1. Write a program to find the average and sum of the N numbers using Command line Argument.

Programiz
Online Java Compiler

SumAverage.java

```java
import java.util.Scanner;

public class SumAverage {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter how many numbers: ");
        int n = sc.nextInt();
        int sum = 0;
        for (int i = 0; i < n; i++) {
            System.out.print("Enter number " + (i + 1) + ": ");
            sum += sc.nextInt();
        }
        double average = (double) sum / n;
        System.out.println("Sum: " + sum);
        System.out.println("Average: " + average);
    }
}
```

Output

```
Enter how many numbers: 4
Enter number 1: 10
Enter number 2: 20
Enter number 3: 30
Enter number 4: 40
Sum: 100
Average: 25.0

=== Code Execution Successful ===
```

2. Write a program to demonstrate type casting.

Programiz
Online Java Compiler

TypeCasting.java

```java
public class TypeCasting {
    public static void main(String[] args) {
        int i = 100;
        double d = i; // Implicit
        double x = 45.67;
        int y = (int)x; // Explicit
        System.out.println("Implicit: " + d);
        System.out.println("Explicit: " + y);
    }
}
```

Output

```
Implicit: 100.0
Explicit: 45

=== Code Execution Successful ===
```

3. Write a program to generate prime numbers between 1 & given number.

Programiz
Online Java Compiler

PrimeNumbers.java

Share Run

```java
import java.util.Scanner;
public class PrimeNumbers {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the upper limit: ");
        int n = sc.nextInt();
        for (int i = 2; i <= n; i++) {
            boolean isPrime = true;
            for (int j = 2; j <= i / 2; j++) {
                if (i % j == 0) {
                    isPrime = false;
                    break;
                }
            }
            if (isPrime) System.out.print(i + " ");
        }
    }
}
```

Output Clear

```
Enter the upper limit: 5
2 3 5
=== Code Execution Successful ===
```

4. Write a program to generate pyramid of stars using nested for loops.

Programiz
Online Java Compiler

StarPyramid.java

Share Run

```java
public class StarPyramid {
    public static void main(String[] args) {
        int rows = 5;
        for (int i = 1; i <= rows; i++) {
            for (int j = i; j < rows; j++) System.out.print(" ");
            for (int k = 1; k <= (2 * i - 1); k++) System.out.print("*");
            System.out.println();
        }
    }
}
```

Output Clear

```
    *
   ***
  *****
 *******
*********

=== Code Execution Successful ===
```

5. Write a program to reversed pyramid using for loops & decrement operator.

Programiz
Online Java Compiler

ReversePyramid.java

Share Run

```java
public class ReversePyramid {
    public static void main(String[] args) {
        int rows = 5;
        for (int i = rows; i >= 1; i--) {
            for (int j = rows; j > i; j--) System.out.print(" ");
            for (int k = 1; k <= (2 * i - 1); k++) System.out.print("*");
            System.out.println();
        }
    }
}
```

Output Clear

```
*********
 *******
  *****
   ***
    *

=== Code Execution Successful ===
```

6. Write a program for demonstrate Nested Switch .

Programiz
Online Java Compiler

NestedSwitch.java

Share Run

```java
public class NestedSwitch {
    public static void main(String[] args) {
        int dept = 1;
        int year = 2;
        switch (dept) {
            case 1:
                switch (year) {
                    case 1: System.out.println("CSE First Year");
                        break;
                    case 2: System.out.println("CSE Second Year");
                        break;
                }
                break;
            default:
                System.out.println("Invalid Dept");
        }
    }
}
```

Output Clear

```
CSE Second Year

=== Code Execution Successful ===
```

7. Write a program to calculate area of a circle using radius .

Programiz
Online Java Compiler

CircleArea.java

```java
import java.util.Scanner;
public class CircleArea {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter radius: ");
        double radius = sc.nextDouble();
        double area = Math.PI * radius * radius;
        System.out.println("Area: " + area);
    }
}
```

Output

```
Enter radius: 5
Area: 78.53981633974483

=== Code Execution Successful ===
```

8. Write a program to find G.C.D of the number.

Programiz
Online Java Compiler

GCD.java

```java
import java.util.Scanner;
public class GCD {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter two numbers: ");
        int a = sc.nextInt();
        int b = sc.nextInt();
        while (b != 0) {
            int temp = b;
            b = a % b;
            a = temp;
        }
        System.out.println("GCD: " + a);
    }
}
```

Output

```
Enter two numbers: 4 5
GCD: 1

=== Code Execution Successful ===
```

9. Write a program to design a class account using the inheritance and static members which show all functions of a bank (Withdrawl, deposit)

OnlineGDB
online compiler and debugger for c/c++

code. compile. run. debug. share.

IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login

Run | Debug | Stop | Share | Save | { } Beautify

BankApp.java

```java
import java.util.Scanner;

class Bank {
    static String bankName = "State Bank of India";

    void showBankName() {
        System.out.println("Bank: " + bankName);
    }
}

class Account extends Bank {
    int accNumber;
    String holderName;
    double balance;

    Account(int accNumber, String holderName, double balance) {
        this.accNumber = accNumber;
        this.holderName = holderName;
        this.balance = balance;
    }

    void deposit(double amount) {
        balance += amount;
        System.out.println("₹" + amount + " deposited. Current balance: ₹" + balance);
    }

    void withdraw(double amount) {
        if (amount <= balance) {
            balance -= amount;
            System.out.println("₹" + amount + " withdrawn. Current balance: ₹" + balance);
        } else {
            System.out.println("Insufficient balance!");
        }
    }

    void display() {
```

input

```
Enter account number: 8527926166
Exception in thread "main" java.util.InputMismatchException: For input string: "8527926166"
        at java.base/java.util.Scanner.nextInt(Scanner.java:2290)
        at java.base/java.util.Scanner.nextInt(Scanner.java:2238)
        at BankApp.main(BankApp.java:48)


...Program finished with exit code 1
Press ENTER to exit console.
```

code. compile. run. debug. share.

IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login

```java
    }

    void display() {
        System.out.println("Account Number: " + accNumber);
        System.out.println("Holder Name: " + holderName);
        System.out.println("Balance: ₹" + balance);
    }
}

public class BankApp {
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter account number: ");
        int accNo = sc.nextInt();

        sc.nextLine();
        System.out.print("Enter account holder name: ");
        String name = sc.nextLine();

        System.out.print("Enter initial balance: ");
        double bal = sc.nextDouble();

        Account acc = new Account(accNo, name, bal);
        acc.showBankName();
        acc.display();

        System.out.print("Enter amount to deposit: ");
        double dep = sc.nextDouble();
        acc.deposit(dep);

        System.out.print("Enter amount to withdraw: ");
        double with = sc.nextDouble();
        acc.withdraw(with);
    }
}
```

input

```
Enter account number: 8527926166
Exception in thread "main" java.util.InputMismatchException: For input string: "8527926166"
        at java.base/java.util.Scanner.nextInt(Scanner.java:2290)
        at java.base/java.util.Scanner.nextInt(Scanner.java:2238)
        at BankApp.main(BankApp.java:48)


...Program finished with exit code 1
Press ENTER to exit console.
```

10. Write a program to create a simple class to find out the area and perimeter of rectangle

Rectangle.java

```java
class Shape {
    int length = 10;
}

class Rectangle extends Shape {
    int length = 20;
    void display() {
        System.out.println("Length from super: " + super.length);
        System.out.println("Length from this: " + this.length);
        int breadth = 5;
        System.out.println("Area: " + (this.length * breadth));
        System.out.println("Perimeter: " + 2 * (this.length + breadth));
    }
    public static void main(String[] args) {
        new Rectangle().display();
    }
}
```

```
Length from super: 10
Length from this: 20
Area: 100
Perimeter: 50

...Program finished with exit code 0
Press ENTER to exit console.
```

11. Write a program to find the factorial of a given number using recursion.

Programiz
Online Java Compiler

Factorial.java

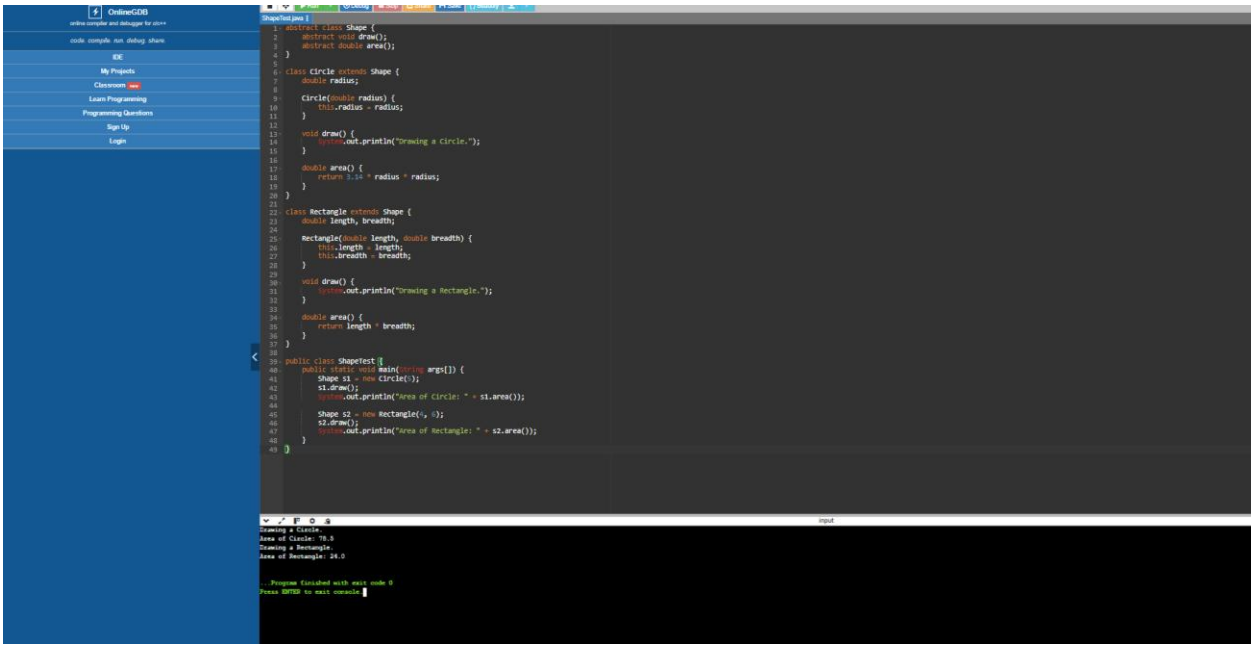
```java
import java.util.Scanner;
public class Factorial {
    static int fact(int n) {
        if (n == 0) return 1;
        return n * fact(n - 1);
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        System.out.println("Factorial: " + fact(num));
    }
}
```

