Slip1:

A) Write a ‘java’ program to display characters from ‘A’ to ‘Z’. [15 M]

```
public class Slip1a {
    public static void main(String[] args) {
        for (char ch = 'A'; ch <= 'Z'; ch++) {
            System.out.print(ch + " ");
        }
    }
}
```

B) Write a ‘java’ program to copy only non-numeric data from one file to another file. [25 M]

```
import java.io.*;

public class Slip1b {
    public static void main(String[] args) throws IOException {
        BufferedReader reader = new BufferedReader(new FileReader("input.txt"));
        BufferedWriter writer = new BufferedWriter(new FileWriter("output.txt"));
        int ch;
        while ((ch = reader.read()) != -1) {
            if (!Character.isDigit(ch)) {
                writer.write(ch);
            }
        }
        reader.close();
        writer.close();
    }
}
```

Slip2:

A) Write a java program to display all the vowels from a given string. [15 M]

---referNoteBook---

B) Design a screen in Java to handle the Mouse Events such as MOUSE_MOVED and MOUSE_CLICK and display the position of the Mouse_Click in a TextField. [25 M]

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

public class Slip2b extends JFrame implements MouseListener, MouseMotionListener {
    JTextField textField;

    public Slip2b() {
        textField = new JTextField();
        textField.setBounds(50, 50, 200, 30);
        add(textField);
        addMouseListener(this);
        addMouseMotionListener(this);
        setSize(300, 300);
        setLayout(null);
        setVisible(true);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public void mouseMoved(MouseEvent e) {
        textField.setText("Mouse Moved: " + e.getX() + ", " + e.getY());
    }

    public void mouseClicked(MouseEvent e) {
        textField.setText("Mouse Clicked at: " + e.getX() + ", " + e.getY());
    }

    public void mousePressed(MouseEvent e) {}
    public void mouseReleased(MouseEvent e) {}
    public void mouseEntered(MouseEvent e) {}
    public void mouseExited(MouseEvent e) {}
    public void mouseDragged(MouseEvent e) {}

    public static void main(String[] args) {
        new Slip2b();
    }
}
```

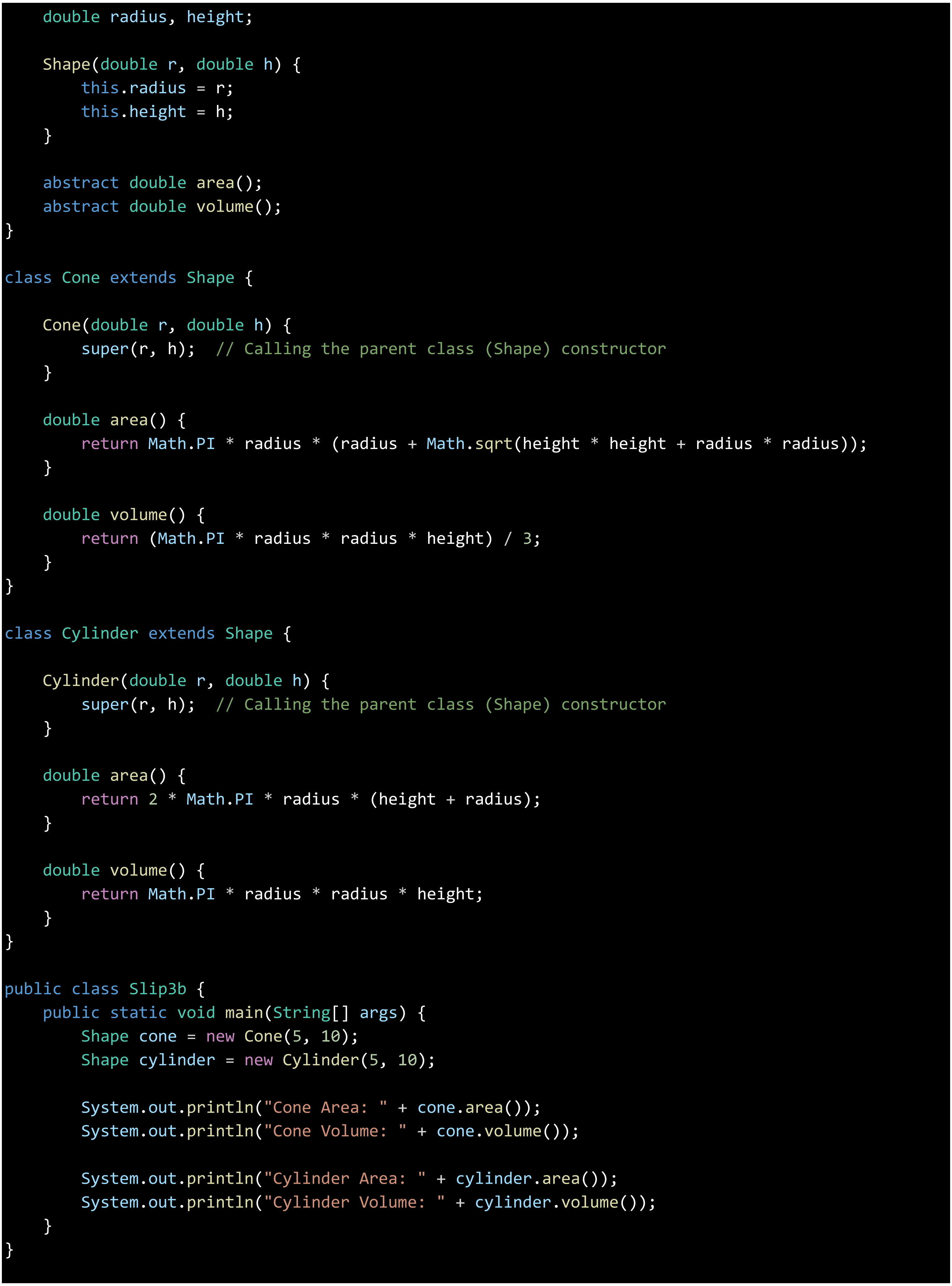
Slip3:

A) Write a ‘java’ program to check whether given number is Armstrong or not. (Use static keyword) [15 M]

```
public class Slip3a{
    public static void main(String[] args){
        int num = 370;
        int temp = num;
        int rem, sum = 0;

        while(num > 0){
            rem = num % 10;
            num = num / 10;
            sum = sum + rem*rem*rem;
        }

        if(temp == sum)
            System.out.println("it is an armstrong number");
        else
            System.out.println("it is not an armstrong number");
    }
}
```

B) Define an abstract class Shape with abstract methods area () and volume (). Derive abstract class Shape into two classes Cone and Cylinder. Write a java Program to calculate area and volume of Cone and Cylinder.(Use Super Keyword.) [25 M]

```
abstract class Shape {
```



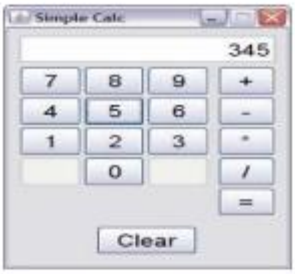
```java
    double radius, height;

    Shape(double r, double h) {
        this.radius = r;
        this.height = h;
    }

    abstract double area();
    abstract double volume();
}

class Cone extends Shape {

    Cone(double r, double h) {
        super(r, h);  // Calling the parent class (Shape) constructor
    }

    double area() {
        return Math.PI * radius * (radius + Math.sqrt(height * height + radius * radius));
    }

    double volume() {
        return (Math.PI * radius * radius * height) / 3;
    }
}

class Cylinder extends Shape {

    Cylinder(double r, double h) {
        super(r, h);  // Calling the parent class (Shape) constructor
    }

    double area() {
        return 2 * Math.PI * radius * (height + radius);
    }

    double volume() {
        return Math.PI * radius * radius * height;
    }
}

public class Slip3b {
    public static void main(String[] args) {
        Shape cone = new Cone(5, 10);
        Shape cylinder = new Cylinder(5, 10);

        System.out.println("Cone Area: " + cone.area());
        System.out.println("Cone Volume: " + cone.volume());

        System.out.println("Cylinder Area: " + cylinder.area());
        System.out.println("Cylinder Volume: " + cylinder.volume());
    }
}
```

Slip4:

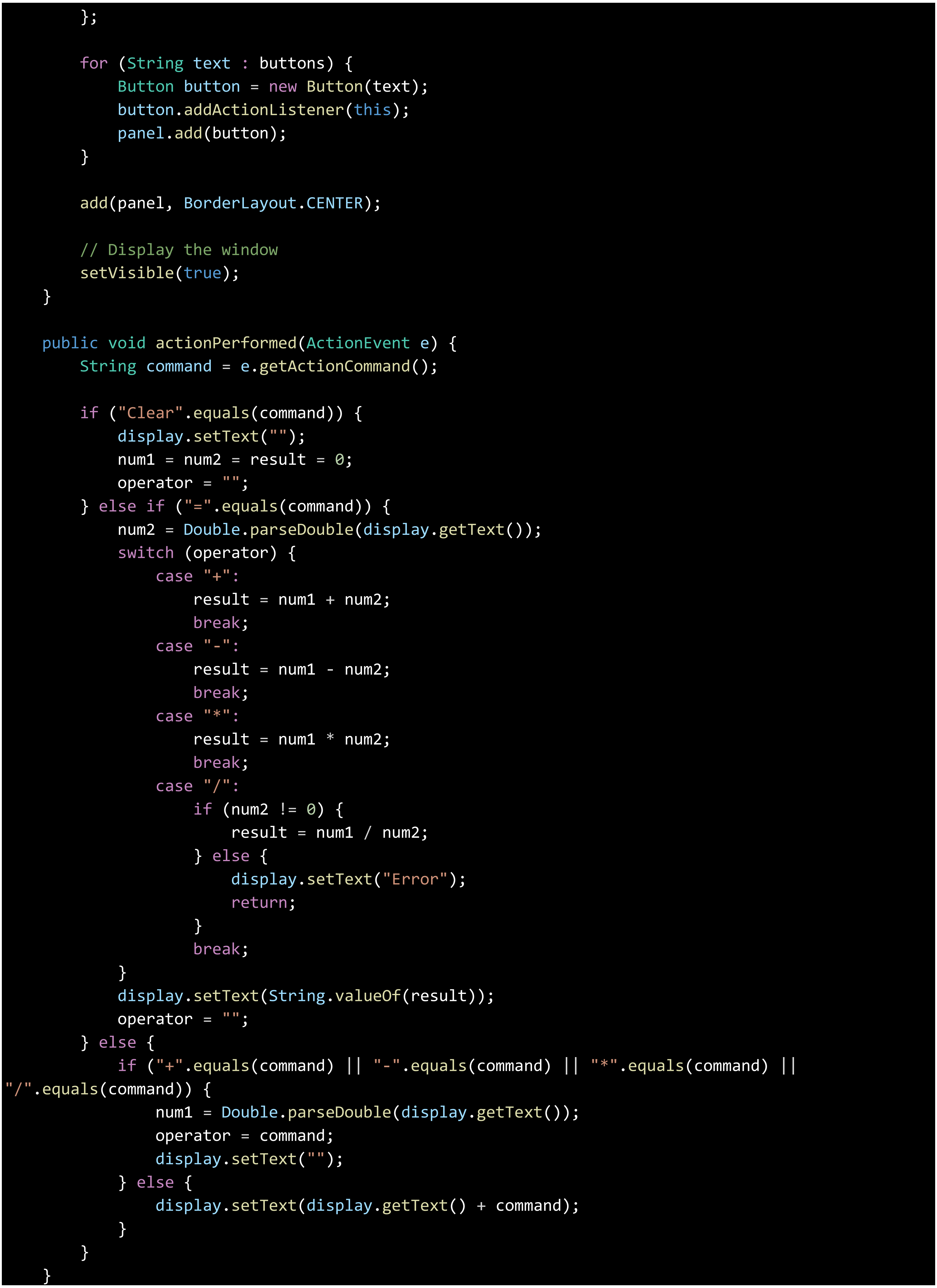
A) Write a java program to display alternate character from a given string. [15 M]

