21BCE0611

1. Shapes

**CODE**

// JAVA DA - 3

// by Dhruv Rajeshkumar Shah

// 21BCE0611

// Circle class

class Circle {

    public double radius;

    public double area() {

        return Math.PI \* radius \* radius;

    }

    public double perimeter() {

        return 2 \* Math.PI \* radius;

    }

    public double circumference() {

        return perimeter();

    }

}

// Rectangle class

class Rectangle {

    public double length;

    public double breadth;

    public double area() {

        return length \* breadth;

    }

    public double perimeter() {

        return 2 \* (length + breadth);

    }

    public boolean isSquare() {

        return length == breadth;

    }

}

// Square class

class Cylinder {

    public double radius;

    public double height;

    public double lidArea() {

        return Math.PI \* radius \* radius;

    }

    public double circumference() {

        return 2 \* Math.PI \* radius;

    }

    public double totalSurfaceArea() {

        return 2 \* lidArea() + circumference() \* height;

    }

    public double volume() {

        return lidArea() \* height;

    }

}

// Main class

public class Shapes {

    public static void main(String[] args) {

        // Circle

        Circle c = new Circle();

        c.radius = 5;

        System.out.println("Circle");

        System.out.println("Area: " + c.area());

        System.out.println("Perimeter: " + c.perimeter());

        System.out.println("Circumference: " + c.circumference());

        System.out.println();

        // Rectangle

        Rectangle r = new Rectangle();

        r.length = 5;

        r.breadth = 10;

        System.out.println("Rectangle");

        System.out.println("Area: " + r.area());

        System.out.println("Perimeter: " + r.perimeter());

        System.out.println("Is Square: " + r.isSquare());

        System.out.println();

        // Cylinder

        Cylinder cy = new Cylinder();

        cy.radius = 5;

        cy.height = 10;

        System.out.println("Cylinder");

        System.out.println("Lid Area: " + cy.lidArea());

        System.out.println("Circumference: " + cy.circumference());

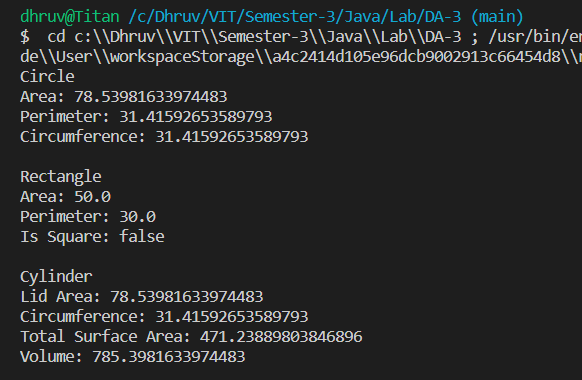
        System.out.println("Total Surface Area: " + cy.totalSurfaceArea());

        System.out.println("Volume: " + cy.volume());

    }

}

**OUTPUT**

****

1. Student marks

**CODE**

// JAVA DA - 3

// by Dhruv Rajeshkumar Shah

// 21BCE0611

class Student {

    public String name;

    public int rollNo;

    public String course;

    public int m1, m2, m3;

    public int total() {

        return m1 + m2 + m3;

    }

    public float average() {

        return (float) total() / 3;

    }

    public char grade() {

        if (average() >= 60) {

            return 'A';

        } else {

            return 'B';

        }

    }

    public String details() {

        return "Name: " + name + "\nRoll No: " + rollNo + "\nCourse: " + course + "\nMarks: " + m1 + ", " + m2 + ", "

                + m3 + "\nTotal: " + total() + "\nAverage: " + average() + "\nGrade: " + grade();

    }

    public String toString() {

        return details();

    }

}

public class Marks {

    public static void main(String[] args) {

        Student s1 = new Student();

        s1.name = "Dhruv";

        s1.rollNo = 0611;

        s1.course = "B.Tech";

        s1.m1 = 99;

        s1.m2 = 88;

        s1.m3 = 98;

        // Printing details by calling details() method

        System.out.println(s1.details());

        System.out.println();

