SPIT109/SPCA111

POSTGRADUATE COURSE

M.Sc., IT/MCA

FIRST YEAR

SECOND SEMESTER

CORE PAPER - VIII / PAPER - X

PRACTICAL-III : JAVA PROGRAMMING LAB
PRACTICAL-IV : PROGRAMMING IN JAVA LAB

INSTITUTE OF DISTANCE EDUCATION
UNIVERSITY OF MADRAS

MCA/Msc., IT
FIRST YEAR - SECOND SEMESTER

CORE PAPER - VIII/ PAPER - X
PRACTICAL-III : JAVA PROGRAMMING LAB
PRACTICAL-IV : PROGRAMMING IN JAVA LAB

WELCOME

Warm Greetings.

It is with a great pleasure to welcome you as a student of Institute of Distance Education, University of Madras. It is a proud moment for the Institute of Distance education as you are entering into a cafeteria system of learning process as envisaged by the University Grants Commission. Yes, we have framed and introduced Choice Based Credit System(CBCS) in Semester pattern from the academic year 2018-19. You are free to choose courses, as per the Regulations, to attain the target of total number of credits set for each course and also each degree programme. What is a credit? To earn one credit in a semester you have to spend 30 hours of learning process. Each course has a weightage in terms of credits. Credits are assigned by taking into account of its level of subject content. For instance, if one particular course or paper has 4 credits then you have to spend 120 hours of self-learning in a semester. You are advised to plan the strategy to devote hours of self-study in the learning process. You will be assessed periodically by means of tests, assignments and quizzes either in class room or laboratory or field work. In the case of PG (UG), Continuous Internal Assessment for 20(25) percentage and End Semester University Examination for 80 (75) percentage of the maximum score for a course / paper. The theory paper in the end semester examination will bring out your various skills: namely basic knowledge about subject, memory recall, application, analysis, comprehension and descriptive writing. We will always have in mind while training you in conducting experiments, analyzing the performance during laboratory work, and observing the outcomes to bring out the truth from the experiment, and we measure these skills in the end semester examination. You will be guided by well experienced faculty.

I invite you to join the CBCS in Semester System to gain rich knowledge leisurely at your will and wish. Choose the right courses at right times so as to erect your flag of success. We always encourage and enlighten to excel and empower. We are the cross bearers to make you a torch bearer to have a bright future.

With best wishes from mind and heart,

DIRECTOR

MCA/Msc., IT
FIRST YEAR - SECOND SEMESTER

CORE PAPER - VIII/ PAPER - X
PRACTICAL-III : JAVA PROGRAMMING LAB
PRACTICAL-IV : PROGRAMMING IN JAVA LAB

COURSE WRITERS

Dr. S. SASIKALA, M.C.A., M.Phil., Ph.D.,
Asst. Prof. in Computer Science
Institute of Distance Education
University of Madras
Chennai - 600 005.

COORDINATION AND EDITING

Dr. S. SASIKALA, M.C.A., M.Phil., Ph.D.,
Asst. Prof. in Computer Science
Institute of Distance Education
University of Madras
Chennai - 600 005.

MCA/MSc., IT., DEGREE COURSE

FIRST YEAR

SECOND SEMESTER

Core Paper - VIII / Core Paper - X

PRACTICAL- III : JAVA PROGRAMMING LAB

PRACTICAL - IV : PROGRAMMING IN JAVA LAB

SYLLABUS

APPLICATION

1. Generating random numbers using Random Class.
2. Implementation of Point Class for Image manipulation.
3. Usage of Calendar Class and manipulation.
4. String Manipulation using Char Array.
5. Database Creation for storing e-mail addresses and manipulation.
6. Usage of Vector Classes.
7. Implementing Thread based applications & Exception Handling (Synchronization & asynchronization).

APPLETS

8. Working with Frames and various controls.
9. Working with Dialogs and Menus.
10. Working with Panel and Layout.
11. Incorporating Graphics (Scaling Only).
12. Create a payroll application using Swings.

APPLICATION FOR EVENTS HANDLING

13. Application Using JDBC Connectivity

INSTITUTE OF DISTANCE EDUCATION
RECORD OF PRACTICALS

M.Sc., IT

(First Year)

2018-2019

Practical – III

PROGRAMMING IN JAVA Lab

Name :

Enrolment Number :

Group No :

UNIVERSITY OF MADRAS
CHENNAI - 600 005

INSTITUTE OF DISTANCE EDUCATION
UNIVERSITY OF MADRAS

CHENNAI - 600 005.

Certified that this is the Bonafide Record of work done by ____________________ with Enrolment Number _________________________ of First Year M.C.A. / M.Sc. (IT) Degree Course in the Institute of Distance Education, University of Madras during the year ____________ respect of Practical under Paper ____________________

Date: **Co-ordinator**

Submitted for First Year M.C.A. / M.Sc. (IT) Degree Course Practical Examination held on _____________________ at IDE, University of Madras.

Date: **Examiners**

1. **Name:**

Signature:

2. **Name:**

Signature:

MCA/MSc., IT., DEGREE COURSE

FIRST YEAR

SECOND SEMESTER

Core Paper - VIII / Core Paper - X

PRACTICAL- III : JAVA PROGRAMMING LAB

PRACTICAL - IV : PROGRAMMING IN JAVA LAB

SCHEME OF LESSONS

1. Random class:

```
import java.io.*;
import java.util.Random;
class rand
{
public static void main(String args[]) throws IOException
{
Random r=new Random();
BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
int temp,n,i,j;
int arr[]=new int[100];
System.out.println(“How many Random Numbers to Generate? “);
n=Integer.parseInt(br.readLine());
System.out.println(“The Generated Random Numbers Are:”);
//Code for Random Number Generation
for(i=0;i<=n;)
{
temp=r.nextInt();
if(temp>0 &&temp<=100)
{
arr[i]=temp;
i++;
System.out.print(temp+”   “);
}
}
//Code for Sorting Random Numbers
for(i=0;i<n;i++)
{
for(j=0; j<n-1;j++)
{
if(arr[j]>arr[j+1])
{
temp=arr[j];
arr[j]=arr[j+1];
arr[j+1]=temp;
}
}
}
System.out.println( );
System.out.println(“Numbers in Ascending Order: “);
for(i=0;i<n;i++)
{
System.out.print(arr[i]+”     “);
}
System.out.println( );
System.out.println(“Numbers in Reverse Order: “);
for(i=n-1;i>=0;i—)
{
System.out.print(arr[i]+”     “);
}
}
}
```

OUTPUT

****

```
How many Random Numbers to Generate?

10

The Generated Random Numbers Are:

75   5   58   54   70   15   41   5   66   69   31

Numbers in Ascending Order:

5    5    15    41    54    58    66    69    70    75

Numbers in ReverseOrder:

75          70    69    66    58    54    41    15    5    5
```

