GR20 Regulations

II B.Tech II Semester

Java Programming Lab

(GR20A2080)

Department of Computer Science and Engineering

(Artificial Intelligence and Machine Learning)

GOKARAJU RANGARAJU

INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

SYLLABUS

Gokaraju Rangaraju Institute of Engineering and Technology

Java Programming Lab

Course Code:GR20A2080 **L/T/P/C:0/0/4/2**

II Year II Semester

Course Objectives:

1. Understand Object Oriented Programming concepts and apply them inproblem solving.
2. Get knowledge on Abstract classes, Interfaces and Multithreading.
3. Developing java applications and handle the exceptions.
4. Design applications for solving real world problems using Collectionframework.
5. Building java GUI based applications using Applets, AWT and Swing.

Course Outcomes:

At the end of the course, the student will be able to

1. Analyze a problem, identify and define the computing requirements appropriate to its solution using object-oriented programming concepts.
2. Design the applications using Inheritance, Polymorphism andSynchronization concepts.
3. Handle exceptions at Compile time and Run time.
4. Solve the real-world problems using Java Collection framework.
5. Develop GUI applications using Applets, AWT and Swings.

TASK 1

Write java programs that implement the following

a. Constructor
b. Parameterized constructor
c. Method overloading
d. Construct or overloading.

TASK 2

a. Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome.
b. Write a Java program for sorting a given list of names in ascending order.
c. Write a Java Program that reads a line of integers, and then displays eachinteger and the sum of allthe integers (Use StringTokenizer class of java.util)

TASK 3

Write java programs that uses the following keywords

a) This b)super c)static d)final

TASK 4

a. Write a java program to implement method overriding
b. Write a java program to implement dynamic method dispatch.
c. Write a Java program to implement multiple inheritance.
d. Write a java program that uses access specifiers.

TASK 5

a. Write a Java program that reads a file name from the user, then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.
b. Write a Java program that reads a file and displays the file on the screen, with a line number before each line.
c. Write a Java program that displays the number of characters, lines and words in a text file.

TASK 6

a. Write a Java program for handling Checked Exceptions.
b. Write a Java program for handling Unchecked Exceptions.

TASK 7

a. Write a Java program that creates three threads. First thread displays “GoodMorning” every one second, the second thread displays “Hello” every two seconds and the third thread displays “Welcome” every three seconds.
b. Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.

TASK 8

Write a program illustrating following collections framework

a) Array List b) Vector c) Hash Table d) Stack

TASK 9

a. Develop an applet that displays a simple message.
b. Develop an applet that receives an integer in one text field and compute its factorial value and return it in another text field, when the button named “Compute” is clicked.
c. Write a Java program that works as a simple calculator. Use a grid layout to arrange button for the digits and for the +, -,*, % operations. Add a text field to display the result.

TASK 10

a. Write a Java program for handling mouse events.
b. Write a Java program for handling key events.

TASK 11

a. Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields Num1 and Num 2.
b. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1or Num2 were not an integer, the program would throw Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception and display the exception in a message dialog box.

TASK 12

a. Write a java program that simulates traffic light. The program lets the user select one of three lights: red, yellow or green. When a radio button is selected, the light is turned on, and only one light can be on at a time. No light is on when the program starts.
b. Write a Java program that allows the user to draw lines, rectangles andovals.

TASK 13

Create a table in Table.txt file such that the first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using JTable component.

Text Books/ References:

1. Java: The Complete Reference, 10thedition, Herbert Schildt, McGraw-Hill.
2. Java Fundamentals- A Comprehensive introduction, Herbert schildtandDale skrien, TMH.
3. Java for programming, P.J.Dietel Pearson education (OR) Java: How to Program P.J.Dietel and H.M.Dietel,PHI
4. Object Oriented Programming through java, P.Radha Krishna, UniversitiesPress.
5. Thinking in Java, Bruce Eckel, PearsonEducation
6. Programming in Java, S.Malhotra and S.Choudhary, Oxford UniversityPress

INDEX

8	Write a program illustrating following collections framework a) ArrayList b) Vector c) HashTable d) Stack	36
9	a) Develop an applet that displays a simple message. b) Develop an applet that receives an integer in one text field and compute itsfactorial value and return it in another text field, when the button named “Compute” is clicked. c) Write a Java program that works as a simple calculator. Use a grid layout toarrange button for the digits and for the +, -,*, % operations. Add a text field to display the result.	44
10	a) Write a Java program for handling mouse events. b) Write a Java program for handling key events.	50
11	Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields Num1 and Num 2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1or Num2 were not an integer, the program would throw Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception and display the exception in a message dialog box.	54
12	a) Write a java program that simulates traffic light. The program lets the user select one of three lights: red, yellow or green. When a radio button is selected, the light is turned on, and only one light can be on at a time No l Light is on when the program starts. b) Write a Java program that allows the user to draw lines, rectangles and ovals.	57
13	Create a table in Table.txt file such that the first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using JTable component.	63

TASK 1

Task 1(a): Write a java program that implements Default constructor.

Aim: To write a java program that implements Default constructor.

Program:

```
class DefaultConstructor
{
int Sid;
String Sname;
//Default Constructor methodDefaultConstructor()
{
System.out.println("Default Constructor method gets calledautomatically
whenever object of the class gets created.");
Sid=846;
Sname="Sowmya";
}
void showDetails()
{
System.out.println("Sid:"+Sid+"Sname:"+Sname);
}
public static void main(String[] args)
{
//creating object
DefaultConstructor object = new DefaultConstructor();object.showDetails();
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac DefaultConstructor.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java DefaultConstructor
Sid:846
 Sname:Sowmya
```

Task 1(b): Write a java program that implements Parameterized constructor

Aim: To write a java program that implement parameterized constructor

Program:

```
class ParamConstructor
{

int Sid;
String Sname;
ParamConstructor()
{
Sid=125;
Sname="Sindhu";
}
ParamConstructor(int x,String n)
{
Sid=x;
Sname=n;
}
void showDetails()
{
System.out.println("Sid:"+Sid+"Sname:"+Sname);
}
public static void main(String[]args)
{
ParamConstructor p=new ParamConstructor();
p.showDetails();
ParamConstructor p1=new ParamConstructor(1458,"Pooja");
p1.showDetails();
ParamConstructor p2=new ParamConstructor(1459,"Anand");
p2.showDetails();
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac ParamConstruct.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java ParamConstruct
Sid:125
Sname:Sindhu
Sid:1458
Sname:Pooja
 Sid:1459
Sname:Anand
```

Task 1(c): Write a java program that implements method overloading

Aim: To write a java program that implements method overloading

Program:

```
class Task1c
{
public void disp(char c)
{
System.out.println("Value of charcter c="+c);
}
public void disp(char c, int num)
{
System.out.println("Value of character c="+c);
System.out.println("Value of integer num="+num);
}
public void disp(int no, double d)
{
System.out.println("Value of Integer number No="+no);
System.out.println("Value of Floating Number ="+no);

}
}

class MethodOverloading

{
public static void main(String args[])
{
Task1c obj = new Task1c();
obj.disp('a');
obj.disp('a',10);
obj.disp(10,20.40);
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac MethodOverloading.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java MethodOverloading
Value of character c=a
Value of character c=a
Value of integer num=10
Value of Integer number No=10
Value of Floating no value=20.4
```

Task 1(d): Write a java program that implements constructor overloading

Aim: To write a java program that implements constructor overloading

Program:

```
class ConstructOverloading
{
int i, j;
public ConstructOverloading()
{
//i=100;//j=200;
System.out.print("Inside default constructor");
System.out.println(" Value of i "+i+"and j "+j);
}
public ConstructOverloading(int q)
{
System.out.println("Inside single parameter constructor with int value="+q);
}
public ConstructOverloading(String str)
{
System.out.println("Inside single parameter constructor with String object");
System.out.println("String Value="+str);
}
public ConstructOverloading(int p, double k)
{
System.out.println("Inside double parameter constructor  value ofp="+p+""+"and k"+k);
}
public static void main(String args[])
{

ConstructOverloading mco = new ConstructOverloading();

 ConstructOverloading spmco = new ConstructOverloading(10);

ConstructOverloading dpmco = new ConstructOverloading(10,20.20);
ConstructOverloading dpmco1 = new ConstructOverloading(" Task1");
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac Construct Overloading.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java Construct Overloading
Inside default constructor value of i and j is 0
Inside single parameter constructor with int value=10
Inside double parameter constructor value of p=10 and k=20.2
Inside single parameter constructor with string object string value= Task1
```