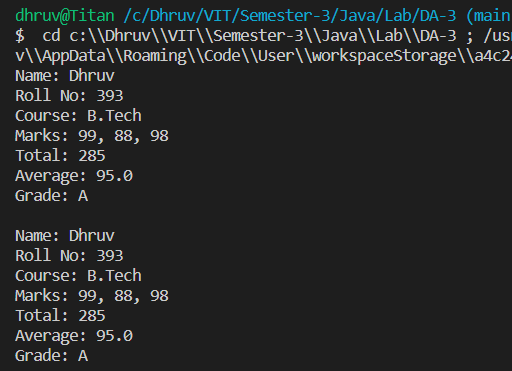
        // Printing details by calling toString() method

        System.out.println(s1);

    }

}

**OUTPUT**

****

1. Exercise 1 (Bank account, Television, car classes)

**CODE**

// JAVA DA - 3

// by Dhruv Rajeshkumar Shah

// 21BCE0611

// Bank account class

class BankAccount {

    public String name;

    public int accountNo;

    public double balance;

    public void deposit(double amount) {

        balance += amount;

    }

    public void withdraw(double amount) {

        if (amount > balance) {

            System.out.println("Insufficient balance");

        } else {

            balance -= amount;

        }

    }

    public void transfer(BankAccount other, double amount) {

        if (amount > balance) {

            System.out.println("Insufficient balance");

        } else {

            balance -= amount;

            other.balance += amount;

        }

    }

    public String toString() {

        return "Name: " + name + "\nAccount No: " + accountNo + "\nBalance: " + balance;

    }

}

// Telivision class

class Television {

    public String brand;

    public int size;

    public int volume;

    public int chanNo;

    public boolean isSmart;

    public void turnOn() {

        System.out.println("Turning on the TV");

    }

    public void turnOff() {

        System.out.println("Turning off the TV");

    }

    public void changeChannel(int ch) {

        chanNo = ch;

        System.out.println("Changing channel to " + chanNo);

    }

    public void changeVolume(int v) {

        volume = v;

        System.out.println("Changing volume to " + volume);

    }

    public String toString() {

        return "Brand: " + brand + "\nSize: " + size + "\nIs Smart: " + isSmart;

    }

}

// Car class

class Car {

    public String brand;

    public String model;

    public int year;

    public int speed;

    public void start() {

        System.out.println("Starting the car");

    }

    public void stop() {

        System.out.println("Stopping the car");

    }

    public void accelerate(int s) {

        speed += s;

        System.out.println("Accelerating to " + speed);

    }

    public void brake(int s) {

        speed -= s;

        System.out.println("Braking to " + speed);

    }

    public String toString() {

        return "Brand: " + brand + "\nModel: " + model + "\nYear: " + year;

    }

}

// Main class

public class Exercise1 {

    public static void main(String[] args) {

        // Bank account

        System.out.println("Bank account");

        BankAccount acc1 = new BankAccount();

        acc1.name = "Dhruv Shah";

        acc1.accountNo = 123456789;

        acc1.balance = 10000;

        System.out.println(acc1);

        acc1.deposit(1000);

        System.out.println(acc1);

        acc1.withdraw(5000);

        System.out.println(acc1);

        acc1.transfer(acc1, 1000);

        System.out.println(acc1);

        System.out.println();

        // Television

        System.out.println("Television");

        Television tv1 = new Television();

        tv1.brand = "Samsung";

        tv1.size = 32;

        tv1.volume = 10;

        tv1.chanNo = 1;

        tv1.isSmart = true;

        System.out.println(tv1);

        tv1.turnOn();

        tv1.changeChannel(5);

        tv1.changeVolume(15);

        tv1.turnOff();

        System.out.println();

        // Car

        System.out.println("Car");

        Car car1 = new Car();

        car1.brand = "Lamborghini";

        car1.model = "Veneno";

        car1.year = 2014;

        car1.speed = 0;

        System.out.println(car1);

        car1.start();

        car1.accelerate(100);

