

**Java Programming**

**MCA-272**

**Assignment – 08**

***BY***

**HIMANSHU HEDA (24225013)**

**SUBMITTED TO**

**Dr. Manjula Shannhog**

**SCHOOL OF SCIENCES**

**2024-25**

1. **Thread Priority : --**

package Threads\_and\_Swings;

class thread1 extends Thread {

    thread1(String name) {

        super(name);

    }

    public void run() {

        for (int i = 0; i < 10; i++) {

            System.out.println("Good Morning");

            try {

            } catch (Exception e) {

                System.out.println(e);

            }

        }

    }

}

class thread2 extends Thread {

    thread2(String name) {

        super(name);

    }

    public void run() {

        for (int i = 0; i < 10; i++) {

            System.out.println("Myself Himanshu Heda");

            try {

            } catch (Exception e) {

                System.out.println(e);

            }

        }

    }

}

public class Thread\_priority {

    public static void main(String[] args) {

        thread1 t1 = new thread1("This is the program on Thread Priority");

        thread2 t2 = new thread2("Have a Look on it.");

        // Print initial thread names and priorities

        System.out.println("Thread 1 name is " + t1.getName());

        System.out.println("Thread 2 name is " + t2.getName());

        System.out.println("Thread 1 Priority is " + t1.getPriority());

        System.out.println("Thread 2 Priority is " + t2.getPriority());

        // Set new priorities

        t1.setPriority(9);

        t2.setPriority(3);

        // Start the threads

        t1.start();

        t2.start();

        try {

            // Wait for threads to complete

            t1.join();

            t2.join();

        } catch (InterruptedException e) {

            System.out.println(e);

        }

        // Print final priorities after threads have completed

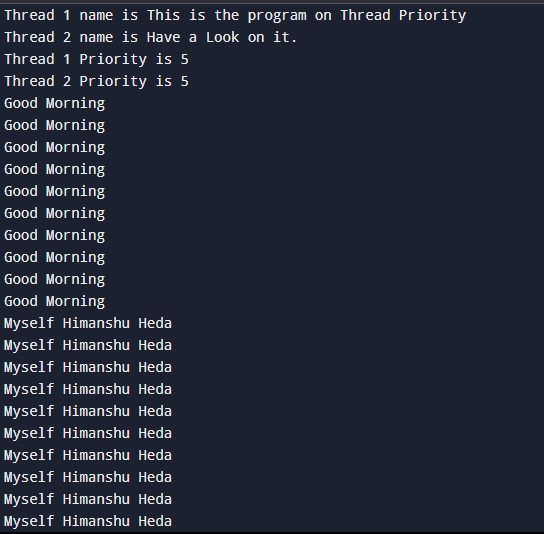
        System.out.println("Thread 1 Priority is " + t1.getPriority());

        System.out.println("Thread 2 Priority is " + t2.getPriority());

    }

}

Output : --



1. **Synchronization Thread : --**

package Threads\_and\_Swings;

class SharedCounter {

    private int count = 0;

    // Synchronized method to ensure thread safety

    public synchronized void increment() {

        count++;

    }

    public int getCount() {

        return count;

    }

}

class IncrementThread extends Thread {

    SharedCounter counter;

    public IncrementThread(SharedCounter counter) {

        this.counter = counter;

    }

    @Override

    public void run() {

        for (int i = 0; i < 100; i++) {

            counter.increment();

        }

    }

}

public class Synchronization\_thread {

    public static void main(String[] args) {

        SharedCounter counter = new SharedCounter();

        IncrementThread thread1 = new IncrementThread(counter);

        IncrementThread thread2 = new IncrementThread(counter);

        thread1.start();

        thread2.start();

        // Wait for both threads to finish

        try {

            thread1.join();

            thread2.join();

        } catch (InterruptedException e) {

            System.out.println("Thread interrupted: " + e.getMessage());

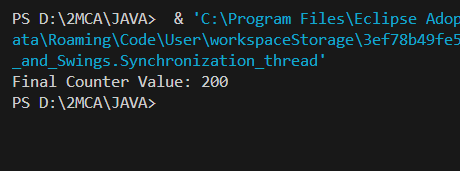
        }

        System.out.println("Final Counter Value: " + counter.getCount());

    }

}

**Output : --**

****

1. **Swings : --**

package Threads\_and\_Swings;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class Login extends JFrame implements ActionListener {

    // Now we have to define the buttons Globally to access each of them outside the constructor also