TASK 2

Task 2(a): Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome

Aim: To write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome

Program:

```
import java.util.Scanner;
public class Palindrome
{
public static void main(String[] args)
{
Scanner input=new Scanner(System.in);
System.out.println("Enter the string ");
String s=input.nextLine();

if(isPalindrome(s))
System.out.println("String "+s +"is palindrome");

else
System.out.println("String "+s+" is not palindrome");
}
public static boolean isPalindrome(String s1)
{
int low=0;
int high=s1.length()-1;while(low<high)
{
if(s1.charAt(low) !=s1.charAt(high))
return false;
low++;
high--;
}
return true;
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac Palindrome.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java Palindrome
Enter the string: MADAM String MADAM is palindrome
```

Program:

```
//Palindrome using reverse()method.
import  java.util.*;
class PalindromeString
{

public static void main(String args[])
{
Scanner s=new Scanner(System.in);// Create a Scanner object
System.out.println("Enter the string");// Take the data from the user.
String st1=s.nextLine();//Read the data entered by the user.
StringBuffer sb=new StringBuffer(st1);// Create StringBuffer obj for st1
sb.reverse();// Reverse the letters
// Check & Print if palindrome
if(st1.equals(sb.toString()))
{
System.out.println("String "+st1+" is Palindrome");
}
else
{
System.out.println("String "+st1+" is notPalindrome");
}
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac PalindromeString.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java  PalindromeString
 Enter the string: pooja
String pooja is not a palindrome
```

Task 2(b): Write a Java program for sorting a given list of names in ascendingorder

Aim: To write a Java program for sorting a given list of names in ascendingorder

Program:

```
import java.util.Scanner;
class SortString
{
public static void main(String args[])
{
String temp;
Scanner SC = new Scanner(System.in);
System.out.print("Enter the value of N: ");
int N= SC.nextInt();
SC.nextLine(); //ignore next line characterString names[] = new String[N];
System.out.println("Enter names: ");
for(int i=0; i<N; i++)
{
System.out.print("Enter name [ " + (i+1) +" ]:");
names[i] = SC.nextLine();
for(int i=0; i<N; i++)
{
for(int j=1; j<N; j++)
{
  if(names[j].compareTo(names[j-1])>0)
 {
  temp=names[j-1];
 names[j-1]=names[j];
names[j]=temp;
 }
}
}
System.out.println("\nSorted names are in Ascending Order: ");
for(int i=0;i<N;i++)
{
System.out.println(names[i]);
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac SortString.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java  SortString
Enter the value of N: 4
Enter names:
Enter name [ 1 ] : z
Enter name [ 2 ] : w
Enter name [ 3 ] : a
Enter name [4 ] : r
Sorted names are in Ascending Order:
a r wz
```

Task 2(c): Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers (Use StringTokenizer class of java.util)

Aim: To write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers (Use StringTokenizer class of java.util)

Program:

```
import java.util.StringTokenizer;
import java.util.Scanner;
class tokens
{
public static void main(String[] args)
{
String temp; int k,total=0;
Scanner input=new Scanner(System.in);//Take the input from the user.
 System.out.println("Enter the Numbers : "); //Display the message to user.
String sentence=input.nextLine();  //read the numbers entered by the user.
//System.out.println("Number="+sentence);
StringTokenizer s1=new StringTokenizer(sentence);  //Create object of theString
//Tokenizer and pass the numbers entered by the user to it.
 System.out.println("Total Number of tokens:"+s1.countTokens());
while(s1.hasMoreTokens())
{
temp=s1.nextToken();

//nextToken() takes the next token and returns string
k=Integer.parseInt(temp);//Make use of parseInt() method from the Integer class
//to convert the String token into integer token.
total=total+k;//total+=k;
//Perform the addtion for obtaining final sum of all theSystem.out.print(k+"\t")
}
System.out.println("Sum of tokens :"+total); //Display the sum of all the tokensentered by
//the user.
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac tokens.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java  tokens
Enter the Numbers :
1 2 3 4 5
Total Number of tokens : 5
12345
Sum of tokens : 15
```

TASK 3

Task 3(a): Write a java program that implements the concept of this keyword.

Aim: To write a java program that implements the concept of this keyword**.**

Program:

```
//This keyword is used to point to the instance variable instead of pointing to the localvariable.
class ThisExample
{
int variable = 5;
public static void main(String args[])
{
ThisExample obj = new ThisExample();
obj.method(20);
obj.method();
}
void method(int variable)
{
variable = 10;
System.out.println("Value of Instance variable :" +this.variable);
System.out.println("Value of Local variable :" + variable);
}
void method()
{
int variable = 40;

System.out.println("Value of Instance variable :" + this.variable);
System.out.println("Value of Local variable :" + variable);
}}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac ThisExample.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java ThisExample
Value of Instance variable : 5
Value of Local variable : 10
Value of Instance variable: 5
Value of Local variable : 40
```

Task 3(b): Write a java program that implements the concepts of superkeyword

Aim: To write a java program that implements the concepts of super keyword

Program:

```
class Parentclass
{
int num=100;
}
//Child class or subclass
class SuperKeyword extends Parentclass
{
int num=110;

void printNumber()
{

//Super.variable_name
System.out.println("Accessing Parentclass variable using super keyword="+super.num);
System.out.println("Without using super keyword the value of num ="+num);
}
public static void main(String args[])

{
SuperKeyword obj= new SuperKeyword();
obj.printNumber();

}

}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac SuperKeyword.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java SuperKeyword
Accessing Parentclass variable using super keyword = 100
 without using super keyword the value of num is = 110
```

Task 3(c): Write a java program that implements the concepts of statickeyword

Aim: To write a java program that implements the concepts of static keyword

Program:

```
class Counter2
{
static int count=0;
//int count=0;
Counter2()
{
count++;
//System.out.println("Count value = " +count);
}
}
class staticCounter
{
public static void main(String args[])
{
Counter2 c1=new Counter2();
System.out.println("c1 count="+c1.count);
Counter2 c2=new Counter2();
System.out.println("c2 count="+c2.count);
Counter2 c3=new Counter2();
System.out.println("c3 count="+c3.count);
Counter2 c4=new Counter2();
 System.out.println("c4 count="+c4.count);
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac staticCounter.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java staticCounter
c1 count=1
c2 count=2
c3 count=3
c4 count=4
```

Task 3(d): Write a JAVA program that implements final keyword

Aim: To write a JAVA program that implements final keyword

Program:

```
class FinalKeyword
{
final int MAX_VALUE=99;
void myMethod()
{
System.out.println("MAX_VALUE ="+MAX_VALUE);
//MAX_VALUE=101;
}
public static void main(String args[])
{
FinalKeyword obj=new FinalKeyword();
obj.myMethod();
}

}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac FinalKeyword.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java FinalKeyword
MAX_VALUE=99
```

TASK 4

Task 4(a): Write a java program that implements the concept of methodoverriding.

Aim: To write a java program that implements the concept of methodoverriding.

Program:

```
class Bank

{
int getROI()
{
return 0;
}
}
class SBI extends Bank
{
int getROI()
{
return 8;
}
}
class ICICI extends Bank
{
int getROI()
{
return 7;
}
}
class Axis extends Bank
{
int getROI()
{
return 9;
}
}
class NewMethodOverriding
{
public static void main(String[] args)
{
SBI s=new
SBI();
ICICI i=new ICICI();
Axis a=new Axis ();
System.out.println("SBI rate of interest=" + s.getROI());
System.out.println("ICICI rate of interest=" + i.getROI());
System.out.println("Axis rate of interest=" + a.getROI());
Bank b=new ICICI ();
System.out.println("Example of Base class can hold the object of any child class:"+b.getROI());
/*Reference of Base class can hold the object of any child class.But vice-versa as shown below is
not possible.**/
/*ICICI i1=new Bank ();
System.out.println("ROI:"+i1.getROI());*/
//Here i1 is reference of ICICI bank but it is trying to hold object of base classwhich
//is not allowed.
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac NewMethodOverriding.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java NewMethodOverriding
SBI rate of interest=8
ICICI rateof interest=7
Axis rate of interest=9

Example of Base class reference can hold the object of any child class = 7
```