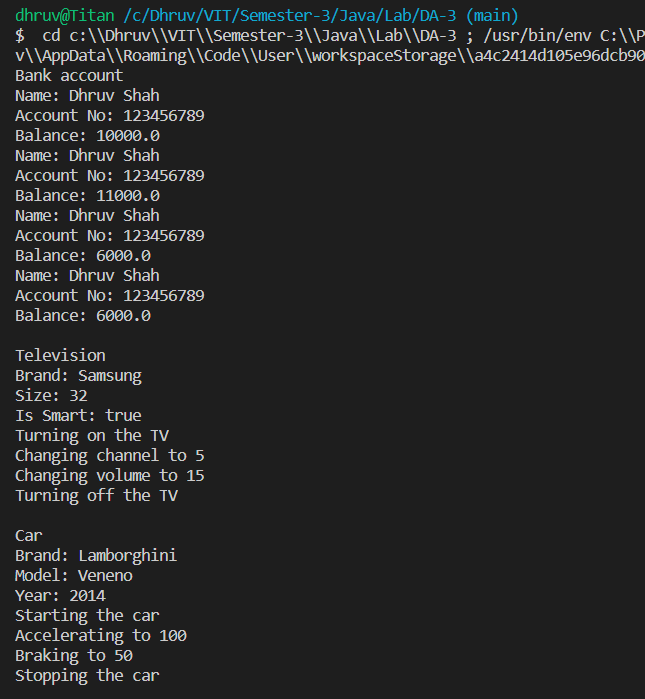
        car1.brake(50);

        car1.stop();

    }

}

**OUTPUT**

****

1. Access modifiers and read and write methods (Set and get)

**CODE**

// JAVA DA - 3

// by Dhruv Rajeshkumar Shah

// 21BCE0611

class PrivateRectangle {

    private double length;

    private double breadth;

    public double getLength() {

        return length;

    }

    public double getBreadth() {

        return breadth;

    }

    public double getArea() {

        return length \* breadth;

    }

    public double getPerimeter() {

        return 2 \* (length + breadth);

    }

    public void setLength(double length) {

        if (length > 0) {

            this.length = length;

        } else {

            this.length = 0;

        }

    }

    public void setBreadth(double breadth) {

        if (breadth > 0) {

            this.breadth = breadth;

        } else {

            this.breadth = 0;

        }

    }

    public boolean isSquare() {

        return length == breadth;

    }

}

public class AccessModifiers {

    public static void main(String[] args) {

        PrivateRectangle r1 = new PrivateRectangle();

        r1.setLength(10);

        r1.setBreadth(10);

        System.out.println("Length: " + r1.getLength());

        System.out.println("Breadth: " + r1.getBreadth());

        System.out.println("Area: " + r1.getArea());

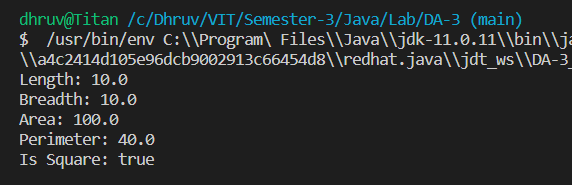
        System.out.println("Perimeter: " + r1.getPerimeter());

        System.out.println("Is Square: " + r1.isSquare());

    }

}

**OUTPUT**

****

1. Constructors (Default, parameterized and constructor overloading for rectangle and cylinder)

**CODE**

// JAVA DA - 3

// by Dhruv Rajeshkumar Shah

// 21BCE0611

// Rectangle class

class ConstructorRectangle {

    public double length;

    public double breadth;

    // Default constructor

    public ConstructorRectangle() {

        length = 0;

        breadth = 0;

    }

    // Parameterized constructor

    public ConstructorRectangle(double length, double breadth) {

        this.length = length;

        this.breadth = breadth;

    }

    // Overloading constructor

    public ConstructorRectangle(double side) {

        length = side;

        breadth = side;

    }

    public void setBreadth(double breadth) {

        this.breadth = breadth;

    }

    public void setLength(double length) {

        this.length = length;

    }

    public double getBreadth() {

        return breadth;

    }

    public double getLength() {

        return length;

    }

    public double area() {

        return length \* breadth;

    }

    public double perimeter() {

        return 2 \* (length + breadth);

    }

    public boolean isSquare() {

        return length == breadth;

    }

}

// Cylinder class

class ConstructorCylinder {

    public double radius;

    public double height;

    // Default constructor

    public ConstructorCylinder() {

        radius = 0;

        height = 0;

    }

    // Parameterized constructor

    public ConstructorCylinder(double radius, double height) {

        this.radius = radius;

        this.height = height;

    }

    // Overloading constructor

    public ConstructorCylinder(double radius) {

        this.radius = radius;

        height = 0;

    }

    public void setRadius(double radius) {

        this.radius = radius;

    }

    public void setHeight(double height) {

        this.height = height;

    }

    public double getRadius() {

        return radius;

    }

    public double getHeight() {

        return height;