Output

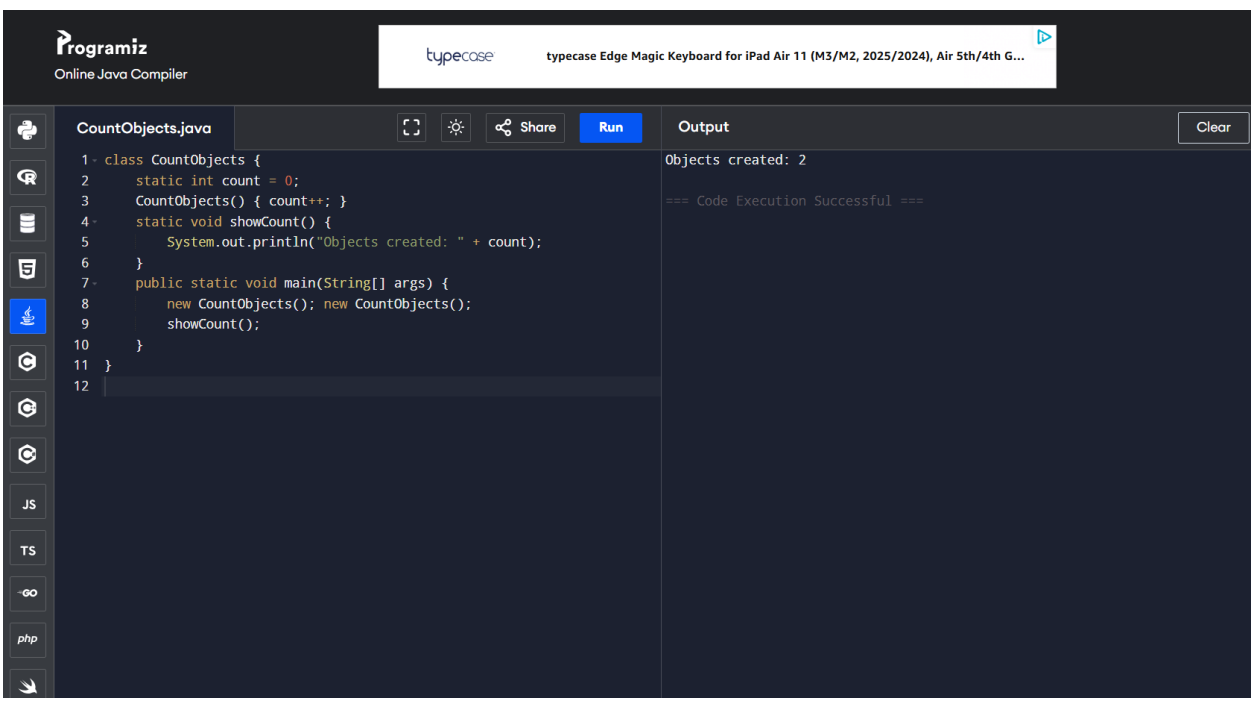
```
Enter a number: 5
Factorial: 120

=== Code Execution Successful ===
```

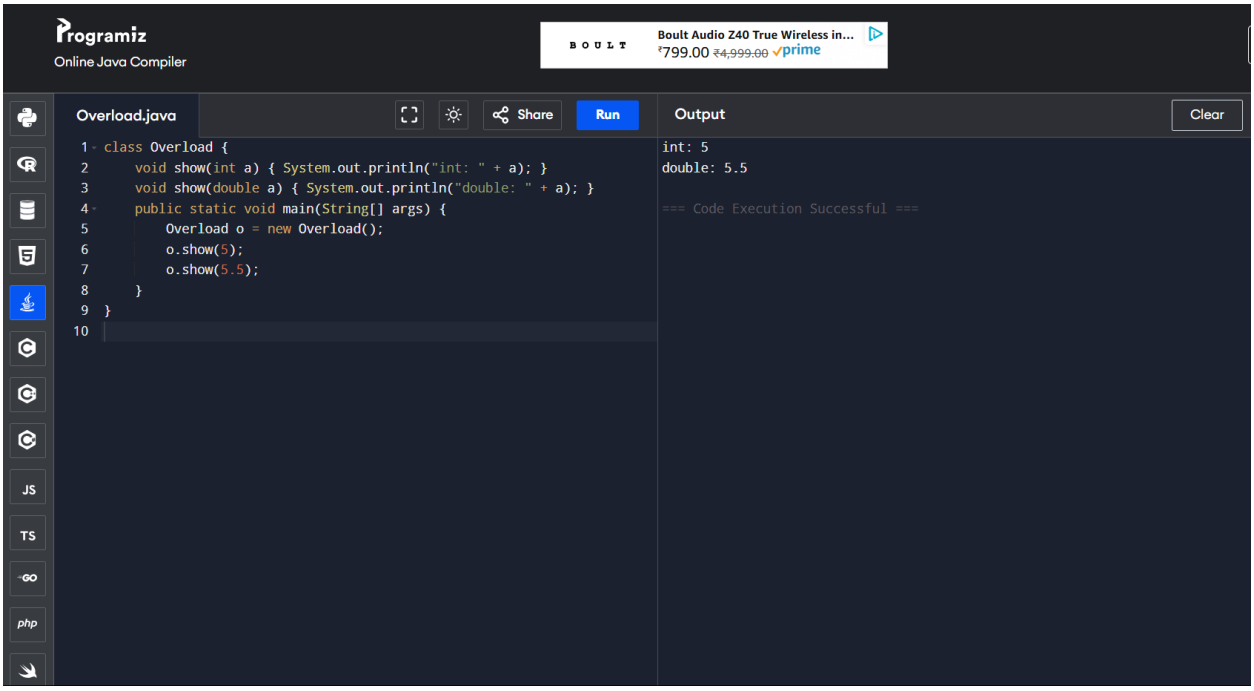
12. Write a program to design a class using abstract methods and abstract classes.