Task 4(b): Write a Java Program to demonstrate use of Dynamic Method

Aim: To write a Java Program to demonstrate use of DynamicMethod

Program:

```
Dispatch class A
{
void callme()
{

System.out.println("Inside A's callme method");

}

}
class B extends A
{
// override callme()

void callme()

{

System.out.println("Inside B's callme method");


}

}

class C extends A
{
// override callme()

void callme()

{

System.out.println("Inside C's callme method");

}
}
class Dispatch
{
public static void main(String args[])
{
A a = new A(); // object of type A
B b = new B(); // object of type B
C c = new C(); // object of type C

A r; // obtain a reference of type

A r = a; // r refers to an A object
r.callme(); // calls A's version of callme
r = b; // r refers to a B object
r.callme(); // calls B's version of callme
r = c; // r refers to a C object
 r.callme()// calls C's version of callme

}

}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac Dispatch.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java Dispatch
Inside A’s callme method
Inside B’s callme method
Inside C’s callme method
```

Task 4(c): Write a java program that implement multiple inheritance

Aim: To write a java program that implement multiple inheritance

Program:

```
import java.lang.*;

import java.io.*;
interfaceExam
{
void percent_cal();
}
class Student
{
String name;
int roll_no,mark1,mark2;
Student(String n, int r, intm1, int m2)
{
name=n;
roll_no=r;
 mark1=m1;
 mark2=m2;

}
void display()
{
System.out.println ("Name of Student: "+name);
System.out.println ("Roll No. of Student: "+roll_no);
System.out.println ("Marks of Subject 1: "+mark1);
System.out.println ("Marks of Subject 2: "+mark2);
}
}
class Result extends Student implements Exam
{
Result(String n, int r, int m1, int m2)
{
super(n,r,m1,m2);
}
public void percent_cal()
{

int total=(mark1+ mark2);
 float percent=total*100/200;
System.out.println ("Percentage: "+percent+"%");
}
void display()
{
```

```
super.display();
}
}
class q10Multiple
{
public static void main(String args[])
{
Result R = new Result("Ra.one",12,93,84);
R.display();
R.percent_cal();
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac q10Multiple.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java q10Multiple
Name of Student: Ra.one
Roll No. of Student: 12
Marks of Subject 1: 93
Marks of Subject 2: 84
Percentage: 88.0%
```

Task 4(d): Write a java program that implements all Accessspecifiers.

Aim: To write a java program that implements all Access specifiers.

Program:

```
class A
{

Private int a=5;

publicint b=6;
protected int c=1;int d=2;
void print()
{
a=20;
System.out.println(a+" "+b+" "+c+" "+d);
}
}
class B
{
A x=new A();void print1()
{
x.b=7;//accessing public variable of class A
//x.a=30;error  can't  access  private  variableoutside the class
x.d=35;//accessing  default  access  variableoutside the class
x.print();
}

}
class C
{
public static void main(String []args)
{
B y=new B();y.print1();
}
}
```

OUTPUT:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac C.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java C
20 7 1 35
```

TASK 5

Task 5(a): Write a Java program that reads a file name from the user, then display information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes

Aim: To write a Java program that reads a file name from the user, then display Information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes

Program:

```
import java.io.*;
 import java.util.*;
class Week5a
{
public static void main (String [] args) throws IOException
{
BufferedReader br =new BufferedReader(new InputStreamReader(System.in));
System.out.println("Enter a file name");
String s=new String(br.readLine());
File f=new File(s);
if(f.exists())
System.out.println("File exists");
if(f.canRead())
System.out.println("File isreadable");
if(f.canWrite())
System.out.println("File iswritable");
if(f.isFile())
System.out.println("File iswritable");
else if(f.isDirectory())
System.out.println("It isdirectory");
System.out.println("The length of the file is "+length()+" bytes");
System.out.println("path:"+f.getAbsolutePath());
long l=f.lastModified();
Date d=new  Date(l);
int date=d.getDate();
int month=d.getMonth();
int year=d.getYear();
int hh=d.getHours();
int mm=d.getMinutes();
int ss=d.getSeconds();
System.out.println(date+"/"+(month+1)+"/"+(1900+year));
System.out.println(hh+":"+mm+":"+ss);
}
}
```

Output:

```
D:\education\java\programs>javac Week5a.java
D:\education\java\programs>java Week5a
Enter file name D:\education\java\programs\hello.java
File exists
File is readable

 File is writable

It is a file

The length of the file is 173 bytes
```

Task 5(b): Write a Java program that reads a file and displays the file on thescreen, with a line number before each line

Aim: To write a Java program that reads a file and displays the file on thescreen, with a line number before each line

Program:

```
import java.io.*;

class Week5b

{

public static void main(String[] args) throws IOException
{
BufferedReaderbr=new BufferedReader(new InputStreamReader(System.in));
System.out.println("Enter a file name");
String s=br.readLine();
int s1,i=1;
FileInputStream fin=new FileInputStream(s);
System.out.println(i+++" ");
do
{
s1= fin.read();if(s1 !=-1)
System.out.println((char)s1);

if((char)s1=='\n')


System.out.println(i+++" ");

}while(s1!= -1);

fin.close();

}

}
```

Output:

```
D:\education\java\programs>javac Week5b.java
D:\education\java\programs>java Week5b
Enter a file nameA.java
1 class A
2 {
3 public static void main(String args[])
4{
5System.out.println("hi");
6}
7 }
```

Task 5(c): Write a Java program that displays the number of characters, linesand words in a text file.

Aim: To write a Java program that displays the number of characters, lines and words in a text file

Program:

```
import java.io. *;

class task5c

{
public static void main (String [] args) throws IOException
{

BufferedReader br=new BufferedReader(newInputStreamReader(System.in));
System.out.println("Enter a file name");
String s=br.readLine();
int s1,ch=0,ln=0,wd=0;
FileInputStream fin=new FileInputStream(s);do
{
s1= fin.read();if(s1 !=-1)
ch++;
if((char)s1==' '||(char)s1=='\n')
wd++;
if((char)s1=='\n')
ln++;

} while(s1!= -1);
System.out.println("The number of characters is"+ch);
System.out.println("The number of words is"+wd);
System.out.println("The number of lines is"+ln);

}
}
```

Output:

```
D:\education\java\programs>javac task5c.java
D:\education\java\programs>java task5c
Enter a file nameA.java
The number of characters are 6800
The number of wordsare 580
The number of lines are264
```

TASK 6

Task 6(a): Write a java program for handling Checked Exceptions

Aim: To write a java program for handling Checked Exceptions

Program:

```
import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;
public class CheckedExceptionExample
{
private static String filepath = "E:/Demo/ArrayBR.java";
public static void main(String[] args)
{
BufferedReader br = null;
 String line;
try
{
br = new BufferedReader(new FileReader(filepath));
while ((line = br.readLine()) != null)
{
System.out.println(line);
 }
 } catch (IOException e)
 {
System.err.println("An IOException was caught:"+e.getMessage());
 }
/** finally
{
try
{
if(br != null)
br.close();
  } catch (IOException e)
{
e.printStackTrace();
  }
} */
} }
```

Output:

/* if the specified file in the file path string variable doesn’t exist, it throws an IOException stating that “the system cannot find the specified file” otherwise this program prints thecontents of the file*/

Task 6(b): Write a java program for handling Unchecked Exceptions

Aim: To write a java program for handling Unchecked Exceptions

Program:

```
class ArrayIndexOutOfBound
{
public static void main(String args[])
{
try
{
int arr[] ={1,2,3,4,5};
System.out.println(arr[7]);
}catch(ArrayIndexOutOfBoundsException e)
{
System.out.println("The specified index does not exist”+ "in array. Please correct
the  error.");
}
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac ArrayIndexOutOfBound.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java ArrayIndexOutOfBound
The specified index does not exist in array. Please correct the error.
```

Program:

```
class ClassNotFoundExceptionExample
{
public static void main(String args[])

{

try
{

Class x= Class.forName("UE");
System.out.println("Specified Class " + x + " found successfully!");
}
catch(ClassNotFoundException e)
{
//System.out.println("Sorry, below specified class is not found");
System.err.println("A ClassNotFoundException was caught: " +
e.getMessage());
e.printStackTrace();
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac ClassNotFoundExceptionExample.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java ClassNotFoundExceptionExample
Specified class java.lang.String found successfully!
```

Program:

