

MALLA REDDY COLLEGE OF ENGINEERING & TECHNOLOGY (autonomous institution – ugc, govt. of india)



# **Department of CSE** (Emerging Technologies)



## **Object Oriented Programming through** JAVA

LAB MANUAL

## **Object Oriented Programming through** JAVA



## LABORATORY MANUAL

B.TECH (R-20 Regulation) (II YEAR – II SEM) (2021-22)

## DEPARTMENT OF CSE (EMERGING TECHNOLOGIES)



# MALLA REDDY COLLEGE OF ENGINEERING & TECHNOLOGY

(Autonomous Institution – UGC, Govt. of India)

Recognized under 2(f) and 12 (B) of UGC ACT 1956 (Affiliated to JNTUH, Hyderabad, Approved by AICTE - Accredited by NBA & NAAC – 'A' Grade - ISO 9001:2015 Certified) Maisammaguda, Dhulapally (Post Via. Hakimpet), Secunderabad – 500100, Telangana State, India

#### **Department of Computer Science and Engineering**

#### Vision

To acknowledge quality education and instill high patterns of discipline making the students technologically superior and ethically strong which involves the improvement in the quality of life in human race.

#### Mission

- To achieve and impart holistic technical education using the best of infrastructure, outstanding technical and teaching expertise to establish the students in to competent and confident engineers.
- Evolving the center of excellence through creative and innovative teaching learning practices for promoting academic achievement to produce internationally accepted competitive and world class professionals.

#### **PROGRAM OUTCOMES (POs)**

#### Engineering Graduates will be able to:

- 1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- 2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- 3. **Design / development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, andthe cultural, societal, and environmental considerations.
- 4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- 5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
- 6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
- 7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- 8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- 9. **Individual and team work**: Function effectively as an individual, and as a memberor leader in diverse teams, and in multidisciplinary settings.
- 10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- 11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multi disciplinary environments.
- 12. Life- long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

#### Lab Objectives:

- To prepare students to become familiar with the Standard Java technologies of J2SE
- To prepare students to excel in Object Oriented programming and to succeed as a Java Developer through global rigorous education.
- To provide Students with a solid foundation in OOP fundamentals required to solve programming problems and also to learn Advanced Java topics like J2ME, J2EE, JSP, JavaScript
- To train Students with good OOP programming breadth so as to comprehend, analyze, design and create novel products and solutions for the real life problems.
- To inculcate in students professional and ethical attitude, multidisciplinary approach and an ability to relate java programming issues to broader application context.
- To provide student with an academic environment aware of excellence, written ethical codes and guidelines and lifelong learning needed for a successful professional career.

#### Lab Outcomes:

Upon successful completion of this course, the students will be able to:

- Able to analyze the necessity for Object Oriented Programming paradigm and over structured programming and become familiar with the fundamental concepts in OOP.
- Demonstrate an ability to design and develop java programs, analyze, and interpret object oriented data and report results.
- Demonstrate an ability to design an object oriented system, AWT components or multithreaded process as per needs and specifications.
- Demonstrate an ability to visualize and work on laboratory and multidisciplinary tasks like console and windows applications both for standalone and Applets programs

#### Introduction about lab

System configurations are as follows:

- Hardware / Software's installed: Intel® CORE™ i3-3240 CPU@3.40GHZRAM: 4GB / C, C++ Compiler, JAVA JDK 1.8, EditPlus.
- Systems are provided for students in the 1:1 ratio.
- Systems are assigned numbers and same system is allotted for students when they do the lab.
- All Systems are configured in LINUX, it is open source and students can use any different programming environments through package installation.

#### **Guidelines to students**

#### A. Standard operating procedure

- a) Explanation on today's experiment by the concerned faculty using PPT covering the following aspects:
- 1) Name of the experiment
- 2) Aim
- 3) Software/Hardware requirements
- 4) Writing the java programs by the students
- 5) Commands for executing programs

#### Writing of the experiment in the Observation Book

The students will write the today's experiment in the Observation book as per the following format:

- a) Name of the experiment
- b) Aim
- c) Writing the program
- d) Viva-Voce Questions and Answers
- e) Errors observed (if any) during compilation/execution

Signature of the Faculty

#### **B.** Guide Lines to Students in Lab

#### Disciplinary to be maintained by the students in the Lab

• Students are required to carry their lab observation book and record book with completed experiments while entering the lab.

- Students must use the equipment with care. Any damage is caused student ispunishable.
- Students are not allowed to use their cell phones/pen drives/ CDs in labs.
- Students need to be maintain proper dress code along with ID Card

• Students are supposed to occupy the computers allotted to them and are not supposed to talk or make noise in the lab.

• Students, after completion of each experiment they need to be updated in observation notes and same to be updated in the record.

• Lab records need to be submitted after completion of experiment and get itcorrected with the concerned lab faculty.

• If a student is absent for any lab, they need to be completed the same experiment in the free time before attending next lab.

#### Steps to perform experiments in the lab by the student

Step1: Students have to write the date, aim and forthat experiment in the observation book.Step2: Students have to listen and understand the experiment explained by the faculty and note down the important points in the observation book.

Step3: Studentsneed to write procedure/algorithm in the observation book.