2. Point class:

****

```
import java.io.*;

import java.awt.*;

import java.lang.*;

class po

{

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

{

int x,y,x1,y1;

String s;

BufferedReader br=new BufferedReader

(new InputStreamReader(System.in));

System.out.println(“Implementation of

Point Class For Image Manipulation”);
```

```
System.out.println("********************
*********************************");
System.out.println(" ");
System.out.println(" ");
s=br.readLine();
x1=Integer.parseInt(s);
System.out.println("The x1 coordinate : "+x1);


System.out.println("");
s=br.readLine();
y1=Integer.parseInt(s);
System.out.println("The y1 coordinate : "+y1);
System.out.println(" ");
s=br.readLine();
x=Integer.parseInt(s);
System.out.println(" The column displacement : "+x);
s=br.readLine();
y=Integer.parseInt(s);
System.out.println("The row displacement:"+y);
System.out.println(" ");
Point p=new Point();
Point p1=new Point(x1,y1);
if(p.getX()==p1.getX() && p.getY()==p1.getY())
{
System.out.println(" ");
System.out.println(" Invalid Cordinate ");
System.out.println(“ “);
}
else if(p.getX()==p1.getX())
{
System.out.println(“ “);
System.out.println(“ Vertical Line “);
System.out.println(“ “);
}
else if(p.getY()==p1.getY())
{
System.out.println(“ “);
System.out.println(“ Horizontal Line “);
System.out.println(“ “);
}
else if(p.getX()==p1.getY())
{
System.out.println(“ “);
System.out.println(“ Square “);
System.out.println(“ “);
}
else
{
System.out.println(“ “);
System.out.println(“ Rectangle “);
System.out.println(“ “);
}
```

```
p.translate(x,y);
p1.translate(x,y);
System.out.println(“ “);

System.out.println(“ The Translated Coordinates:
 (“ +p.getX()+ “,” +p.getY()+”)”+”(“+p1.getX()+”,”+p1.getY()+”) “);
}
}
```

OUTPUT

******

```
Implementation of Point Class For Image Manipulation
****************************************************

The x1 coordinate : 0
The y1 coordinate : 0
The column displacement : 2
The row displacement:3
Invalid Cordinate
The Translated Coordinates: (2.0,3.0)(2.0,3.0)
The x1 coordinate : 0
The y1 coordinate : 2
The column displacement : 9
The row displacement:6
Vertical Line
```

```
The Translated Coordinates: (9.0,6.0)(9.0,8.0)

The x1 coordinate : 2

The y1 coordinate : 0

The column displacement : 6

The row displacement:5

Horizontal Line

The Translated Coordinates: (6.0,5.0)(8.0,5.0)

The x1 coordinate : 4

The y1 coordinate : 5

The column displacement : 8

The row displacement:3

Rectangle

The Translated Coordinates: (8.0,3.0)(12.0,8.0)
```

3. Calendar Class:

****

```
import java.io.*;

import java.util.*;

class employee

{

public int c;

public int b;

public int n,n1,n2,nn,nn1,n3,n4,age,exp,ret,

date,month,year,d1,d2,m1,m2,yy1,yy2;

int d[]={31,28,31,30,31,30,31,31,30,31,30,31};
```

```
String s,ss,ss1,s1,s2,s3,s4,s5,s6,a;
GregorianCalendar g=new GregorianCalendar();
int y=g.get(Calendar.YEAR);
int yy=g.get(Calendar.MONTH);
int y2=g.get(Calendar.DATE);
int y1=yy+1;
public void get() throws IOException
{
BufferedReader br=new BufferedReader
(new InputStreamReader(System.in));
System.out.println(“\t\t\tCalender class”);
System.out.println(“Enter the Empno.:”);
s=br.readLine();
n=Integer.parseInt(s);
System.out.println(“Enter the Name:”);
s1=br.readLine();
System.out.println(“Enter the Design:”);
s2=br.readLine();
System.out.println(“Enter the Year of Birth:”);
ss1=br.readLine();
nn1=Integer.parseInt(ss1);
do
{
System.out.println(“Enter the Month:”);
ss=br.readLine();
nn=Integer.parseInt(ss);
if(nn>12)

System.out.println(“Invalid Entry!!!Try again.”);
}
while(nn>12);
if(g.isLeapYear(nn1)&&(nn==2))
b=29;
else
b=d[nn-1];
do
{
System.out.println(“Enter the Date of birth:”);
s3=br.readLine();
n1=Integer.parseInt(s3);
if(n1>b)
System.out.println(“Invalid Entry!!!try Again”);
}
while(n1>b);
do
{
System.out.println(“Enter the Year of Joining”);
s6=br.readLine();
n4=Integer.parseInt(s6);
if(n4>y)
System.out.println(“Invalid Entry!!!Try Again”);
}
```

```
while(n4>y);
do
{
System.out.println("Enter the month:");
s5=br.readLine();
n3=Integer.parseInt(s5);
if(n3>12)
System.out.println("Invalid Entry!!!Try Again");
}
while(n3>12);
if(g.isLeapYear(n4)&&(n3==2))
c=29;
else

c=d[n3-1];
do
{
System.out.println("Enter the Date:");
s4=br.readLine();
n2=Integer.parseInt(s4);
if(n2>c)
System.out.println("Invalid Entry!!!try again");
}
while(n2>c);
}
public void cal()throws IOException
```

```
{
if(y2>=n1)
d1=y2-n1;
else
d1=n1-y2;
if(y1>=nn & y2>=n1)
m1=y1-nn;
else
{
m1=y1-nn;
m1—;
b=d[y1-2];
d1=(b-n1)+y2;
}
if(y>nn1)
yy1=y-nn1;
if(y2>=n2)
d2=y2-n2;
else
d2=n2-y2;
if(y1>=n3 && y2>=n2)
m2=y1-n3;
else
{
m2=y1-n3;
m2—;
c=d[y1-y2];
d2=(c-n2)+y2;
}
if(y>n4)
yy2=y-n4;
ret=nn1+58;
}

public void disp() throws IOException
{
System.out.println(“Details of   :”+s1);
System.out.println(“Employee no. :”+n);
System.out.println(“Name         :”+s1);
System.out.println(“Designation  :”+s2);
System.out.println(“Age          :”+yy1+”Yrs”
+ “ “ +m1+ “Months” +” “+ d1 + “Days”);
System.out.println(“Experience   :”+yy2+”Yrs”
 + “ “ +m2+ “Months” +” “+ d2 + “Days”);
System.out.println(“Ret Date     :”+n1+ “:” +nn+ “:” +ret);
System.out.println(“              “);
}
}
class emp
{
public static void main(String args[])throws
 IOException,NullPointerException
```

```
{
int x;
BufferedReader b=new BufferedReader(new
InputStreamReader(System.in));
System.out.println(“Enter no. of Employees: “);
String a=b.readLine();
x=Integer.parseInt(a);
employee e[]=new employee[x];
for(int i=0;i<x;i++)
{
e[i]=new employee();
e[i].get();
e[i].cal();
}

System.out.println(“Output”);
System.out.println(“******”);
System.out.println(“      “);
for(int i=0;i<x;i++)
{
e[i].disp();
}
}
}
```