```
class DivideByZero
{
public static void main(String args[])
{
int x = 0; int y =10;
try
{
  int z = y/x;
 }
catch(ArithmeticException ae)
 {
    System.out.println("Divisor cannot be zero");
}
}
 }
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac DivideByZero.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>java DivideByZero
Divisor cannot be zero
```

TASK 7

Task 7(a): Write a Java program that creates three threads. First thread displays "Good Morning" every one second, the second thread displays "Hello" every two seconds and the third thread displays "Welcome" every three seconds

Aim: To write a Java program that creates three threads. First thread displays"Good Morning" every one second, the second thread displays "Hello" every two seconds and the third thread displays "Welcome" every three seconds

Program:

```
class FT implements Runnable
{
public void run()
{
while(true)
{
try
{
System.out.println("Good Morning");
Thread.sleep(1000);
}
catch(InterruptedException e)
{
System.out.println("First Thread is interrupted when sleeping");
}
}
}
}

class ST implements Runnable
{
public void run()
{
while(true)
{
 try
{
System.out.println("Hello");
Thread.sleep(2000);
}
catch(InterruptedException e)
{
 System.out.println("Second Thread is interrupted when sleeping");
}}
}}
class TT implements Runnable
{
public void run()
{
while(true)
{
try
{
System.out.println("Welcome");
Thread.sleep(3000);
}
catch(InterruptedException e)
{
System.out.println("Third Thread is interrupted when sleeping");
}
}
}
}
public class task7a
{
 public static void main(String[] args)
{
FT ft=new FT();
ST st = new ST();
 TT tt=new TT();
 Thread thread1=new Thread(ft);
 thread1.start();
Thread thread2=new Thread(st);
thread2.start();
Thread thread3=new Thread(tt);
thread3.start();
}
}
```

Output:

```
D:\education\java\programs>javac task7a.java
D:\education\java\programs>java task7b GoodMorning
Hello
Welcome
GoodMorning
Hello
GoodMorning
Welcome
```

Task 7(b):

Write a Java program that correctly implements producer consumer problemusing the concept of inter thread communication

Aim: To write a Java program that correctly implements producer consumer problem using the concept of inter thread communication

Program

```
class Q

int n;
boolean valueSet=false;
synchronized int get()
if(!valueSet)
try
{
wait();
}
catch(InterruptedException e)
{
System.out.println("Interrupted Exception caught");
}
System.out.println("Got:"+n);
valueSet=false;
notify();
return n;
}
synchronized void put(int n)
{

if(valueSet)
try
{
wait();
}
catch(InterruptedException e)
{
System.out.println("Interrupted Exception caught");
}
this.n=n; valueSet=true;
System.out.println("Put:"+n);notify();
}
}

class Producer implements Runnable
{
```

```
Q q;
Producer(Q q)
{
this.q=q;
new Thread(this,"Producer").start();
}
public void run()
{
int i=0;
while(true)
{
q.put(i++);
 }
 }
 }
class Consumer implements Runnable
{

Q q;
Consumer(Q q)
{
this.q=q;
new Thread(this,"Consumer").start();
}
public void run()
{
while(true)
{
q.get();
}
}
}
class task7b
{
public static void main(String[] args)
{

Q q=new Q(); new Producer(q); new Consumer(q);
System.out.println("Press Control-c to stop");
}
}
```

Output:

```
D:\education\java\programs>javac task7b.java
D:\education\java\programs>java  task7b
Put:1
Got:1
Put:2
Got:2
Put:3
Got:3
Put:4
Got:4
Put:5
Got:5
```

TASK 8

Task 8: Write a program illustrating following collections framework

Task 8(a): Write a program using ArrayList

Aim: Write a program using ArrayList

Program:

```
import java.util.*;
public class JavaExample {
public static void main(String args[]) {
/* Creating ArrayList of type "String" which means
* we can only add "String" elements

*/
ArrayList<String> obj = new ArrayList<String>();
/*This is how we add elements to an ArrayList*/

obj.add("Ajeet");
obj.add("Harry");
obj.add("Chaitanya");
obj.add("Steve");
obj.add("Anuj");

// Displaying elements
System.out.println("Original ArrayList:");
for(String str:obj)
System.out.println(str);

/* Add element at the given index
*                    obj.add(0, "Rahul") - Adding element "Rahul" at first position
*                    obj.add(1, "Justin") - Adding element "Justin" at second position
*/
obj.add(0, "Rahul");
obj.add(1, "Justin");
// Displaying elements
System.out.println("ArrayList after add operation:");

for(String str:obj)
System.out.println(str);
//Remove elements from ArrayList like this

obj.remove("Chaitanya"); //Removes "Chaitanya" from ArrayList

obj.remove("Harry"); //Removes "Harry" from ArrayList
// Displaying elements
System.out.println("ArrayList after remove operation:");

for(String str:obj)
System.out.println(str);
//Remove element from the specified index

obj.remove(1); //Removes Second element from the List
// Displaying elements

System.out.println("Final ArrayList:");

for(String str:obj)

 System.out.println(str);
}
}
```

Output:

```
Original ArrayList:
Ajeet Harry ChaitanyaSteve Anuj
ArrayList after add operation:
Rahul Justin Ajeet Harry ChaitanyaSteve Anuj
ArrayList after remove operation:
RahulJustinAjeetSteveAnuj
Final ArrayList:Rahul
Ajeet Steve

Anuj
```

Task 8(b): Write a program using Vector

Aim: To Write a program using Vector

Program:

```
import java.util.*;
public class VectorExample {
public static void main(String args[]) {
/* Vector of initial capacity(size) of 2 */

Vector<String> vec = new Vector<String>(2);
/* Adding elements to a vector*/
vec.addElement("Apple");
 vec.addElement("Orange");
vec.addElement("Mango");
vec.addElement("Fig");
/* check size and capacityIncrement*/
System.out.println("Size is: "+vec.size());
System.out.println("Default capacity increment is: "+vec.capacity());
vec.addElement("fruit1");
vec.addElement("fruit2");
vec.addElement("fruit3");
/*size and capacityIncrement after two insertions*/
System.out.println("Size after addition: "+vec.size());
System.out.println("Capacity after increment is: "+vec.capacity());
/*Display Vector elements*/
Enumeration en = vec.elements();
System.out.println("\nElements are:");
while(en.hasMoreElements())
System.out.print(en.nextElement() + " ");
}
}
```

Output:

```
Size is: 4
Default capacity increment is: 4

Size after addition: 7

Capacity after increment is:

Elements are:

Apple Orange Mango Fig fruit1 fruit2 fruit3
```

Task 8(c): Write a program using HashTable

Aim: Write a program using HashTable

Program:

```
import java.util.Hashtable;
import java.util.Enumeration;
 public class HashtableExample
{
public static void main (String[] args)
{
Enumeration names;
String key;
// Creating a Hashtable

 Hashtable<String, String>

hashtable =new Hashtable<String, String>();


// Adding Key and Value pairs to Hashtablehashtable.put("Key1","Chaitanya");
hashtable.put("Key2","Ajeet");
hashtable.put("Key3","Peter");
hashtable.put("Key4","Ricky");

hashtable.put("Key5","Mona");

names = hashtable.keys();

while(names.hasMoreElements())

 {

 key = (String) names.nextElement();
System.out.println("Key: " +key+ " & Value: " +hashtable.get(key));
}
}
}
```

Output:

```
Key: Key4 & Value: Ricky

Key: Key3 & Value: Peter

Key: Key2 & Value: Ajeet

Key: Key1 & Value: Chaitanya
Key: Key5 & Value: Mona
```

Task 8(d): Write a java program using Stack

Aim: Write a program using Stack

Program:

```
import java.util.*;
class StackDemo
{
public static void main(String args[])
{
StackDemo s=new StackDemo();

System.out.println("content of s="+s);

System.out.println("size of s="+s.size()); //10

System.out.println("Is empty?="s.empty());  //true
//add the data to ss.push(10);
s.push(20);
s.push(30);
s.push(40);
System.out.println("content of s="+s); //[10,20,30,40]

System.out.println("size of s="+s.size()); //4

System.out.println("Is s empty ?=s.empty()");  //false

//remove the top most element

System.out.println("delete element="+s.pop());  //40
System.out.println("content of s after pop="+s);//  [10,20,30]
//extract the top most element
System.out.println("top most element="+s.peek()); //30

 System.out.println("content of s after peek="+s);//[10 20 30]
//Search the element 10 and 100int srp=s.search(10);
System.out.println("stack relative pos.of 10 is="+srp);//3
int srp1=s.search(100);
System.out.println("stack relative pos.of 100 is="+srp1);//-1
}
}
```

Output