```
public class Slip4a{
    public static void main(String[] args){
        String str = "hello world";
        for(int i = 0; i < str.length(); i += 2){
            System.out.println(str.charAt(i));
        }
    }
}
```

B) Write a java program using Applet to implement a simple arithmetic calculator. [25 M]



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

public class Slip4b extends JFrame implements ActionListener {
    private TextField display;
    private String operator;
    private double num1, num2, result;

    public Slip4b() {
        // Set up the frame
        setTitle("Simple Calc");
        setSize(300, 400);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new BorderLayout());

        // Create display field
        display = new TextField();
        display.setEditable(false);
        add(display, BorderLayout.NORTH);

        // Create panel for buttons
        Panel panel = new Panel();
        panel.setLayout(new GridLayout(4, 4, 10, 10)); // 4x4 grid with gaps

        // Create buttons
        String[] buttons = {
            "7", "8", "9", "+",
            "4", "5", "6", "-",
            "1", "2", "3", "/",
            "0", "Clear", "=", ""
```



```
    };

    for (String text : buttons) {
      Button button = new Button(text);
      button.addActionListener(this);
      panel.add(button);
    }

    add(panel, BorderLayout.CENTER);

    // Display the window
    setVisible(true);
  }

  public void actionPerformed(ActionEvent e) {
    String command = e.getActionCommand();

    if ("Clear".equals(command)) {
      display.setText("");
      num1 = num2 = result = 0;
      operator = "";
    } else if ("=".equals(command)) {
      num2 = Double.parseDouble(display.getText());
      switch (operator) {
        case "+":
          result = num1 + num2;
          break;
        case "-":
          result = num1 - num2;
          break;
        case "*":
          result = num1 * num2;
          break;
        case "/":
          if (num2 != 0) {
            result = num1 / num2;
          } else {
            display.setText("Error");
            return;
          }
          break;
      }
      display.setText(String.valueOf(result));
      operator = "";
    } else {
      if ("+".equals(command) || "-".equals(command) || "*".equals(command) ||
"/".equals(command)) {
        num1 = Double.parseDouble(display.getText());
        operator = command;
        display.setText("");
      } else {
        display.setText(display.getText() + command);
      }
    }
  }
```

```
    public static void main(String[] args) {
        new Slip4b(); // Create an instance of the calculator
    }
}
```

Slip5:

A) Write a java program to display following pattern: [15 M]

5

4 5

3 4 5

2 3 4 5

1 2 3 4 5

---referNoteBook---

B) Write a java program to accept list of file names through command line. Delete the files having extension .txt. Display name, location and size of remaining files. [25 M]

```
import java.io.*;

class Slip5b{
    public static void main(String[] args){
        for(int i=0; i<args.length; i++){
            File file = new File(args[i]);
            if(file.isFile()){
                String name = file.getName();
                if(name.endsWith(".txt")){
                    file.delete();
                    System.out.println("file is deleted successfully!" +file);
                }
                else{
                    System.out.println("file name: " +name + "\nfile location: "
+file.getAbsolutePath() + "\nfile size: " +file.length() +"bytes");
                }
            }
            else{
                System.out.println(args[i] + "is not a file");
            }
        }
    }
}
```

Slip6:

A) Write a java program to accept a number from user, if it zero then throw user defined Exception “Number Is Zero”, otherwise calculate the sum of first and last digit of that number. (Use static keyword). [15 M]

```java
import java.util.Scanner;

class NumberIsZeroException extends Exception {
    public NumberIsZeroException(String message) {
        super(message);
    }
}

public class Slip6a {
    static int number;

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

        try {
            System.out.print("Enter a number: ");
            number = scanner.nextInt();

            // Check if the number is zero
            if (number == 0) {
                throw new NumberIsZeroException("Number is zero.");
            } else {
                // Calculate and print the sum of the first and last digits
                int sum = calculateSumOfFirstAndLastDigit(number);
                System.out.println("The sum of the first and last digits is: " + sum);
            }
        } catch (NumberIsZeroException e) {
            System.out.println(e.getMessage());
        } finally {
            scanner.close();
        }
    }

    // Static method to calculate the sum of the first and last digits
    private static int calculateSumOfFirstAndLastDigit(int num) {
        int lastDigit = Math.abs(num % 10); // Last digit (absolute value for negative numbers)
        int firstDigit = Character.getNumericValue(Integer.toString(num).charAt(0)); // First
digit

        return firstDigit + lastDigit;
    }
}
```

B) Write a java program to display transpose of a given matrix. [25 M]

```java
import java.util.Scanner;

public class Slip6b {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter number of rows and columns:");
        int rows = sc.nextInt();
        int cols = sc.nextInt();
        int[][] matrix = new int[rows][cols];
        System.out.println("Enter matrix elements:");
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                matrix[i][j] = sc.nextInt();
            }
        }
        System.out.println("Transpose:");
        for (int i = 0; i < cols; i++) {
            for (int j = 0; j < rows; j++) {
                System.out.print(matrix[j][i] + " ");
            }
            System.out.println();
        }
        sc.close();
    }
}
```

Slip7:

A) Write a java program to display Label with text “Dr. D Y Patil College”, background color Red and font size 20 on the frame. [15 M]

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

public class Slip7a extends JFrame {
    public Slip7a() {
        JLabel label = new JLabel("Dr. D Y Patil College", JLabel.CENTER);
        label.setFont(new Font("Serif", Font.PLAIN, 20));
        label.setOpaque(true);
        label.setBackground(Color.RED);
        add(label);
        setSize(400, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }

    public static void main(String[] args) {
        new Slip7a();
    }
}
```

B) Write a java program to accept details of ‘n’ cricket player (pid, pname, totalRuns, InningsPlayed, NotOuttimes). Calculate the average of all the players. Display the details of player having maximum average. (Use Array of Object) [25 M]

```
import java.util.Scanner;

class Player {
    int pid;
    String name;
    int totalRuns;
    int inningsPlayed;
    int notOutTimes;

    Player(int pid, String name, int totalRuns, int inningsPlayed, int notOutTimes) {
        this.pid = pid;
        this.name = name;
        this.totalRuns = totalRuns;
        this.inningsPlayed = inningsPlayed;
        this.notOutTimes = notOutTimes;
    }

    float calculateAverage() {
        return (float) totalRuns / (inningsPlayed - notOutTimes);
    }
}

public class Slip7b {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter number of players: ");
        int n = sc.nextInt();
        Player[] players = new Player[n];
        float maxAverage = 0;
        Player maxAvgPlayer = null;

        for (int i = 0; i < n; i++) {
            System.out.print("Enter pid, name, totalRuns, inningsPlayed, notOutTimes: ");
            int pid = sc.nextInt();
            String name = sc.next();
            int totalRuns = sc.nextInt();
            int inningsPlayed = sc.nextInt();
            int notOutTimes = sc.nextInt();

            players[i] = new Player(pid, name, totalRuns, inningsPlayed, notOutTimes);
            float avg = players[i].calculateAverage();
            if (avg > maxAverage) {
                maxAverage = avg;
                maxAvgPlayer = players[i];
            }
        }
        System.out.println("Player with max average: " + maxAvgPlayer.name);
        sc.close();
    }
}
```

Slip8:

A) Define an Interface Shape with abstract method area(). Write a java program to calculate an area of Circle and Sphere.(use final keyword) [15 M]

```
// Define the Shape interface
interface Shape {
    double area();  // Abstract method to calculate area
}

// Circle class implementing Shape interface
```

```
class Circle implements Shape {
    private final double radius;  // Final variable for radius

    public Circle(double radius) {
        this.radius = radius;  // Initialize radius
    }

    @Override
    public double area() {
        return Math.PI * radius * radius;  // Area of the circle
    }
}

// Sphere class implementing Shape interface
class Sphere implements Shape {
    private final double radius;  // Final variable for radius

    public Sphere(double radius) {
        this.radius = radius;  // Initialize radius
    }

    @Override
    public double area() {
        return 4 * Math.PI * radius * radius;  // Surface area of the sphere
    }
}

// Main class to test the program
public class Slip8a {
    public static void main(String[] args) {
        Shape circle = new Circle(5);  // Create Circle object with radius 5
        Shape sphere = new Sphere(3);   // Create Sphere object with radius 3