14

```
INPUT
******
Enter no. of Employees:
2
                    Calender class
Enter the Empno.:
101
Enter the Name:
John
Enter the Design:
Manager
Enter the Year of Birth:
1989
Enter the Month:
5
Enter the Date of birth:
2
Enter the Year of Joining
2001
Enter the month:
6
Enter the Date:
1
                    Calender class
Enter the Empno.:
105
```

```
Enter the Name:
Tom
Enter the Design:
Doctor
Enter the Year of Birth:
1995
Enter the Month:
5
Enter the Date of birth:
2
Enter the Year of Joining
2016
Enter the month:
6
Enter the Date:
3
```

Output

```
******
Details of   :John
Employee no. :101
Name         :John
Designation  :Manager
Age          :28Yrs 3Months 10Days
Experience   :16Yrs 2Months 11Days
```

```
Ret Date     :2:5:2047


Details of   :Tom

Employee no. :105

Name         :Tom

Designation  :Doctor

Age          :22Yrs 3Months 10Days

Experience   :1Yrs 2Months 9Days

Ret Date     :2:5:2053
```

4. String manipulation:

****

```
import java.io.*;

import java.lang.String;

class array

{

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

{

String s=new String();

char chars[];

char vowels[]={‘a’,’e’,’i’,’o’,’u’,’A’,’E’,’I’,’O’,’U’};

int v=0,n=0,sp=0;

System.out.println(“String manipulation using char array”);

System.out.println(“*************************************”);

System.out.println(“Enter the string”);
BufferedReader b=new BufferedReader(new
InputStreamReader(System.in));
s=b.readLine();
chars=s.toCharArray();
for(int j=0;j<s.length();j++)
{
if(chars[j]>47 && chars[j]<58)
n++;
else
if((chars[j]>=65 && chars[j]<90 ||
 chars[j]>95 && chars[j]<122))
{
for(int k=0;k<10;k++)
{
if(chars[j]==vowels[k])
v++;
}
}
else
sp++;
}
System.out.println("vowels:"+v);
System.out.println("Digits:"+n);
System.out.println("Special Characters :"+sp);
}
}
```

```
OUTPUT

******

String manipulation using char array

*************************************

Enter the string

INDIA is my country

vowels:6

Digits:0

Special Characters :3
```

5. Database creation for storing e-mail addresses and manipulation

```
*****************

import java.io.*;

class file1

{

public static void main(String

 args[])throws IOException

{

System.out.println(“\n*************

**************************************************”);

System.out.println(“Database creation

 for storing E-mail addresses and manipulation”);

System.out.println(“\n****************

************************************************”);

BufferedReader b=new BufferedReader(new

 InputStreamReader(System.in));
```

```
String s,name,email,dummy,no;
int ch=0,found=0;
do
{
System.out.println("\n\t\tMENU");
System.out.println("\n\t\t****");
System.out.println("\n\t\t1. ADD");
System.out.println("\n\t\t2. MODIFY");
System.out.println("\n\t\t3. DELETE");
System.out.println("\n\t\t4. FIND");
System.out.println("\n\t\t5. VIEW");
System.out.println("\n\t\t6. EXIT");
s=b.readLine();
ch=Integer.parseInt(s);
switch(ch)
{
case 1:
FileWriter fw=new FileWriter("mail.dat",true);
do
{
do
{
System.out.println("\n\t\tName ('q' to stop): ");
name=b.readLine();
if(name.equals(""))
System.out.println("\n\t\tName cannot be empty");
}
while(name.equals(“”));
if(!name.equals(“q”))
{
do
{
System.out.println(“\t\tNo:”);
no=b.readLine();
if(no.equals(“”))
System.out.println(“\t\tNumber cannot be empty “);
}
while(no.equals(“”));
do
{
System.out.println(“\t\tEmail”);
email=b.readLine();
if(email.equals(“”))
System.out.println(“\t\tEmail id should be entered”);
}
while(email.equals(“”));
s=name+”\n”+no+”\n”+email+”\n”;
fw.write(s);
System.out.println(“\n\tRecord has been stored”);
}
}
while(!name.equals(“q”));
```

```
fw.close();
break;

case 2:
FileReader fr1=new FileReader(“mail.dat”);
FileWriter fw1=new FileWriter(“dummy.dat”);
BufferedReader br=new BufferedReader(fr1);
String name1;
do
{
System.out.println(“\n\t\tEnter the name :”);
name1=b.readLine();
if(name1.equals(“”))
System.out.println(“\t\tName Cannot be empty”);
}
while(name1.equals(“”));
while((s=br.readLine())!=null)
{
if(name1.equals(s))
{
found=1;
do
{
System.out.println(“\t\tEnter new no:”);
no=b.readLine();
if(no.equals(“”))
System.out.println(“\n\t\tNumber cannot be empty”);
}
while(no.equals(“”));

do
{
System.out.print(“\t\tEnter new id:”);
email=b.readLine();
if(email.equals(“”))
System.out.println(“\t\tEmail id Cannot be empty”);
}
while(email.equals(“”));
String s1=name1+”\n”+no+”\n”+email+”\n”;
fw1.write(s1);
System.out.println(“\n\t\tRecord had been modified”);
br.readLine();
br.readLine();
}
else
{
String ss=s+”\n”;
fw1.write(ss);
}
}
if(found!=1)
System.out.println(“\n\t\tSorry!Record not found”);
```

```
fw1.close();
fr1.close();
FileReader fr2=new FileReader(“dummy.dat”);
FileWriter fw2=new FileWriter(“mail.dat”);
BufferedReader br1=new BufferedReader(fr2);
while((s=br1.readLine())!=null)
{
String ss=s+”\n”;
fw2.write(ss);
}
fw2.close();
fr2.close();
break;

case 3:
FileReader fr4=new FileReader(“mail.dat”);
FileWriter fw4=new FileWriter(“dummy.dat”);
BufferedReader br2=new BufferedReader(fr4);
String name2;
found=0;
do
{
System.out.println(“\n\t\tEnter the name”);
name2=b.readLine();
if(name2.equals(“”))
System.out.println(“\t\tName cannot be empty”);
```

```
}
while(name2.equals(“,”));
while((s=br2.readLine())!=null)
{
if(name2.equals(s))
{
found=1;
br2.readLine();
br2.readLine();
System.out.println(“\n\t\tRecord has been deleted”);
}
else
{
String ss=s+”\n”;
fw4.write(ss);
}
}
if(found!=1)
System.out.println(“\n\t\tSorry!Record not found”);
fw4.close();
fr4.close();
FileReader fr3=new FileReader(“dummy.dat”);
FileWriter fw3=new FileWriter(“mail.dat”);
BufferedReader br3=new BufferedReader(fr3);
while((s=br3.readLine())!=null)
{
```

```
String ss=s+"\n";
fw3.write(ss);
}
fw3.close();
fr3.close();
break;

case 4:
FileReader fr5=new FileReader("mail.dat");
BufferedReader br4=new BufferedReader(fr5);
String name3;
found=0;
do
{
System.out.println("\n\t\tEnter the name");
name3=b.readLine();
if(name3.equals(""))
System.out.println("\t\tName cannot be empty");
}
while(name3.equals(","));
while((s=br4.readLine())!=null)
{
if(name3.equals(s))
{
found=1;
System.out.println("\t\tNo.:"+br4.readLine());
System.out.println(“\t\tEmail id:”+br4.readLine());
break;
}
}
if(found==0)
System.out.println(“\n\t\tSorry!Record not found”);
fr5.close();
break;

case 5:
FileReader fr=new FileReader(“mail.dat”);
BufferedReader br5=new BufferedReader(fr);
while(br5.readLine()!=null)
{
System.out.println(“\t\tNo.:”+br5.readLine());
System.out.println(“\t\tEmail id:”+br5.readLine());
break;
}

case 6:
break;
}
}
while(ch!=6);
}
}
```