**Step4:** Analyze and Develop/implement the logic of the program by the student in respective platform

**Step5:** After approval of logic of the experiment by the faculty then the experiment has to be executed on the system.

**Step6:** After successful execution the results are to be shown to the faculty andnoted the same in the observation book.

**Step7:** Students need to attend the Viva-Voce on that experiment and write the same in the observation book.

**Step8:** Update the completed experiment in the record and submit to the concerned faculty in-charge.

#### Instructions to maintain the record

- Before start of the first lab they have to buy the record and bring the record to thelab.
- Regularly (Weekly) update the record after completion of the experiment and get itcorrected with concerned lab in-charge for continuous evaluation. In case the record is lost inform the same day to the faculty in charge and get the new record within 2 days the record has to be submitted and get it corrected by the faculty.
- If record is not submitted in time or record is not written properly, the evaluation marks (5M) will be deducted.

#### Awarding the marks for day to day evaluation

Total marks for day to day evaluation is 15 Marks as per Autonomous (JNTUH). These 15 Marks are distributed as:

Regularity	3 Marks
Program written	3 Marks
Execution & Result	3 Marks
Viva-Voce	3 Marks
Dress Code	3 Marks

#### **Allocation of Marks for Lab Internal**

Total marks for lab internal are 30 Marks as per Autonomous (JNTUH.)

These 30 Marks are distributed as:

Average of day to day evaluation marks: 15 Marks

Lab Mid exam: 10 Marks

VIVA & Observation: 5 Marks

#### **Allocation of Marks for Lab External**

Total marks for lab Internal and External are 70 Marks as per Autonomous / (JNTUH).

These 70 External Lab Marks are distributed as:

Program Written	30 Marks
Program Execution and Result	20 Marks
Viva-Voce	10 Marks
Record	10 Marks

#### **C. General laboratory instructions**

- 1. Students are advised to come to the laboratory at least 5 minutes before (to thestarting time), those who come after 5 minutes will not be allowed into the lab.
- 2. Plan your task properly much before to the commencement, come prepared to the lab with the synopsis / program / experiment details.
- 3. Student should enter into the laboratory with:
- a. Laboratory observation notes with all the details (Problem statement, Aim, Algorithm, Procedure, Program, Expected Output, etc.,) filled in for the lab session.
- b. Laboratory Record updated up to the last session experiments and other utensils (if any) needed in the lab.
- c. Proper Dress code and Identity card.
- 4. Sign in the laboratory login register, write the TIME-IN, and occupy the computer system allotted to you by the faculty.
- 5. Execute your task in the laboratory, and record the results / output in the labobservation note book, and get certified by the concerned faculty.
- 6. All the students should be polite and cooperative with the laboratory staff, mustmaintain the discipline and decency in the laboratory.
- 7. Computer labs are established with sophisticated and high end branded systems, which should be utilized properly.
- 8. Students / Faculty must keep their mobile phones in SWITCHED OFF mode during the lab sessions. Misuse of the equipment, misbehaviors with the staff and systems etc., will attract severe punishment.
- 9. Students must take the permission of the faculty in case of any urgency to go out ; if anybody found loitering outside the lab / class without permission during working hours will be treated seriously and punished appropriately.
- 10. Students should LOG OFF/ SHUT DOWN the computer system before he/she leaves the lab after completing the task (experiment) in all aspects. He/she must ensure the system / seat is kept properly.

#### Head of the Department

#### Principal

**OOPs through JAVA** 

MRCET CAMPUS

### INDEX

S.No	List of Programs	PageNos.
1	Write a java program to find the Fibonacci series using recursive and non recursive functions	1
2	Write a java program to multiply two given matrices.	3
3	Write a java program for Method overloading and Constructor overloading	4
4	Write a java program to display the employee details using Scanner class	6
5	Write a java program that checks whether a given string is palindrome or not	7
6	A)Write a java program to represent Abstract class with example.	8
	B)Write a java program to implement Interface using extends keyword	9
7	A) Write a java program to create inner classes	11
	B) Write a java program to create user defined package	12
8	A)Write a java program for creating multiple catch blocks	13
	B)Write a java program for producer and consumer problem using Threads	14
9	Write a Java program that implements a multi-thread application that has three threads	16
10	A)Write a java program to display File class properties	18
	B)Write a java program to represent ArrayList class	19
	C)Write a Java program loads phone no, name from a text file using hash table	20
11	Write an applet program that displays a simple message	23
12	A)Write a Java program compute factorial value using Applet	24
	B)Write a program for passing parameters using Applet	26
13	A) Write a java program for handling Mouse events and Key events	27
	B) Write a java program for handling Key events	31
14	Write a java program that connects to a database using JDBC	33
15	A)Write a java program to connect to a database using JDBC and insert values into it.	34
	B)Write a java program to connect to a database using JDBC and delete values from it	36
16	Write a java program that works as a simple calculator. Use a Grid Layout	38-45
	to arrange Buttons for digits and for the $+ - *$ % operations. Add a text field to display the result	

#### **WEEK -1**

Date:

## Aim: Write a java program to find the Fibonacci series using recursive and non recursivefunctions

```
Program:
class fib
int a,b,c;
                                      //Non recursive function to find the Fibonacci series.
void nonrecursive(int n)
ł
a=0;
b=1;
System.out.print(a+ "" + b);
c=a+b;
while(c<=n)
System.out.print(c);
a=b:
b=c;
c=a+b;
int recursive(int n)
                                  // Recursive function to find the Fibonacci series.
ł
if(n==0)
return (0);
if(n==1)
return (1);
else
return(recursive(n-1)+recursive(n-2));
ł
                             // Class that calls recursive and non recursive functions
class fib1
public static void main(String args[])
int n=5:
System.out.println("The Fibonacci series using non recursive is");
                           // Creating object for the fib class.
fib f=new fib();
                           // Calling non recursive function oF fib class.
f.nonrecursive(n);
System.out.println("\n The Fibonacci series using recursive is");
for(int i=0;i<=n;i++)
ł
                           // Calling recursive function of fib class.
int F1=f.recursive(i);
System.out.print(F1);
Malla Reddy College of Engineering and Technology (MRCET CAMPUS)
                                                                                         Page | 1
```

Three Test Outputs:

Output1: E:\kalpana>javac fib1.java E:\kalpana>java fib1 5 The Fibonacci series using non recursive is 011235 The Fibonacci series using recursive is 011235

Signature of the faculty

EXERCISE:

- 1. Write a java program to print the multiplication table.
- 2. Write a java program to find the Factorial of a given integer using recursive and nonrecursive functions
- 3. Write a java program that prompts the user for an integer and then printouts all prime numbers up to that integer.

#### **WEEK -2**

Aim: Write a java program to multiply two given matrices. public class MatrixEx { public static void main(String args[]) { //creating two matrices int a[][]={{1,1,1},{2,2,2},{3,3,3}}; int b[][]={{1,1,1},{2,2,2},{3,3,3}};

//creating another matrix to store the multiplication of two matrices
int c[][]=new int[3][3]; //3 rows and 3 columns

```
//multiplying and printing multiplication of 2 matrices
for(int i=0;i<3;i++)
for(int j=0; j<3; j++)
{
c[i][j]=0;
for(int k=0;k<3;k++)
ł
c[i][j] += a[i][k] * b[k][j];
       //end of k loop
}
System.out.print(c[i][j]+" "); //printing matrix element
       //end of j loop
ł
System.out.println(); //new line
ļ
ł
```

Three test outputs:

E:\kalpana>javac MatrixEx.java E:\kalpana>java MatrixEx 6 6 6 12 12 12 18 18 18

Signature of the faculty

EXERCISE:

- 1. Write a java program to Transpose of a matrix is obtained by changing rows to cols & columns to rows.
- 2. Write a java program to check whether the Matrix is Symmetric or Not.
- 3. Write a Java Program to find Matrix is an Identity Matrix or not.
- 4. Write a java program to add and subtract two given matrices.

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

#### <u>WEEK -3</u>

#### Aim: Write a java program for Method overloading and Constructor overloading

#### Method overloading:

```
import java.io.*;
class MethodOverloadingEx {
  static int add(int a, int b)
  ł
   return a + b;
  }
  static int add(int a, int b, int c)
  {
     return a + b + c;
  }
  public static void main(String args[])
  ł
      System.out.println("add() with 2 parameters");
      System.out.println(add(4, 6));
      System.out.println("add() with 3 parameters");
     System.out.println(add(4, 6, 7));
  }
}
```

#### **Output:**



Date:

#### **Constructor overloading**

```
public class Student {
//instance variables of the class
int id;
String name;
Student(){
System.out.println("this a default constructor");
ł
Student(int i, String n){
id = i;
name = n;
ł
public static void main(String[] args) {
//object creation
Student s = new Student();
System.out.println("\nDefault Constructor values: \n");
System.out.println("Student Id : "+s.id + "\nStudent Name : "+s.name);
System.out.println("\nParameterized Constructor values: \n");
Student student = new Student(10, "Kalpana");
System.out.println("Student Id : "+student.id + "\nStudent Name : "+student.name);
ł
```

Three test outputs:



EXERCISE:

Signature of the faculty

Write a java program to find all even and odd integers up to a given integer.
 Write a java program that reads a line of integers and displays each integers and the product of all integers use String Tokenizer.

#### <u>WEEK - 4</u>

Date:

#### Write a java program to display the employee details using Scanner class

```
import java.util.*;
class EmployeeDetails
{
    public static void main(String args[])
    {
        System.out.println("enter name,id,age,salary");Scanner sc=new Scanner(System.in);
        String n=sc.next();int i=sc.nextInt(); int a=sc.nextInt();
        float s=sc.nextFloat();
        System.out.println("name is"+n+"idis"+i+"ageis"+a+"salaryis"+s);
    }
}
```

Three test Outputs:

```
E:\kalpana>javac EmployeeDetails.java
E:\kalpana>java EmployeeDetails
enter name,id,age,salary
Socse
3
2
123456789
name is: Socse
id is :3
age is: 2
salary is:1.23456792E8
```

Signature of the faculty

EXERCISE:

1. Write a java program to Read and display the student details using Scannerclass.

2. Write a java program that displays the number of characters, lines, words, white spaces in a text file.

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

#### <u>WEEK - 5</u>

#### Aim: Write a java program that checks whether a given string is palindrome or not

#### Program:

```
// Class to find whether string is palindrome or not.
class palindrome
{
public static void main(String args[])
// Accepting the string at run time.
String s=args[0];
String s1="";
int le.j;
// Finding the length of the string.
le = s.length();
// Loop to find the reverse of the string.
for(j=le-1;j>=0;j--)
ł
s1=s1+s.charAt(j);
}
// Condition to find whether two strings are equal and display the message.
```

```
if(s.equals(s1))
System.out.println("String "+s+" is palindrome"); else
System.out.println("String "+s+" is not palindrome");
}
```

Three test outputs:

#### E:\kalpana≻javac palindrome.java E:\kalpana≻java palindrome kalpana String kalpana is not palindrome E:\kalpana≻java palindrome madam String madam is palindrome E:\kalpana≻

EXERCISE:

Signature of the faculty

- 1. Write a java program to sort the given integers in ascending/descending order.
- 2. Write a java program to display characters in a string in sorted order.
- 3. Write a program that uses a sequence input stream to output the contents of two files.

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

Date:

#### <u>WEEK – 6</u>A

#### Write a java program to represent Abstract class with example Program

```
abstract class Bank{
abstract int getRateOfInterest();
}
class SBI extends Bank
ł
int getRateOfInterest()
ł
return 7;
}
}
class PNB extends Bank
int getRateOfInterest(){return 8;}
}
class TestBank{
public static void main(String args[]){
Bank b:
b=new SBI();
System.out.println("Rate of Interest is: "+b.getRateOfInterest()+" %");
b=new PNB();
System.out.println("Rate of Interest is: "+b.getRateOfInterest()+" %");
}
}
```

Three Test Outputs:

Rate of Interest is: 7 % Rate of Interest is: 8 %

Signature of the faculty

#### EXERCISE:

Write a java program that reads a file and displays the file on the screen, with an asterisk mark before each line

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

Date:

#### WEEK - 6B

class Person

```
{
  String name;
   Person(String n)
   ł
   name = "Person: " + n;
   ļ
}
interface Mother
  public void FeedChildren();
 }
interface Wife
 ł
  public void CallHusband();
 }
class WifeAndMother extends Person implements Wife, Mother
  WifeAndMother(String n)
   {
    super(n);
    name = "Wife and mother: " + n;
public void FeedChildren()
    System.out.println(name + " is feeding the children.");
public void CallHusband()
   System.out.println(name + " is calling her husband.");
 }
class Test
{
  public static void main(String args[])
      Person p = new Person("SreeRam");
      WifeAndMother w = new WifeAndMother("Seetha");
      System.out.println("p is a " + p.name + "
                                                  ");
      System.out.println("w is a " + w.name);
```

Write a java program to implement Interface using extends keyword

#### w.FeedChildren(); w.CallHusband();

#### } }

}

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

Page 9

B.Tech-CSE (Emerging Technologies)

Three Test Outputs:

#### Output 1:

E:\kalpana>javac Test.java E:\kalpana>java Test p is a Person: SreeRam w is a Wife and mother: Seetha Wife and mother: Seetha is feeding the children. Wife and mother: Seetha is calling her husband.

Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

#### <u>WEEK – </u>7A

# Write a java program to create inner classes class A

{ int a=10; void display( ) { В b=new B(); b.show( ); } class B { int b=20; void show( ) { System.out.println(" a value is " +a); System.out.println(" b value is " +b); } } } class InnerDemo { public static void main(String args[]) { А a=new A(); a.displa y(); } }

#### Three Test Outputs:

E:∖kalpana≻javac InnerDemo.java

E:\kalpana>java InnerDemo a value is 10 b value is 20

Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

R-20

Date:

#### <u>WEEK – 7B</u>

#### Write a java program to create user defined package

#### <u>A.java</u>

```
package pack;
public class A
{
public void msg()
{
System.out.println("Hello");
}
}
```

#### <u>B.java</u>

```
import pack.A;
class B
{
public static void main(String args[])
{
A obj = new A();
obj.msg();
}
}
```

Three Test Outputs:



E:∖kalpana≻javac B.java E:∖kalpana≻java B Hello

Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

Page | 12

Date:

#### <u>WEEK – 8</u>A

#### Date:

#### Write a java program for creating multiple catch blocks public class MultipleCatchBlocks { public static void main(String[ ] args) { try

```
{
int a[]=new
int[5];
a[5]=30/0;
}
catch(ArithmeticException e)
{
System.out.println("Arithmetic Exception occurs");
}
catch(ArrayIndexOutOfBoundsException e)
{
System.out.println("ArrayIndexOutOfBounds Exception occurs");
}
catch(Exception e)
{
System.out.println("Parent Exception occurs");
}
System.out.println("rest of the code");
}
```

#### **Three Test Outputs:**

E:\kalpana>javac MultipleCatchBlocks.java

E:\kalpana>java MultipleCatchBlocks Arithmetic Exception occurs rest of the code

Signature of the faculty

public static void main(String args[])

Producer p1=new Producer(); Consumer c1=new Consumer(p1);