        // Calculate and display areas
        System.out.println("Area of Circle: " + circle.area());  // Output area of circle
        System.out.println("Surface Area of Sphere: " + sphere.area());  // Output surface area of
sphere
    }
}
```

B) Write a java program to display the files having extension .txt from a given directory. [25 M]

```
import java.io.*;

class Slip8b{
    public static void main(String[] args){
        File file = new File("C:\\Users\\DELL\\Documents\\java");
        File [] fl = file.listFiles((d, f)-> f.toLowerCase().endsWith(".txt"));

        for(File f: fl){
            System.out.println(f.getName());
        }
    }
}
```

Slip9:

A) Write a java Program to display following pattern: [15 M]

1

0 1

0 1 0

1 0 1 0

```
public class Slip9a {
    public static void main(String[] args) {

        for (int i = 0; i <4; i++) {
            for (int j = 0; j <= i; j++) {
                if((i+j) % 2 == 0){
                  System.out.print("0");
                }else{
                    System.out.print("1");
                }
            }
            System.out.println();
        }
    }
}
```

B) Write a java program to validate PAN number and Mobile Number. If it is invalid then throw user defined Exception “Invalid Data”, otherwise display it. [25 M]

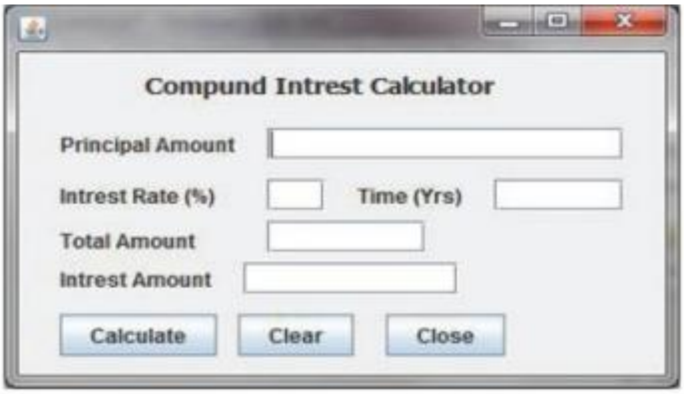
```
import java.util.Scanner;

// User-defined exception
class InvalidDataException extends Exception {
    public InvalidDataException(String message) {
        super(message);
    }
}

public class Slip9b{
    // Method to validate PAN number
    public static void validatePAN(String pan) throws InvalidDataException {
        if (!pan.matches("[A-Z]{5}[0-9]{4}[A-Z]")) {
            throw new InvalidDataException("Invalid PAN number");
        }
    }

    // Method to validate Mobile number
    public static void validateMobile(String mobile) throws InvalidDataException {
        if (!mobile.matches("[789][0-9]{9}")) {
            throw new InvalidDataException("Invalid mobile number");
        }
    }

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

        // Input PAN number
        System.out.print("Enter PAN number: ");
        String pan = scanner.nextLine();

        // Input Mobile number
        System.out.print("Enter Mobile number: ");
        String mobile = scanner.nextLine();

        try {
            // Validate PAN and Mobile number
            validatePAN(pan);
            validateMobile(mobile);
            System.out.println("Valid PAN number: " + pan);
            System.out.println("Valid Mobile number: " + mobile);
        } catch (InvalidDataException e) {
            System.out.println(e.getMessage());
        } finally {
            scanner.close();
        }
    }
}
```

Slip10:

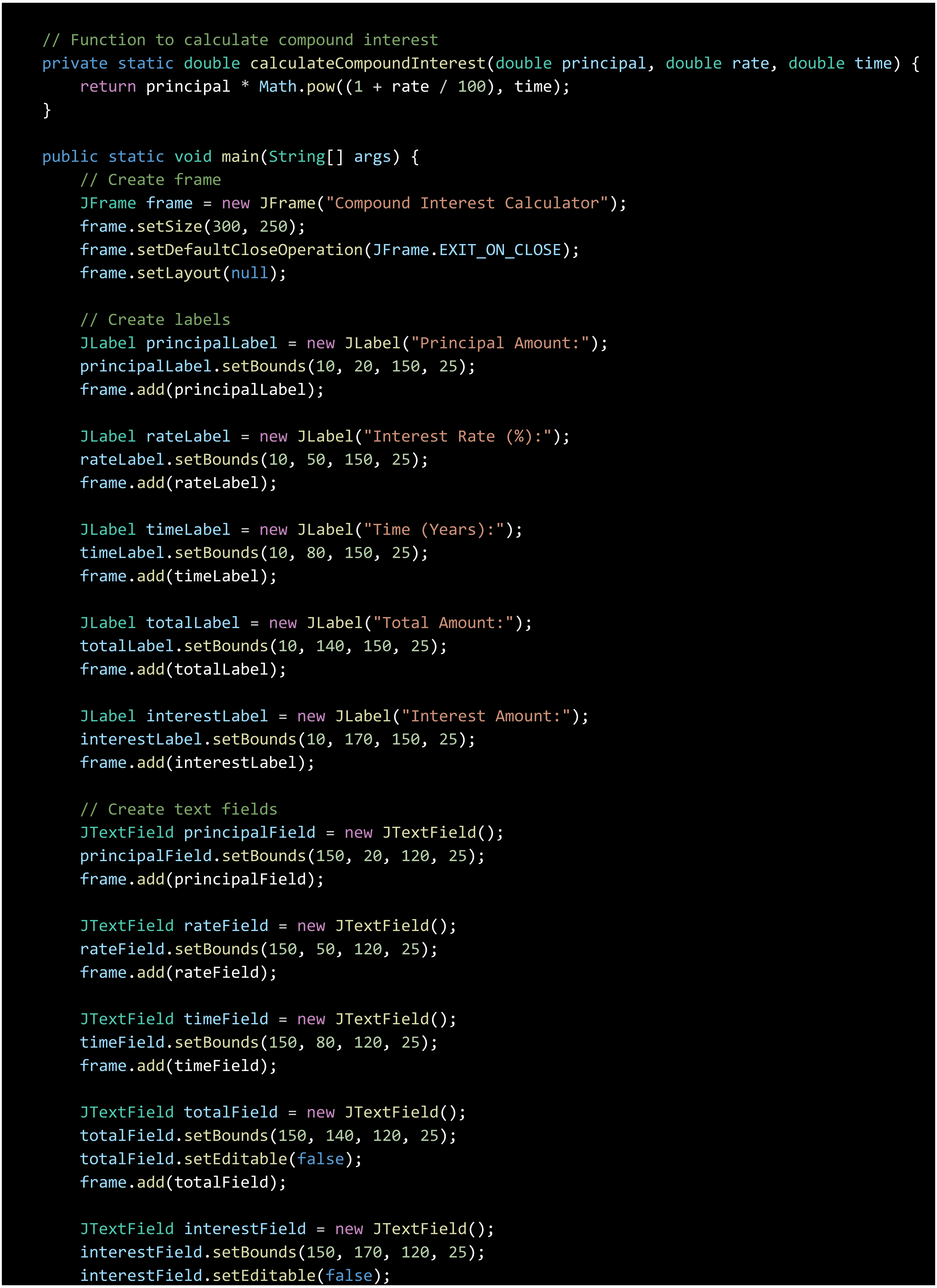
A) Write a java program to count the frequency of each character in a given string. [15 M]

---referNoteBook---

B) Write a java program for the following: [25 M]



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

public class Slip10b {
```



```
    // Function to calculate compound interest
    private static double calculateCompoundInterest(double principal, double rate, double time) {
        return principal * Math.pow((1 + rate / 100), time);
    }

    public static void main(String[] args) {
        // Create frame
        JFrame frame = new JFrame("Compound Interest Calculator");
        frame.setSize(300, 250);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setLayout(null);

        // Create labels
        JLabel principalLabel = new JLabel("Principal Amount:");
        principalLabel.setBounds(10, 20, 150, 25);
        frame.add(principalLabel);

        JLabel rateLabel = new JLabel("Interest Rate (%):");
        rateLabel.setBounds(10, 50, 150, 25);
        frame.add(rateLabel);

        JLabel timeLabel = new JLabel("Time (Years):");
        timeLabel.setBounds(10, 80, 150, 25);
        frame.add(timeLabel);

        JLabel totalLabel = new JLabel("Total Amount:");
        totalLabel.setBounds(10, 140, 150, 25);
        frame.add(totalLabel);

        JLabel interestLabel = new JLabel("Interest Amount:");
        interestLabel.setBounds(10, 170, 150, 25);
        frame.add(interestLabel);

        // Create text fields
        JTextField principalField = new JTextField();
        principalField.setBounds(150, 20, 120, 25);
        frame.add(principalField);

        JTextField rateField = new JTextField();
        rateField.setBounds(150, 50, 120, 25);
        frame.add(rateField);

        JTextField timeField = new JTextField();
        timeField.setBounds(150, 80, 120, 25);
        frame.add(timeField);

        JTextField totalField = new JTextField();
        totalField.setBounds(150, 140, 120, 25);
        totalField.setEditable(false);
        frame.add(totalField);

        JTextField interestField = new JTextField();
        interestField.setBounds(150, 170, 120, 25);
        interestField.setEditable(false);
        frame.add(interestField);

        // Create buttons
        JButton calculateButton = new JButton("Calculate");
        calculateButton.setBounds(30, 110, 100, 25);
        frame.add(calculateButton);

        JButton clearButton = new JButton("Clear");
        clearButton.setBounds(140, 110, 100, 25);
        frame.add(clearButton);

        JButton closeButton = new JButton("Close");
        closeButton.setBounds(90, 200, 100, 25);
        frame.add(closeButton);

        // Action listener for calculate button
        calculateButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                double principal = Double.parseDouble(principalField.getText());
                double rate = Double.parseDouble(rateField.getText());
                double time = Double.parseDouble(timeField.getText());

                double totalAmount = calculateCompoundInterest(principal, rate, time);
                double interestAmount = totalAmount - principal;

                totalField.setText(String.format("%.2f", totalAmount));
                interestField.setText(String.format("%.2f", interestAmount));
            }
        });

        // Action listener for clear button
        clearButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                principalField.setText("");
                rateField.setText("");
                timeField.setText("");
                totalField.setText("");
                interestField.setText("");
            }
        });

        // Action listener for close button
        closeButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                System.exit(0);
            }
        });

        // Set frame visibility
        frame.setVisible(true);
    }
}
```

