**LAB CYCLE-4**

Submitted by:

Nithin Raj

S2MCA

**PROGRAM NO-1**

**AIM:**

Create a Graphics package that has classes and interfaces for figures Rectangle, Triangle, Square and Circle. Test the package by finding the area of these figures.

**ALGORITHM:**

STEP 1: Start

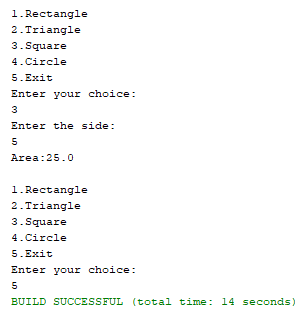
**CODE:**

| Graphics.Shapes.java | package Graphics;  //Create a Graphics package that has classes and interfaces for figures Rectangle, Triangle,  //Square and Circle. Test the package by finding the area of these figures.  interface figures  {  void Rectangle(double a,double b);  void Triangle(double a,double b);  void Square(double a);  void Circle(double a);  }  public class Shapes implements figures  {  @Override  public void Rectangle(double a,double b)  {  System.out.println("Area:"+(a\*b));  }  @Override  public void Triangle(double a,double b)  {  System.out.println("Area:"+((a\*b)/2));  }  @Override  public void Square(double a)  {  System.out.println("Area:"+(a\*a));  }  @Override  public void Circle(double a)  {  double pi = 3.14;  System.out.println("Area:"+(pi\*a\*a));  }  public static void main(String[] args) {    }  } |
| --- | --- |
| CO4Q1.java | package javaprograms2;  //Create a Graphics package that has classes and interfaces for figures Rectangle, Triangle,  //Square and Circle. Test the package by finding the area of these figures.  import Graphics.Shapes;  import java.util.Scanner;  public class CO4Q1  {  public static void main(String[] args)  {  int ch;  double l,b;  Shapes ob = new Shapes();  Scanner sc = new Scanner(System.in);  do  {  System.out.println("\n1.Rectangle\n2.Triangle\n3.Square\n4.Circle\n5.Exit\nEnter your choice:");  ch = sc.nextInt();  switch(ch)  {  case 1:System.out.println("Enter the length and breadth:");  l = sc.nextDouble();  b = sc.nextDouble();  ob.Rectangle(l, b);  break;  case 2:System.out.println("Enter the breadth and height:");  l = sc.nextDouble();  b = sc.nextDouble();  ob.Triangle(l, b);  break;  case 3:System.out.println("Enter the side:");  l = sc.nextDouble();  ob.Square(l);  break;  case 4:System.out.println("Enter the radius:");  l = sc.nextDouble();  ob.Circle(l);  break;  case 5:System.exit(0);  break;  default:System.out.println("Invalid choice");  }  }while(true);  }  } |

**RESULT:**

The above program is successfully executed and the output is obtained.

**OUTPUT:**

****

**PROGRAM NO-2**

**AIM:**

Create an Arithmetic package that has classes and interfaces for the 4 basic arithmetic operations. Test the package by implementing all operations on two given numbers