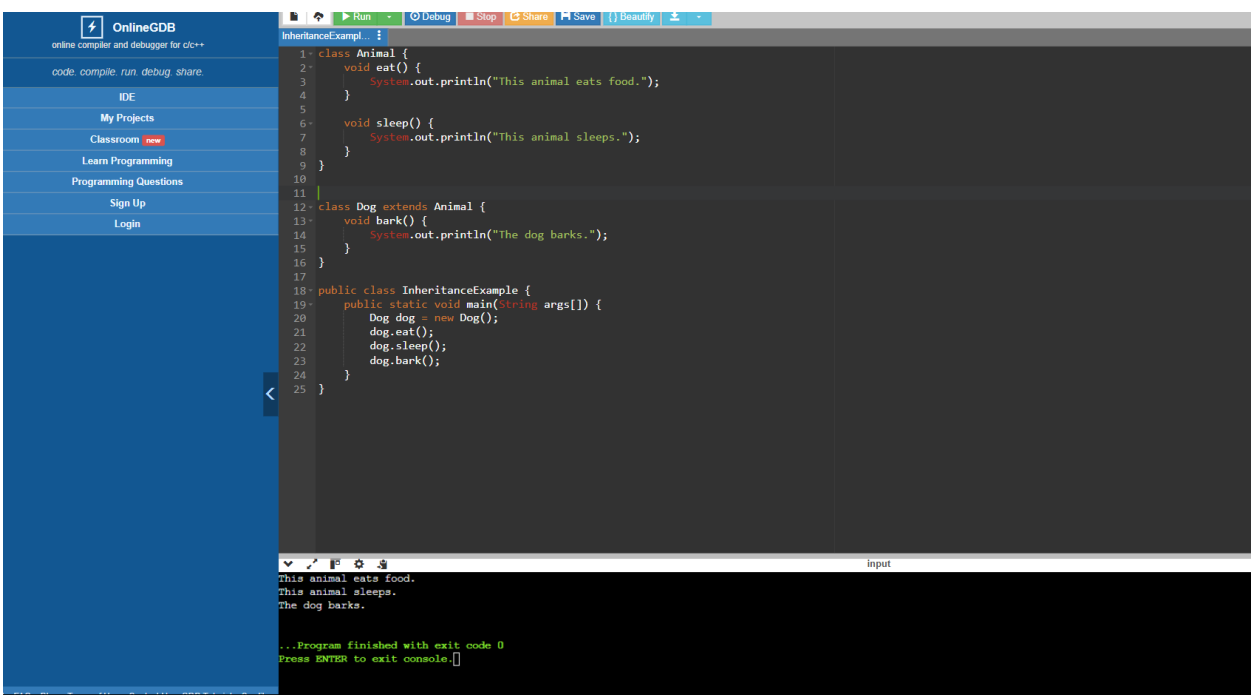
13. Write a program to count the number of objects created for a class using static member



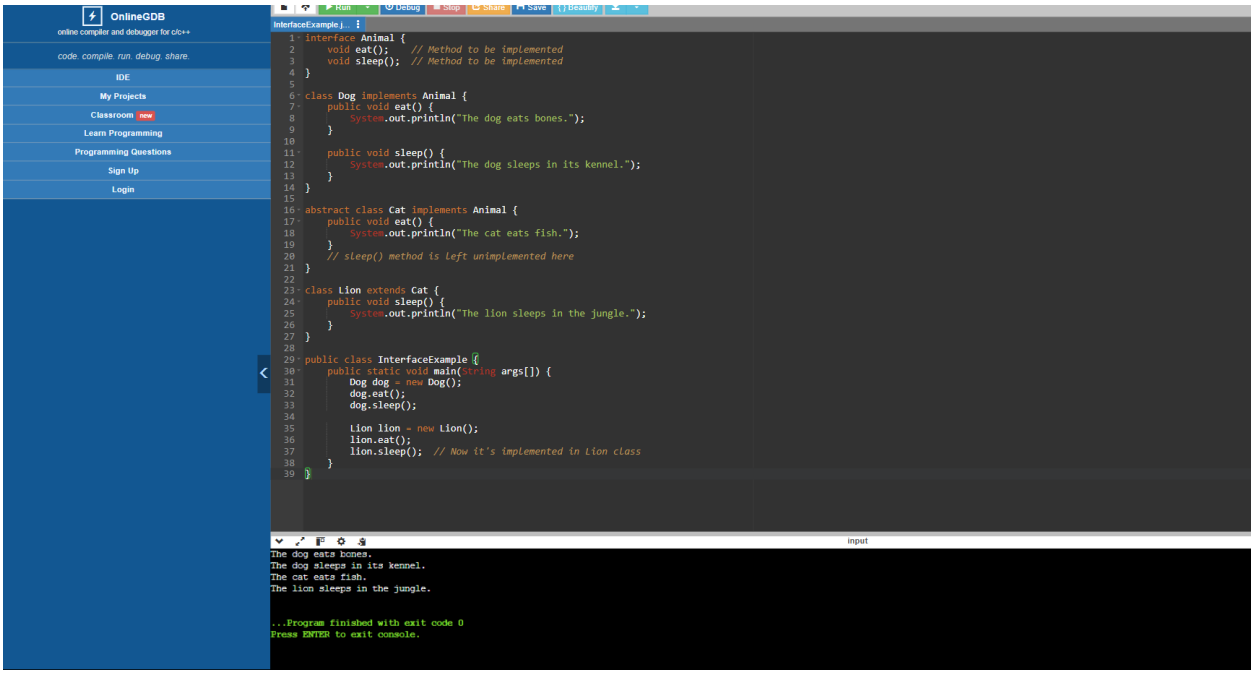
14. Write a program to demonstrate the use of function overloading.