Slip11:

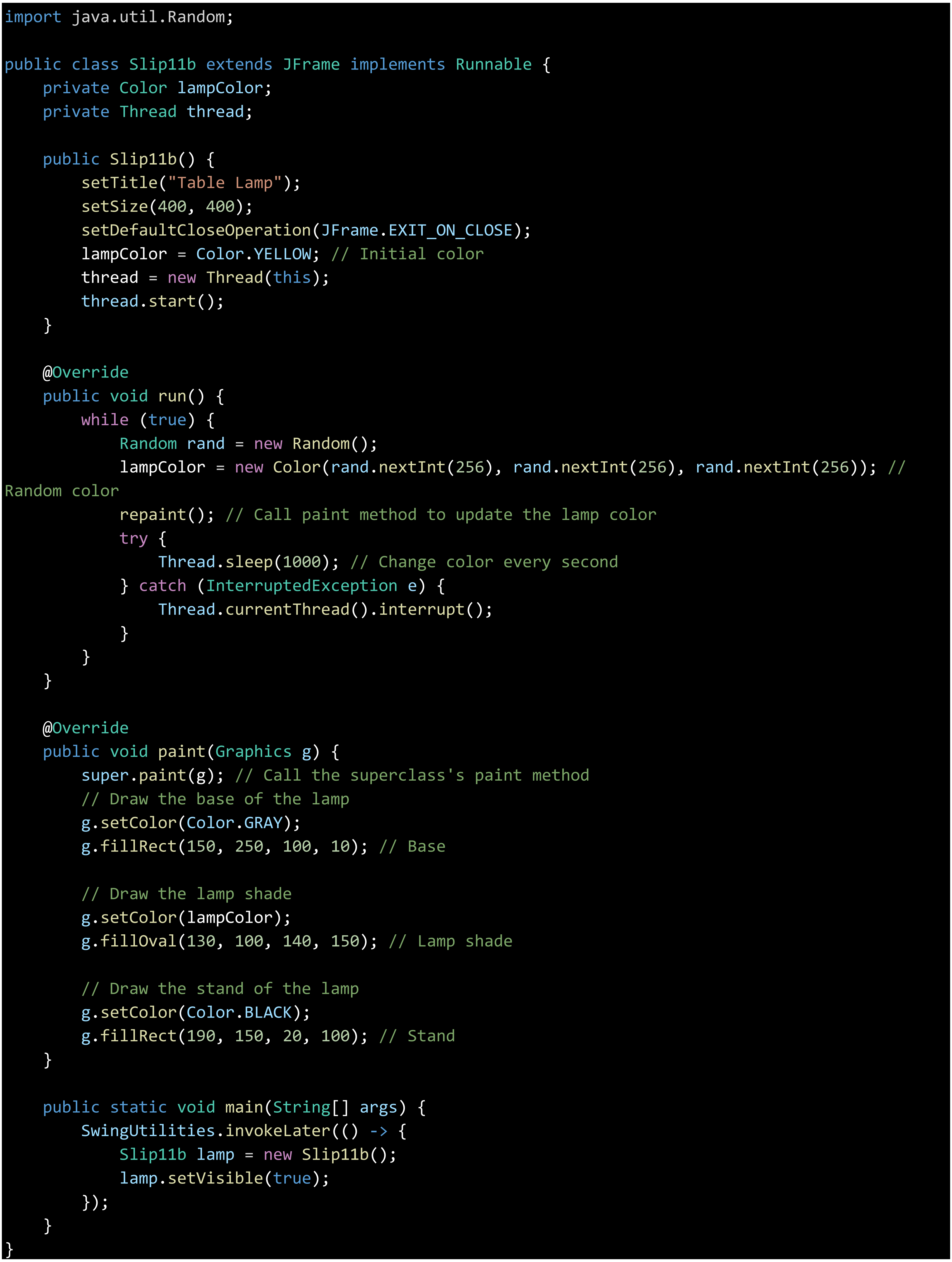
A) Write a menu driven java program using command line arguments for the following: [15 M]

1. Addition 2. Subtraction 3. Multiplication 4. Division.

```
public class Slip11a {
    public static void main(String[] args) {
        if (args.length != 3) {
            System.out.println("Usage: java SimpleCalculator <operation> <num1> <num2>");
            System.out.println("Operation: add, sub, mul, div");
            return;
        }

        String operation = args[0];
        double num1 = Double.parseDouble(args[1]);
        double num2 = Double.parseDouble(args[2]);
        double result;

        switch (operation) {
            case "add":
                result = num1 + num2;
                System.out.println("Result: " + result);
                break;
            case "sub":
                result = num1 - num2;
                System.out.println("Result: " + result);
                break;
            case "mul":
                result = num1 * num2;
                System.out.println("Result: " + result);
                break;
            case "div":
                if (num2 != 0) {
                    result = num1 / num2;
                    System.out.println("Result: " + result);
                } else {
                    System.out.println("Error: Division by zero is not allowed.");
                }
                break;
            default:
                System.out.println("Invalid operation. Use add, sub, mul, or div.");
                break;
        }
    }
}
```

B) Write an applet application to display Table lamp. The color of lamp should get change randomly. [25 M]

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



```
import java.util.Random;

public class Slip11b extends JFrame implements Runnable {
    private Color lampColor;
    private Thread thread;

    public Slip11b() {
        setTitle("Table Lamp");
        setSize(400, 400);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        lampColor = Color.YELLOW; // Initial color
        thread = new Thread(this);
        thread.start();
    }

    @Override
    public void run() {
        while (true) {
            Random rand = new Random();
            lampColor = new Color(rand.nextInt(256), rand.nextInt(256), rand.nextInt(256)); //
Random color
            repaint(); // Call paint method to update the lamp color
            try {
                Thread.sleep(1000); // Change color every second
            } catch (InterruptedException e) {
                Thread.currentThread().interrupt();
            }
        }
    }

    @Override
    public void paint(Graphics g) {
        super.paint(g); // Call the superclass's paint method
        // Draw the base of the lamp
        g.setColor(Color.GRAY);
        g.fillRect(150, 250, 100, 10); // Base

        // Draw the lamp shade
        g.setColor(lampColor);
        g.fillOval(130, 100, 140, 150); // Lamp shade

        // Draw the stand of the lamp
        g.setColor(Color.BLACK);
        g.fillRect(190, 150, 20, 100); // Stand
    }

    public static void main(String[] args) {
        SwingUtilities.invokeLater(() -> {
            Slip11b lamp = new Slip11b();
            lamp.setVisible(true);
        });
    }
}
```

```
<html>
<body>
    <applet code="Slip11b.class" width="300" height="400">
    </applet>
</body>
</html>
```

Slip12:

A) Write a java program to display each String in reverse order from a String array. [15 M]

```
public class Slip12a {
    public static void main(String[] args) {
        String[] strings = {"Hello", "World", "Java", "Programming"};

        System.out.println("Strings in reverse order:");
        for (String str : strings) {
            System.out.println(reverseString(str));
        }
    }

    // Method to reverse a string
    private static String reverseString(String str) {
        return new StringBuilder(str).reverse().toString();
    }
}
```

B) Write a java program to display multiplication table of a given number into the List box by clicking on button.[25 M]

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

public class Slip12b extends JFrame {
    private JTextField numberField;
    private JList<String> tableList;
    private DefaultListModel<String> listModel;

    public Slip12b() {
        setTitle("Multiplication Table");
        setSize(300, 250);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLayout(new FlowLayout());

        numberField = new JTextField(10);
        JButton button = new JButton("Show Table");
        listModel = new DefaultListModel<>();
        tableList = new JList<>(listModel);

        button.addActionListener((ActionEvent e) -> {
            listModel.clear();
            try {
                int number = Integer.parseInt(numberField.getText());
                for (int i = 1; i <= 10; i++) {
                    listModel.addElement(number + " x " + i + " = " + (number * i));
                }
            } catch (NumberFormatException ex) {
                JOptionPane.showMessageDialog(this, "Enter a valid number.");
            }
        });

        add(new JLabel("Enter a number:"));
        add(numberField);
        add(button);
        add(new JScrollPane(tableList));
    }

    public static void main(String[] args) {
        SwingUtilities.invokeLater(() -> {
            new Slip12b().setVisible(true);
        });
    }
}
```

Slip13:

A) Write a java program to accept ‘n’ integers from the user & store them in an ArrayList collection. Display the elements of ArrayList collection in reverse order. [15 M]

```
import java.util.*;

class Slip13a{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.println("enter no. of elements: ");

        int n = sc.nextInt();
        ArrayList<String> list = new ArrayList<>();
        System.out.println("enter the elements of array list collection: ");
        for(int i = 0; i<n; i++){
            String ele = sc.next();
            list.add(ele);
        }

        System.out.println("original array list: " +list);
        Collections.reverse(list);
        System.out.println("reversed array list: " +list);

    }
}
```

B) Write a java program that asks the user name, and then greets the user by name. Before outputting the user's name, convert it to upper case letters. For example, if the user's name is Raj, then the program should respond "Hello, RAJ, nice to meet you!". [25 M]

```
import java.util.Scanner;
```

```
public class Slip13b {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Ask for the user's name
        System.out.print("Enter your name: ");
        String name = scanner.nextLine();

        // Convert the name to upper case
        String upperCaseName = name.toUpperCase();

        // Greet the user
        System.out.println("Hello, " + upperCaseName + ", nice to meet you!");

        // Close the scanner
        scanner.close();
    }
}
```

Slip14:

A) Write a Java program to calculate power of a number using recursion. [15 M]