**ALGORITHM:**

STEP 1: Start

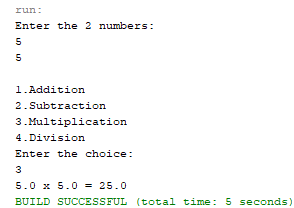
**CODE:**

| Arithmetic.operations.java | package arithmetic;  interface basic  {  void addition(double a,double b);  void subtraction(double a,double b);  void multiplication(double a,double b);  void division(double a,double b);  }  public class operations implements basic  {  @Override  public void addition(double a, double b) {  System.out.println(a+" + "+b+" = "+(a+b));  }  @Override  public void subtraction(double a, double b) {  System.out.println(a+" - "+b+" = "+(a-b));  }  @Override  public void multiplication(double a, double b) {  System.out.println(a+" x "+b+" = "+(a\*b));  }  @Override  public void division(double a, double b) {  System.out.println(a+" / "+b+" = "+(a/b));  }    } |
| --- | --- |
| CO4Q2.java | package c04q2;  //Create an Arithmetic package that has classes and interfaces for the 4 basic arithmetic  //operations. Test the package by implementing all operations on two given numbers  import arithmetic.operations;  import java.util.Scanner;  public class C04Q2  {  public static void main(String[] args)  {  double n1,n2;  int ch;  operations ob = new operations();  Scanner sc = new Scanner(System.in);  System.out.println("Enter the 2 numbers:");  n1 = sc.nextDouble();  n2 = sc.nextDouble();  System.out.println("\n1.Addition\n2.Subtraction\n3.Multiplication\n4.Division\nEnter the choice:");  ch = sc.nextInt();  switch(ch)  {  case 1:ob.addition(n1,n2);  break;  case 2:ob.subtraction(n1,n2);  break;  case 3:ob.multiplication(n1,n2);  break;  case 4:ob.division(n1,n2);  break;  default:System.out.println("Invalid choice");  }  }    } |

**RESULT:**

The above program is successfully executed and the output is obtained.

**OUTPUT:**

****

**PROGRAM NO-3**

**AIM:**

Write a user defined exception class to authenticate the user name and password.

**ALGORITHM:**

STEP 1: Start

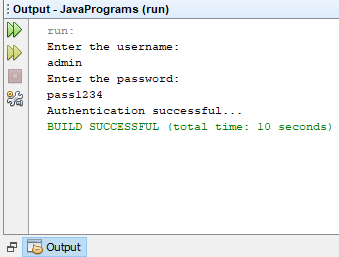
**CODE:**

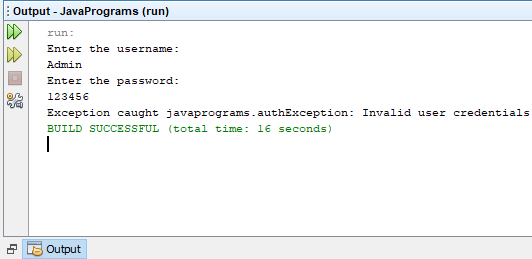
| CO3Q3.java | package javaprograms;  //Write a user defined exception class to authenticate the user name and password  import java.util.Scanner;  class authException extends Exception  {  public authException(String s) {  super(s);  }    }  public class CO4Q3  {  public static void main(String[] args) {  String username = "admin";  String passcode = "pass1234";  String user\_name,password;  Scanner sc = new Scanner(System.in);  try  {  System.out.println("Enter the username:");  user\_name = sc.nextLine();  // sc.nextLine();  System.out.println("Enter the password:");  password = sc.nextLine();  if(username.equals(user\_name) && passcode.equals(password))  {  System.out.println("Authentication successful...");  }  else  throw new authException("Invalid user credentials");    }  catch(authException e)  {  System.out.println("Exception caught "+e);  }  }  } |
| --- | --- |

**RESULT:**

The above program is successfully executed and the output is obtained.

**OUTPUT:**

****

****

**PROGRAM NO-4**

**AIM:**

Find the average of N positive integers, raising a user defined exception for each negative input.