```
Javac StackDemo.java

Java StackDemo
40 30 20 10
stack[] push(10) stack[10] push(20) stack[10 20] push(30) stack[10 20 30]push(40)
stack[10 20 30 40 ]
40
stack[10 20 30 ]
30
stack[ 10 20 ]
20
stack[ 10 ]
10
Stack[  ]
Stack is empty
```

TASK 9

Task 9(a): Develop an applet that displays a simple message.

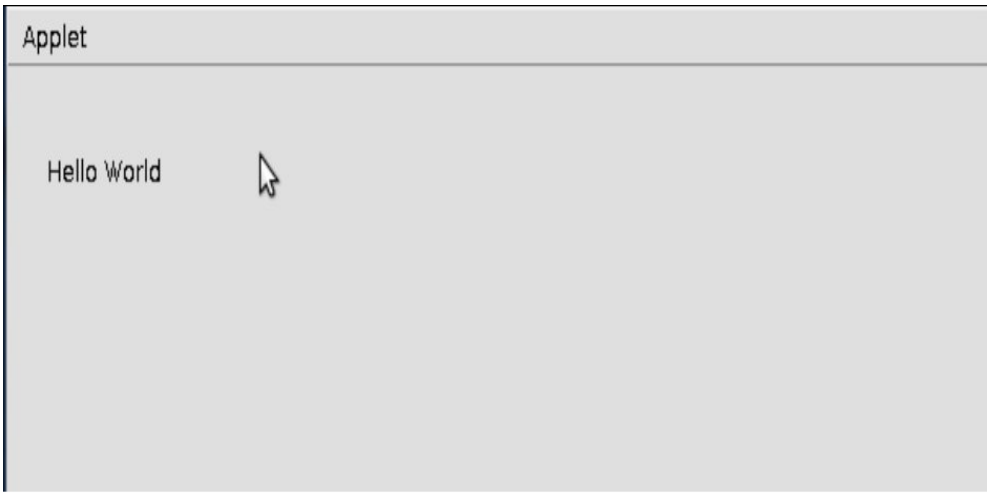
Aim: Develop an applet that displays a simple message.

Program:

```
import java.applet.*;
import java.awt.*;
/*<applet code="task9a.class" height=200 width=200>
</applet>*/
public class task9a extends Applet
{
 public void paint(Graphics g)
{
g.drawString("hello world",25,50);
}
}
```

Output:

```
D:\education\java\programs>javac task9a.java

D:\education\java\programs>appletviewer task9a.java
```



Hello World

Task 9(b): Develop an applet that receives an integer in one text field and compute its factorial value and return it in another text field, when the button named “Compute” is clicked.

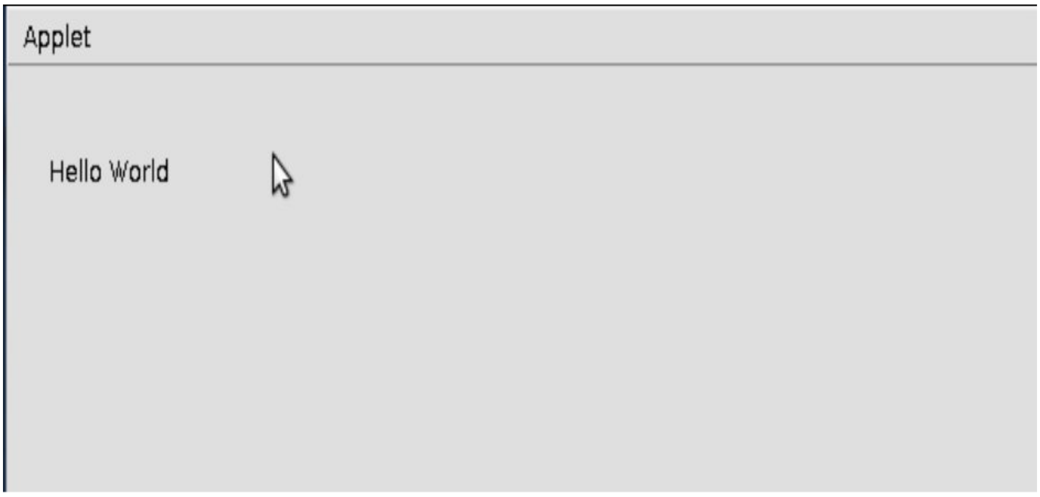
Aim: Develop an applet that receives an integer in one text field and compute its factorial value and return it in another text field, when the button named “Compute” is clicked.

Program:

```
import java.applet.*;
import java.awt.*;
import javax.swing.*;
/*<applet code="task9b.class" height=200 width=200></applet>*/
public class task9b extends Applet implements ActionListener
{
TextField t1,t2;
Label l1,l2;
Button b;
int a,fact;
public void init()
{
l1=new Label("enter a number");
t1= new TextField(5);
l2=new Label("Factorial of the given number is :");
t2=new TextField(10);
b=new Button("compute");
add(l1);
add(t1);
add(b);
add(l2);
add(t2);
b.addActionListener(this);
}
public void actionPerformed(ActionEvent e)
{
a=Integer.parseInt(t1.getText());
fact=1;
if(a<0)
{
t2.setText("wrong Input");
}

else
{
for(int i=a;i>1;i--)
fact*=i;
t2.setText(""+fact);
}
}
}
```

Output:

```
D:\education\java\programs>javac task9b.java

D:\education\java\programs>appletviewer task9b.java
```



Hello World

Task 9(c): Write a Java program that works as a simple calculator. Use a gridlayout to arrange buttons for the digits and for the +, -, *, % operations. Add a text field todisplay the result

Aim: To write a java program that works as a simple calculator

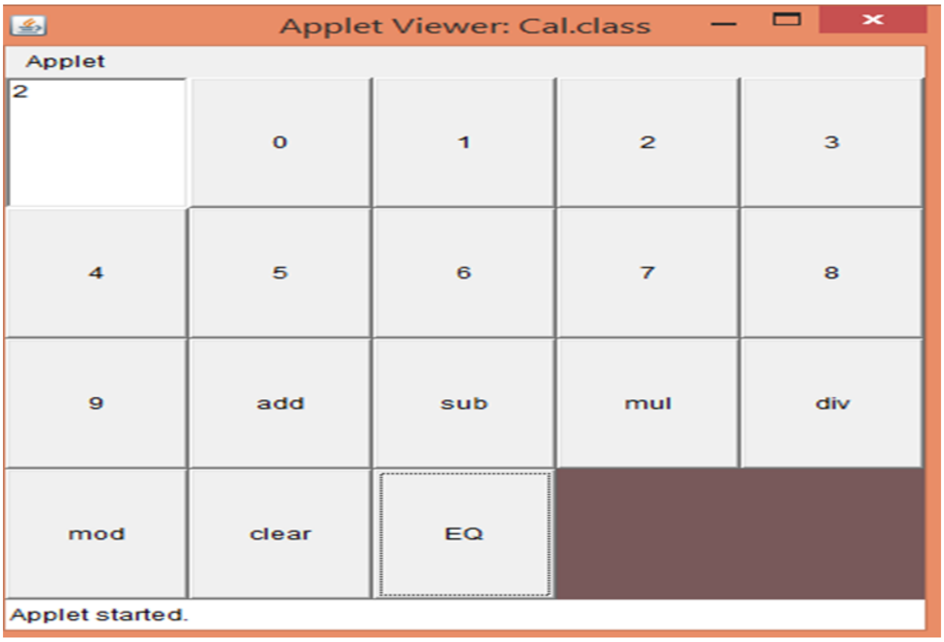
Program:

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code="Cal" width=300 height=300>
</applet>
*/
public class Cal extends Applet implements ActionListener
{
String msg=" ";int v1,v2,result;TextField t1;
Button b[]=new Button[10];
Button add,sub,mul,div,clear,mod,EQ;
char OP;
public void init()
{
Color k=new Color(120,89,90);
setBackground(k);
t1=new TextField(10);
GridLayout gl=new GridLayout(4,5);
setLayout(gl);
for(int i=0;i<10;i++)
{
b[i]=new Button(""+i);
}
add=new Button("add");
sub=new Button("sub");
 mul=new Button("mul");
div=new Button("div");
mod=new Button("mod");
clear=new Button("clear");
EQ=new Button("EQ");
t1.addActionListener(this);
 add(t1);
for(int i=0;i<10;i++)
{
add(b[i]);
add(add);
add(sub);
add(mul);
add(div);
add(mod);
add(clear);
add(EQ);
for (int i=0;i<10;i++)
{
b[i].addActionListener(this);
}

add.addActionListener(this);
sub.addActionListener(this);
mul.addActionListener(this);
div.addActionListener(this);
mod.addActionListener(this);
clear.addActionListener(this);
EQ.addActionListener(this);
}
public void actionPerformed(ActionEvent ae)
{
String str=ae.getActionCommand();
char ch=str.charAt(0);
if ( Character.isDigit(ch))
 t1.setText(t1.getText()+str);
else if(str.equals("add"))
{
v1=Integer.parseInt(t1.getText());
OP='+';
t1.setText("");
}
else if(str.equals("sub"))
{
v1=Integer.parseInt(t1.getText());
OP='-';
t1.setText("");
}
else if(str.equals("mul"))
{

v1=Integer.parseInt(t1.getText());

OP='*';
t1.setText("");
}
else if(str.equals("div"))
{
v1=Integer.parseInt(t1.getText());
OP='/';
t1.setText("");
}
else if(str.equals("mod"))
```

```
{
v1=Integer.parseInt(t1.getText());
OP='%';
t1.setText("");
}
if(str.equals("EQ"))
{
v2=Integer.parseInt(t1.getText());
if(OP=='+')
result=v1+v2;else if(OP=='-')
result=v1-v2;else if(OP=='*')
result=v1*v2;else if(OP=='/')
result=v1/v2;else if(OP=='%')
result=v1%v2;
 t1.setText(""+result);
}
if(str.equals("clear"))
{
t1.setText("");
}
}
}
```