OUTPUT

******

```
**************************************
Database creation for storing E-mail
addresses and manipulation
**************************************
        MENU
       ****
        1. ADD
        2. MODIFY
        3. DELETE
        4. FIND
        5. VIEW
        6. EXIT
1
        Name (‘q’ to stop):
Arasu
        No:
1
        Email
arasu@hotmail.com
   Record has been stored
        Name (‘q’ to stop):
Raja
        No:
```

```
3
        Email
raja@gmail.com
    Record has been stored
        Name (‘q’ to stop):
q
        MENU
        ****
        1. ADD
        2. MODIFY
        3. DELETE
        4. FIND
        5. VIEW
        6. EXIT
2
        Enter the name :
arasu
        Sorry!Record not found
        MENU
        ****
        1. ADD
        2. MODIFY
        3. DELETE
        4. FIND
        5. VIEW
        6. EXIT
```

```
2
          Enter the name :
Arasu
          Enter new no:
3
          Enter new id:ars@abc.com
          Record had been modified
          MENU
          ****
          1. ADD
          2. MODIFY
          3. DELETE
         4. FIND
          5. VIEW
          6. EXIT
3
          Enter the name
Arasu
          Record has been deleted
          MENU
          ****
          1. ADD
          2. MODIFY
          3. DELETE
          4. FIND
          5. VIEW
          6. EXIT
4
          Enter the name
Raja
          No.:3
          Email id:raja@gmail.com
          MENU
          ****
          1. ADD
          2. MODIFY
          3. DELETE
          4. FIND
          5. VIEW
          6. EXIT
5
          No.:1
          Email id:abc@abc.com
          MENU
          ****
          1. ADD
          2. MODIFY
          3. DELETE
          4. FIND
          5. VIEW
          6. EXIT
6
```

5. Usage of Vector class

```
***************

import java.io.*;

import java.util.*;

class vec

{

public static void main(String args[])

throws IOException

{

Vector v=new Vector();

int n=0,no;

String name,s;

System.out.println("***********************");

System.out.println("Usage of Vector Classes");

System.out.println("***********************");

BufferedReader br=new BufferedReader(new

 InputStreamReader(System.in));

do

{

System.out.println("MENU");

System.out.println("****");

System.out.println("1. ADD");

System.out.println("2. INSERT");

System.out.println("3. DELETE");

System.out.println("4. SEARCH");

System.out.println("5. MODIFY");
System.out.println(“6. DISPLAY”);

System.out.println(“7. EXIT”);

System.out.println(“Enter your choice”);

s=br.readLine();

n=Integer.parseInt(s);

switch(n)

{

case 1:

System.out.println(“Enter the Roll No:”);

s=br.readLine();

no=Integer.parseInt(s);

v.addElement(new Integer(no));

System.out.println(“Enter the Name:”);

name=br.readLine();

v.addElement(name);

break;


case 2:

System.out.println(“Enter the position”);

s=br.readLine();

int n1=Integer.parseInt(s);

if(n1%2!=0)

System.out.println(“Invalid Position”);

else

{

System.out.print(“Enter the Roll no.”);
String s1=br.readLine();
no=Integer.parseInt(s1);
v.insertElementAt(new Integer(no),n1);
System.out.println(“Enter the Name:”);
name=br.readLine();
v.insertElementAt(name,n1+1);
}
break;

case 3:
System.out.println(“Enter the Roll no:”);
s=br.readLine();
n1=Integer.parseInt(s);
int m=v.indexOf(new Integer(n1));
if(m==-1)
System.out.println(“Record not found”);
else
{
v.removeElementAt(m);
v.removeElementAt(m);
}
break;
case 4:
System.out.println(“Enter the Roll no.”);
s=br.readLine();
n1=Integer.parseInt(s);
```

```
m=v.indexOf(new Integer(n1));
if(m==-1)
System.out.println("Record not found");
else
{
s=(String)v.elementAt(m+1);
System.out.println("Roll no.:"+n1);
System.out.println("Name "+s);
}
break;

case 5:
System.out.println("Enter the Roll no.");
s=br.readLine();
n1=Integer.parseInt(s);
m=v.indexOf(new Integer(n1));
if(m==-1)
System.out.println("Record not found");
else
{
System.out.println("Enter the New name:");
name=br.readLine();
v.insertElementAt(name,m+1);
s=(String)v.elementAt(m+1);
v.removeElementAt(m+2);
}
```

```
break;

case 6:
Enumeration e=v.elements();
System.out.println(“Displaying the list
of Roll no.s and Names”);
while(e.hasMoreElements())
{
System.out.println(e.nextElement()+” “);
System.out.println(“  “);
}
break;

case 7:
System.out.println(“Bye”);
break;
}
}
while(n!=7);
}
}
```

OUTPUT

```
******
**********************
Usage of Vector Classes
```

```
***********************

MENU

****

1. ADD

2. INSERT

3. DELETE

4. SEARCH

5. MODIFY

6. DISPLAY

7. EXIT

Enter your choice

1

Enter the Roll No:

201

Enter the Name:

Sunil

MENU

****

1. ADD

2. INSERT

3. DELETE

4. SEARCH

5. MODIFY

6. DISPLAY

7. EXIT

Enter your choice
```

```
1
Enter the Roll No:
202
Enter the Name:
Tom
MENU
****
1. ADD
2. INSERT
3. DELETE
4. SEARCH
5. MODIFY
6. DISPLAY
7. EXIT
Enter your choice
2
Enter the position
2
Enter the Roll no.203
Enter the Name:
Jude
MENU
****
1. ADD
2. INSERT
3. DELETE
```

```
4. SEARCH
5. MODIFY
6. DISPLAY
7. EXIT
Enter your choice
4
Enter the Roll no.
202
Roll no.:202
Name Tom
MENU
****
1. ADD
2. INSERT
3. DELETE
4. SEARCH
5. MODIFY
6. DISPLAY
7. EXIT
Enter your choice
6
Displaying the list of Roll no.s and Names
201

Sunil
```

```
203

Jude

202

Tom

MENU
****
1. ADD
2. INSERT
3. DELETE
4. SEARCH
5. MODIFY
6. DISPLAY
7. EXIT
Enter your choice
7
Bye
```