Thread t1=new Thread(p1); Thread t2=new Thread(c1);

class Producer extends Thread

#### <u>WEEK – 8B</u>

{

t2.start();
t1.start();

StringBuffer sb;

} }

class InterThreadDemo

```
Producer()
{
sb=new StringBuffer();
}
public void run()
synchronized(sb)
ł
for(int i=0;i<=10;i++)
ł
try
{
sb.append(i+":");
Thread.sleep(1000);
System.out.println("appending");
}
catch(InterruptedException e)
System.out.println(e);
}
}
sb.notify();
}
}
}
Malla Reddy College of Engineering and Technology (MRCET CAMPUS)
```

Write a java program for producer and consumer problem using Threads

Date:

B.Tech – CSE (Emerging Technologies)

```
class Consumer extends Thread
{
Producer prod;
Consumer(Producer prod)
this.prod=prod;
public void run()
ł
synchronized(prod.sb)
{
try
{
prod.sb.wait();
ł
catch(Exception e)
System.out.println(e);
System.out.println(prod.sb);
Three Test Outputs:
```



Signature of the faculty

#### <u>WEEK – 9</u>

Date:

Write a Java program that implements a multi-thread application that has three threads class Thread1 extends Thread

ł public void run() for(int i=0;i<=5;i++) System.out.println("Thread1:"+ i); class Thread2 extends Thread public void run() for(int j=0;j<=5;j++) System.out.println("Thread2:"+ j); } } class Thread3 extends Thread public void run() for(int k=0;k<=5;k++) System.out.println("Thread3:"+ k); } } } class MultiThreadDemo { public static void main(String args[]) { Thread1 t1=new Thread1(); Thread2 t2=new Thread2(); Thread3 t3=new Thread3();t1.start(); t2.start(); t3.start(); for(int i=0;i<=5;i++) ł System.out.println("main thread:"+ i); } } } Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

Three Test Outputs:

Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

#### <u>WEEK - 10A</u>

#### Write a java program to display File class properties

import java.io.\*; public class FileDemo2 { public static void main(String[] args) { String fname=args[0]; File f=new File(fname); System.out.println("path: "+f.getPath()); System.out.println("parent: "+f.getPath()); System.out.println("parent: "+f.getParent()); System.out.println("exits: "+f.exists()); if(f.exists()) { System.out.println("isWritable: "+f.canWrite()); System.out.println("isReadable: "+f.canRead()); System.out.println("isDirectory: "+f.isDirectory()); System.out.println("size of the file: "+f.length()); } }

#### **Test Output:**



Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

R-20

#### <u>WEEK – 10B</u>

```
Write a java program to represent ArrayList class
import java.util.*;
class TestJavaCollection
{
public static void main(String args[ ])
ArrayList<String> list=new ArrayList<String>();
                                                         //Creating arraylist
list.add("Kalpana");
                                           //Adding object in arraylist
list.add("Venu");
list.add("Suneetha");
list.add("Gayatri");
       //Traversing list through Iterator
Iterator itr=list.iterator();
while(itr.hasNext())
{
 System.out.println(itr.next());
}
}
Test output:
```

```
E:\kalpana>javac TestJavaCollection.java
E:\kalpana>java TestJavaCollection
Kalpana
Venu
Suneetha
Gayatri
```

Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

#### <u>WEEK – 10C</u>

Date:

#### Write a Java program loads phone no, name from a text file using hash table

**Aim:** Implement a program to display the content of a text file where the data is organized as one line per record and each field in a record are separated by a whitespace (\\s means 0 or more repetitions of any whitespace character. It takes a name or phone number as input and prints the corresponding other value from the hash table.

#### Program:

```
import java.util.*;
import java.io.*;
public class Hashtbl {
public static void main(String[] args)
{
try
FileInputStream fs = new FileInputStream("E:\\kalpana\\ph.txt");
Scanner sc = new Scanner(fs).useDelimiter("\\s+");
Hashtable<String, String> ht = new Hashtable<String, String>();
String[] arrayList;
String a;
System.out.println("HASH TABLE IS");
System.out.println("-----");
System.out.println("KEY : VALUE");
while (sc.hasNext())
ł
a = sc.nextLine();
arrayList = a.split("\\s+");
ht.put(arrayList[0], arrayList[1]);
System.out.println(arrayList[0] + ":" + arrayList[1]);
}
System.out.println("----MENU-----");
System.out.println("----1.Search by Name------");
System.out.println("----2.Search by Mobile-----");
System.out.println("----3.Exit-----");
String opt = "";
String name, mobile;
Scanner s = new Scanner(System.in);
while (opt != "3")
System.out.println("Enter Your Option 1,2,3");
opt = s.next();
```

```
switch (opt)
{
case "1":
{
System.out.println("Enter Name");
name = s.next();
if (ht.containsKey(name))
ł
System.out.println("Mobile is " + ht.get(name));
}
else
System.out.println("Not Found");
}
break;
case "2":
{
System.out.println("Enter mobile");
mobile = s.next();
if (ht.containsValue(mobile)) {
for (@SuppressWarnings("rawtypes") Map.Entry e : ht.entrySet()) {
if (mobile.equals(e.getValue())) {
System.out.println("Name is " + e.getKey());
}
}
}
else
System.out.println("Not Found");
ł
break;
case "3":
{
opt = "3";
System.out.println("Menu Successfully Exited");
}
break;
default:
System.out.println("Choose Option betwen 1 and Three");
break;
} } }
catch (Exception ex) {
System.out.println(ex.getMessage());
} } }
```