Output:

```
D:\education\java\programs>javac Cal.java

D:\education\java\programs>appletviewer Cal.java
```



TASK 10

Task 10(a): Write a java program for handling mouse events.

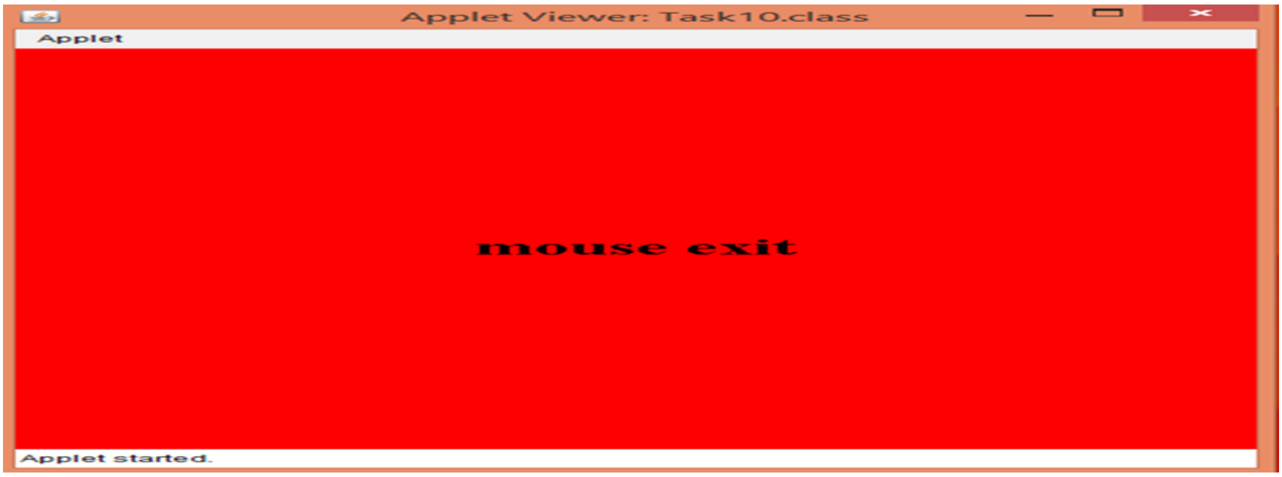
Aim: To write a java program for handling mouse events.

Program:

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

/*<applet code="Task10.class" width=400 height=350></Applet>*/

public class Task10 extends Applet implements MouseListener
{
String msg=""; public void init()
{
addMouseListener(this);
}
public void mouseClicked(MouseEvent obj)
{
msg="mouse clicked";
repaint();
}
public void mouseEntered(MouseEvent obj)
{
msg="mouse entered";
repaint();
}
public void mouseExited(MouseEvent obj)
{
msg="mouse exit";
repaint();
}
public void mousePressed(MouseEvent obj)
{
msg="mouse pressed";
repaint();
}
public void mouseReleased(MouseEvent obj)
{
msg="mouse released";
repaint();
}
public void mouseMoved(MouseEvent obj)
{
msg="mouse moved";
repaint();
}
public void mouseDragged(MouseEvent obj)
{
msg="mouse Dragged";
repaint();
}
public void paint(Graphics g)
{
Font currentFont=new Font("TimesRoman",Font.BOLD,29);
g.setFont(currentFont);
//FontMetrics, Dimension class are used to display the text in themiddle of the
screen.
FontMetrics fm=g.getFontMetrics();
Dimension d= getSize();
int xc,yc;
xc=d.width/2- fm.stringWidth(msg)/2;
yc=d.height/2+fm.getDescent();
//g.drawString(msg,50,50);
g.drawString(msg,xc,yc);
setBackground(Color.red);
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac Task10.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>appletviewer Task10.java
```



mouse exit

Applet started.

Task 10(b): Write a java program for handling key events

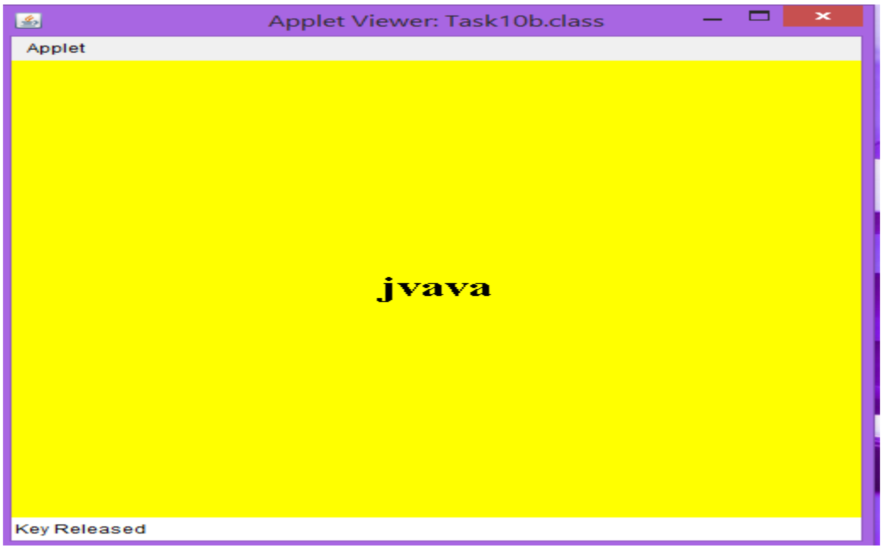
Aim: To write a java program for handling key events

Program:

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

/*<applet code="Task10b.class" width=400 height=350></Applet>*/

public class Task10b extends Applet implements KeyListener
{
String msg=""; public void init()
{
addKeyListener(this);
//Register Listener with Applet.requestFocus();

//requestFocus() method is present in the Component class. This method makesa request to obtain
the Input Focus for the current program.
//Otherwise program will not receive any keyboard events.
}
public void keyPressed(KeyEvent obj)
{
showStatus("Key Pressed");
}
public void keyReleased(KeyEvent obj)
{
showStatus("Key Released");
}
public void keyTyped(KeyEvent obj)
{
showStatus("Key Typed");
msg=msg+obj.getKeyChar();
 repaint();
//getKeyChar() returns the key typed as a character.
//showStatus(String msg) method displays the message on the status bar of theapplet window.
}

public void paint(Graphics g)
{
Font currentFont=new Font("TimesRoman",Font.BOLD,29);
g.setFont(currentFont);
//FontMetrics, Dimension class are used to display the text in the middleof the screen
FontMetrics fm=g.getFontMetrics();
Dimension d= getSize();
int xc,yc;
xc=d.width/2-fm.stringWidth(msg)/2;
yc=d.height/2+fm.getDescent();
//g.drawString(msg,50,50); g.drawString(msg,xc,yc); ssetBackground(Color.red);
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac Task10b.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>appletviewer  Task10b.html
```



TASK 11

Task 11: Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the textfields, Num1 and Num2. The division of Num1 and Num2 isdisplayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer,the program would throw a NumberFormatException. If Num2 were Zero, the programwould throw an ArithmeticException. Display the exception in a message dialog box.