7a. Implementing thread based applications using Exception handling

```
********************
import java.io.*;
import java.lang.*;
import java.util.*;
class thr
{
public static void main(String args[]) throws IOException
{
System.out.println("**********************
***************************************");
System.out.println("Implementing Thread Based
Applications and Exception Handling");
System.out.println("**********************
***************************************");
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
System.out.print("Enter the Number of Paris of values:");
int n=Integer.parseInt(br.readLine());
eval first[]=new eval[n];
eval second[]=new eval[n];
for(int i=0;i<n;++i)
{
first[i]=new eval();
second[i]=new eval();
}
for(int i=0;i<n;i++)
{
System.out.print("\n The "+(i+1)+"Pair is:");
first[i].start();
try
{
Thread.sleep(500);
}
catch(InterruptedException e)
{
}
second[i].start();
try
{
Thread.sleep(1000);
}
catch(InterruptedException e)
{
}
System.out.println("\n The Result is :
"+first[i].evaluate(first[i],second[i]));
}
}
}
class eval extends Thread
{
int r1;
public void run()
{
Random r=new Random();
```

```
r1=r.nextInt();
System.out.println(“ “+r1+” “);
}
static float evaluate(eval a,eval b)
{
float a1;
if(a.r1==b.r1)
{
a1=0;
throw new ArithmeticException(“Both
 the values are one and the same”);
}
else
a1=(a.r1+b.r1)/(a.r1-b.r1);
return a1;
}
}
```

OUTPUT

******

```
****************************************
Implementing Thread Based Applications
and Exception Handling
****************************************
Enter the Number of Paris of values:7
```

```
The 1Pair is: -1821927543
1919282707

The Result is :0.0

The 2Pair is: -1974138470
1053141873

The Result is :0.0

The 3Pair is: 17309012
1950693198

The Result is :-1.0

The 4Pair is: 1724039874
-366351699

The Result is :0.0

The 5Pair is: 1667306329
-1401542247

The Result is :0.0
The 6Pair is: -1119262596

-1401286048


The Result is :6.0


The 7Pair is: -559346896

524090187


The Result is :0.0
```

7b. Implementing thread based application using thread synchronization

****

```
import java.io.*;

import java.lang.*;

class customer extends Thread

{

int val;

float amt;

customer(int a,float b)

{

val=a;

amt=b;

}

public void run()

{
```

```
if(val==1)
{
calc.deposit_amt(amt);
calc.disp();
}
else
{
calc.withdraw_amt(amt);
calc.disp();
}
}
}
class calc
{
static float amount=0;
static synchronized void deposit_amt(float amt)
{
amount=amount+amt;
System.out.println("\nAmount deposited");
}
static synchronized void withdraw_amt(float amt)
{
if(amt>=amount||(amount-amt)<=250)
System.out.println("\nWITHDRAW IS PROHIBITED ");
else
{
```

```
amount=amount-amt;
System.out.println(“\nAmount withdrawn “);
}
}
static void disp()
{
System.out.println(“Balance Rs”+amount+”\nThank You”);
}
}
class abc
{
customer first[]=new customer[2];
void func()throws IOException
{
BufferedReader b=new BufferedReader(new
InputStreamReader(System.in));
for(int i=0;i<2;i++)
{
System.out.println(“\nCustomer”+
(i+1)+”\n1.Deposit\n2.Withdraw”);
System.out.println(“Enter your choice : “);
int choice=Integer.parseInt(b.readLine());
System.out.println(“Enter the amount : “);
first[i]=new customer(choice,Float.parseFloat(b.readLine()));
}
first[0].start();
```

```
try
{
Thread.sleep(1000);
}
catch(InterruptedException e)
{

}
first[1].start();
}
}
class atm
{
public static void main(String a[])
throws IOException
{
System.out.println(“\n******Banking
Transaction*******”);
System.out.println(“Managing a combined
Account Transaction”);
BufferedReader b=new BufferedReader
(new InputStreamReader(System.in));
abc thr1=new abc();
int choice;
do
{
```

```
thr1.func();
try
{
Thread.sleep(1000);
}
catch(InterruptedException e)
{
}
System.out.println(“\n\nEnter 1 to continue :”);
choice=Integer.parseInt(b.readLine());
}
while(choice==1);
}
}
```

OUTPUT

```
******
******Banking Transaction*******
Managing a combined Account Transaction

Customer1
1.Deposit
2.Withdraw
Enter your choice :
1
Enter the amount :
```

```
2000

Customer2
1.Deposit
2.Withdraw
Enter your choice :
1
Enter the amount :
3000

Amount deposited
Balance Rs2000.0
Thank You

Amount deposited
Balance Rs5000.0
Thank You


Enter 1 to continue :
1

Customer1
1.Deposit
2.Withdraw
Enter your choice :
```

```
2
Enter the amount :
1000

Customer2
1.Deposit
2.Withdraw
Enter your choice :
2
Enter the amount :
1000

Amount withdrawn
Balance Rs4000.0
Thank You

Amount withdrawn
Balance Rs3000.0
Thank You


Enter 1 to continue :
1

Customer1
1.Deposit
```

```
2.Withdraw
Enter your choice :
2
Enter the amount :
2000

Customer2
1.Deposit
2.Withdraw
Enter your choice :
2
Enter the amount :
3000

Amount withdrawn
Balance Rs1000.0
Thank You

WITHDRAW IS PROHIBITED
Balance Rs1000.0
Thank You



Enter 1 to continue :
2
```