    }

    public double lidArea() {

        return Math.PI \* radius \* radius;

    }

    public double drumArea() {

        return 2 \* Math.PI \* radius \* height;

    }

    public double circumference() {

        return 2 \* Math.PI \* radius;

    }

    public double totalSurfaceArea() {

        return 2 \* lidArea() + circumference() \* height;

    }

    public double volume() {

        return lidArea() \* height;

    }

}

public class Constructor {

    public static void main(String[] args) {

        // Rectangle

        System.out.println("Rectangle");

        ConstructorRectangle r1 = new ConstructorRectangle();

        ConstructorRectangle r2 = new ConstructorRectangle(10, 20);

        ConstructorRectangle r3 = new ConstructorRectangle(10);

        System.out.println("Area of r1: " + r1.area());

        System.out.println("Perimeter of r1: " + r1.perimeter());

        System.out.println("Is r1 a square: " + r1.isSquare());

        System.out.println();

        System.out.println("Area of r2: " + r2.area());

        System.out.println("Perimeter of r2: " + r2.perimeter());

        System.out.println("Is r2 a square: " + r2.isSquare());

        System.out.println();

        System.out.println("Area of r3: " + r3.area());

        System.out.println("Perimeter of r3: " + r3.perimeter());

        System.out.println("Is r3 a square: " + r3.isSquare());

        System.out.println();

        System.out.println("----------------------------------------");

        // Cylinder

        System.out.println("Cylinder");

        ConstructorCylinder c1 = new ConstructorCylinder();

        ConstructorCylinder c2 = new ConstructorCylinder(10, 20);

        ConstructorCylinder c3 = new ConstructorCylinder(10);

        System.out.println("Lid area of c1: " + c1.lidArea());

        System.out.println("Drum area of c1: " + c1.drumArea());

        System.out.println("Circumference of c1: " + c1.circumference());

        System.out.println("Total surface area of c1: " + c1.totalSurfaceArea());

        System.out.println("Volume of c1: " + c1.volume());

        System.out.println();

        System.out.println("Lid area of c2: " + c2.lidArea());

        System.out.println("Drum area of c2: " + c2.drumArea());

        System.out.println("Circumference of c2: " + c2.circumference());

        System.out.println("Total surface area of c2: " + c2.totalSurfaceArea());

        System.out.println("Volume of c2: " + c2.volume());

        System.out.println();

        System.out.println("Lid area of c3: " + c3.lidArea());

        System.out.println("Drum area of c3: " + c3.drumArea());

        System.out.println("Circumference of c3: " + c3.circumference());

        System.out.println("Total surface area of c3: " + c3.totalSurfaceArea());

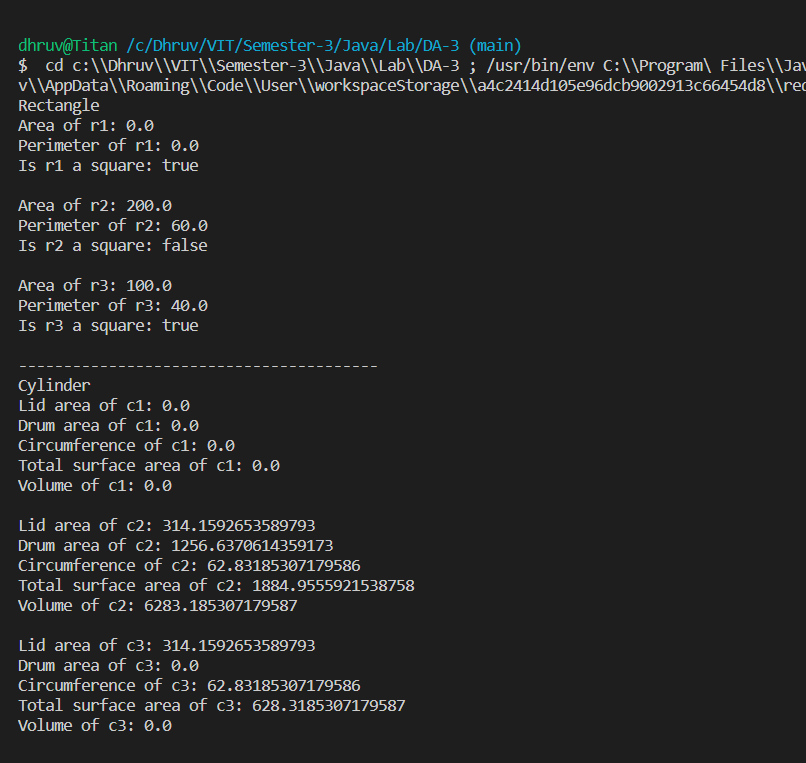
        System.out.println("Volume of c3: " + c3.volume());