**ALGORITHM:**

STEP 1: Start

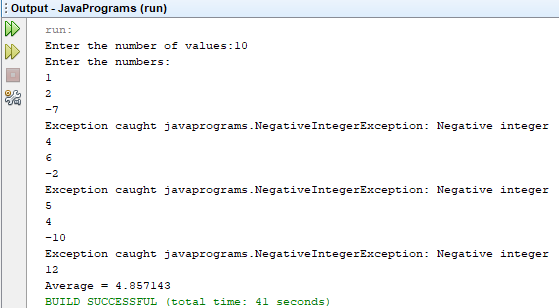
**CODE:**

| CO4Q4.java | package javaprograms;  //Find the average of N positive integers, raising a user defined exception for each negative  //input  import java.util.Scanner;  class NegativeIntegerException extends Exception  {  public NegativeIntegerException(String s)  {  super(s);  }  }  public class CO4Q4 {  public static void sample()  {  try {  int n,count=0;  float num[];  float total=0;  Scanner sc = new Scanner(System.in);  System.out.print("Enter the number of values:");  n = sc.nextInt();  num = new float[n];  System.out.println("Enter the numbers:");  for(int i=0;i<n;i++)  {  num[i] = sc.nextInt();  try{  if(num[i]<0)  {  throw new NegativeIntegerException("Negative integer");  }  else  {  total += num[i];  count++;  }  }catch(NegativeIntegerException e)  {  System.out.println("Exception caught "+e);  }  }  System.out.println("Average = "+(total/count));  } catch (Exception e) {  System.out.println("Exception caught "+e);  }  }  public static void main(String[] args) {  try {  sample();  } catch (Exception e) {  }  }  } |
| --- | --- |

**RESULT:**

The above program is successfully executed and the output is obtained.

**OUTPUT:**

****

**PROGRAM NO-5**

**AIM:**

Define 2 classes; one for generating multiplication table of 5 and other for displaying first N prime numbers. Implement using threads. (Thread class)

**ALGORITHM:**

STEP 1: Start

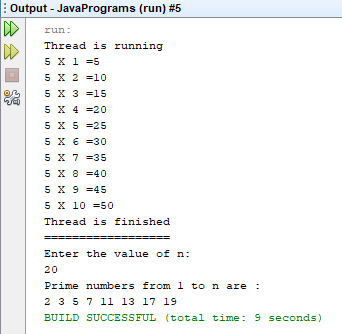
**CODE:**

| CO4Q5.java | //Define 2 classes; one for generating multiplication table of 5 and other for displaying first  //N prime numbers. Implement using threads. (Thread class)  package javaprograms;  import java.util.Scanner;  import java.util.logging.Level;  import java.util.logging.Logger;  class MultiplicationTable extends Thread  {  public void run()  {  System.out.println("Thread is running");  for(int i=1;i<=10;i++)  {  try  {  System.out.println("5 X "+i+" ="+(5\*i));  Thread.sleep(200);  } catch (InterruptedException ex) {    }  }  System.out.println("Thread is finished");  System.out.println("==================");  }  }  class PrimeNumbers extends Thread  {  public void run()  {  Scanner sc = new Scanner(System.in);  int i =0;  int num =0;  String primeNumbers = "";  System.out.println("Enter the value of n:");  int n = sc.nextInt();  for (i = 1; i <= n; i++)  {  int counter=0;  for(num =i; num>=1; num--)  {  if(i%num==0)  {  counter = counter + 1;  }  }  if (counter ==2)  {  primeNumbers = primeNumbers + i + " ";  }  }  System.out.println("Prime numbers from 1 to n are :");  System.out.println(primeNumbers);  }  }  public class CO4Q5 {  public static void main(String[] args) throws InterruptedException {  MultiplicationTable MTOb = new MultiplicationTable();  MTOb.start();  Thread.sleep(5000);  PrimeNumbers PNob = new PrimeNumbers();  PNob.start();  }  } |
| --- | --- |

**RESULT:**

The above program is successfully executed and the output is obtained.

**OUTPUT:**

****

**PROGRAM NO-6**

**AIM:**

Define 2 classes; one for generating Fibonacci numbers and other for displaying even numbers in a given range. Implement using threads. (Runnable Interface)

**ALGORITHM:**

STEP 1: Start

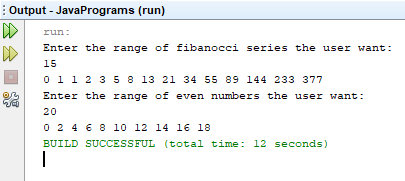
**CODE:**

| CO4Q6.java | //Define 2 classes; one for generating Fibonacci numbers and other for displaying even  //numbers in a given range. Implement using threads. (Runnable Interface)  package javaprograms;  import java.util.Scanner;  class Fibonacci implements Runnable  {  int n,first,second,t;  String str;    public Fibonacci(int num)  {  n = num;  first = 0;  second = 1;  }    @Override  public void run()  {  str = first+" "+second;  for(int i=0;i<=n-3;i++)  {  t = first + second;  first = second;  second = t;  str += " "+t;  }  System.out.println(str);  }    }  class Even implements Runnable  {  int n;  String str;  public Even(int n)  {  this.n = n;  str = "";  }  @Override  public void run()  {  for(int i=0;i<n;i=i+2)  if(i%2==0)  {  str+=i+" ";  }  System.out.println(str);  }    }  public class CO4Q6 {  public static void main(String[] args) throws InterruptedException {  int n1,n2;  Scanner sc = new Scanner(System.in);  System.out.println("Enter the range of fibanocci series the user want:");  n1 = sc.nextInt();  Fibonacci fibob = new Fibonacci(n1);  Thread th = new Thread(fibob);  th.start();  Thread.sleep(400);  System.out.println("Enter the range of even numbers the user want:");  n2 = sc.nextInt();  Even evob = new Even(n2);  Thread th2 = new Thread(evob);  th2.start();  }  } |
| --- | --- |

**RESULT:**

The above program is successfully executed and the output is obtained.

**OUTPUT:**

****

**PROGRAM NO-7**

**AIM:**

Producer/Consumer using ITC

**ALGORITHM:**

STEP 1: Start

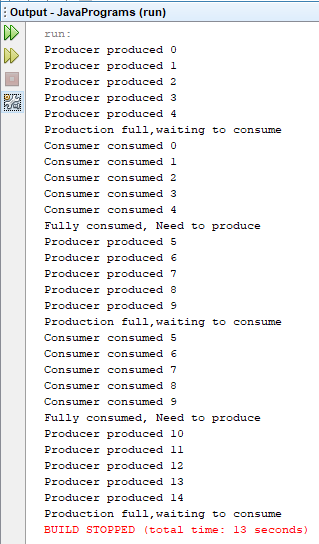
**CODE:**

| CO4Q7.java | package javaprograms;  //Producer/Consumer using ITC  import java.util.ArrayList;  import java.util.List;  class Producer implements Runnable  {  List<Integer> flist;  int max\_size = 5;  int i=0;  Producer(List<Integer> flist)  {  this.flist = flist;  }  @Override  public void run()  {  while(true)  {  try  {  produce(i++);  } catch (Exception e)  {  System.out.println("Intteruption "+e);  }  }  }  public void produce(int i) throws InterruptedException  {  synchronized (flist)  {  while(flist.size()==max\_size)  {  System.out.println("Production full,waiting to consume");  flist.wait();  }  }  synchronized(flist)  {  System.out.println("Producer produced "+i);  flist.add(i);  flist.notify();  }  }    }  class Consumer implements Runnable  {  List<Integer> flist;  Consumer(List<Integer> flist)  {  this.flist = flist;  }  @Override  public void run()  {  while(true)  {  try  {  consume();  } catch (Exception e)  {  System.out.println("Exception "+e);  }  }  }  public void consume() throws InterruptedException  {  synchronized (flist)  {  while(flist.isEmpty())  {  System.out.println("Fully consumed, Need to produce");  flist.notify();  Thread.sleep(500);  flist.wait();  }  }  synchronized(flist)  {  Thread.sleep(1000);  System.out.println("Consumer consumed "+flist.remove(0));  }  }    }  public class CO4Q7 {  public static void main(String[] args)  {  List<Integer> flist = new ArrayList<Integer>();  Thread th1 = new Thread(new Producer(flist));  Thread th2 = new Thread(new Consumer(flist));  th1.start();  th2.start();  }  } |
| --- | --- |

**RESULT:**

The above program is successfully executed and the output is obtained.

**OUTPUT:**

****

**PROGRAM NO-8**

**AIM:**

Program to create a generic stack and do the Push and Pop operations.

**ALGORITHM:**

STEP 1: Start

**CODE:**

| CO4Q8.java |  |
| --- | --- |