8. Frames and controls

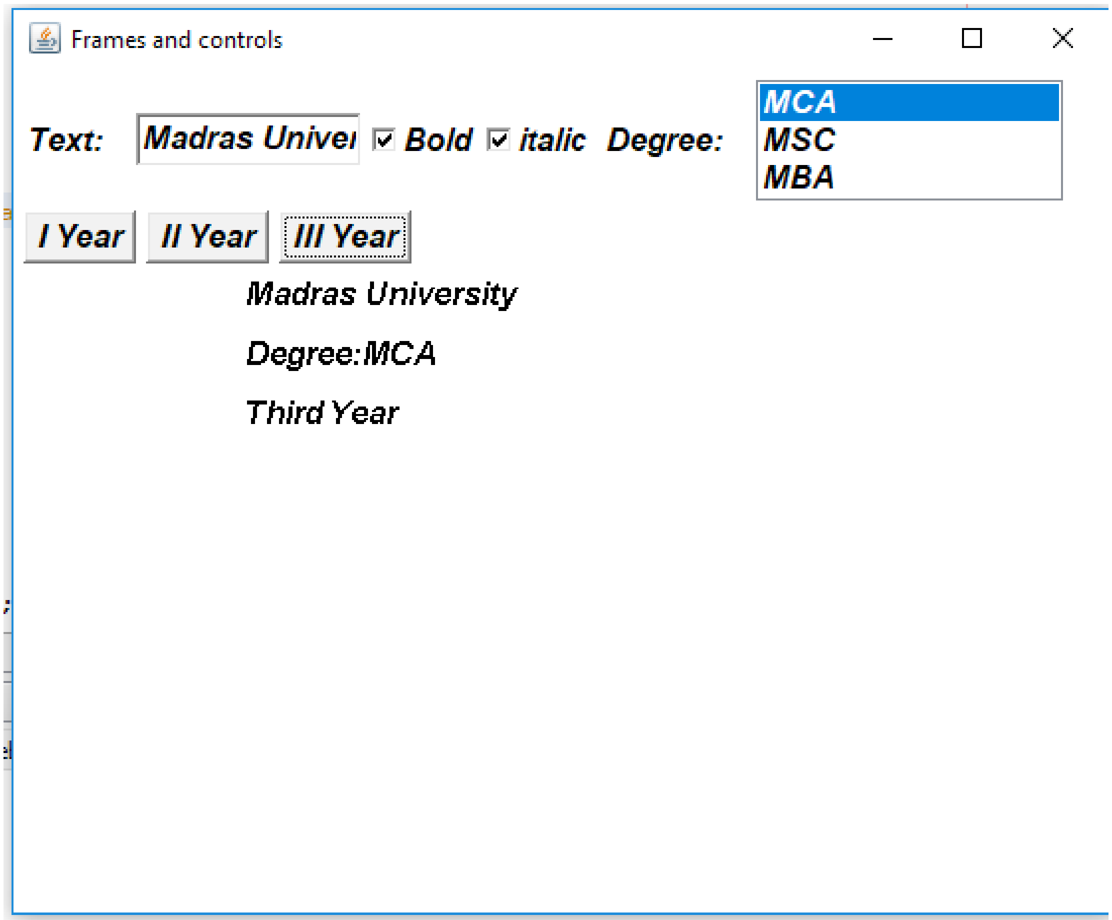
```
********************

import java.awt.*;

import java.awt.event.*;

import java.applet.*;

/*<applet code=”framx” width=250 height=200>

</applet>

*/

class fram extends Frame

 implements ActionListener,ItemListener

{

Button b1,b2,b3;

List os;

Checkbox bold,italic;

TextField name;

Font fo;

String msg=””;

String msg1=””;

fram(String s)

{

super(s);

setLayout(new FlowLayout(FlowLayout.LEFT));

os=new List(3,false);

os.add(“MCA”);

os.add(“MSC”);

os.add(“MBA”);
```

```
Label text=new Label(“Text:”,Label.LEFT);
Label text1=new Label(“Degree:”,Label.LEFT);
name=new TextField(10);
bold=new Checkbox(“Bold”,null,false);
italic=new Checkbox(“italic”);
b1=new Button(“I Year”);
b2=new Button(“II Year”);
b3=new Button(“III Year”);
add(text);
add(name);
add(bold);
add(italic);
add(text1);
add(os);
add(b1);
add(b2);
add(b3);
os.addActionListener(this);
b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
name.addActionListener(this);
bold.addItemListener(this);
italic.addItemListener(this);
mywindowadapter adap=new mywindowadapter(this);
addWindowListener(adap);
```

```
}
public void paint(Graphics g)
{
g.drawString(name.getText(),125,150);
int idx[];
msg=”Degree:”;
idx=os.getSelectedIndexes();
for(int i=0;i<idx.length;i++)
msg+=os.getItem(idx[i])+””;
g.drawString(msg,125,180);
g.drawString(msg1,125,210);
}
public void actionPerformed(ActionEvent ae)
{
String str=ae.getActionCommand();
if(str.equals(“I Year”))
{
msg1=”First Year”;
}
if(str.equals(“II Year”))
{
msg1=”Second Year”;
}
if(str.equals(“III Year”))
{
msg1=”Third Year”;
}
repaint();
}
public void itemStateChanged(ItemEvent ie)
{
if(bold.getState() & italic.getState())
fo=new Font(“Dialog”,Font.BOLD|Font.ITALIC,16);
else if(bold.getState())
fo=new Font(“Dialog”,Font.BOLD,16);
else if(italic.getState())
fo=new Font(“Dialog”,Font.ITALIC,16);
setFont(fo);
repaint();
}
}
class mywindowadapter extends WindowAdapter
{
fram fr;
public mywindowadapter(fram fr)
{
this.fr=fr;
}
public void windowClosing(WindowEvent we)
{
fr.dispose();
}
```

```
}
public class framz extends Applet
{
Frame F;
public void init()
{
F=new fram(“Frames and controls”);
setSize(300,250);
F.setSize(300,250);
F.setVisible(true);
}
public void start()
{
F.setVisible(true);
}
public void stop()
{
F.setVisible(false);
}
}
```

OUTPUT:



Text: Madras Univer ☑ Bold ☑ italic Degree: MCA

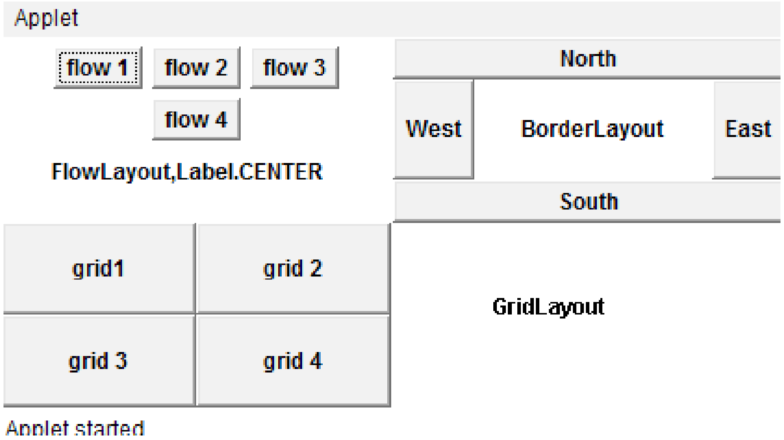
```
Madras University
Degree:MCA
Third Year
```

9. Panels and Layouts

```
***********************

import java.awt.*;

import java.awt.event.*;

import java.applet.*;

/*

<applet code=”pan” width=500 height=200>

</applet>

*/

public class pan extends Applet

{

public void init()

{

setLayout(new GridLayout(2,2));

setFont(new Font(“comic sans”,Font.BOLD,12));

Panel flow=new Panel();

flow.setLayout(new FlowLayout());

flow.add(new Button(“flow 1”));

flow.add(new Button(“flow 2”));

flow.add(new Button(“flow 3”));

flow.add(new Button(“flow 4”));

flow.add(new Label(“FlowLayout,Label.CENTER”));

add(flow);

Panel border=new Panel();

border.setLayout(new BorderLayout());

border.add(new Button(“North”),BorderLayout.NORTH);

border.add(new Button(“South”),BorderLayout.SOUTH);
border.add(new Button(“East”),BorderLayout.EAST);
border.add(new Button(“West”),BorderLayout.WEST);
border.add(new Label(“BorderLayout”,Label.CENTER));
add(border);
Panel grid=new Panel();
grid.setLayout(new GridLayout(2,2));
grid.add(new Button(“grid1”));
grid.add(new Button(“grid 2”));
grid.add(new Button(“grid 3”));
grid.add(new Button(“grid 4”));
add(grid);
}
public void paint(Graphics g)
{
g.drawString(“GridLayout”,265,150);
}
}
```

