

# Java Cycle 2

1. Write a program that reads a CSV file(Let csv file have Some information relating to student like Name,RollNo and MobileNo) and display the contents in console window.

```
import java.io.*;

public class P1StudentsCSV {

    public static void main(String[] args) throws IOException {
        FileReader fr = new FileReader("data.csv");
        BufferedReader br = new BufferedReader(fr);
        String line;
        while ((line = br.readLine()) != null) {
            String[] sd = line.split(",");
            System.out.println("Student - Name:" + sd[0] + ", Roll.no:" +
sd[1] + ", Mobile:" + sd[2] + "\n");
        }
    }
}
```

*data.csv:*

Abc,1,0

Xyz,2,1

Pqr,3,2

2. Write a program that reads two numbers from the user to perform integer division into Num1 and Num2 variables. The division of Num1 and Num2 is displayed if they are integers. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception.

```
import java.util.Scanner;

public class P2NumberExceptions {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String _num1 = sc.next();
        String _num2 = sc.next();
        try {
            int num1 = Integer.parseInt(_num1);
            int num2 = Integer.parseInt(_num2);
            System.out.println(num1 / num2);
        }
    }
}
```

```

        } catch (NumberFormatException | ArithmeticException e) {
            System.out.println(e);
        }
    }
}

```

3. Write a program that accepts age from the user. Throw a user defined exception when age is <0 and age >120.

```

import java.util.Scanner;

class AgeException extends Exception {
    AgeException(int age) {
        super("AgeException " + age + " is not in between 0-120");
    }
}

public class P3UDEException {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int age = sc.nextInt();
        try {
            if (age < 0 || age > 120) {
                throw new AgeException(age);
            } else {
                System.out.println(age + " is ok");
            }
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}

```

4. Write a program that creates 3 threads by extending the Thread class. First thread displays “Good Morning” every 1 sec, the second thread displays “Hello” every 2 seconds and the third displays Welcome” every 3 seconds. (Repeat the same by implementing Runnable).

```

public class P4Threads {
    public static void main(String[] args) {
        new Thread(() -> {
            while (true) {
                try {
                    Thread.sleep(1000);
                    System.out.println("Good morning!");
                } catch (Exception e) {

```

```

        e.printStackTrace();
    }
}
}).start();
new Thread(() → {
    while (true) {
        try {
            Thread.sleep(2000);
            System.out.println("Hello!");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}).start();
new Thread(() → {
    while (true) {
        try {
            Thread.sleep(3000);
            System.out.println("Welcome!");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}).start();
}
}

```

5. Write a Java program that illustrates producer consumer problem using Interthread communication.

```

// from geeks4geeks
import java.util.LinkedList;
class Data {
    LinkedList<Integer> list = new LinkedList<>();
    int capacity = 2;

    void produce() throws InterruptedException {
        int value = 0;
        while (true) {
            while (list.size() == capacity) wait();
            System.out.println("Produced - " + value++);
            list.add(value);
            notify();
            Thread.sleep(1000);
        }
    }
}

```

```

    void consume() throws InterruptedException {
        while (true) {
            while (list.size() == 0) wait();
            System.out.println("Consumed - " + list.pop());
            notify();
            Thread.sleep(1000);
        }
    }
}

public class P5ProducerConsumer {
    public static void main(String[] args) throws InterruptedException {
        Data data = new Data();
        Thread p_thread = new Thread(() → {
            try {
                data.produce();
            } catch (InterruptedException e) {
                System.out.println(e);
            }
        });
        Thread c_thread = new Thread(() → {
            try {
                data.consume();
            } catch (InterruptedException e) {
                System.out.println(e);
            }
        });
        p_thread.start();
        c_thread.start();
        p_thread.join();
        c_thread.join();
    }
}

```

6. Create a JApplet that displays a message which is scrolling from left to right.

```

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

public class P6ScrollingText extends JFrame {
    Label text = new Label("Hello world");

    P6ScrollingText() {
        super("Scrolling Text");
        setSize(300, 300);
        text.setLocation(150, 0);
        add(text);
    }

    int multiplier = 1;

    @Override

```

```

public void paint(Graphics g) {
    super.paint(g);
    int offset = 10;
    int x = text.getLocation().x;
    int y = text.getLocation().y;

    if (x == 0) multiplier = 1;
    if (x == 300) multiplier = -1;

    text.setLocation(x + multiplier * offset, y);

    try {
        Thread.sleep(50);
        repaint();
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
}

public static void main(String[] args) {
    P6ScrollingText scrollingText = new P6ScrollingText();
    scrollingText.setVisible(true);
}
}