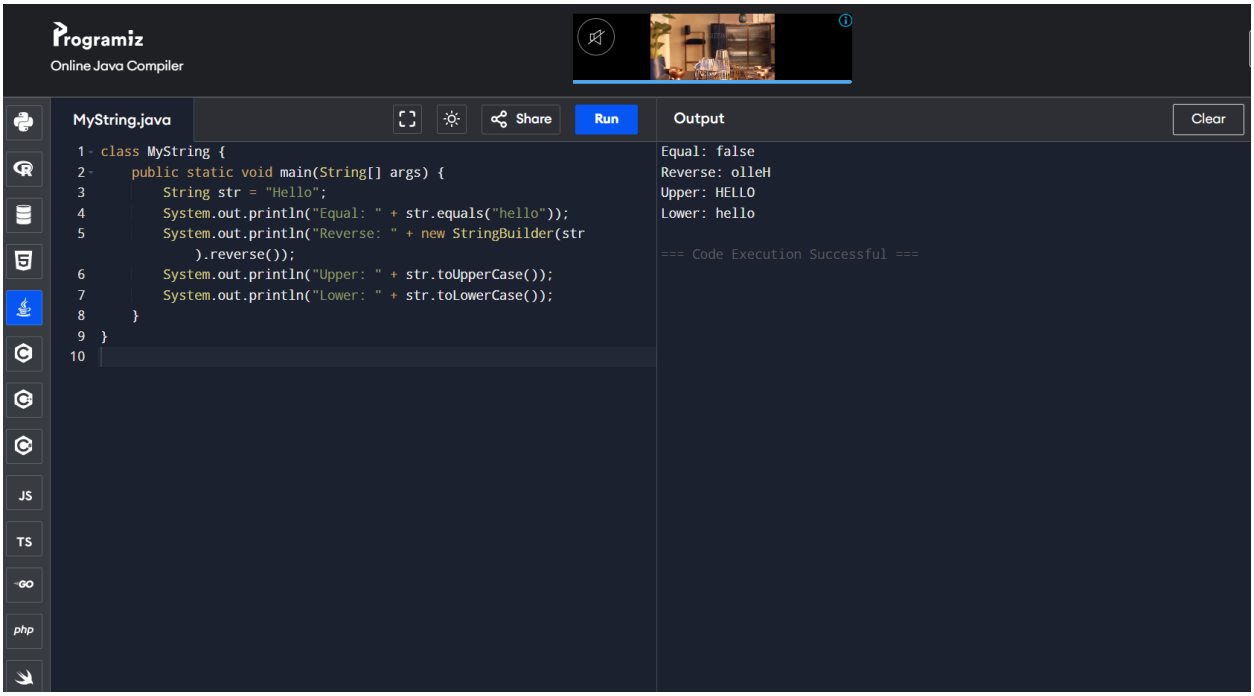
15. Write a program to demonstrate the use of inheritance



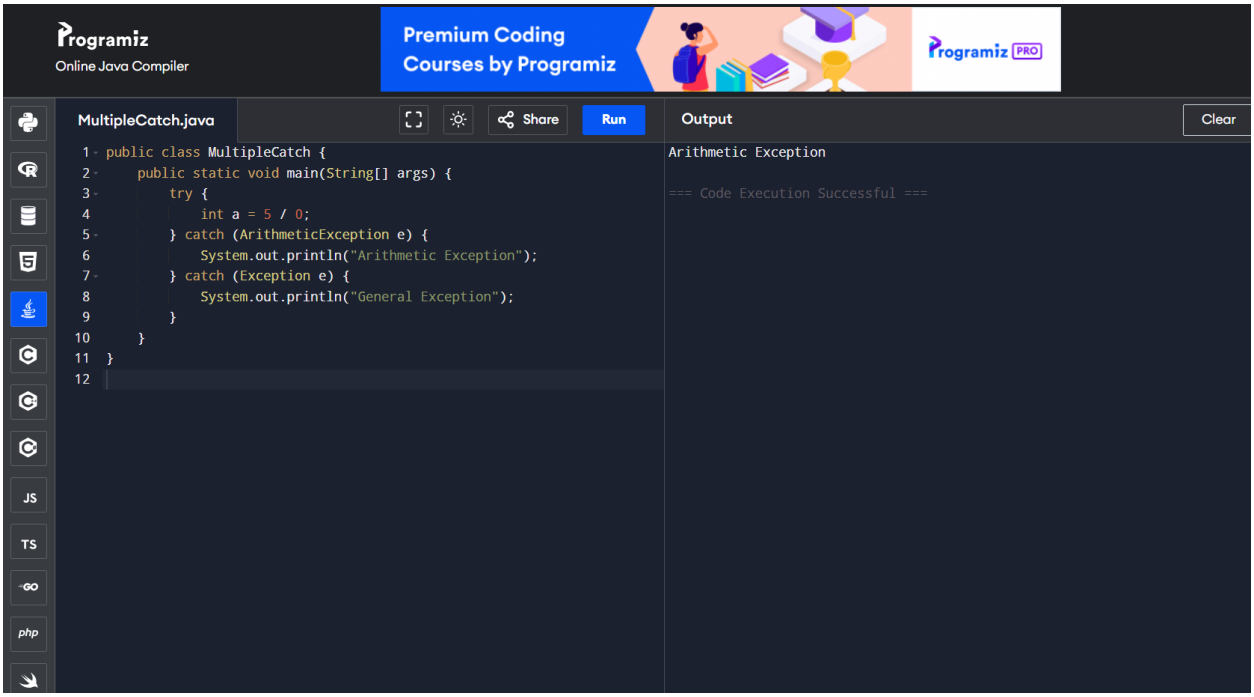
16. Write a program that show the partial implementation of Interface