    // If we declear the JButton Globally then we do not need to mentioned it Locally

    JButton login, signup, clear;

    JTextField cardTextField;

    JPasswordField pinTextField;

    // Lets Define a Constructor Named Login

    Login(){

        // This is Title

        setTitle("AUTOMATED TELLER MACHINE");

        // This is the layout which is use for the customizations

        setLayout(null);

        // This is the logo

        ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/logo.jpg"));

        Image i2 = i1.getImage().getScaledInstance(100,100,Image.SCALE\_DEFAULT);

        ImageIcon i3 = new ImageIcon(i2);

        JLabel label = new JLabel(i3);

        label.setBounds(70, 10, 100, 100);

        add(label);

        // This is the Text

        JLabel text = new JLabel("Welcome To ATM");

        text.setFont(new Font("Osward", Font.BOLD, 38));

        text.setBounds(200, 40, 400, 40);

        add(text);

        // This is the Card No.

        JLabel cardno = new JLabel("Card No. :");

        cardno.setFont(new Font("Raleway", Font.BOLD, 28));

        cardno.setBounds(120, 150, 150, 30);

        add(cardno);

        // TextBox for the Card No.

        cardTextField = new JTextField();

        cardTextField.setBounds(300, 150, 230, 30);

        cardTextField.setFont(new Font("Arial", Font.BOLD, 14));

        add(cardTextField);

        // This is the pin

        JLabel pin = new JLabel("Pin :");

        pin.setFont(new Font("Osward", Font.BOLD, 28));

        pin.setBounds(120, 220, 250, 40);

        add(pin);

        // TextBox for the pin

        pinTextField = new JPasswordField();

        pinTextField.setBounds(300, 220, 230, 30);

        pinTextField.setFont(new Font("Arial", Font.BOLD, 14));

        add(pinTextField);

        // Lets Create a Button of Sign In

        login = new JButton("SIGN IN");

        login.setBounds(300, 300, 100, 30);

        login.setBackground(Color.BLACK);

        login.setForeground(Color.WHITE);

        login.addActionListener(this);

        add(login);

        // Lets Create a Button of Clear

        clear = new JButton("CLEAR");

        clear.setBounds(430, 300, 100, 30);

        clear.setBackground(Color.BLACK);

        clear.setForeground(Color.WHITE);

        clear.addActionListener(this);

        add(clear);

        // Lets Create a Button of Sign Up

        signup = new JButton("SIGN UP");

        signup.setBounds(300, 350, 230, 30);

        signup.setBackground(Color.BLACK);

        signup.setForeground(Color.WHITE);

        signup.addActionListener(this);

        add(signup);

        // It is use the change the background color

        getContentPane().setBackground(Color.WHITE);

        // This is use the create a basic frame in which we can design everything

        setSize(800,480);

        setVisible(true);

        setLocation(350,200);

    }

    // Abstact Method Override

    // ActionEvent ae is use to define what action you need to perform or on what component it is performaed

    public void actionPerformed(ActionEvent ae){

        if (ae.getSource() == clear){

            cardTextField.setText("");

            pinTextField.setText("");

        }

    }

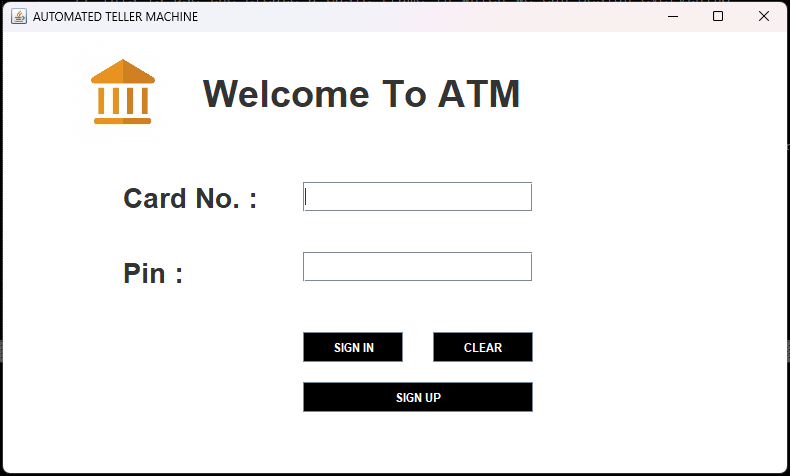
    public static void main(String[] args) {

        new Login();

    }

}

**Output : --**

****