1.Write a Java program to check whether the given number is Palindrome or not

```
import java.util.Scanner;
public class PalindromeCheck {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int number = scanner.nextInt();
        int originalNumber = number;
        int reversedNumber = 0;
        while (number != 0) {
            int digit = number % 10;
            reversedNumber = reversedNumber * 10 + digit;
            number /= 10;
        }
        if (originalNumber == reversedNumber) {
            System.out.println(originalNumber + " is a Palindrome.");
        } else {
            System.out.println(originalNumber + " is not a Palindrome.");
        }
        scanner.close();
    }
}
```

2. Write a Java program

```
import java.util.Scanner;
public class CircleAreaAndPrimeCheck {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the radius of the circle: ");
double radius = scanner.nextDouble();
double area = Math.PI * radius * radius;
double circumference = 2 * Math.PI * radius;
System.out.println("Area of the circle: " + area);
System.out.println("Circumference of the circle: " + circumference);
System.out.print("Enter a number: ");
int number = scanner.nextInt();
boolean isPrime = true;
if (number <= 1) {
isPrime = false; }
else {
for (int i = 2; i <= Math.sqrt(number); i++)  {
if (number % i == 0)  {
isPrime = false;
break; } } }
if (isPrime)
System.out.println(number + " is a prime number.");
else
System.out.println(number + " is not a prime number.");
scanner.close(); } }
```

3. Write a Java program to demonstrate a division by zero exception.

```
public class DivisionByZeroExceptionDemo {
   public static void main(String[] args) {
 try {
          int numerator = 10;
          int denominator = 0;
          int result = numerator / denominator;
          System.out.println("Result: " + result);
       } catch (ArithmeticException e) {
          System.out.println("Exception caught: Division by zero is not allowed!"); } } }
```

4. Write a Java program to implement Inner class and demonstrate its Access protection.

```
class Outer {
int a = 10;
public int b = 20;
private int c = 30;
protected int d = 40;
class inner { int p = 5;
public int q = 15;
 private int r = 25;
protected int s = 35;
void display() {
System.out.println("Inner class");
System.out.println("Value of a=" + a);
System.out.println("Value of public variable" + b);
System.out.println("Value of private variable" + c);
System.out.println("Value of protected variable" + d); } }
void outermet() {
inner inn = new inner();
inn.display();
 System.out.println("Outer class");
System.out.println("Value of p=" + inn.p);
System.out.println("Value of public variable" + inn.q);
System.out.println("Value of private variable" + inn.r);
System.out.println("Value of protected variable" + inn.s); } }
public class lab02 {
public static void main(String[] args) { {
Outer ot = new Outer();
ot.outermet();
} } }
```

5. Constructor Overloading and Method Overloading.

```
class cube {
double l, b, h;
 cube() {
System.out.println("Constructor with no arguments\n");
 l = 0; b = 0; h = 0; }
cube(double i) {
System.out.println("Constructor with one arguments\n");
 l = b = h = i; }
cube(double x, double y, double z)  {
 System.out.println("Constructor with three arguments\n");
l = x; b = y; h = z; }
void area() {
System.out.println("Method with no arguments");
double a = l * b * h;
System.out.println("Area of Cube is " + a + "\n"); }
void area(double t) {
System.out.println("Method with one arguments");
double a = t * t * t;
System.out.println("Area of Cube is " + a + "\n"); }
void area(double p, double q, double r) {
System.out.println("Method with three arguments");
double a = p * q * r;
System.out.println("Area of Cube is " + a + "\n"); } }
public class lab01 {
public static void main(String[] args)  {
 cube A = new cube(5, 6, 7);
A.area();
cube B = new cube(5);
```

```
B.area(); cube C = new cube();
 C.area(); C.area(6);
C.area(6, 7, 8); } }
```

7. Write a JAVA program to create an enumeration Day of Week

```
public class DayOfWeekTest {
   public enum DayOfWeek {
      SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY;
      public boolean isWorkday() {
         return this != SATURDAY && this != SUNDAY;
      } }
   public static void main(String[] args) {
      System.out.println(DayOfWeek.SUNDAY + " is workday: " +
DayOfWeek.SUNDAY.isWorkday());
      System.out.println(DayOfWeek.MONDAY + " is workday: " +
DayOfWeek.MONDAY.isWorkday());
      System.out.println(DayOfWeek.FRIDAY + " is workday: " +
DayOfWeek.FRIDAY.isWorkday());
      System.out.println(DayOfWeek.SATURDAY + " is workday: " +
DayOfWeek.SATURDAY.isWorkday());
   } }
```