```

7. Write a program for handling mouse events with adapter classes.

```

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

public class P7MouseEvents extends JFrame implements MouseListener {

    JLabel click_label = new JLabel("Click to see change");
    JLabel press_label = new JLabel("Press/release to see change");
    JLabel enter_label = new JLabel("Enter/exit to see change");

    P7MouseEvents() {
        super("Mouse events");
        addMouseListener(this);
        setSize(500, 300);
        setLayout(new GridLayout());
        add(click_label);
        add(press_label);
        add(enter_label);
    }

    public static void main(String[] args) {
        P7MouseEvents events = new P7MouseEvents();
        events.setVisible(true);
    }

    @Override

```

```

    public void mouseClicked(MouseEvent e) {
        click_label.setText("Mouse click at (" + e.getX() + "," + e.getY() +
        ")");
    }

    @Override
    public void mousePressed(MouseEvent e) {
        press_label.setText("Mouse press at (" + e.getX() + "," + e.getY() +
        ")");
    }

    @Override
    public void mouseReleased(MouseEvent e) {
        press_label.setText("Mouse released at (" + e.getX() + "," + e.getY()
        + ")");
    }

    @Override
    public void mouseEntered(MouseEvent e) {
        enter_label.setText("Mouse entered at (" + e.getX() + "," + e.getY()
        + ")");
    }

    @Override
    public void mouseExited(MouseEvent e) {
        enter_label.setText("Mouse exited at (" + e.getX() + "," + e.getY() +
        ")");
    }
}

```

8. Create an interface containing 3 radio buttons named line, rectangle and oval. Based on the radio button selected, allow user to draw lines, rectangles or ovals.

```

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

public class P8GraphicPrinter extends JFrame {

    public static void main(String[] args) {
        P8GraphicPrinter gp = new P8GraphicPrinter();
        gp.setSize(500, 500);
        gp.setVisible(true);
    }

    String mode = "";

    P8GraphicPrinter() {
        super("Graphics Printer");
        setLayout(new FlowLayout());
        JRadioButton rb1 = new JRadioButton("Line");
        JRadioButton rb2 = new JRadioButton("Rectangle");
        JRadioButton rb3 = new JRadioButton("Oval");
    }
}

```

```

        rb1.addActionListener((ActionEvent e) → {
            rb2.setSelected(false);
            rb3.setSelected(false);
            mode = "line";
            repaint();
        });
        rb2.addActionListener((ActionEvent e) → {
            rb1.setSelected(false);
            rb3.setSelected(false);
            mode = "rect";
            repaint();
        });
        rb3.addActionListener((ActionEvent e) → {
            rb1.setSelected(false);
            rb2.setSelected(false);
            mode = "oval";
            repaint();
        });
        add(rb1);
        add(rb2);
        add(rb3);
    }

    @Override
    public void paint(Graphics g) {
        super.paint(g);
        switch (mode) {
            case "line":
                g.drawLine(100, 200, 400, 250);
                break;
            case "rect":
                g.drawRect(100, 200, 200, 200);
                break;
            case "oval":
                g.drawOval(100, 200, 200, 150);
                break;
        }
    }
}

```

9. Write a Java program to display students information using JTable.(Let students name,rollno and mobile number be the table headings).

```

import javax.swing.*;

public class P9StudentsTable extends JFrame {

    P9StudentsTable() {
        super("Student's table");
        setSize(500, 500);
        String[][] students = {
            {"101", "Jon", "+9110010010001"},

```

```

        {"102", "Rob", "+9110810810802"},
        {"103", "Daenerys", "+9110410410403"},
    };
    String[] head = {"Roll no.", "Name", "Phone"};
    JTable table = new JTable(students, head);
    JScrollPane pane = new JScrollPane(table);
    add(pane);
}

public static void main(String[] args) {
    P9StudentsTable studentsTable = new P9StudentsTable();
    studentsTable.setVisible(true);
}
}

```

10. Write a program that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result, and then sends the result back to the client. The client displays the result on the console. For ex: The data sent from the client is the radius of a circle, and the result produced by the server is the area of the circle.

```

import java.io.*;
import java.net.*;
import java.util.*;

public class P10ServerClient {

    static int port = 3000;

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Choose 1 for server, other for client: ");
        try {
            int op = (sc.nextInt());
            if ((op == 1)) server();
            else client(sc);
        } catch (Exception e) {
            System.out.print(e);
        }
    }

    static synchronized void server() throws Exception {
        ServerSocket serverSocket = new ServerSocket(port);
        System.out.println("Server started successfully");
        while (true) {
            Socket clientSocket = serverSocket.accept();
            DataInputStream dis = new
DataInputStream(clientSocket.getInputStream());
            DataOutputStream dos = new
DataOutputStream(clientSocket.getOutputStream());
            float radius = dis.readFloat();
            System.out.println("Client says " + radius);
            dos.writeFloat((float) (Math.PI * radius * radius));