```
import java.util.Scanner;

public class Slip14a {
    static int power(int base, int exp) {
        if (exp == 0) return 1;
        return base * power(base, exp - 1);
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter base: ");
        int base = sc.nextInt();
        System.out.print("Enter exponent: ");
        int exp = sc.nextInt();
        System.out.println(base + " raised to " + exp + " is: " + power(base, exp));
        sc.close();
    }
}
```

B) Write a java program to accept the details of employee (Eno, EName, Sal) and display it on next frame using appropriate event . [25 M]

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

public class Slip14b extends JFrame implements ActionListener {
    JTextField enoField, enameField, salField;
    JButton submitButton;

    public Slip14b() {
        setLayout(new FlowLayout());
        add(new JLabel("Employee No:"));
        enoField = new JTextField(10);
        add(enoField);

        add(new JLabel("Employee Name:"));
        enameField = new JTextField(10);
        add(enameField);

        add(new JLabel("Salary:"));
        salField = new JTextField(10);
        add(salField);

        submitButton = new JButton("Submit");
        add(submitButton);
        submitButton.addActionListener(this);

        setSize(300, 150);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        JFrame frame = new JFrame("Employee Details");
        JTextArea details = new JTextArea();
        details.setText("Employee No: " + enoField.getText() + "\n" +
                        "Employee Name: " + enameField.getText() + "\n" +
                        "Salary: " + salField.getText());
        frame.add(details);
        frame.setSize(300, 200);
        frame.setVisible(true);
    }

    public static void main(String[] args) {
        new Slip14b();
    }
}
```

Slip15:

A) Write a java program to search given name into the array, if it is found then display its index otherwise display appropriate message. [15 M]

```
import java.util.Scanner;

public class Slip15a {
    public static void main(String[] args) {
        String[] names = {"Alice", "Bob", "Charlie", "David", "Eve"};
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter a name to search: ");
        String nameToSearch = scanner.nextLine();

        int index = -1; // Initialize index to -1 (not found)
        for (int i = 0; i < names.length; i++) {
            if (names[i].equalsIgnoreCase(nameToSearch)) {
                index = i; // Store the index if found
                break;
            }
        }

        // Display result
        if (index != -1) {
            System.out.println("Name found at index: " + index);
        } else {
            System.out.println("Name not found.");
        }

        scanner.close();
    }
}
```

B) Write an applet application to display smiley face. [25 M]

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

public class Slip15b extends JFrame {
    public Slip15b() {
        setTitle("Smile Face");
        setSize(300, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }

    @Override
    public void paint(Graphics g) {
        super.paint(g); // Call the superclass's paint method

        // Face
        g.setColor(Color.YELLOW);
        g.fillOval(50, 50, 200, 200);

        // Eyes
        g.setColor(Color.BLACK);
        g.fillOval(90, 100, 30, 30); // Left eye
        g.fillOval(180, 100, 30, 30); // Right eye

        // Smile
        g.drawArc(110, 150, 80, 50, 0, -180);
    }

    public static void main(String[] args) {
        new Slip15b(); // Create an instance of the JFrame
    }
}
```

Slip16:

A) Write a java program to calculate sum of digits of a given number using recursion. [15 M]

```
import java.util.Scanner;

public class Slip16a {
    static int sumDigits(int num) {
        if (num == 0) return 0;
        return num % 10 + sumDigits(num / 10);
    }

    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("Sum of digits: " + sumDigits(num));
        sc.close();
    }
}
```

B) Write a java program to accept n employee names from user. Sort them in ascending order and Display them.(Use array of object and Static keyword) [25 M]

```
import java.util.Arrays;
import java.util.Scanner;

class Employee {
    String name;

    Employee(String name) {
        this.name = name;
    }
}

public class Slip16b {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the number of employees: ");
        int n = sc.nextInt();
        sc.nextLine(); // consume newline
        Employee[] employees = new Employee[n];

        for (int i = 0; i < n; i++) {
            System.out.print("Enter employee name: ");
            employees[i] = new Employee(sc.nextLine());
        }

        Arrays.sort(employees, (e1, e2) -> e1.name.compareTo(e2.name));

        System.out.println("Sorted employee names:");
        for (Employee emp : employees) {
```

```java
            System.out.println(emp.name);
        }
        sc.close();
    }
}
```

Slip17:

A) Write a java Program to accept ‘n’ no’s through command line and store only armstrong no’s into the array and display that array. [15 M]

```java
public class Slip17a {
    static boolean isArmstrong(int num) {
        int original = num, sum = 0;
        while (num != 0) {
            int digit = num % 10;
            sum += digit * digit * digit;
            num /= 10;
        }
        return sum == original;
    }

    public static void main(String[] args) {
        int[] armstrongNums = new int[args.length];
        int count = 0;

        for (String arg : args) {
            int num = Integer.parseInt(arg);
            if (isArmstrong(num)) {
                armstrongNums[count++] = num;
            }
        }

        System.out.println("Armstrong numbers:");
        for (int i = 0; i < count; i++) {
            System.out.println(armstrongNums[i]);
        }
    }
}
```

B) Define a class Product (pid, pname, price, qty). Write a function to accept the product details, display it and calculate total amount. (use array of Objects) [25 M]

```java
import java.util.Scanner;

class Product {
    int pid;
    String pname;
    double price;
    int qty;

    Product(int pid, String pname, double price, int qty) {
        this.pid = pid;
        this.pname = pname;
        this.price = price;
        this.qty = qty;
    }

    void display() {
        System.out.println("Product ID: " + pid + ", Name: " + pname + ", Price: " + price + ", 
Qty: " + qty + ", Total: " + (price * qty));
    }
}

public class Slip17b {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter number of products: ");
        int n = sc.nextInt();
        Product[] products = new Product[n];

        for (int i = 0; i < n; i++) {
            System.out.print("Enter product ID, name, price, and qty: ");
            int pid = sc.nextInt();
            String pname = sc.next();
            double price = sc.nextDouble();
            int qty = sc.nextInt();
            products[i] = new Product(pid, pname, price, qty);
        }

        System.out.println("Product details:");
        for (Product product : products) {
            product.display();
        }
        sc.close();
    }
}
```

Slip18:

A) Write a Java program to calculate area of Circle, Triangle & Rectangle.(Use Method Overloading) [15 M]

```
import java.util.Scanner;

public class Slip18a {

    // Method to calculate area of Circle
    public static double area(double radius) {
        return Math.PI * radius * radius;
    }

    // Method to calculate area of Triangle
    public static double area(double base, double height) {
        return 0.5 * base * height;
    }

    // Method to calculate area of Rectangle
    public static double area(double[] dimensions) {
        if (dimensions.length == 2) {
            return dimensions[0] * dimensions[1]; // length * width
        }
        return 0; // Invalid case
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter shape (1: Circle, 2: Triangle, 3: Rectangle): ");
        int choice = scanner.nextInt();

        double area = 0;
        switch (choice) {
            case 1:
                System.out.print("Radius: ");
                area = area(scanner.nextDouble());
                break;
            case 2:
                System.out.print("Base: ");
                double base = scanner.nextDouble();
                System.out.print("Height: ");
                area = area(base, scanner.nextDouble());
                break;
            case 3:
                System.out.print("Length: ");
                double length = scanner.nextDouble();
                System.out.print("Width: ");
                double width = scanner.nextDouble();
                area = area(new double[]{length, width});
                break;
            default:
                System.out.println("Invalid choice.");
                return; // Exit the program
        }

        System.out.println("Area: " + area);
        scanner.close();
    }
}
```

B) Write a java program to copy the data from one file into another file, while copying change the case of characters in target file and replaces all digits by '*' symbol. [25 M]

```
import java.io.*;

class Slip18b{
    public static void main(String[] args) throws IOException{
        FileReader fr = new FileReader("old.txt");
        FileWriter fw = new FileWriter("new.txt");

        int c;

        while((c = fr.read()) != -1){
            if(Character.isDigit(c) == false){
                if(Character.isUpperCase(c)){
                    fw.write(Character.toLowerCase(c));
                }
                else if(Character.isLowerCase(c)){
                    fw.write(Character.toUpperCase(c));
                }
            }
            else{
                fw.write("*");
            }
        }
        fr.close();
        fw.close();
    }
}
```

Slip19:

A) Write a Java program to display Fibonacci series using function. [15 M]

```
import java.util.Scanner;

public class Slip19a {
    static void fibonacci(int n) {
        int a = 0, b = 1;
        System.out.print(a + " " + b + " ");
        for (int i = 2; i < n; i++) {
            int next = a + b;
            System.out.print(next + " ");
            a = b;
            b = next;
        }
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the number of terms: ");
        int n = sc.nextInt();
        fibonacci(n);
        sc.close();
    }
}
```

B) Create an Applet that displays the x and y position of the cursor movement using Mouse and Keyboard. (Use appropriate listener) [25 M]

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

public class Slip19b extends JFrame implements MouseListener, KeyListener {
    JLabel label;

    public Slip19b() {
        label = new JLabel("Move or Click mouse, press any key...");
        label.setBounds(50, 50, 300, 50);
        add(label);
        addMouseListener(this);
        addKeyListener(this);
        setSize(400, 200);
        setLayout(null);
        setVisible(true);
        setFocusable(true);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public void mouseClicked(MouseEvent e) {
        label.setText("Mouse clicked at: " + e.getX() + ", " + e.getY());
    }

    public void mouseMoved(MouseEvent e) {}
    public void mousePressed(MouseEvent e) {}
    public void mouseReleased(MouseEvent e) {}
    public void mouseEntered(MouseEvent e) {}
    public void mouseExited(MouseEvent e) {}

    public void keyPressed(KeyEvent e) {
        label.setText("Key Pressed: " + e.getKeyChar());
    }

    public void keyReleased(KeyEvent e) {}
    public void keyTyped(KeyEvent e) {}

    public static void main(String[] args) {
        new Slip19b();
    }
}
```

Slip20:

A) Write a java program using AWT to create a Frame with title “TYBBACA”, background color RED. If user clicks on close button then frame should close. [15 M]

```
import javax.swing.*;


import java.awt.*;



class Slip20a {

    public static void main(String args[]) {
        JFrame frame = new JFrame("TYBBACA");

        frame.setSize(400, 400);

        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        frame.getContentPane().setBackground(Color.RED);

        frame.setVisible(true);

    }

}
```