B.Tech-CSE (Emerging Technologies)

#### **Test output:**

∖kalpana>javac Hashtbl.java Е: java Hashtbl IS ABLE SH<sup>ka</sup> A nnetha 230 lpana:8 ŏ yа 9898989 Ŕ NÚ-Search Search Exit--Name-----Mobile---L Ēxit<sup>.</sup> Your Option 1,2,3 ter ter Name is 9898989898 our Option 1,2,3 le pana Option 1,2,3 enu Successfully Exited



Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

#### <u>WEEK – 11</u>

#### Write an applet program that displays a simple message

```
import java.awt.*;
import java.applet.*;
/*
<applet code="FirstApplet" width=200 height=300>
</applet>*/
public class FirstApplet extends Applet
{
    public void init()
    {
    setBackground(Color.red);
    }
    public void paint(Graphics g)
    {
      g.drawString("this is first applet",50,30);
      showStatus("welcome");
    }
}
```

Three test outputs:



Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

R-20

#### <u>WEEK – 12A</u>

```
Write a Java program compute factorial value using Applet
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code="FactorialApplet" width=300 height=300>
</applet>
*/
public class FactorialApplet extends Applet implements ActionListener
Label L1,L2;
TextField T1,T2;
Button B1;
public void init()
ł
setLayout(new FlowLayout(FlowLayout.LEFT));
L1=new Label("enter the value");
add(L1);
T1=new TextField(10);add(T1);
L2=new Label("factorial value is");
add(L2);
T2=new TextField(10);add(T2);
B1=new Button("compute");
add(B1);
B1.addActionListener(this);
}
public void actionPerformed(ActionEvent e)
if(e.getSource()==B1)
int value=Integer.parseInt(T1.getText());
int fact=factorial(value);
T2.setText(String.valueOf(fact));
}
int factorial(int n)
if(n==0)return 1;else
return n*factorial(n-1);
}
}
```

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

Date:

E:∖kalpana>javac FactorialApplet.java			
E:\kalpana>appletviewer FactorialApplet.java			
	Applet Viewer: FactorialApplet		
	Applet		
	enter the value 5		
	factorial value is 120 compute		
	Applet started.		

Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

#### <u>WEEK – 12B</u>

```
Write a program for passing parameters using Applet
import java.awt.*; import java.applet.*;
/*
<applet code="MyApplet" width=200 height=300>
<param name="t1" value="Ravi">
<param name="t2" value="102">
</applet>
*/
public class MyApplet extends Applet
String n; String id;
public void init()
{
n=getParameter("t1");
id=getParameter("t2");
}
public void paint(Graphics g)
ł
g.drawString("name is"+n,100,100);
g.drawString("id is"+id,100,150);
}
}
Three test outputs:
   E:∖kalpana≻javac MyApplet.java
                                       💪 Applet Viewer: M... 📃 🔳 🗾
   E:∖kalpana>appletviewer MyApplet.java
                                        Applet
```

name is: Kalpana id is: 101

Applet started.

Date:

Signature of the faculty

#### <u>WEEK – 13A</u>

```
Write a java program for handling Mouse events
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code="MouseEvents" width=300 height=100>
</applet>
*/
public class MouseEvents extends Applet implements MouseListener, MouseMotionListener
ł
String msg = "";
int mouse X = 0, mouse Y = 0; // coordinates of mouse
public void init() {
addMouseListener(this);
addMouseMotionListener(this);
}
// Handle mouse clicked.
public void mouseClicked(MouseEvent me) {
// save coordinates
mouse X = 0;
mouseY = 10;
msg = "Mouse clicked.";
repaint();
}
// Handle mouse entered.
public void mouseEntered(MouseEvent me) {
// save coordinates
mouse X = 0:
mouseY = 10;
msg = "Mouse entered.";
repaint();
}
// Handle mouse exited.
public void mouseExited(MouseEvent me) {
// save coordinates
mouseX = 0;
mouseY = 10;
msg = "Mouse exited.";
repaint();
}
```

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

Date:

```
// Handle button pressed.
public void mousePressed(MouseEvent me) {
// save coordinates mouseX = me.getX();
mouseY = me.getY();
msg = "Down";
repaint();
}
// Handle button released.
public void mouseReleased(MouseEvent me) {
// save coordinates
mouseX = me.getX();
mouseY = me.getY();
msg = "Up";
repaint();
}
// Handle mouse dragged.
public void mouseDragged(MouseEvent me) {
// save coordinates
mouseX = me.getX();
mouseY = me.getY();
msg = "*";
showStatus("Dragging mouse at " + mouseX + ", " + mouseY);
repaint();
}
// Handle mouse moved.
public void mouseMoved(MouseEvent me)
{
// show status
showStatus("Moving mouse at " + me.getX() + ", " + me.getY());
}
// Display msg in applet window at current X,Y location.
public void paint(Graphics g)
```

```
{
g.drawString(msg, mouseX, mouseY);
}
```





Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

#### **WEEK – 13B**

#### Write a program for handling Key Events

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
```

/\*

```
<applet code="SimpleKey" width=300 height=100>
</applet> */
public class SimpleKey extends Applet implements KeyListener
String msg = "";
int X = 10, Y = 20; // output coordinates
public void init()
{
addKeyListener(this);
requestFocus(); // request input focus
}
public void keyPressed(KeyEvent ke)
 showStatus("Key Down");
}
public void keyReleased(KeyEvent ke)
showStatus("Key Up");
public void keyTyped(KeyEvent ke)
   msg += ke.getKeyChar();
repaint();
}
    // Display keystrokes.
    public void paint(Graphics g)
    {
      g.drawString(msg, X, Y);
}
```



Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

#### <u>WEEK – 14</u>

```
Aim: Write a java program that connects to a database using JDBC
 Program:
 import java.sql.Connection;
 import java.sql.DriverManager;
 public class PostgreSQLJDBC
 {
 public static void main(String args[ ])
 ł
 Connection c = null;try
 {
 Class.forName("org.postgresql.Driver");
  c = DriverManager .getConnection
     ("jdbc:postgresql://localhost:5432/testdb", "postgres", "123");
} catch (Exception e)
{
  e.printStackTrace();
 System.err.println(e.getClass().getName()+": "+e.getMessage());System.exit(0);
 System.out.println("Opened database successfully");
 }
 }
```

Three test outputs:

Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

Date:

#### <u>WEEK – 15A</u>

Date:

#### Write a java program to connect to a database using JDBC and insert values into it

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;
public class PostgreSQLJDBC
public static void main(String args[ ])
Connection c = null; Statement stmt = null;try {
Class.forName("org.postgresql.Driver");c = DriverManager
 .getConnection("jdbc:postgresql://localhost:5432/testdb", "manisha", "123");
c.setAutoCommit(false); System.out.println("Opened database successfully");stmt =
c.createStatement();
String sql = "INSERT INTO COMPANY (ID, NAME, AGE, ADDRESS, SALARY) "
 + "VALUES (1, 'Paul', 32, 'California', 20000.00 );";
stmt.executeUpdate(sql);
sql = "INSERT INTO COMPANY (ID, NAME, AGE, ADDRESS, SALARY) "
  + "VALUES (2, 'Allen', 25, 'Texas', 15000.00);";
stmt.executeUpdate(sql);
sql = "INSERT INTO COMPANY (ID, NAME, AGE, ADDRESS, SALARY) "
  + "VALUES (3, 'Teddy', 23, 'Norway', 20000.00);";
stmt.executeUpdate(sql);
sql = "INSERT INTO COMPANY (ID,NAME,AGE,ADDRESS,SALARY) "
   + "VALUES (4, 'Mark', 25, 'Rich-Mond ', 65000.00);";
stmt.executeUpdate(sql);
stmt.close();
c.commit();
c.close();
}
catch (Exception e)
System.err.println( e.getClass().getName()+": "+ e.getMessage());
System.exit(0);
System.out.println("Records created successfully");
ł
}
```

Three test outputs:

Signature of the faculty

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

#### <u>WEEK – 15B</u>

Date:

```
Write a java program to connect to a database using JDBC and delete values from it
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
  public class PostgreSQLJDBC6 { public static void main( String args[ ] )
{
Connection c = null;
Statement stmt = null;
try {
Class.forName("org.postgresql.Driver");
c = DriverManager.getConnection
             ("jdbc:postgresql://localhost:5432/testdb", "manisha", "123");
c.setAutoCommit(false);
System.out.println("Opened database successfully");
stmt = c.createStatement();
String sql = "DELETE from COMPANY where ID=2;";
stmt.executeUpdate(sql);
c.commit():
ResultSet rs = stmt.executeQuery( "SELECT * FROM COMPANY;");
while (rs.next())
{
int id = rs.getInt("id");
String name = rs.getString("name");
int age = rs.getInt("age");
String address = rs.getString("address");
float salary = rs.getFloat("salary");
System.out.println( "ID = " + id );
System.out.println( "NAME = " + name );
System.out.println( "AGE = " + age );
System.out.println( "ADDRESS = " + address );
System.out.println( "SALARY = " + salary );
System.out.println();
}
rs.close();
stmt.close();
c.close();
}
```

```
catch ( Exception e )
{
System.err.println( e.getClass().getName()+": "+ e.getMessage());
System.exit(0);
}
System.out.println("Operation done successfully");
}
```

Three test outputs:

Signature of the faculty

#### <u>WEEK – 16</u>

```
Date:
```

Write a java program that works as a simple calculator. Use a Grid Layout to arrange Buttons for digits and for the + - \* % operations. Add a text field to display the result import javax.swing.\*;

import javax.swing.event.\*;

import java.awt.\*;

import java.awt.event.\*;

class A extends JFrame implements ActionListener

```
{
```

public JButton b1, b2, b3, b4, b5, b6, b7, b8, b9, b10, b11, b12, b13, b14, b15, b16; public JTextField tf1; public JPanel p; public String v = ""; public String v1 = "0"; public String op = ""; public A() { setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE); setSize(400, 400); p = new JPanel(new FlowLayout()); tf1 = new JTextField(10); p.add(tf1); add(p); setLayout(new GridLayout(0, 3)); b1 = new JButton("1"); b1.addActionListener(this); add(b1);b2 = new JButton("2");b2.addActionListener(this); add(b2);b3 = new JButton("3"); b3.addActionListener(this); add(b3); b4 = new JButton("4");b4.addActionListener(this); add(b4);

```
b5 = new JButton("5");
b5.addActionListener(this);
add(b5);
b6 = new JButton("6");
b6.addActionListener(this);
add(b6);
b7 = new JButton("7");
b7.addActionListener(this);
add(b7);
b8 = new JButton("8");
b8.addActionListener(this);
add(b8);
b9 = new JButton("9");
b9.addActionListener(this);
add(b9);
b10 = new JButton("0");
b10.addActionListener(this);
add(b10);
b11 = new JButton("+");
b11.addActionListener(this);
add(b11);
b12 = new JButton("-");
b12.addActionListener(this);
add(b12);
b13 = new JButton("*");
b13.addActionListener(this);
add(b13);
b14 = new JButton("/");
b14.addActionListener(this);
add(b14);
b16 = new JButton("%");
b16.addActionListener(this);
add(b16);
b15 = new JButton("=");
b15.addActionListener(this);
add(b15);
setVisible(true);
Malla Reddy College of Engineering and Technology (MRCET CAMPUS)
```

```
public void actionPerformed(ActionEvent ae)
{
String b = ae.getActionCommand();
switch (b)
{
case "1":
{
v = v + "1";
tf1.setText(v);
}
break;
case "2": {
v = v + "2";
tf1.setText(v);
}
break;
case "3": {
v = v + "3";
tf1.setText(v);
}
break;
case "4": {
v = v + "4";
tf1.setText(v);
}
break;
case "5": {
v = v + "5";
tf1.setText(v);
}
break;
case "6": {
v = v + "6";
tf1.setText(v);
}
break;
```

```
case "7": {
v = v + "7";
tf1.setText(v);
}
break;
case "8": {
v = v + "8";
tf1.setText(v);
}
break;
case "9": {
v = v + "9";
tf1.setText(v);
}
break;
case "0": {
v = v + "0";
tf1.setText(v);
}
break;
case "+": {
op = "+";
v1 = tf1.getText();
v = "";
}
break;
case "-": {
op = "-";
v1 = tf1.getText();
v = "";
}
break;
case "*": {
op = "*";
v1 = tf1.getText();
v = "";
}
```

```
R-20
```

```
break;
case "/": {
op = "/";
v1 = tf1.getText();
v = "";
}
break;
case "%": {
op = "%";
v1 = tf1.getText();
v = "";
}
break;
case "=": {
switch (op) {
case "+": {
v = tf1.getText();
if (v.equals("")) {
v = "0";
}
long i = Long.parseLong(v1) + Long.parseLong(v);
tf1.setText(String.valueOf(i));
v="";
}
break;
case "-": {
v = tf1.getText();
if (v.equals("")) {
v = "0";
}
long i = Long.parseLong(v1) - Long.parseLong(v);
tf1.setText(String.valueOf(i));
v="";
}
break;
```

Malla Reddy College of Engineering and Technology (MRCET CAMPUS)

```
case "*": {
v = tf1.getText();
if (v.equals("")) {
v = "0";
}
long i = Long.parseLong(v1) * Long.parseLong(v);
tf1.setText(String.valueOf(i));
v="";
}
break;
case "/": {
try {
v = tf1.getText();
if (v.equals("")) {
v = "0";
}
long i = Long.parseLong(v1) / Long.parseLong(v);
tf1.setText(String.valueOf(i));
v="";
} catch (Exception ex) {
JOptionPane.showMessageDialog(this, ex.getMessage());
}
}
break;
case "%": {
try {
v = tf1.getText();
if (v.equals("")) {
v = "0";
}
long i = Long.parseLong(v1) % Long.parseLong(v);
tf1.setText(String.valueOf(i));
v="";
} catch (Exception ex) {
JOptionPane.showMessageDialog(this, ex.getMessage());
}
}
Malla Reddy College of Engineering and Technology (MRCET CAMPUS)
```

```
break;
}
}
break;
}
break;
}
public class Calc
{
public static void main(String[] args)
{
A a = new A();
}
```

#### **Test outputs:**

E:\kalpana≻javac Calc.java F:\kalpana≻java Calc				
	98	1	2	
	3	4	5	
	6	7	8	
	9	0	+	
	-	B	I	
	%	-		

E:∖kalpana≻javac Calc.java			
E:∖kalpana>java Calc			
	<u>\$</u>		
	98	1	2
	3	4	5
	6	7	8
	9	0	•
	-	8	I
	%	-	
	%	-	

E:\kalpana≻javac Calc.java				
El (Kalpana) java oa				
	8	1	2	
	3	4	5	
	6	7	8	
	9	0	•	
	-	*	I	
	%	=		

E:∖kalpana>javac Calc.java			
E:∖kalpana≻java Calc			
	<u></u>		
	784	1	2
	3	4	5
	6	7	8
	9	0	+
	-	*	1
	%	B	

Signature of the faculty