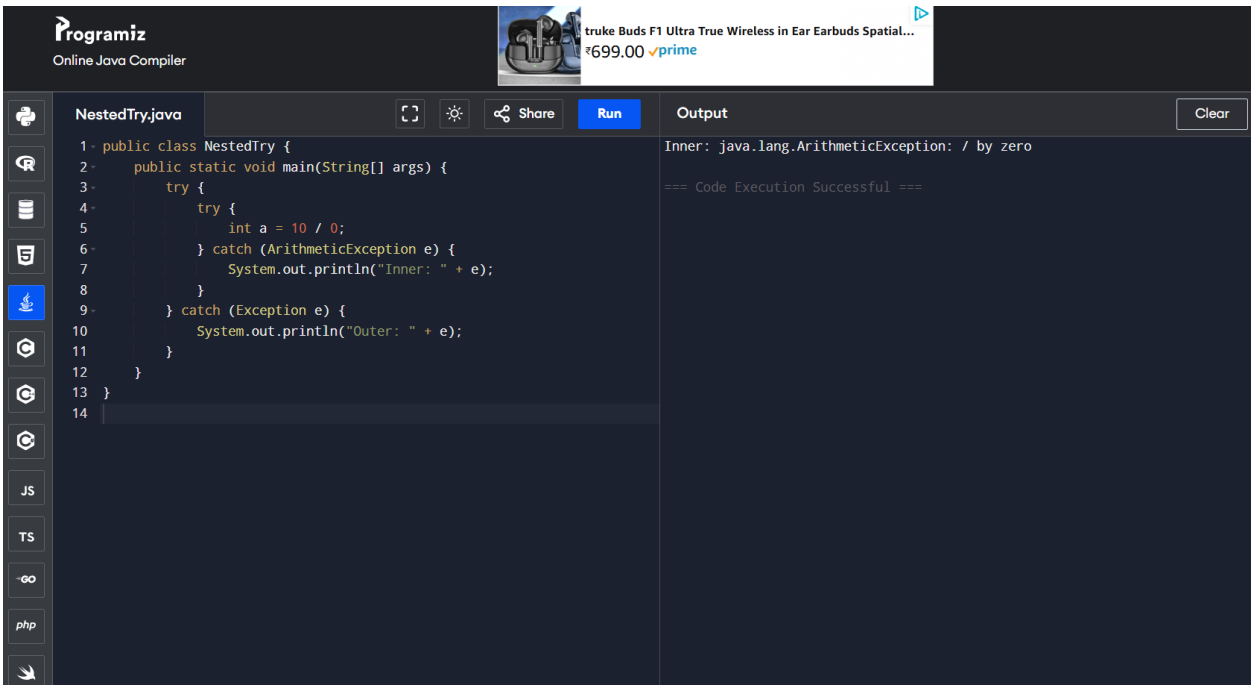
17. Write a program to design a string class that perform string method (Equal, Reverse the string, change case).



18. Write a program to handle the exception using try and multiple catch block.



19. Write a program that implement the Nested Try Statements.



20. Write a program to create a package that access the member of External class as well as

Programiz
Online Java Compiler

MyFirstJavaProgram.java

Share Run

```java
public class MyFirstJavaProgram {


    public static void main(String[] args) {

        System.out.println("Hello MY NAME IS ANSH SHARMA");
        System.out.println("This is my first Java program.");


        int myNumber = 10;


        System.out.println("My Number is: " + myNumber);


        int result = myNumber * 2;


        System.out.println("Result of multiplication: " + result);
    }
}
```

Output Clear

```
Hello MY NAME IS ANSH SHARMA
This is my first Java program.
My Number is: 10
Result of multiplication: 20

=== Code Execution Successful ===
```