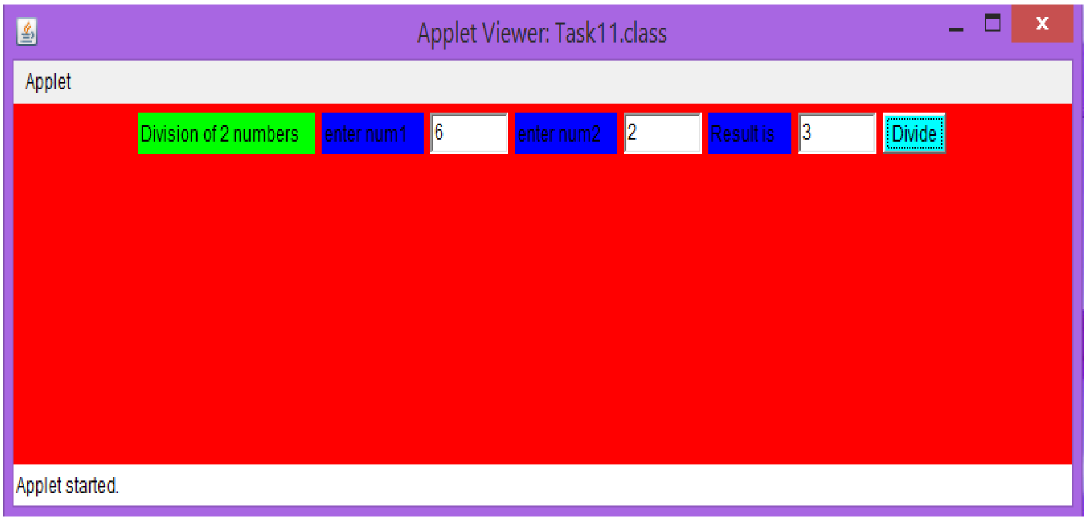
Aim: To create a user interface for division of two numbers

Program:

```
import java.awt.*;
import javax.swing.*;
import java.applet.Applet;
import java.awt.event.*;
public class Task11 extends Applet implements ActionListener
{
TextField t1,t2,t3;
Button b;
Label L1,L2,L3,L4;
String s;
Task11 e;
public void init()
{
e=this;
//setLayout(new GridLayout(3,2));
t1=new TextField(5);
t2=new TextField(5);
t3=new TextField(5);
L1=new Label("enter num1");
L2=new Label("enter num2");
L3=new Label("Result is");
L4=new Label("Division of 2 numbers");
b=new Button("Divide");
add(L4);
add(L1);
add(t1);
add(L2);
add(t2);
add(L3);
add(t3);
add(b);
 b.addActionListener(this);
}
public void paint(Graphics g)
{
setBackground(Color.red);
setForeground(Color.black);
L1.setBackground(Color.blue);
L2.setBackground(Color.blue);
L3.setBackground(Color.blue);
L4.setBackground(Color.green);
//L4.setAlignment(Label.CENTER);
b.setBackground(Color.cyan);
}
public void actionPerformed(ActionEvent ae)
{
try
{
int num1=Integer.parseInt(t1.getText());
int num2=Integer.parseInt(t2.getText());s=""+(num1/num2);
t3.setText(s);
}
catch(ArithmeticException a)
{
JOptionPane.showMessageDialog(null,"Divide by zero");
}
catch(NumberFormatException b)
{
JOptionPane.showMessageDialog(null,"Number FormateException");
}
}
}
```

Output:

```
C:ProgramFiles/Java/jdk1.7..0_09/bin>javac Task11.java
C:ProgramFiles/Java/jdk1.7..0_09/bin>appletviewer Task11.java
```



TASK 12

Task 12(a): Write a java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time. No light is on when the program starts.

Aim: To write a java program that simulates a traffic light.

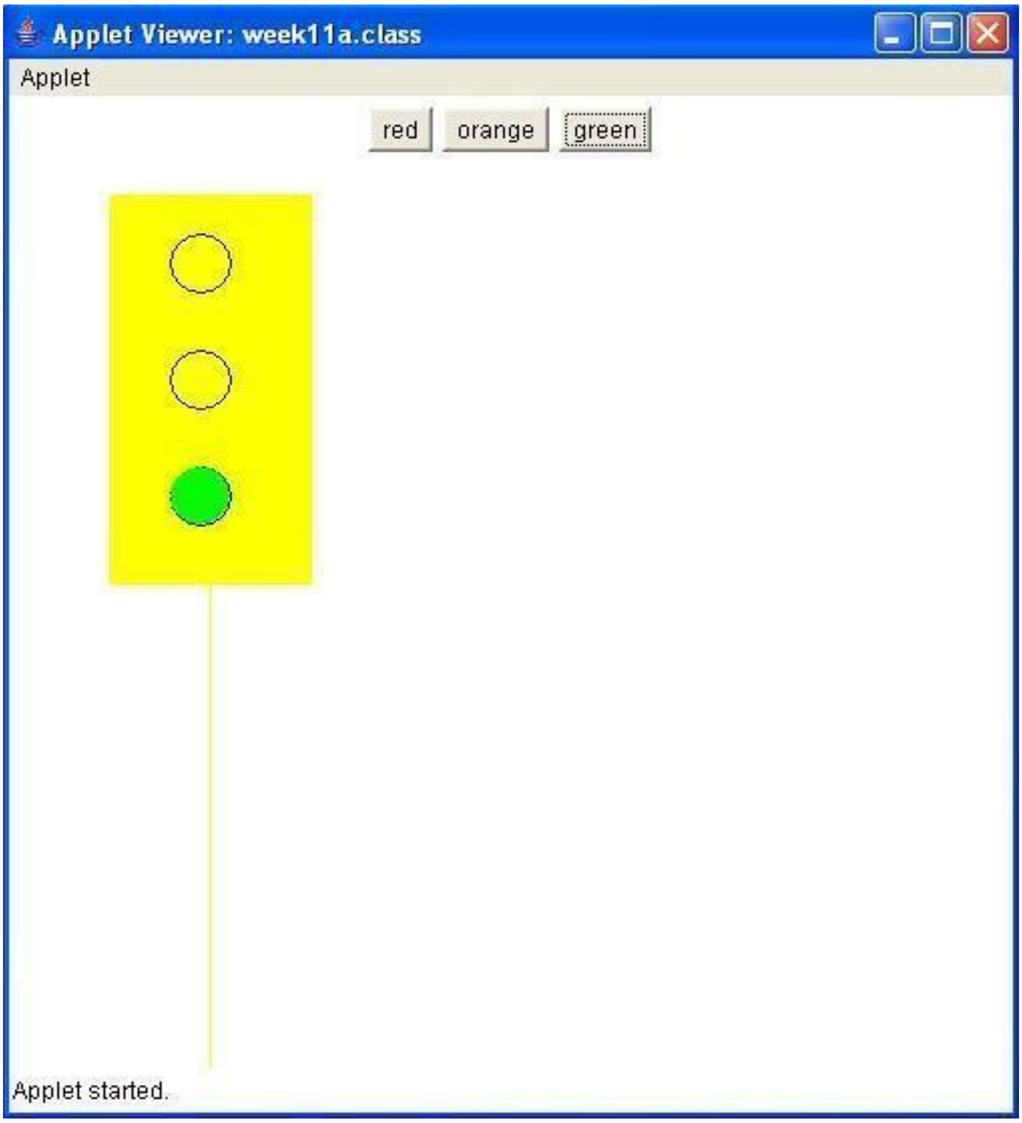
Program:

```
import java.awt.*;

 import java.applet.*;
import java.awt.event.*; /*
<applet code="task12a.class" width=500 height=500> </applet>
*/
public class task12a extends Applet implements ActionListener
{

int i=0;
Button R,O,G;
public void init()
{
setBackground(Color.white);
setForeground(Color.black);
R=new Button("red");
O=new Button("orange");
 G=new Button("green");
add(R);
add(O);
add(G);
R.addActionListener(this);
O.addActionListener(this);
G.addActionListener(this);
}
public void actionPerformed(ActionEvent ae)
{
String s=ae.getActionCommand();
if(s.equals("red"))
i=1;
if(s.equals("orange"))
i=2;
if(s.equals("green"))
i=3;
repaint();
}

public void paint(Graphics g)
{
```

```
g.setColor(Color.yellow);
g.drawRect(50,50,100,200);
g.fillRect(50,50,100,200);
g.setColor(Color.black);
g.drawOval(80,70,30,30);
g.drawOval(80,130,30,30);
g.drawOval(80,190,30,30);
g.setColor(Color.yellow);
g.drawLine(100,250,100,900);
if(i==1)
{
g.setColor(Color.red);
g.fillOval(80,70,30,30);
}
if(i==2)
{
g.setColor(Color.orange);
g.fillOval(80,130,30,30);
}
if(i==3)
{
g.setColor(Color.green);
g.fillOval(80,190,30,30);
}
}
}
```

Output:

```
Javac task12a.java
Appletviewer task12a.java
```



Applet started.