        System.out.println();

    }

}

**OUTPUT**

****

1. Exercise 2 (Product and customer classes)

**CODE**

// JAVA DA - 3

// by Dhruv Rajeshkumar Shah

// 21BCE0611

// Product class

class Product {

    private int itemNo;

    private String name;

    private double price;

    private int quantity;

    // Default constructor

    public Product() {

        itemNo = 0;

        name = "";

        price = 0;

        quantity = 0;

    }

    // Parameterized constructor

    public Product(int itemNo, String name, double price, int quantity) {

        this.itemNo = itemNo;

        this.name = name;

        this.price = price;

        this.quantity = quantity;

    }

    public void setPrice(double price) {

        this.price = price;

    }

    public void setQuantity(int quantity) {

        this.quantity = quantity;

    }

    public int getItemNo() {

        return itemNo;

    }

    public String getName() {

        return name;

    }

    public double getPrice() {

        return price;

    }

    public int getQuantity() {

        return quantity;

    }

}

// Customer class

class Customer {

    private int custId;

    private String name;

    private String address;

    private String phoneNo;

    // Default constructor

    public Customer() {

        custId = 0;

        name = "";

        address = "";

        phoneNo = "";

    }

    // Parameterized constructor

    public Customer(int custId, String name, String address, String phoneNo) {

        this.custId = custId;

        this.name = name;

        this.address = address;

        this.phoneNo = phoneNo;

    }

    public void setAddress(String address) {

        this.address = address;

    }

    public void setPhoneNo(String phoneNo) {

        this.phoneNo = phoneNo;

    }

    public int getCustId() {

        return custId;

    }

    public String getName() {

        return name;

    }

    public String getAddress() {

        return address;

    }

    public String getPhoneNo() {

        return phoneNo;

    }

    public String toString() {

        return "Customer ID: " + custId + "\nName: " + name + "\nAddress: " + address + "\nPhone Number: " + phoneNo;

    }

}

// Main class

public class Exercise2 {

    public static void main(String[] args) {

        // Create a product object

        Product product = new Product(1, "Laptop", 50000, 1);

        // Create a customer object

        Customer customer = new Customer(1, "Dhruv", "Mumbai", "1234567890");

        // Print the details of the product

        System.out.println("Product Details:");

        System.out.println("Item Number: " + product.getItemNo());

        System.out.println("Name: " + product.getName());

        System.out.println("Price: " + product.getPrice());

        System.out.println("Quantity: " + product.getQuantity());

        // Print the details of the customer

        System.out.println("\nCustomer Details:");

        System.out.println(customer);

        // Update the price and quantity of the product

        product.setPrice(60000);

        product.setQuantity(2);

        // Update the address and phone number of the customer

        customer.setAddress("Pune");

        customer.setPhoneNo("0987654321");

        // Print the updated details of the product

        System.out.println("\nUpdated Product Details:");

        System.out.println("Item Number: " + product.getItemNo());

        System.out.println("Name: " + product.getName());

        System.out.println("Price: " + product.getPrice());

        System.out.println("Quantity: " + product.getQuantity());

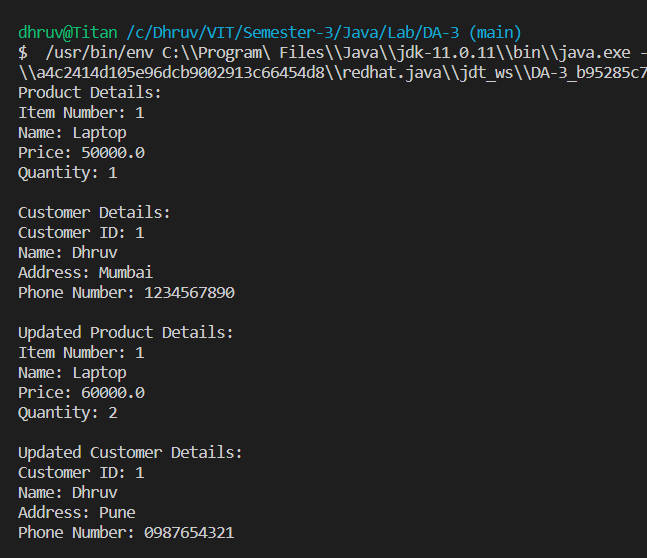
        // Print the updated details of the customer