B) Construct a Linked List containing name: CPP, Java, Python and PHP. Then extend your java program to do the following: [25 M]

i. Display the contents of the List using an Iterator ii. Display the contents of the List in reverse order using a ListIterator.

```
import java.util.*;

public class Slip20b{
    public static void main(String[] args){
        LinkedList<String> ll = new LinkedList<>();
        ll.add("CPP");
        ll.add("JAVA");
        ll.add("Python");
        ll.add("PHP");

        System.out.println("display content using iterator: ");
        Iterator<String> it = ll.iterator();
        while(it.hasNext()){
            System.out.println(it.next());
        }

        System.out.println("display content using list iterator: ");
        ListIterator<String> lit = ll.listIterator();
        while(lit.hasNext()){
            lit.next();
        }

        while(lit.hasPrevious()){
            System.out.println(" " +lit.previous());
        }
    }
}
```

Slip21:

A) Write a java program to display each word from a file in reverse order. [15 M]

```
import java.io.*;
import java.util.*;
```

```
class Slip21a{
    public static void main(String[] args) throws IOException{
        FileReader fr = new FileReader("old.txt");
        FileWriter fw = new FileWriter("new2.txt");

        try{
            Scanner sc = new Scanner(fr);

            while(sc.hasNextLine()){
                String s = sc.nextLine();
                StringBuffer buffer = new StringBuffer(s);
                buffer = buffer.reverse();
                String res = buffer.toString();
                fw.write(res);
            }
        }
        catch(Exception e){}
        fr.close();
        fw.close();
    }
}
```

B) Create a hashtable containing city name & STD code. Display the details of the hashtable. Also search for a specific city and display STD code of that city. [25 M]

```
import java.io.*;
import java.util.*;

public class Slip21b{
    public static void main(String[] args){
        Hashtable<String, Integer> hash1 = new Hashtable<>();
        Enumeration<String> en;
        int i, n, std, val;
        String nm, cname, str;
        Scanner sc = new Scanner(System.in);

        try{
            System.out.println("enter no. of records: ");
            n = sc.nextInt();
            System.out.println("enter city name & std code: ");
            for(i = 0; i<n; i++){
                cname = sc.next();
                std = sc.nextInt();
                hash1.put(cname, std);
            }
            System.out.println("enter city name to search: ");
            nm = sc.next();

            en = hash1.keys();
            while(en.hasMoreElements()){
                str = en.nextElement();
                val = hash1.get(str);
                if(str.equals(nm)){
                    System.out.println("std code: " +val);
                }
            }

        }
        catch(Exception e){}
    }
}
```

Slip22:

A) Write a Java program to calculate factorial of a number using recursion. [15 M]

```java
import java.util.Scanner;

public class Slip22a {

    // Recursive method to calculate factorial
    public static long factorial(int n) {
        return (n == 0 || n == 1) ? 1 : n * factorial(n - 1);
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a positive integer: ");
        int number = scanner.nextInt();
        if (number < 0) {
            System.out.println("Factorial is not defined for negative numbers.");
        } else {
            System.out.println("Factorial of " + number + " is: " + factorial(number));
        }
        scanner.close();
    }
}
```

B) Write a java program for the following: [25 M]

1. To create a file. 2. To rename a file. 3. To delete a file. 4. To display path of a file.

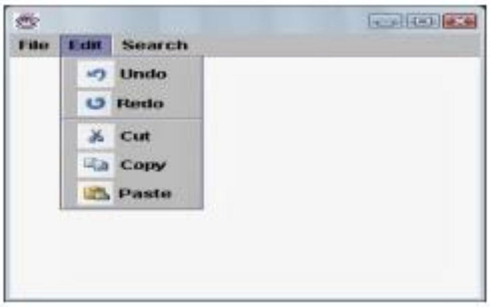
```java
import java.io.*;
import java.util.*;

class Slip22b{
    public static void main(String[] args) throws IOException{
        Scanner sc = new Scanner(System.in);

        System.out.println("1. create a file \n2. rename a file \n3. delete a file \n4. display
file path");
        System.out.println("enter a file name: ");

        String str = sc.nextLine();
        File file = new File(str);

        System.out.print("choose your option: ");
        int ch = sc.nextInt();
        sc.nextLine();

        switch(ch){
            case 1:
                if(file.createNewFile()){
                    System.out.println("file created: " +file.getName());
                }
                else{
                    System.out.println("file already exists");
                }
                break;
            case 2:
                System.out.print("enter new file name: ");
                String nf = sc.nextLine();
                File nf1 = new File(nf);

                if(file.renameTo(nf1)){
                    System.out.println("file renamed");
                }
                else{
                    System.out.println("file can't be renamed");
                }
                break;
            case 3:
                if(file.delete()){
                    System.out.println("delete the file: " +file.getName());
                }
                else{
                    System.out.println("failed to delete the file");
                }
                break;
            case 4:
                System.out.println("file location: " +file.getAbsolutePath());
                break;
            default:
                System.out.println("please choose the correct option");
                break;
        }
    }
}
```

Slip23:

A) Write a java program to check whether given file is hidden or not. If not then display its path, otherwise display appropriate message. [15 M]

```
import java.io.*;
import java.util.*;

public class Slip23a{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);

        try{
            System.out.print("enter file name: ");
            String str = sc.nextLine();
            File file = new File(str);

            if(file.isHidden()){
                System.out.println("file is hidden");
            }
            else{
                System.out.println("file location: " +file.getAbsolutePath());
            }
        }
        catch(Exception e){}
    }
}
```

B) Write a java program to design following Frame using Swing. [25 M]



Slip24:

A) Write a java program to count number of digits, spaces and characters from a file. [15 M]

```
import java.io.*;

class Slip24a{
    public static void main(String[] args) throws IOException{
        FileReader fr = new FileReader("old.txt");
        FileWriter fw = new FileWriter("out.txt");

        int c, letter = 0, spaces = 0, num = 0, other = 0;
        while((c = fr.read())!= -1){
            if(Character.isDigit(c)){
                num++;
            }
            else if(Character.isLetter(c)){
                letter++;
            }
            else if(Character.isSpaceChar(c)){
                spaces++;
            }
            else{
                other++;
            }
        }
        fw.write("number: " +num+ "\nletter: " +letter+ "\nspaces: " +spaces+ "\nspecial 
characters: " +other);

        fr.close();
        fw.close();
    }
}
```

B) Create a package TYBBACA with two classes as class Student (Rno, SName, Per) with a method disp() to display details of N Students and class Teacher (TID, TName, Subject) with a method disp() to display the details of teacher who is teaching Java subject. (Make use of finalize() method and array of Object) [25 M]

Slip25:

A) Write a java program to check whether given string is palindrome or not. [15 M]

```
import java.util.Scanner;

public class Slip25a {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String input = sc.nextLine();
        String reversed = new StringBuilder(input).reverse().toString();
        if (input.equalsIgnoreCase(reversed)) {
            System.out.println("The string is a palindrome.");
        } else {
            System.out.println("The string is not a palindrome.");
        }
        sc.close();
    }
}
```

B) Create a package named Series having three different classes to print series: i. Fibonacci series ii. Cube of numbers iii. Square of numbers Write a java program to generate ‘n’ terms of the above series. [25 M]

Slip26:

A) Write a java program to display ASCII values of the characters from a file. [15 M]

```
import java.util.*;
import java.io.*;

class Slip26a{
    public static void main(String[] agrs) throws IOException{
        char ch;
        FileReader fr = new FileReader("old.txt");
        int c;
        while((c = fr.read()) != -1){
            ch = (char)c;
            if(Character.isDigit(ch) == false && (Character.isSpaceChar(c) == false)){
                System.out.println("ASCII " +ch+ ":" +c);
            }
        }
        fr.close();
    }
}
```

B) Write a java program using applet to draw Temple. [25 M]

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

public class Slip26b extends JPanel {
    @Override
    protected void paintComponent(Graphics g) {
        super.paintComponent(g);
        // Draw temple structure
        g.drawRect(100, 100, 200, 200); // Base
        g.drawLine(100, 100, 200, 50); // Left slope
        g.drawLine(200, 50, 300, 100); // Right slope
        g.drawRect(175, 200, 50, 100); // Door
    }

    public static void main(String[] args) {
        JFrame frame = new JFrame("Temple Drawing");
        Slip26b temple = new Slip26b();
        frame.add(temple);
        frame.setSize(400, 400);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}
```

Slip27:

A) Write a java program to accept a number from user, If it is greater than 1000 then throw user defined exception “Number is out of Range” otherwise display the factors of that number. (Use static keyword) [15 M]

```
import java.util.Scanner;

class NumberOutOfRangeException extends Exception {
    public NumberOutOfRangeException(String message) {
        super(message);
    }
}

public class Slip27a {
    static int number;
    // Method to display the factors of a number
    private static void displayFactors(int number) {
        System.out.print("Factors of " + number + " are: ");
        for (int i = 1; i <= number; i++) {
            if (number % i == 0) {
                System.out.print(i + " ");
            }
        }
        System.out.println(); // For a newline after displaying all factors
    }

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        try {
            System.out.print("Enter a number: ");
            int number = scanner.nextInt();

            if (number > 1000) {
                throw new NumberOutOfRangeException("Number is out of Range.");
            } else {
                displayFactors(number);
            }
        } catch (NumberOutOfRangeException e) {
            System.out.println(e.getMessage());
        } finally {
            scanner.close();
        }
    }
}
```

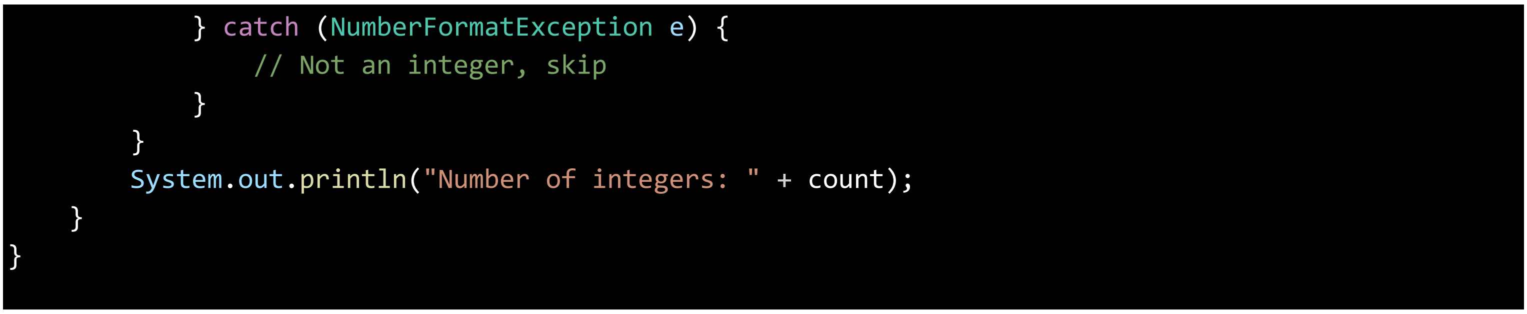
B) Write a java program to accept directory name in TextField and display list of files and subdirectories in List Control from that directory by clicking on Button. [25 M]