Task 12(b): Write a Java program that allows the user to draw lines,rectangles and ovals.

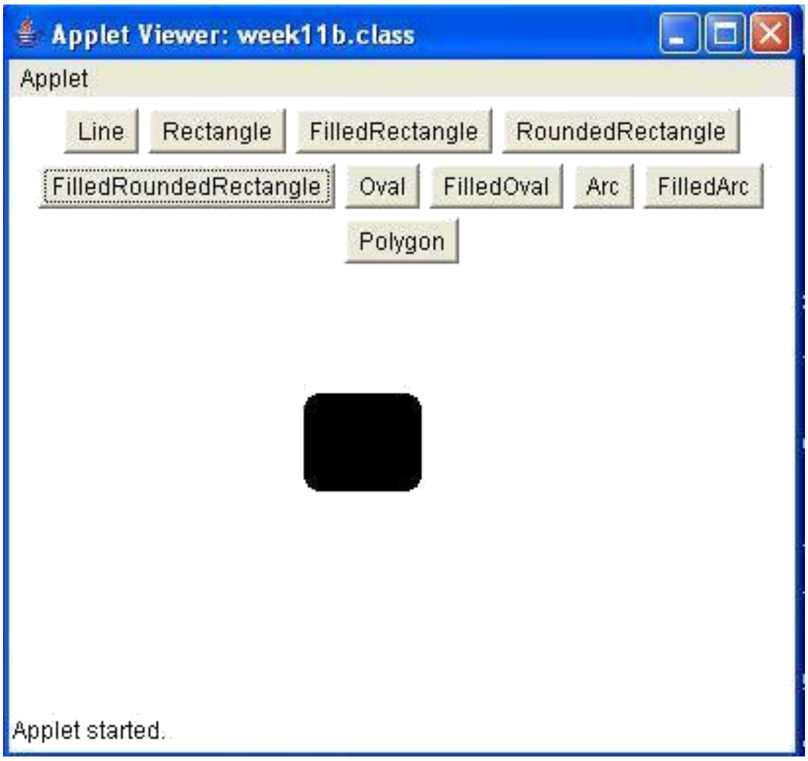
Aim: To write a Java program that allows the user to draw lines, rectanglesand ovals.

Program:

```
import java.applet.*;

import java.awt.event.*;

import java.awt.*;

/*<applet code="task12b.class" height=310 width=400> </applet>*/
public class task12b extends Applet implements ActionListener
{
Button b[]=new Button[10];
int in;
public void init()
{

b[0]=new Button("Line");
b[1]=new Button("Rectangle");
b[2]=new Button("FilledRectangle");
b[3]=new Button("RoundedRectangle");
b[4]=new Button("FilledRoundedRectangle");
b[5]=new Button("Oval");
b[6]=new Button("FilledOval");
b[7]=new Button("Arc");
 b[8]=new Button("FilledArc");
b[9]=new Button("Polygon");
for(int i=0;i<10;i++)
{
add(b[i]);
b[i].addActionListener(this);
}
}
public void actionPerformed(ActionEvent ae)
{
for(int j=0;j<10;j++)
{

if(ae.getSource()==b[j])
{

in=j+1; break;
}
}
repaint();
}
public void paint(Graphics g)
{
if(in==1)
g.drawLine(150, 150, 250,300);
if(in==2)
g.drawRect(150, 150, 60,50);
if(in==3)
g.fillRect(150,150, 60,50);
if(in==4)
g.drawRoundRect(150, 150, 60, 50, 15, 15);
if(in==5)
g.fillRoundRect(150, 150, 60, 50, 15, 15);
if(in==6)
g.drawOval(150, 150, 60, 50);
if(in==7)
g.fillOval(150, 150, 60,50);
if(in==8)
g.drawArc(150, 150, 60, 50, 0, 75);
if(in==9)
g.fillArc(150, 150, 60, 50, 0,75);
if(in==10)
{
int xpoints[] = {50, 200,250, 250,200};
int ypoints[] = {250, 200,250,300,300};
int num = 5;
g.drawPolygon(xpoints, ypoints, num);
}
}

}
```

Output:

```
D:\>javac task12b.java
D:\>appletviewer task12b.java
```



Applet started.

TASK 13

Task 13: Create a table in Table.txt file such that the first line in the file is the headerand the remaining lines corresponds to rows in the table. The elements are separatedby commas. Write a java program to display the table using JTable component.

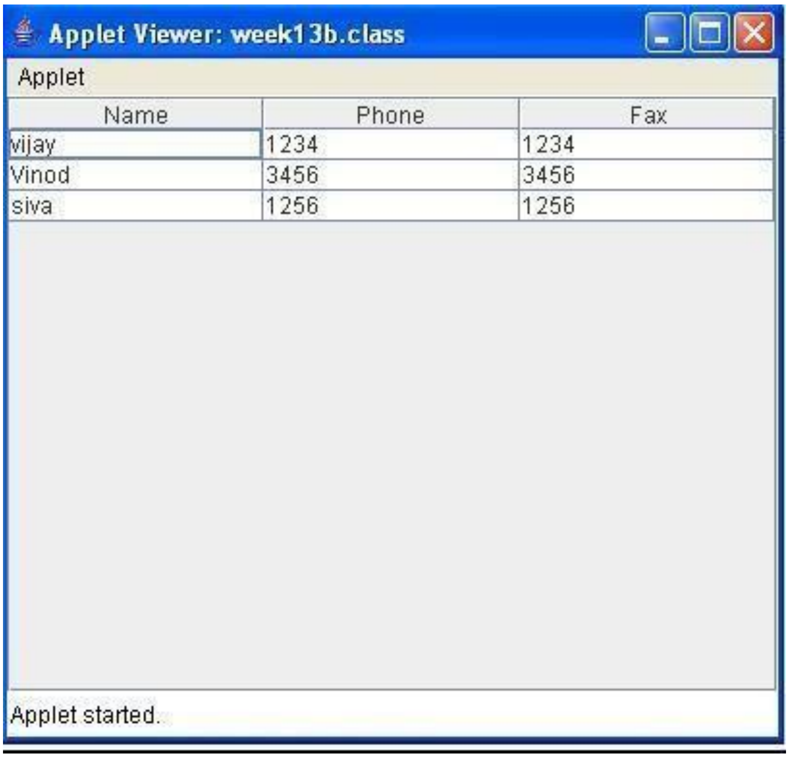
Aim: To write a java program to display the table using JTable component.

Program:

```
import java.applet.*;

import java.awt.event.*;

import java.awt.*;
/*<applet code="task13.class" height=310 width=400></applet>*/
public class task13 extends Japplet
{
public void init()
{
Container contentpane=getContentPane();

Contentpane.setLayout(new  BorderLayout());

final String[] colheads={“Name”,”Phone”,”Fax”};

final object[][]data={{“vijay”,”1234”,”1234”},{“Vinod”,”3456”,”3456”},

{“siva”,”1256”,”1256”}};
Jtable table=new Jtable(data,colheads);

int v=ScrollPaneConstants.VERTICAL_SCROLLBAR_AS_NEEDED;

int h=ScrollPaneConstants.HORIZANTAL_SCROLLBAR_AS_NEEDED;

JScrollPane jsp=new JScrollpane(table,v,h);

contentPane.add(jsp,BorderLayout.CENTER);
}
}
```

Output:

```
D:\>javac task13.java
 D:\>appletviewer task13.java
```



Name	Phone	Fax
vijay	1234	1234
Vinod	3456	3456
siva	1256	1256

Applet started.