

LAB CYCLE:1

DATE:13/03/23

EXPERIMENT NO:1

WELCOME JAVA

Aim: Write a java program to display message “Welcome Java”.

Algorithm:

Step 1: Define main class.

Step 2: Write the main method, public static void main(String args[])

Step 3: Inside the main method write the display statement and message using, System.out.println()

Step 4: End of the main class.

Result: Program executed successfully and output is verified.

LAB CYCLE:1

DATE:13/03/23

EXPERIMENT NO:2

AREA OF RECTANGLE

Aim: Create a class Rectangle with instance variable length and breadth. Define a method setdata for setting values of instance variables and a method getArea to return the area of Rectangle using the class. Find the area of the rectangle using the values length = 12.48 and breadth = 13.

Algorithm:

Step 1: Define a class Rectangle with instance variables length and breadth.

Step 2: Define methods setData() and getArea() as member functions of Rectangle

class to set the values of instance variables and to calculate the area respectively.

Step 3: Create an object to access the members of Rectangle class.

Step 4: Invoke the method setData(12.48,13).

Step 5: Invoke method getArea().

Step 6: Print area of the rectangle.

Result: Program executed successfully and output is verified.

LAB CYCLE:1

DATE:13/03/23

EXPERIMENT NO:3

EVEN OR ODD

Aim: Write a program to read integer from keyboard and check whether the number is even or odd.

Algorithm:

Step 1: Import the Java 'Scanner' class.

Step 2: Read an integer (num) from user.

Step 3: If $\text{num} \% 2 == 0$, then print "num is even".

Step 4: Else, print "num id odd".

Result: Program executed successfully and output is verified.

LAB CYCLE:1

DATE:20/03/23

EXPERIMENT NO:4

CONSTRUCTOR IMPLEMENTATION

Aim: Define a class 'product' with data members pcode, pname and price.
Create three objects of the class and find the product having the lowest price.

Algorithm:

- Step 1: Define a class Product with data members pcode,pname,price.
- Step 2: Read the details (product code,name,price) of three products from the user.
- Step 3: Create three objects for Product class and initialize it with the details read from the user.
- Step 4: Compare the prices of three products.
- Step 5: Print the product which has the lowest price.

Result: Program executed successfully and output is verified.

LAB CYCLE:1

DATE:20/03/23

EXPERIMENT NO:5

MATRIX ADDITION

Aim: Write a java program to read two matrices from the console and perform matrix addition.

Algorithm:

Step 1: Import the Scanner class.

Step 2: Define the class and read the count of rows and columns (r,c).

Step 3: Declare two matrices and read the values of the matrices,A[r][c],B[r][c].

Step 4: Declare a new matrix to store the sum of other two matrices,C[r][c].

Step 5: Print matrix C[r][c].

Result: Program executed successfully and output is verified.

LAB CYCLE:1

DATE:20/03/23

EXPERIMENT NO:6

COMPLEX NUMBER ADDITION

Aim: Write a program to perform complex number addition.

Algorithm:

Step 1: Import the Scanner class.

Step 2: Define the class and read the real and imaginary part of two complex numbers.

Step 3: Create two objects to access the complex numbers and temp object to store the sum.

Step 4: Create static method with class type to perform the addition, add().

Step 5: Return temp.

Step 6: Call the class method add().

Step 7: Print the sum.

Result: Program executed successfully and output is verified.

LAB CYCLE:1

DATE:20/03/23

EXPERIMENT NO:7

SYMMETRIC MATRIX

Aim: Write a java program to read a matrix from the console and check whether it is symmetric or not.

Algorithm:

Step 1: Import the Scanner class.

Step 2: Define the class and read the count of rows, rows and columns, cols.

Step 3: Read the values of matrix.

Step 4: Check if number of rows and columns are equal, then read the values of matrix.

Step 5: Check each element with transpose of matrix are equal, then the matrix is symmetric.

Step 6: If not equal then matrix is not symmetric.

Result: Program executed successfully and output is verified.

LAB CYCLE:1

DATE:24/03/23

EXPERIMENT NO:8

LEAP YEAR

Aim: Write a program to print the leap years within the given range.

Algorithm:

Step 1: Import the Scanner class.

Step 2: Define the class and read the start and end range.

Step 3: Check for the leap year condition also for century year.

```
if(i%4==0||(i%100!=0)&&(i%400==0)) then  
    print "Leap year".
```

Step 4: Print the leap year within the range.

Result: Program executed successfully and output is verified.

LAB CYCLE:1

DATE:27/03/23

EXPERIMENT NO:9

NESTED CLASS

Aim: Create CPU with attribute price. Create an inner class processor (no of cores, manufacturer) and static nested class RAM (memory, manufacturer). Create an object of CPU and print information of processor and RAM.

Algorithm:

Step 1: Import the Scanner class.

Step 2: Define the class CPU with data member price, define an inner class Processor with attributes ncores and manft and define a static class RAM with attributes manf and mm.

Step 3: Create the object for CPU and call the Processor and its methods.

Step 4: Create an object for static class RAM and call its methods .

Step 5: Print the details of Processor and RAM.

Result: Program executed successfully and output is verified.