OUT PUT:

PANELS AND LAYOUTS:



Applet started

10. Dialogs and Menus

```
********************

import java.awt.*;

import java.applet.*;

import java.awt.event.*;

/*<applet code=”dmenu” width=300 height=200>

</applet>

*/

class samp extends Dialog implements ActionListener

{

samp(Frame par,String title)

{

super(par,title,false);

setLayout(new FlowLayout());

setSize(200,200);

add(new Label(“modeless Dialog Box”));

Button b1;

add(b1=new Button(“ok”));

b1.addActionListener(this);

}

public void actionPerformed(ActionEvent ae)

{

dispose();

}


}
```

```
class samp1 extends Dialog implements ActionListener
{
samp1(Frame par1,String title1)
{
super(par1,title1,true);
setLayout(new FlowLayout());
setSize(200,200);
add(new Label("modal Dialog Box"));
Button b;
add(b=new Button("ok"));
b.addActionListener(this);
}
public void actionPerformed(ActionEvent ae)
{
dispose();
}
}
class men extends Frame
{
String msg="";
men(String title)
{
super(title);
MenuBar mbar=new MenuBar();
setMenuBar(mbar);
Menu file=new Menu("File");
```

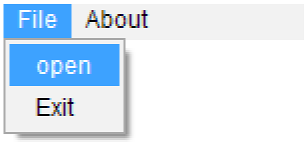
```
MenuItem it1,it2;
file.add(it1=new MenuItem(“open”));
file.add(it2=new MenuItem(“Exit”));
mbar.add(file);
Menu about=new Menu(“About”);
MenuItem it3;
about.add(it3=new MenuItem(“Help”));
mbar.add(about);
menhand hand=new menhand(this);
it1.addActionListener(hand);
it2.addActionListener(hand);
it3.addActionListener(hand);
mywind adap=new mywind(this);
addWindowListener(adap);
}
}
class mywind extends WindowAdapter
{
men m;
public mywind(men m)
{
this.m=m;
}
public void windowClosing(WindowEvent we)
{
m.dispose();
```

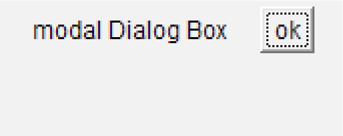
```
}
}
class menhand implements ActionListener
{
men m;
public menhand(men m)
{
this.m=m;
}
public void actionPerformed(ActionEvent ae)
{
String str=(String)ae.getActionCommand();
if(str.equals(“open”))
{
samp1 d1=new samp1(m,”Open Dialog box”);
d1.setVisible(true);
}
else if(str.equals(“Exit”))
{
m.dispose();
}
else if(str.equals(“Help”))
{
samp d2=new samp(m,”Help Dialog Box”);
d2.setVisible(true);
}
```

```
}
}
public class dmenu extends Applet
{
Frame f;
public void init()
{
f=new men(“Menu Demo”);
f.setSize(200,200);
f.setVisible(true);
}
public void start()
{
f.setVisible(true);
}
public void stop()
{
f.setVisible(false);
}
}
```

OUTPUT:

DIALOGUES AND MENUS:





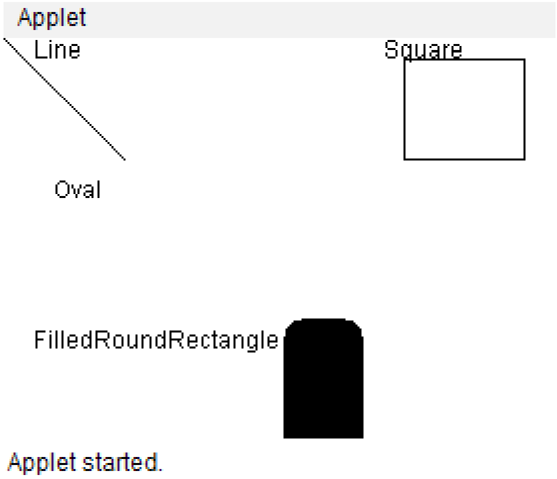
modal Dialog Box

11a. Working with Graphics

```
************************

import java.awt.*;

import java.applet.*;

/*<applet code=”graph” width=300 height=200>

</applet>

*/

public class graph extends Applet

{

public void init()

{

setBackground(Color.white);

}

public void paint(Graphics g)

{

g.drawString(“Line”,15,10);

g.drawLine(0,0,60,60);

g.drawString(“Oval”,25,80);

g.drawString(“FilledRoundRectangle”,15,155);

g.fillRoundRect(140,140,40,70,20,20);

g.drawString(“Circle”,360,50);

g.fillOval(300,20,50,50);
g.drawString(“FilledArc”,130,240);

g.fillArc(180,220,60,80,0,175);

g.drawString(“Square”,190,10);

g.drawRect(200,10,60,50);

int x[]={250,300,250,300,250};

int y[]={250,250,300,300,250,};

int num=5;

g.drawString(“polygon”,260,240);

g.drawPolygon(x,y,num);

}

}
```

OUTPUT:

WORKING WITH GRAPHICS



11b. Point class using Applet

```
import java.awt.*;  // for Graphics
/*<applet code=”Point” width=250 height=200>
</applet>
*/

public class Point {
 // encapsulation - only Point objects can access their own fields directly
 private int x;    // EACH Point object should have a variable
 private int y;    // inside it named x, and a variable named y

  public Point(int initialX, int initialY) {
  x = initialX;
  y = initialY;
 }

 // Initializes the state of a new Point object at the origin (0, 0).
 public Point() {
  x = 0;
  y = 0;
 }

 // Returns the point’s x-coordinate.
 // Provides a read-only access to the point’s state.
 public int getX() {
  return x;
```

```
}
// Returns the point’s y-coordinate.
public int getY() {
 return y;
}

// Draws a point on a DrawingPanel.
public void draw(Graphics g) {
 g.fillOval(x, y, 3, 3);
 g.drawString(toString(), x, y);
}

// Shifts the point’s x/y coordinates by the given amounts.
public void translate(int dx, int dy) {
 x += dx;
 y += dy;    // or, setLocation(x + dx, y + dy);
}

// Sets the point’s x/y coordinates to be the given values.
public void setLocation(int newX, int newY) {
 x = newX;
 y = newY;
}

// Computes the distance between this point and the given other point p2.
public double distance(Point p2) {
  int dx = x - p2.x;
  int dy = y - p2.y;
  double distance = Math.sqrt(dx*dx + dy*dy);
  return distance;
 }

 // Computes the distance between this point and the origin (0, 0).
 public double distanceFromOrigin() {
  Point origin = new Point();  // 0, 0
  return distance(origin);
 }

 // Returns a string representation of the point, such as “(5, -2)”.
 public String toString() {
  return “(“ + x + “, “ + y + “)”;
 }
}
```

