|  |  |
| --- | --- |
| **Name** | Lekh Sanatan Nayak |
| **UID no.** | 2023800068 |
| **Experiment No.** |  |

|  |  |
| --- | --- |
| **AIM:** |  |
| **Program 1** | |
| **PROBLEM STATEMENT :** |  |
| **PROGRAM:** | import java.util.\*; **// Importing all important methods**  public class Date {  **// Private member variables to store year, month, date, hours, minutes, and seconds**  private int year, month, date, hrs, min, sec;  **// Default constructor for Date class**  Date() {  **// Initializing default values**  this.month = 1;  this.date = 1;  this.year = 2000;  this.hrs = 00;  this.min = 00;  this.sec = 00;  }  **// Setter method to set date with 3 arguments**  public void setdate(int year, int month, int date){  this.month = month;  this.date = date;  this.year = year;  this.hrs = 00;  this.min = 00;  this.sec = 00;  }  **// Setter method to set date with 5 arguments**  public void setdate(int year, int month, int date, int hrs, int min){  this.month = month;  this.date = date;  this.year = year;  this.hrs = hrs;  this.min = min;  this.sec = 00;  }  **// Setter method to set date with 6 arguments**  public void setdate(int year, int month, int date, int hrs, int min, int sec){  this.month = month;  this.date = date;  this.year = year;  this.hrs = hrs;  this.min = min;  this.sec = sec;  }  **// Method to display the date and time**  public void Display() {  System.out.println(month + " " + date + ", " + year + " " + hrs + ":" + min + ":" + sec );  }  **// Main method**  public static void main (String[] args) {  **// Creating an object for default constructor**  Date def = new Date();    **// Creating objects for different setdate methods**  Date date1 = new Date();  Date date2 = new Date();  Date date3 = new Date();    **// Setting dates using different setdate methods**  date1.setdate(2024, 2, 9);  date2.setdate(2024, 2, 9, 10, 0);  date3.setdate(2024, 2, 9, 10, 10, 0);    **// Displaying dates**  def.Display();  date1.Display();  date2.Display();  date3.Display();  }  } |
| **RESULT:** | |
| **Program 2** | |
| **PROBLEM STATEMENT :** |  |
| **PROGRAM:** | import java.util.\*;  public class Lines {  **// Instance variables to store line parameters**  double slope, y\_int, x1, x2, y1, y2;  **// Default constructor initializes values for a default line**  Lines() {  x1 = 0;  y1 = 0;  x2 = 1;  y2 = 1;  slope = 1;  y\_int = 0;  }  **// Method to calculate slope between two points**  double getSlope() {  return (y2 - y1) / (x2 - x1);  }  **// Constructor for initializing line with slope-intercept form**  Lines(double m, double y\_int) {  this.slope = m;  this.y\_int = y\_int;  y\_int = y1 - slope \* x1**; // This line doesn't seem necessary and will not update y\_int**  **// Printing equation in slope-intercept form**  System.out.printf("The Equation of line in slope-intercept form is: y = %.2fx + %.2f\n", slope, y\_int);  }  **// Constructor for initializing line with two-point form**  Lines(double x1, double y1, double x2, double y2) {  this.x1 = x1;  this.x2 = x2;  this.y1 = y1;  this.y2 = y2;  **// Printing equation in two-point form**  System.out.printf("The Equation of line in two point form is : (y - %.2f)/%.2f = %.2f (x - %.2f)/(%.2f)\n", y1, y1 - y2, getSlope(), x1, x1 - x2);  }  **// Constructor for initializing line with one point and slope**  Lines(double x1, double y1, double m) {  this.x1 = x1;  this.y1 = y1;  this.slope = m;  **// Printing equation in one-point form**  System.out.printf("The Equation of line in one point form is : (y - %.2f) = %.2f (x - %.2f)\n", y1, getSlope(), x1);  }  public static void main(String[] args) {  Scanner sc = new Scanner(System.in);  double x1, x2, y1, y2, m, y\_int;  loop: while (true) {  **// Display the menu**  System.out.print("Menu:\n" + "1. Print two point form\n" + "2. Print one point form\n"  + "3. Print slope intercept form\n" + "4. Exit\n" + "Enter your choice: ");  **// Read the user's choice**  int choice = sc.nextInt();  switch (choice) {  case 1:  **// Read two points for the line**  System.out.println("Enter the x1,y1,x2,y2 values for line");  x1 = sc.nextDouble();  y1 = sc.nextDouble();  x2 = sc.nextDouble();  y2 = sc.nextDouble();  **// Create Line object with two-point form**  Lines l1 = new Lines(x1, y1, x2, y2);  break;  case 2:  **// Read one point and slope for the line**  System.out.println("Enter the values of x1,y1 and slope for line");  x1 = sc.nextDouble();  y1 = sc.nextDouble();  m = sc.nextDouble();  **// Create Line object with one-point form**  Lines l2 = new Lines(x1, y1, m);  break;  case 3:  **// Read slope and y-intercept for the line**  System.out.println("Enter values of slope and y-intercept for line");  m = sc.nextDouble();  y\_int = sc.nextDouble();  **// Create Line object with slope-intercept form**  Lines l3 = new Lines(m, y\_int);  break;  case 4:  **// Exit the loop and end the program**  System.out.println("Exiting...");  break loop;  default:  **// Handle invalid choice**  System.out.println("Invalid choice.");  }  }  sc.close(); **// Close scanner to prevent resource leak**  }  } |
| **RESULT:** | |
| **Program 3** | |
| **PROBLEM STATEMENT:** |  |
| **PROGRAM:** | import java.util.\*;  **// Define a class named Test**  public class Test {  **// Declare instance variables**  double base; **// base for power calculation**  int power; **// power for exponentiation**  int logBase; **// base for logarithmic calculation**  int argument; **// argument for logarithmic calculation**  **// Constructor for initializing default values**  Test(){  base = 2; **// Default base value**  power = 2; **// Default power value**  logBase = 2; **// Default logarithmic base value**  argument = 2; **// Default argument value**  }  **// Method to calculate power of a number**  public double calculate(double base, int power){  this.base = base;  this.power = power;  double value = Math.pow(base, power); **// Calculate base^power**  System.out.println(value); **// Print the result**  return value; **// Return the result**  }  **// Method to calculate logarithm of a number**  public double calculate(int logBase, int argument) {  this.logBase = logBase;  this.argument = argument;  double value = Math.log(argument) / Math.log(logBase); **// Calculate log\_base(argument)**  System.out.println(value); **// Print the result**  return value;  **// Return the result**  }  **// Method to display a message**  void display() {  System.out.println("Result after the Mathematical operation is: " );  }  }  **// Main class to execute the program**  class TestMain{  public static void main(String[] args) {  **// Create two Test objects**  Test obj1 = new Test();  Test obj2 = new Test();  **// Display message and calculate power for obj1**  obj1.display();  obj1.calculate(2.0, 4);  **// Display message and calculate logarithm for obj2**  obj2.display();  obj2.calculate(2, 4);  }  } |
| **RESULT:** | |
| **Program 4** | |
| **PROBLEM STATEMENT:** |  |
| **PROGRAM:** |  |
| **RESULT:** | |
| **Program 5** | |
| **PROBLEM STATEMENT:** |  |
| **PROGRAM:** |  |
| **RESULT:** | |
| **CONCLUSION:** |  |