8. Write a Java program that displays the number of characters, lines and words in a text file

```
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;
public class TextFileAnalyzer {
   public static void main(String[] args) {
      String fileName = "C:\\Users\\diwak\\OneDrive\\Desktop\\Files\\College Data\\DSATM\\NBA MCA DSATM\\New Text Document.txt"; // Change this to your file name
      try {
         BufferedReader reader = new BufferedReader(new FileReader(fileName));
         int charCount = 0;
         int wordCount = 0;
         int lineCount = 0;
         String line;
         while ((line = reader.readLine()) != null) {
            lineCount++;
            String[] words = line.split("\\s+"); // Split by whitespace
            wordCount += words.length;
            for (String word : words) {
               charCount += word.length(); } }
         reader.close();
         System.out.println("Number of characters: " + charCount);
         System.out.println("Number of words: " + wordCount);
         System.out.println("Number of lines: " + lineCount);
      } catch (IOException e) {
         System.err.println("Error reading the file: " + e.getMessage());
      }
   } }
```

9. Create a package named shape and Create some classes in the package

Program (Rectangle1.java)

```
package Shape;
public class Rectangle1 {
private double length, breadth;
public void setRectangle(double len, double br) {
length = len; breadth = br; }
public void area() {
double area = length * breadth;
System.out.println("Area of Rectangle =" + area); } }
```

Program (Square.java)

```
package Shape;
public class Square {
private double side;
public void setSquare(double val) {
side = val; }
public void area() {
System.out.println("Area of Square=" + (side * side)); } }
```

Program(Circle1.java)

```
package Shape;
public class Circle1 {
private double rad;
public void setCircle(double radius) {
rad = radius; }
public void area() {
double area =  3.14 * rad * rad;
System.out.println("Area of Rectangle =" + area); } }
```

Program(Program10.java)

```
import Shape.Rectangle1;
```

```
import Shape.Square;
import Shape.Circle1;
public class Program10 {
public static void main(String args[]) {
Rectangle1 rect = new Rectangle1();
rect.setRectangle(5.6, 6.4);
rect.area();
Square sq = new Square();
sq.setSquare(10.5);
sq.area();
Circle1 round = new Circle1();
round.setCircle(5.6);
round.area();
}
}
```

10. Write a Java applet program, which handles keyboard event.

```
import java.awt.*;
import java.awt.event.*;
public class Program9 extends Frame implements KeyListener {
Label lbl;
Program9() {
   addKeyListener(this);
   requestFocus();
   lbl = new Label();
   lbl.setBounds(100, 100, 200, 40);
   lbl.setFont(new Font("Calibri", Font.BOLD, 16));
   add(lbl);
   setSize(400, 300);
   setLayout(null);
   setVisible(true);  }
public void keyPressed(KeyEvent e) {
if (e.getKeyChar() == 'M' || e.getKeyChar() == 'm')
lbl.setText("Good morning");
else if (e.getKeyChar() == 'A' || e.getKeyChar() == 'a')
lbl.setText("Good afternoon");
else if (e.getKeyChar() == 'E' || e.getKeyChar() == 'e')
lbl.setText("Good evening");
else if (e.getKeyChar() == 'N' || e.getKeyChar() == 'n')
lbl.setText("Good night"); }
public void keyReleased(KeyEvent e) { }
public void keyTyped(KeyEvent e) { }
public static void main(String[] args) {
new Program9();
} }
```

6. Write a JAVA program to demonstrate Inheritance.

```
abstract class Shape {
    String name;
    Shape(String name) {
        this.name = name; }
    abstract double calculateArea();
    void displayInfo() {
        System.out.println("Shape: " + name); } }
class Rectangle extends Shape {
    double length, width;
    Rectangle(double length, double width) {
        super("Rectangle");
        this.length = length;
        this.width = width; }
    @Override
    double calculateArea() {
        return length * width; } }
class Circle extends Shape {
    double radius;
    Circle(double radius) {
        super("Circle");
        this.radius = radius; }
    @Override
    double calculateArea() {
        return Math.PI * radius * radius; } }
public class ShapeInheritanceDemo {
    public static void main(String[] args) {
        Rectangle rectangle = new Rectangle(5, 3);
        rectangle.displayInfo();
```

```
        System.out.println("Area: " + rectangle.calculateArea());
        System.out.println();
        Circle circle = new Circle(4);
        circle.displayInfo();
        System.out.println("Area: " + circle.calculateArea());
    }
}
```

Output:

```
Shape: Rectangle
Area: 15.0
Shape: Circle
Area: 50.26548245743669
```

@B?

```
interface Engine {
    void startEngine(); }
interface Transmission {
    void changeGear(int gear);  }
class Car implements Engine, Transmission {
    private int currentGear;
    @Override
    public void startEngine() {
        System.out.println("Engine started."); }
    @Override
    public void changeGear(int gear) {
        currentGear = gear;
        System.out.println("Gear changed to: " + currentGear); }
    void drive() {
        System.out.println("Car is driving.");
    }
```

```
}
public class MultipleInheritanceDemo {
    public static void main(String[] args) {
        Car myCar = new Car();
        myCar.startEngine();
        myCar.changeGear(2);
       myCar.drive();
    }
}
```