        System.out.println("\nUpdated Customer Details:");

        System.out.println(customer);

    }

}

**OUTPUT**

1. Exercise 3 (Marks in subject)

**CODE**

// JAVA DA - 3

// by Dhruv Rajeshkumar Shah

// 21BCE0611

class Subject {

    private String subId;

    private String name;

    private int maxMarks;

    private int marksObtained;

    public Subject(String subId, String name, int maxMarks) {

        this.subId = subId;

        this.name = name;

        this.maxMarks = maxMarks;

    }

    public String getSubId() {

        return subId;

    }

    public String getName() {

        return name;

    }

    public int getMaxMarks() {

        return maxMarks;

    }

    public int getMarksObtained() {

        return marksObtained;

    }

    public void setMarksObtained(int marksObtained) {

        this.marksObtained = marksObtained;

    }

    public void setMaxMarks(int maxMarks) {

        this.maxMarks = maxMarks;

    }

    boolean isQualified() {

        return marksObtained >= (maxMarks \* 0.4);

    }

    public String toString() {

        return "Subject ID: " + subId + "\nSubject Name: " + name + "\nMaximum Marks: " + maxMarks

                + "\nMarks Obtained: " + marksObtained + "\nPassed: " + isQualified() + "\n";

    }

}

public class Exercise3 {

    public static void main(String[] args) {

        Subject sub1 = new Subject("CS101", "Computer Programming", 100);

        Subject sub2 = new Subject("CS102", "Data Structures", 100);

        Subject sub3 = new Subject("CS103", "Discrete Mathematics", 100);

        sub1.setMarksObtained(80);

        sub2.setMarksObtained(70);

        sub3.setMarksObtained(30);

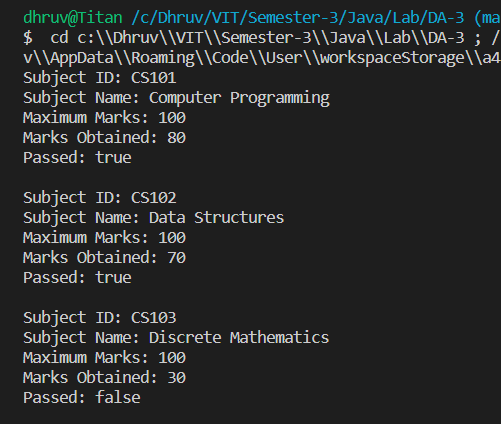
        System.out.println(sub1);

        System.out.println(sub2);

        System.out.println(sub3);

    }

}

**OUTPUT**

1. Static and final members and blocks

**CODE**

// JAVA DA - 3

// by Dhruv Rajeshkumar Shah

// 21BCE0611

class StaticCar {

    // Static variable

    static long carCount = 0;

    // Final variable

    final long CARID = 100002343;

    // Static block

    static {

        System.out.println("Static block called");

    }

    // Constructor

    StaticCar() {

        carCount++;

    }

    // Static method

    static void printCarCount() {

        System.out.println("Number of cars: " + carCount);

    }

    // Final method

    final void printCarId() {

        System.out.println("Car ID: " + CARID);

    }

}

public class Static {

    public static void main(String[] args) {

        StaticCar.printCarCount();

        StaticCar car1 = new StaticCar();

        StaticCar car2 = new StaticCar();

        StaticCar car3 = new StaticCar();

        StaticCar.printCarCount();

        car1.printCarId();

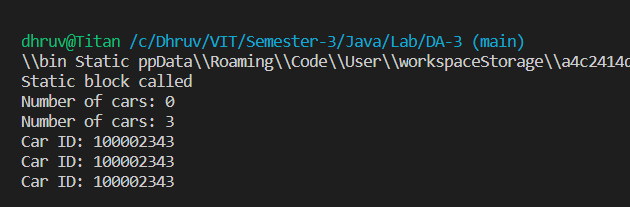
        car2.printCarId();

        car3.printCarId();

    }

}

**OUTPUT**

****