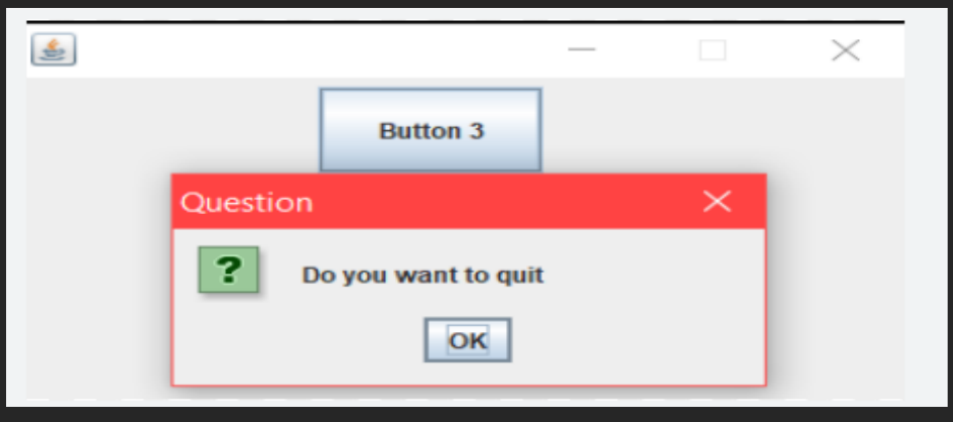
21. Write a program that import the user define package and access the Member variable of classes that contained by package

```
import javax.swing.; import java.awt.event.;

public class DialogBoxExample { public static void main(String[] args) { // Create frame
JFrame frame = new JFrame("Dialog Demo"); frame.setSize(300, 200);
frame.setLayout(null);

    // Create button
     JButton button = new JButton("Button 3");
     button.setBounds(100, 60, 100, 30);
     // Add action listener
     button.addActionListener(new ActionListener() {
         public void actionPerformed(ActionEvent e) {
             JOptionPane.showMessageDialog(
                 frame,
                 "Do you want to quit",
                 "Question",
                 JOptionPane.QUESTION_MESSAGE
             );
         }
     });
     // Add button and frame settings
     frame.add(button);
     frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
     frame.setVisible(true);
}}
```

Result/Output:



22. Write a program to handle the user define exception using throw keyworld.

Programiz
Online Java Compiler

Main.java

```java
class MyException extends Exception {
    public MyException(String message) {
        super(message);
    }
}

public class Main {

    public static void checkAge(int age) throws MyException {
        if (age < 18) {
            throw new MyException("Age is less than 18. Cannot vote.");
        } else {
            System.out.println("Age is valid. You can vote.");
        }
    }

    public static void main(String[] args) {

        try {
            checkAge(15);
            System.out.println("This line will not be printed if age < 18.");
        } catch (MyException e) {
            System.out.println("Caught an exception: " + e.getMessage());
```

Output

```
Caught an exception: Age is less than 18. Cannot vote.
Program continues after the catch block.
Age is valid. You can vote.
Program ends

=== Code Execution Successful ===
```

Programiz
Online Java Compiler

Main.java

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

        try {
            checkAge(15);
            System.out.println("This line will not be printed if age < 18.");
        } catch (MyException e) {
            System.out.println("Caught an exception: " + e.getMessage());
        }

        System.out.println("Program continues after the catch block.");

        try {
            checkAge(20);
        } catch (MyException e) {
            System.out.println("Caught an exception: " + e.getMessage());
        }

        System.out.println("Program ends");
    }
}
```

Output

```
Caught an exception: Age is less than 18. Cannot vote.
Program continues after the catch block.
Age is valid. You can vote.
Program ends

=== Code Execution Successful ===
```

23. Write a program to create a class component that show control and event handling on that control .

Experiment No: 23

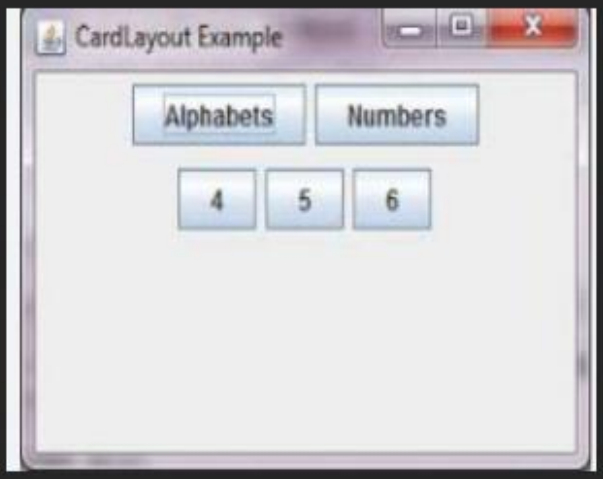
```
import java.awt.; import java.awt.event.; import javax.swing.*;

public class CombinedLayoutExample extends JFrame implements ActionListener
{ CardLayout cardLayout; JPanel cardPanel;

public CombinedLayoutExample() {
    setTitle("CardLayout + GridLayout Example");
    setSize(400, 300);
    setDefaultCloseOperation(EXIT_ON_CLOSE);
    setLayout(new BorderLayout());
    // Top buttons to switch cards
    JPanel buttonPanel = new JPanel();
    JButton btnAlphabet = new JButton("Alphabets");
    JButton btnNumbers = new JButton("Numbers");
    buttonPanel.add(btnAlphabet);
    buttonPanel.add(btnNumbers);
    // Card panel with CardLayout
    cardLayout = new CardLayout();
    cardPanel = new JPanel(cardLayout);
    // Alphabet card using GridLayout (3x1)
    JPanel alphabetPanel = new JPanel(new GridLayout(1, 3, 10,
10));
    alphabetPanel.add(new JButton("A"));
    alphabetPanel.add(new JButton("B"));
    alphabetPanel.add(new JButton("C"));
    // Number card using GridLayout (3x3)
    JPanel numberPanel = new JPanel(new GridLayout(3, 3, 10, 10));
    for (int i = 1; i <= 9; i++) {
        numberPanel.add(new JButton(String.valueOf(i)));
    }
    // Add both cards
    cardPanel.add(alphabetPanel, "Alphabets");
    cardPanel.add(numberPanel, "Numbers");
    // Add action listeners
    btnAlphabet.addActionListener(this);
    btnNumbers.addActionListener(this);
    // Add panels to frame
    add(buttonPanel, BorderLayout.NORTH);
    add(cardPanel, BorderLayout.CENTER);
    setVisible(true);
}