```
import java.awt.*;
import java.awt.event.*;
import java.io.*;

public class Slip27b extends Frame implements ActionListener{
    Graphics g;
    List l;
    TextField t1;
    Button b1;
    Label l1;

    public Slip27b(){
        this.setLayout(new FlowLayout());
        this.setSize(300, 300);
        this.setVisible(true);

        l1 = new Label("enter directory: ");
        t1 = new TextField(20);
        l = new List(10);
        b1 = new Button("click me");

        l1.setBounds(50, 100, 80, 50);
        t1.setBounds(50, 150, 80, 80);
        b1.setBounds(50, 200, 80, 80);
        l.setBounds(50, 300, 100, 100);

        add(l1);
        add(t1);
        add(b1);
        add(l);

        b1.addActionListener(this);

        // Add a window listener to close the frame
        addWindowListener(new WindowAdapter(){
            public void windowClosing(WindowEvent we){
                dispose(); // Close the frame
            }
        });
    }

    public void actionPerformed(ActionEvent ae){
        if (ae.getSource() == b1){
            try{
                String nm = t1.getText();
                File f1 = new File(nm);
                String s1[] = f1.list();

                if(s1 == null){
                    l.add("dir not exist");
                }else{
                    for (int i = 0; i < s1.length; i++){
                        l.add(s1[i]);
                    }
                }
            }catch(Exception e){
                //l.add("Error: " + e.getMessage());
            }
        }
    }

    public static void main(String[] args){
        new Slip27b();
    }
}
```

Slip28:

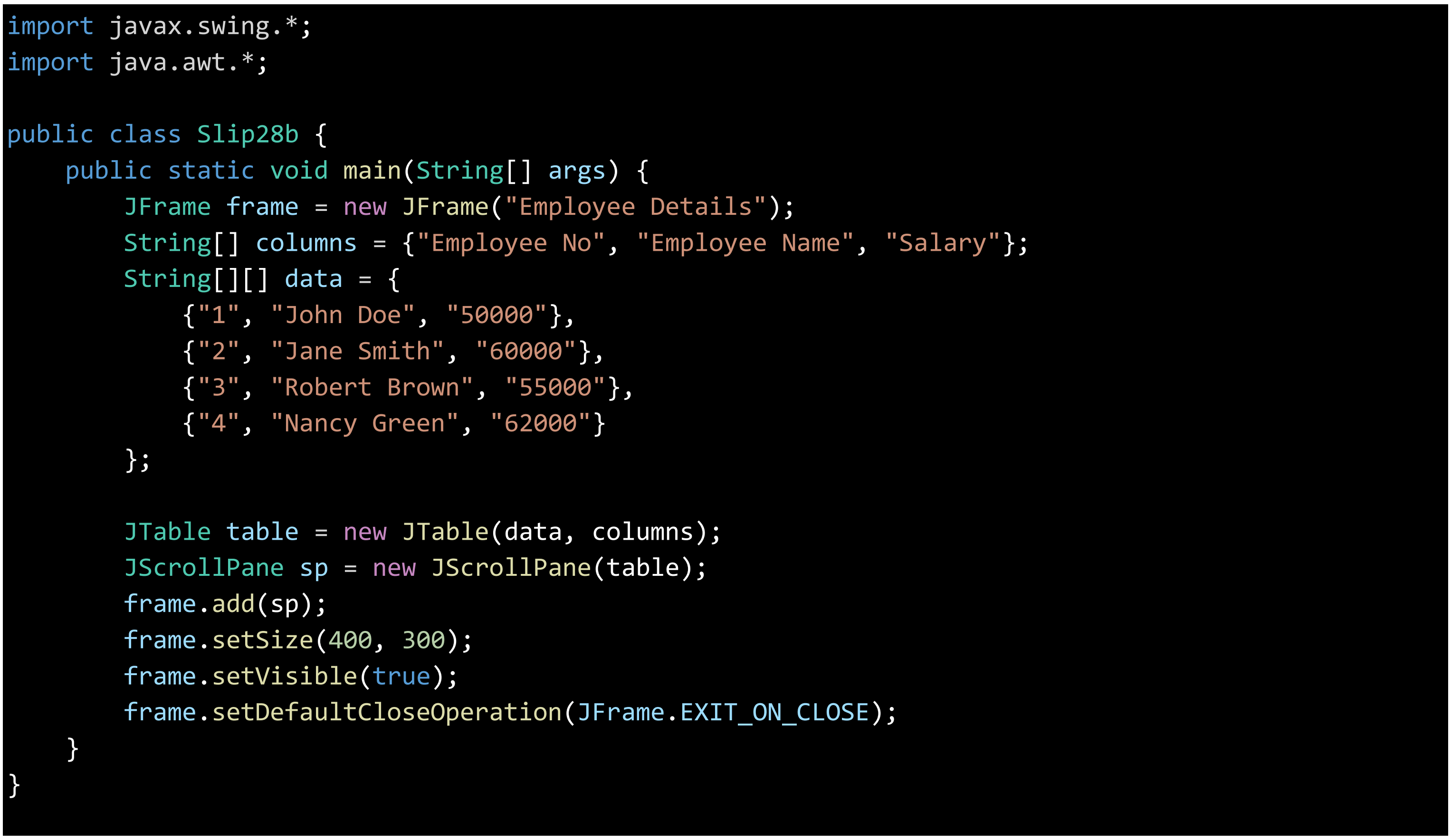
A) Write a java program to count the number of integers from a given list. (Use Command line arguments). [15 M]

```
public class Slip28a {
    public static void main(String[] args) {
        int count = 0;
        for (String arg : args) {
            try {
                Integer.parseInt(arg);
                count++;
```



```
            } catch (NumberFormatException e) {
                // Not an integer, skip
            }
        }
        System.out.println("Number of integers: " + count);
    }
}
```

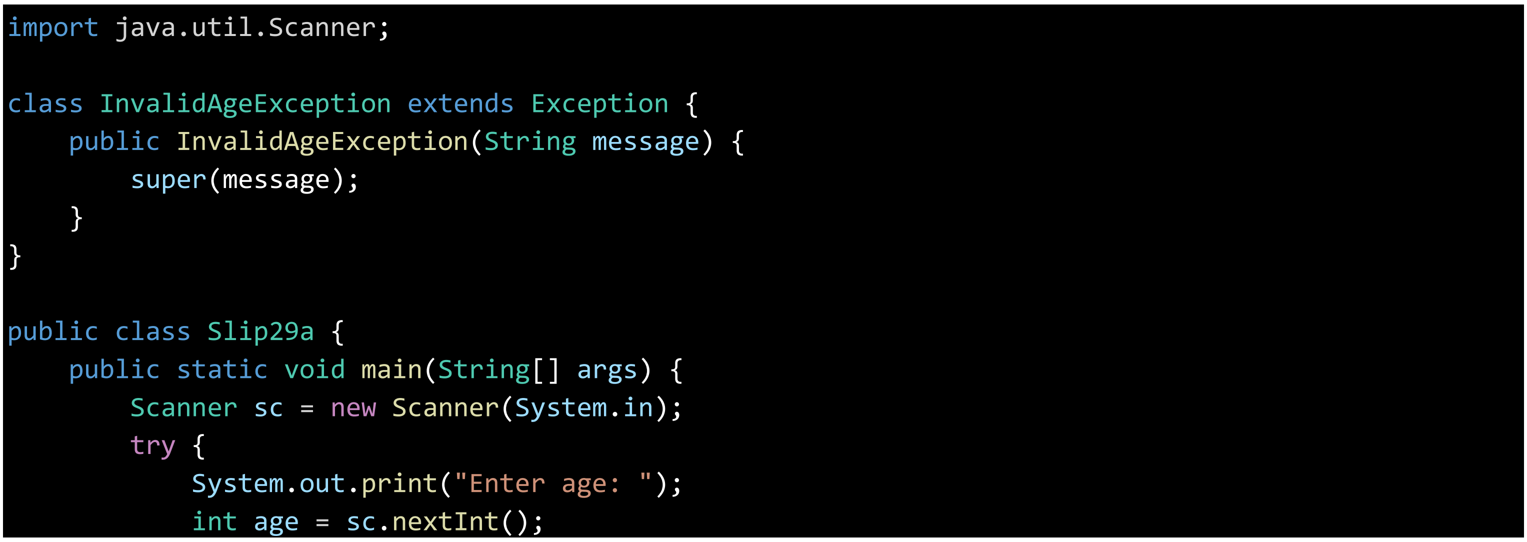
B) Write a java Program to accept the details of 5 employees (Eno, Ename, Salary) and display it onto the JTable. [25 M]



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

public class Slip28b {
    public static void main(String[] args) {
        JFrame frame = new JFrame("Employee Details");
        String[] columns = {"Employee No", "Employee Name", "Salary"};
        String[][] data = {
            {"1", "John Doe", "50000"},
            {"2", "Jane Smith", "60000"},
            {"3", "Robert Brown", "55000"},
            {"4", "Nancy Green", "62000"}
        };

        JTable table = new JTable(data, columns);
        JScrollPane sp = new JScrollPane(table);
        frame.add(sp);
        frame.setSize(400, 300);
        frame.setVisible(true);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}
```

Slip29:

A) Write a java program to check whether given candidate is eligible for voting or not. Handle user defined as well as system defined Exception. [15 M]



```
import java.util.Scanner;

class InvalidAgeException extends Exception {
    public InvalidAgeException(String message) {
        super(message);
    }
}

public class Slip29a {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        try {
            System.out.print("Enter age: ");
            int age = sc.nextInt();
            if (age < 0) throw new IllegalArgumentException("Age cannot be negative.");
            if (age >= 18) {
                System.out.println("Eligible for voting.");
            } else {
                throw new InvalidAgeException("Not eligible for voting.");
            }
        } catch (InvalidAgeException e) {
            System.out.println(e.getMessage());
        } catch (IllegalArgumentException e) {
            System.out.println(e.getMessage());
        }
        sc.close();
    }
}
```

B) Write a java program using Applet for bouncing ball. Ball should change its color for each bounce. [25 M]

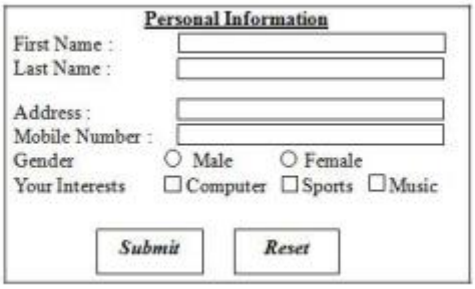
```
import javax.swing.*;
import java.awt.*;
import java.util.Random;

public class Slip29b extends JPanel implements Runnable {
    int x = 50, y = 50, dx = 5, dy = 5;
    Random rand = new Random();
    Color ballColor = Color.RED;

    public Slip29b() {
        setPreferredSize(new Dimension(400, 400));
        Thread t = new Thread(this);
        t.start();
    }

    public void run() {
        while (true) {
            x += dx;
            y += dy;
            if (x < 0 || x > getWidth() - 50) {
                dx = -dx;
                ballColor = new Color(rand.nextInt(256), rand.nextInt(256), rand.nextInt(256));
            }
            if (y < 0 || y > getHeight() - 50) {
                dy = -dy;
                ballColor = new Color(rand.nextInt(256), rand.nextInt(256), rand.nextInt(256));
            }
            repaint();
            try {
                Thread.sleep(50);
            } catch (InterruptedException e) {}
        }
    }

    @Override
    protected void paintComponent(Graphics g) {
        super.paintComponent(g);
        g.setColor(ballColor);
        g.fillOval(x, y, 50, 50);
    }

    public static void main(String[] args) {
        JFrame frame = new JFrame("Bouncing Ball");
        Slip29b bouncingBall = new Slip29b();
        frame.add(bouncingBall);
        frame.pack();
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}
```

Slip30:

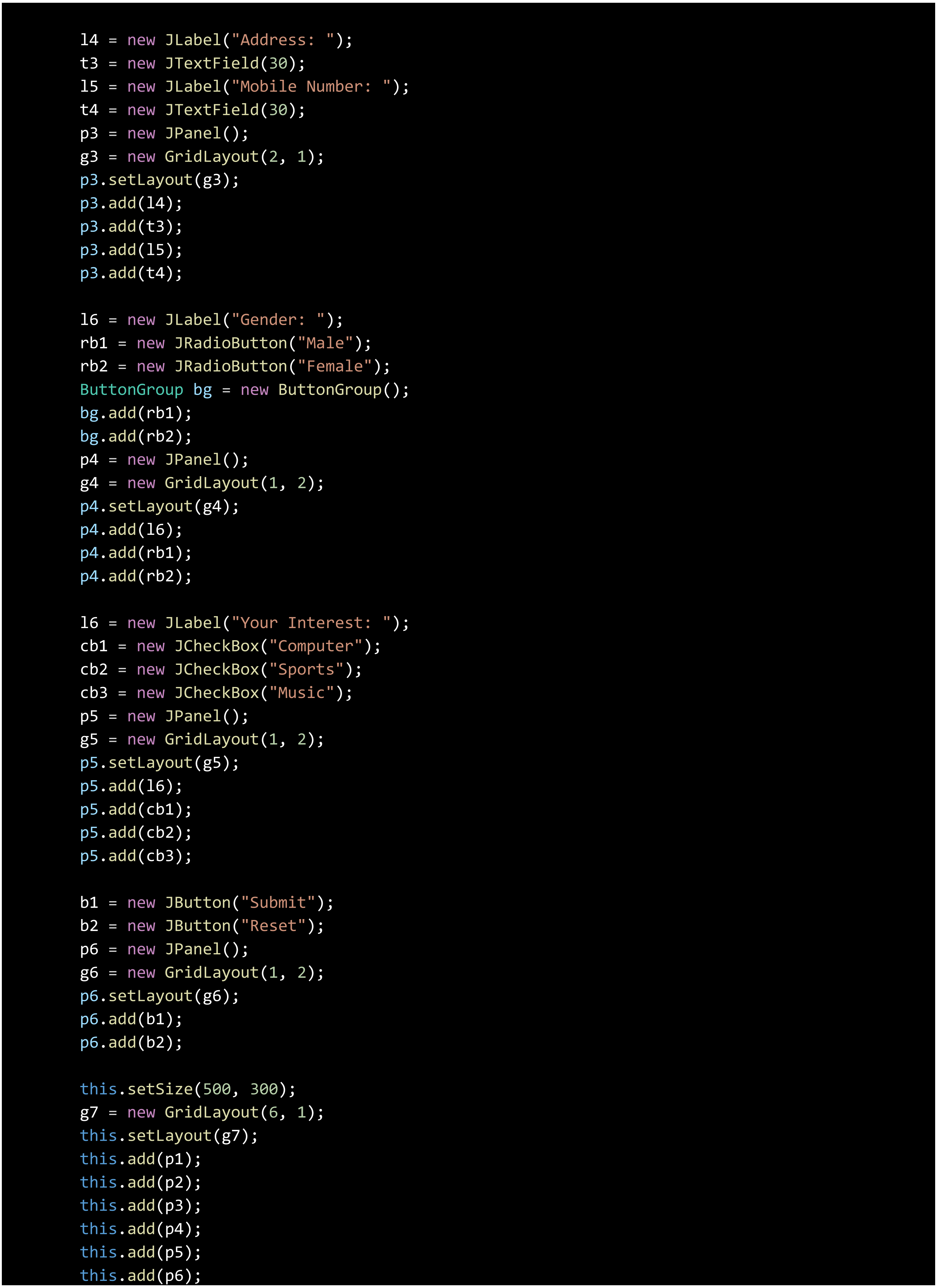
A) Write a java program to accept a number from a user, if it is zero then throw user defined Exception “Number is Zero”. If it is non-numeric then generate an error “Number is Invalid” otherwise check whether it is palindrome or not. [15 M]

```
import java.util.*;

class numIsZero extends Exception{}

class Slip30a{
    public static void main(String[] args){
        int r, sum = 0, temp;
        int n;
        Scanner sc = new Scanner(System.in);

        try{
            System.out.print("enter number: ");
            n = sc.nextInt();

            if(n == 0){
                throw new numIsZero();
            }
            else{
                temp = n;

                while(n > 0){
                    r = n % 10;
                    sum = (sum * 10) + r;
                    n = n/10;
                }

                if(temp == sum){
                    System.out.println("palindrome number");
                }
                else{
                    System.out.println("not a palindrome number");
                }
            }
        }
        catch(numIsZero niz){
            System.out.println("number is zero");
        }
        catch(NumberFormatException e){
            System.out.println("invalid number");
        }
        catch(Exception e){}
    }
}
```

B) Write a java program to design a following GUI (Use Swing). [25 M]



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

class Slip30b extends JFrame {
    JLabel l1, l2, l3, l4, l5, l6;
    JTextField t1, t2, t3, t4;
    JButton b1, b2;
    JRadioButton rb1, rb2;
    JCheckBox cb1, cb2, cb3;
    JPanel p1, p2, p3, p4, p5, p6;
    GridLayout g1, g2, g3, g4, g5, g6, g7;

    public Slip30b() {
        setTitle("Personal Information Form");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        l1 = new JLabel("<html><u>Personal Information</u></html>", JLabel.CENTER);
        p1 = new JPanel();
        g1 = new GridLayout(1, 1);
        p1.setLayout(g1);
        p1.add(l1);
        l1.setFont(new Font("Arial", Font.BOLD, 20));

        l2 = new JLabel("First Name: ");
        t1 = new JTextField(30);
        l3 = new JLabel("Last Name: ");
        t2 = new JTextField(30);
        p2 = new JPanel();
        g2 = new GridLayout(2, 1);
        p2.setLayout(g2);
        p2.add(l2);
        p2.add(t1);
        p2.add(l3);
        p2.add(t2);
```



```
        l4 = new JLabel("Address: ");
        t3 = new JTextField(30);
        l5 = new JLabel("Mobile Number: ");
        t4 = new JTextField(30);
        p3 = new JPanel();
        g3 = new GridLayout(2, 1);
        p3.setLayout(g3);
        p3.add(l4);
        p3.add(t3);
        p3.add(l5);
        p3.add(t4);

        l6 = new JLabel("Gender: ");
        rb1 = new JRadioButton("Male");
        rb2 = new JRadioButton("Female");
        ButtonGroup bg = new ButtonGroup();
        bg.add(rb1);
        bg.add(rb2);
        p4 = new JPanel();
        g4 = new GridLayout(1, 2);
        p4.setLayout(g4);
        p4.add(l6);
        p4.add(rb1);
        p4.add(rb2);

        l6 = new JLabel("Your Interest: ");
        cb1 = new JCheckBox("Computer");
        cb2 = new JCheckBox("Sports");
        cb3 = new JCheckBox("Music");
        p5 = new JPanel();
        g5 = new GridLayout(1, 2);
        p5.setLayout(g5);
        p5.add(l6);
        p5.add(cb1);
        p5.add(cb2);
        p5.add(cb3);

        b1 = new JButton("Submit");
        b2 = new JButton("Reset");
        p6 = new JPanel();
        g6 = new GridLayout(1, 2);
        p6.setLayout(g6);
        p6.add(b1);
        p6.add(b2);

        this.setSize(500, 300);
        g7 = new GridLayout(6, 1);
        this.setLayout(g7);
        this.add(p1);
        this.add(p2);
        this.add(p3);
        this.add(p4);
        this.add(p5);
        this.add(p6);
        this.setVisible(true);
    }

    public static void main(String[] args) {
        new Slip30b();
    }
}
```