```



```

        System.out.println("Responded to client");
    }
}

static synchronized void client(Scanner sc) throws Exception {
    while (true) {
        System.out.print("Enter radius of circle: ");
        float radius = sc.nextFloat();
        Socket socket = new Socket("localhost", port);
        DataInputStream dis = new
DataInputStream(socket.getInputStream());
        DataOutputStream dos = new
DataOutputStream(socket.getOutputStream());
        dos.writeFloat(radius);
        float result = dis.readFloat();
        System.out.println("Result is " + result);
    }
}
}

```

11. Create a registration form comprising of name(JTextField), gender(JRadioButton) and year of birth(JComboBox). Write a program that displays registration details in a JTextArea when user clicks on submit button.

```

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

public class P11RegBox extends JFrame {

    public static void main(String[] args) {
        P11RegBox regBox = new P11RegBox();
        regBox.setSize(500, 500);
        regBox.setVisible(true);
    }

    int year = 1990;

    public P11RegBox() {
        super("Registration Box");
        setLayout(new BorderLayout(getContentPane(), BorderLayout.PAGE_AXIS));

        JLabel l1 = new JLabel("Enter your name:");
        JTextField nameInp = new JTextField();
        add(l1);
        add(nameInp);

        JLabel l2 = new JLabel("Select your gender:");
        JPanel grp = new JPanel(new FlowLayout());
        JRadioButton rb1 = new JRadioButton("Male");
        JRadioButton rb2 = new JRadioButton("Female");
        grp.add(rb1);
    }
}

```

```

        grp.add(rb2);
        add(l2);
        add(grp);

        Vector<Integer> years = new Vector<>();
        for (int i = 1990; i < 2022; i++) years.add(i);

        JComboBox<Integer> cBox = new JComboBox<>(years);
        cBox.addActionListener((ActionEvent event) → year = (Integer)
cBox.getSelectedItem());

        JLabel l3 = new JLabel("Select your birth year:");
        add(l3);
        add(cBox);

        JTextArea textArea = new JTextArea();
        textArea.setEditable(false);

        JButton btn = new JButton("Submit");
        btn.addActionListener((ActionEvent e) → {
            String op = "Details:\n" +
                "Name: " + nameInp.getText().trim() + "\n" +
                "Gender: " + (rb1.isSelected() ? "Male" :
rb2.isSelected() ? "Female" : "") + "\n" +
                "Year of birth: " + year;
            textArea.setText(op);
        });

        add(btn);
        add(textArea);
    }
}

```

12. Create an Swing application with 3 Labels(with labels 1st Number, 2nd Number and Result), 3 TextFields and 4 Buttons(with labels +, -, \*, /). Write a Program that works as a simple calculator(User will give 2 values in first two textfields and result must be displayed in third textfield when user clicks on appropriate button).

```

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

public class P12Calcy extends JFrame implements ActionListener {

    JTextField firstInp = new JTextField(10);
    JTextField secondInp = new JTextField(10);
    JLabel resultLabel = new JLabel();

    public static void main(String[] args) {
        P12Calcy calcy = new P12Calcy();
        calcy.setSize(500, 300);
        calcy.setVisible(true);
    }
}

```

```

public P12Calcy() {
    super("Calculator");
    setLayout(new BorderLayout(getContentPane(), BorderLayout.Y_AXIS));

    JPanel inputPanel = new JPanel();
    inputPanel.setLayout(new GridLayout(3, 2));

    inputPanel.add(new JLabel("Enter your first number:"));
    inputPanel.add(firstInp);

    inputPanel.add(new JLabel("Enter your second number:"));
    inputPanel.add(secondInp);

    inputPanel.add(new JLabel("Your result:"));
    inputPanel.add(resultLabel);

    JPanel buttonsPanel = new JPanel();
    buttonsPanel.setLayout(new FlowLayout());

    String[] commands = new String[]{"+", "-", "*", "/"};

    for (String command : commands) {
        JButton button = new JButton(command);
        button.setActionCommand(command);
        button.addActionListener(this);
        buttonsPanel.add(button);
    }

    add(inputPanel);
    add(buttonsPanel);
}

```

@Override

```

public void actionPerformed(ActionEvent e) {
    try {
        int i1 = Integer.parseInt(firstInp.getText());
        int i2 = Integer.parseInt(secondInp.getText());
        switch (e.getActionCommand()) {
            case "+":
                resultLabel.setText("" + (i1 + i2));
                break;
            case "-":
                resultLabel.setText("" + (i1 - i2));
                break;
            case "*":
                resultLabel.setText("" + (i1 * i2));
                break;
            case "/":
                resultLabel.setText("" + (i1 / (float) i2));
                break;
        }
    }
}

```

```
    } catch (Exception ex) {  
        resultLabel.setText(ex.toString());  
    }  
}
```