public void actionPerformed(ActionEvent e) {
    cardLayout.show(cardPanel, e.getActionCommand());
}
public static void main(String[] args) {
```

Result/Output:



Experiment No: 24

```
import javax.swing.; import javax.swing.table.DefaultTableModel; import java.awt.;
import java.awt.event.*;

public class StudentRecordForm extends JFrame implements ActionListener { // Form
components JTextField rollNoField, nameField, classField, sectionField; JTextArea
addressArea; JTable table; DefaultTableModel model; JButton insertBtn, viewBtn,
clearBtn, exitBtn;

public StudentRecordForm() {
    setTitle("School Record Management System");
    setSize(700, 500);
    setLayout(null);
    setDefaultCloseOperation(EXIT_ON_CLOSE);
    // Title label
    JLabel title = new JLabel("School Record Management System",
JLabel.CENTER);
    title.setFont(new Font("Serif", Font.BOLD | Font.ITALIC, 20));
    title.setBounds(150, 10, 400, 30);
    add(title);
    // Labels and text fields
    JLabel rollLabel = new JLabel("Roll No:");
    rollLabel.setBounds(50, 60, 100, 25);
    add(rollLabel);
    rollNoField = new JTextField();
    rollNoField.setBounds(150, 60, 100, 25);
    add(rollNoField);
    JLabel classLabel = new JLabel("Class:");
    classLabel.setBounds(400, 60, 50, 25);
    add(classLabel);
    classField = new JTextField();
    classField.setBounds(460, 60, 50, 25);
    add(classField);
    JLabel nameLabel = new JLabel("Name:");
    nameLabel.setBounds(50, 100, 100, 25);
    add(nameLabel);
    nameField = new JTextField();
    nameField.setBounds(150, 100, 200, 25);
    add(nameField);
    JLabel sectionLabel = new JLabel("Section:");
    sectionLabel.setBounds(400, 100, 60, 25);
    add(sectionLabel);
    sectionField = new JTextField();
    sectionField.setBounds(460, 100, 50, 25);
    add(sectionField);
    JLabel addressLabel = new JLabel("Address:");
    addressLabel.setBounds(50, 140, 100, 25);
```

```
    add(addressLabel);
    addressArea = new JTextArea();
    JScrollPane scrollPane = new JScrollPane(addressArea);
    scrollPane.setBounds(150, 140, 400, 60);
    add(scrollPane);
    // Table
    model = new DefaultTableModel(new String[]{"Roll No.", "Name",
"Class", "Section", "Address"}, 0);
    table = new JTable(model);
    JScrollPane tablePane = new JScrollPane(table);
    tablePane.setBounds(50, 220, 600, 100);
    add(tablePane);
    // Buttons
    insertBtn = new JButton("Insert");
    insertBtn.setBounds(50, 340, 100, 30);
    insertBtn.addActionListener(this);
    add(insertBtn);
    viewBtn = new JButton("View data");
    viewBtn.setBounds(180, 340, 100, 30);
    viewBtn.addActionListener(this);
    add(viewBtn);
    clearBtn = new JButton("Clear");
    clearBtn.setBounds(310, 340, 100, 30);
    clearBtn.addActionListener(this);
    add(clearBtn);
    exitBtn = new JButton("Exit");
    exitBtn.setBounds(440, 340, 100, 30);
    exitBtn.addActionListener(this);
    add(exitBtn);
    setVisible(true);
}
public void actionPerformed(ActionEvent e) {
    if (e.getSource() == insertBtn) {
        String roll = rollNoField.getText();
        String name = nameField.getText();
        String cls = classField.getText();
        String section = sectionField.getText();
        String address = addressArea.getText();
        if (!roll.isEmpty() && !name.isEmpty()) {
            model.addRow(new Object[]{roll, name, cls, section,
address});
            JOptionPane.showMessageDialog(this, "Record Added
Successfully", "Message", JOptionPane.INFORMATION_MESSAGE);
        } else {
            JOptionPane.showMessageDialog(this, "Please fill in
Roll No and Name", "Warning", JOptionPane.WARNING_MESSAGE);
        }
    } else if (e.getSource() == clearBtn) {
        rollNoField.setText("");
```

```
        nameField.setText("");
        classField.setText("");
        sectionField.setText("");
        addressArea.setText("");
    } else if (e.getSource() == exitBtn) {
        System.exit(0);
    } else if (e.getSource() == viewBtn) {
        // View logic already handled by JTable model
        JOptionPane.showMessageDialog(this, "Table updated with
current records.", "Info", JOptionPane.INFORMATION_MESSAGE);
    }
public static void main(String[] args) {
    new StudentRecordForm();
}}
```

Result/Output: