 GALGOTIAS UNIVERSITY

**Plot No.2, Sector -17 A, Yamuna Expressway,**

**Greater Noida, Gautam Buddha Nagar, U.P., India**

**SCHOOL OF COMPUTING SCIENCE & ENGINEERING**

**“LAB PRACTICAL FILE”**

##### **Course Name: JAVA PROGRAMMING**

**Course Code:  BCSE2333**

**School: SCSE**

**Program: B. Tech**

**Year: 2nd Semester: 3**

**Session: 2022-2023**

|  |  |
| --- | --- |
| **Submitted By:** | **Submitted To:** |
| **NEERAJ SINGH**  **21SCSE1011675**  **SEC-6- P1** | **Sachin Minocha** **SIR** |

**Index**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Title of Lab Experiments | DATE | SIGN |
| 1 | a) Write a JAVA program to print “Hello World.”  b)WAP in java to print the values of various primitive data types. | 1/12/2022 |  |
| 2 | WAP in java to demonstrate operator precedence. | 1/12/2022 |  |
| 3 | WAP in java to create a class Addition and a method add() to add two numbers. | 1/12/2022 |  |
| 4 | Write a program that uses length property for displaying any number of command line arguments. | 1/12/2022 |  |
| 5 | WAP in java to find the average of N numbers. | 1/12/2022 |  |
| **6** | WAP in java to find out factorial of a number | 2/12/2022 |  |
| 7 | WAP in java to find out the Fibonacci series | 2/12/2022 |  |
| 8 | WAP in java to sort elements in 1D array in ascending order. | 2/12/2022 |  |
| 9 | WAP in java to perform matrix addition using two 2D arrays. | 2/12/2022 |  |
| 10 | WAP in java to find out the number of characters in a string. | 2/12/2022 |  |
| 11 | WAP in java that implements method overloading. | 2/12/2022 |  |
| 12 | Create a class Shape and override area() method to calculate area of rectangle, square and circle | 3/12/2022 |  |
| 13 | WAP in java to demonstrate the properties of public private and protected variables and methods (with package). | 3/12/2022 |  |
| 14 | WAP that illustrates method overriding | 3/12/2022 |  |
| 15 | a) WAP in java demonstrating arithmetic exception and Array Out Of Bounds Exception using try catch block  b) WAP in java to demonstrate the use of nested try block and nested catch block. | 3/12/2022 |  |
| 16 | Write a program to count the number of times a character appears in a File.  [Note : The character check is case insensitive... i.e, ‘a’ and ‘A’ are considered to be the same] | 3/12/2022 |  |
| 17 | WAP in java in two different ways that creates a thread by (i) extending thread class (ii) implementing runnable interface which displays 1st 10 natural numbers. |  |  |
| 18 | a) WAP in java that connects to a database using JDBC & insert values into table.  b) WAP to connect to a database using JDBC and delete values from table. |  |  |

# 



# List of Programs

|  |  |
| --- | --- |
| Sr. No. | Title of Lab Experiments |
| 1 | a) Write a JAVA program to print “Hello World.”  b)WAP in java to print the values of various primitive data types.  public class jt\_06\_helloWorld {        public static void main(String[] args){          System.out.println("Hello Neeraj Singh");      }    }    **b)**  public class jt\_07\_print\_primitive {      byte a;      short b;      int c=9;      long d;      float e;      double f;      char g;      boolean h;      public static void main(String[] args){          jt\_07\_print\_primitive neeraj=new jt\_07\_print\_primitive();          System.out.println(neeraj.a);          System.out.println(neeraj.b);          System.out.println(neeraj.c);          System.out.println(neeraj.d);          System.out.println(neeraj.e);          System.out.println(neeraj.f);          System.out.println(neeraj.g);          System.out.println(neeraj.h);      }  } |
| 2 | WAP in java to demonstrate operator precedence.  public class jt\_08\_operator\_precedence {      public static void main(String [] args){          int a=2,b=4,c=3,d=5,result;          result=b+(a^c)-d;          System.out.println(result);      }  } |
| 3 | WAP in java to create a class Addition and a method add() to add two numbers.  public class jt\_09\_Addition {      int a;      int b;      int add(int a,int b){          int sum;          sum=a+b;          return sum;      }      public static void main(String [] args){          jt\_09\_Addition sum1=new jt\_09\_Addition();          sum1.a=45;          sum1.b=34;          System.out.println("sum of two numbers is : "+sum1.add(sum1.a,sum1.b));      }  } |
| 4 | Write a program that uses length property for displaying any number of command line arguments  public class jt\_010\_CommandLine {      public static void main(String [] args){          int sum=0;            for(int i=0;i<args.length;i++){              sum+=Integer.parseInt(args[i]);          }          System.out.println("sum of all elements passed in command line is : "+sum);      }  }  . |
| 5 | **WAP in java to find the average of N numbers.**  **import java.util.Scanner;**  **public class jt\_011\_Average {**  **public static void main(String [] args){**  **System.out.println("enter the numebr of inputs : ");**  **Scanner sc=new Scanner(System.in);**  **float sum=0;**  **float num;**  **int length=sc.nextInt();**  **for(int i=0;i<length;i++){**  **num =sc.nextInt();**  **sum+=num;}**  **System.out.println("average of numbers is :"+sum/length);**  **}**  **}** |
| **6** | WAP in java to find out factorial of a number  import java.util.Scanner;  public class jt\_012\_Factorial {      public static int factorial(int n) {          if (n == 1 || n == 0) {              return 1;          }          return n \* factorial(n - 1);      }      public static void main(String [] args){          Scanner sc=new Scanner(System.in);          System.out.println("Neeraj Singh sec-6");;          System.out.println("enter a number to get its factorail: ");          int num=sc.nextInt();          System.out.println(factorial(num));      }  } |
| 7 | WAP in java to find out the Fibonacci series  import java.util.Scanner;  public class jt\_013\_fibonacci {            public static void main(String args[]) {              Scanner sc=new Scanner(System.in);              System.out.println("enter number to find fibonacci serires till there: ");              int num=sc.nextInt();              int n1 = 0, n2 = 1, n3, i;              System.out.print(n1 + " " + n2);              for (i = 2; i <num; ++i)              {                  n3 = n1 + n2;                  System.out.print(" " + n3);                  n1 = n2;                  n2 = n3;              }              System.out.println("\nNeeraj singh sec-6");          }  } |
| 8 | WAP in java to sort elements in 1D array in ascending order.  import java.util.Scanner;  public class jt\_014\_sorting {      public static int[] Sort(int length,int arr[]){          for(int i=0;i<length;i++){              for(int j=i+1;j<length;j++){                  if(arr[i]>arr[j]){                      int temp=arr[i];                      arr[i]=arr[j];                      arr[j]=temp;                  }              }          }          return arr;      }      public static void main (String [] args){          Scanner sc = new Scanner(System.in);          System.out.println("enter length of array : ");          int length=sc.nextInt();          int [] arr=new int[length];          for(int i=0;i<length;i++){              arr[i]=sc.nextInt();          }          Sort(length, arr);          System.out.println("Neeraj singh sec-6");          for(int i=0;i<length;i++){              System.out.print(arr[i]+" ");          }      }  } |
| 9 | WAP in java to perform matrix addition using two 2D arrays.  import java.util.Scanner;  public class jt\_015\_matrix\_addition {      public static void add(int[][] mat1, int[][] mat2,int row,int col){          int [][] mat3=new int[row][col];          for (int i = 0; i < row; i++) {              for (int j = 0; j < col; j++) {                  mat3[i][j]=mat1[i][j]+mat2[i][j];              }          }          for (int i = 0; i < row; i++) {              for (int j = 0; j < col; j++) {                  System.out.print(mat3[i][j] +" ");              }              System.out.println();          }      }      public static void main(String [] args){          Scanner sc=new Scanner(System.in);          System.out.println("enter row and column of a matrix : ");          int row=sc.nextInt();          int col=sc.nextInt();          int[][] mat1=new int[row][col];          int[][] mat2 = new int[row][col];          int[][] mat3 = new int[row][col];          System.out.println("enter elements for matrix 1");;          for(int i=0;i<row;i++){              for(int j=0;j<col;j++){                  mat1[i][j]=sc.nextInt();              }          }          System.out.println("enter elements for matrix 2");          ;          for (int i = 0; i < row; i++) {              for (int j = 0; j < col; j++) {                  mat2[i][j] = sc.nextInt();              }          }          add(mat1, mat2, row, col);          System.out.println("from neeraj singh-sec-6");      }  } |
| 10 | WAP in java to find out the number of characters in a string.  import java.util.Scanner;  public class jt\_016\_character {      public static int character(String str,int length){          int count=0;          for(int i=0;i<length;i++){              if((str.charAt(i) >= 65 && str.charAt(i)<=90) || (str.charAt(i)>=97 && str.charAt(i)<=122)){                  count++;              }          }          return count;      }      public static void main(String[] args){          Scanner sc=new Scanner(System.in);          System.out.print("enter a string :");          String str=sc.nextLine();          int strLen=str.length();          System.out.println("Nunber of character in tn this string is : "+character(str, strLen));      }  } |
| 11 | WAP in java that implements method overloading.  public class jt\_017\_method\_overloading {      public static int sum(int x,int y){          return x+y;      }      public static int sum(int x, int y,int z) {          return x + y+ z;      }      public static void main(String [] args){          System.out.println("two parameter sum : "+sum(1,3));          System.out.println("three parameter sum : "+sum(1, 3,7));      }  } |
| 12 | Create a class Shape and override area() method to calculate area of rectangle, square and circle  class shape {      void area(float x) {      System.out.println("The area of the square is "+Math.pow(x, 2)+" sq units");  }      void area(float x, float y) {          System.out.println("The area of the rectangle is " + x \* y + " sq units");      }      void area(double x) {          double z = 3.14 \* x \* x;          System.out.println("The area of the circle is " + z + " sq units");      }    }  public class jt\_018\_Shape{      public static void main(String[] args){          shape ob=new shape();          System.out.println("\nneeraj singh sec-6\n");          ob.area(5);          ob.area(5);          ob.area(11, 12);          ob.area(2.5);      }  } |
| 13 | WAP in java to demonstrate the properties of public private and protected variables and methods (with package).  **1st file**  package p1;  public class jt\_019\_accessModifier {      public void msgPublic(){          System.out.println("this is public msg");      }      private void msgPrivate(){          System.out.println("this is private msg");      }      protected void msgProtected(){          System.out.println("this is protected msg");      }      void msgDefault(){          System.out.println("this is default msg");      }      public static void main(String [] args){      }  }  **2nd file**  import p1.\*;  public class importPackage {      public static void main(String [] args){          jt\_019\_accessModifier ob=new jt\_019\_accessModifier();          ob.msgDefault();          // ob.msgPrivate();          ob.msgPublic();          ob.msgProtected();      }  } |
| 14 | WAP that illustrates method overriding  class neeraj1{      public static void say(){          System.out.println("Neeraj singh");      }  }  class neeraj2 extends neeraj1{      public static void say(){          System.out.println("neeraj is override");      }  }  public class jt\_020\_overriding {      public static void main (String [] args){          neeraj1 obj1=new neeraj1();          obj1.say();          neeraj2 obj2=new neeraj2();          obj2.say();      }  } |
| 15 | a) WAP in java demonstrating arithmetic exception and ArrayOutOf Bounds Exception using try catch block  b) WAP in java to demonstrate the use of nested try block and nested catch block.  public class jt\_021\_tryCatch {      public static void main(String [] args){          try{              int a=7;              int b=0;              int c=a/b;              System.out.println("division is: "+c);          }catch(Exception e){              System.out.println(e);          }          finally{              try{              int[] arr={1,2,3};                System.out.println(arr[4]);              }catch(Exception e){              System.out.println(e);          }          }      }}    **b)**  public class jt\_022\_nestedTry {      public static void main(String[] args) {          try{              System.out.println("nested try catch neeraj singh");              try {              int a = 7;              int b = 0;              int c = a / b;              System.out.println("division is: " + c);          } catch (Exception e) {              System.out.println(e);          } finally {              try {                  int[] arr = { 1, 2, 3 };                  System.out.println(arr[4]);              } catch (Exception e) {                  System.out.println(e);              }          }          }catch(Exception e){              System.out.println(e);          }      }  } |
| 16 | Write a program to count the number of times a character appears in a File.  [Note : The character check is case insensitive... i.e, ‘a’ and ‘A’ are considered to be the same]  import java.util.\*;  public class jt\_023\_countChar {      public static void main(String [] args){          Set<String> hash\_set = new HashSet <String>();          Scanner sc=new Scanner(System.in);          System.out.println("enter a string: ");          String str=sc.nextLine();          str=str.toLowerCase();          for (int i = 0; i < str.length(); i++) {              if ( (str.charAt(i) >= 97 && str.charAt(i) <= 122)) {                  String st=String.valueOf(str.charAt(i));                  hash\_set.add(st);              }          }          System.out.println("the number of time a unique character appear is : "+hash\_set.size());      }  } |