12. Communication between HTML and Servlet

****

Html:

****

```
<html>
<head>
<title>Payroll Information</title>
<script language=”javascript”>
```

```
function calc()
{
document.frm.da.value=(parseInt(document.frm.bpay.value)*40)/100;
document.frm.hra.value=(parseInt(document.frm.bpay.value)*5)/100;
document.frm.pf.value=(parseInt(document.frm.bpay.value)*10)/100;
document.frm.npay.value=parseInt(document.frm.bpay.value)+
parseInt(document.frm.da.value)+parseInt(document.frm.hra.value);
document.frm.gpay.value=parseInt(document.frm.pf.value);
}
</script>
</head>
<body bgcolor=”white”>
<form name=”frm” action=”http://localhost:8080/WebApplication1/NewServlet”>
<table border=”1" align=”center”>
<tr><th>Payroll Information</th></tr>
<tr><td>Employee No<input type=”text” name=”eno”></td></tr>
<tr><td>Employee Name<input type=”text” name=”ename”></td></tr>
<tr><td>Sex<input type=”text” name=”sex”></td></tr>
<tr><td>Address<input type=”text” name=”address”></td></tr>
<tr><td>Basic Pay<input type=”text” name=”bpay”></td></tr>
<tr><td>Dearness Allowances<input type=”text” name=”da”></td></tr>
<tr><td>House Rent Allowances<input type=”text” name=”hra”></td></tr>
<tr><td>Provident Fund<input type=”text” name=”pf”></td></tr>
<tr><td>Net Pay<input type=”text” name=”npay”></td></tr>
<tr><td>Gross Pay<input type=”text” name=”gpay”></td></tr>
```

```
<tr><td><input type=”button” value=”Calculate”
name=”calculate” onclick=”calc()”>
<input type=”submit” value=”submit”>
</td></tr>
</table>
</form>

</body>
</html>
```

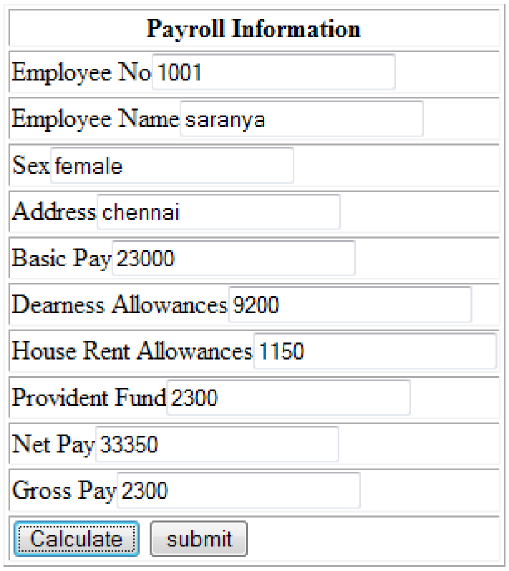
servlet:

****

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet(urlPatterns = {“/NewServlet”})
public class NewServlet extends HttpServlet {
protected void processRequest(HttpServletRequest req,
 HttpServletResponse res)
throws ServletException, IOException {
res.setContentType(“text/html”);
try (PrintWriter out = res.getWriter()) {
int eno=Integer.parseInt(req.getParameter(“eno”));
```

```
String ename=req.getParameter(“ename”);
String sex=req.getParameter(“sex”);
String address=req.getParameter(“address”);
int bpay=Integer.parseInt(req.getParameter(“bpay”));
int da=Integer.parseInt(req.getParameter(“da”));
int hra=Integer.parseInt(req.getParameter(“hra”));
int pf=Integer.parseInt(req.getParameter(“pf”));
int npay=Integer.parseInt(req.getParameter(“npay”));
int gpay=Integer.parseInt(req.getParameter(“gpay”));
out.println(“<!DOCTYPE html>”);
out.println(“<html>”);
out.println(“<head>”);
out.println(“<title>Payroll Information</title>”);
out.println(“</head>”);
out.println(“<body bgcolor=\”white\”>”);
out.println(“<p align=\”center\”><b>Pay Slip</b></p>”);
out.println(“<table border=\”1\” align=\”center\”bgcolor=\”white\”>”);
out.println(“<tr><td>Employee No</td><td>”);
out.println(eno+”</td></tr>”);
out.println(“<tr><td>Employee Name</td><td>”);
out.println(ename+”</td></tr>”);
out.println(“<tr><td>Sex</td><td>”);
out.println(sex+”</td></tr>”);
out.println(“<tr><td>Address</td><td>”);
out.println(address+”</td></tr>”);
out.println(“<tr><td>Basic Pay</td><td>”);
out.println(bpay+”</td></tr>”);
out.println(“<tr><td>Dearness Allowances</td><td>”);
out.println(da+”</td></tr>”);
out.println(“<tr><td>House Rent Allowances</td><td>”);
out.println(hra+”</td></tr>”);
out.println(“<tr><td>Providet Fund</td><td>”);
out.println(pf+”</td></tr>”);
out.println(“<tr><td>Net Pay</td><td>”);
out.println(npay+”</td></tr>”);
out.println(“<tr><td>Gross Pay</td><td>”);
out.println(gpay+”</td></tr>”);
out.println(“</body>”);
out.println(“</html>”);
}
}
```

Html to servlet:



Payroll Information

Employee No 1001

Employee Name saranya

Sex female

Address chennai

Basic Pay 23000

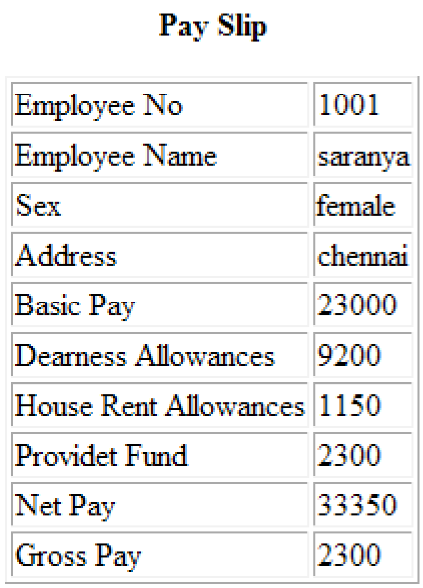
Dearness Allowances 9200

House Rent Allowances 1150

Provident Fund 2300

Net Pay 33350

Gross Pay 2300



Pay Slip

Employee No	1001
Employee Name	saranya
Sex	female
Address	chennai
Basic Pay	23000
Dearness Allowances	9200
House Rent Allowances	1150
Providet Fund	2300
Net Pay	33350
Gross Pay	2300

13. Communication between Applet and Servelet

******

```
import java.applet.Applet;

import java.awt.event.*;

import java.io.*;

import java.io.DataInputStream;

import java.awt.*;

import java.net.*;

import java.awt.event.ActionListener;

import java.net.URL;

import java.net.URLConnection;

import java.awt.TextArea;

import java.awt.Button;
```

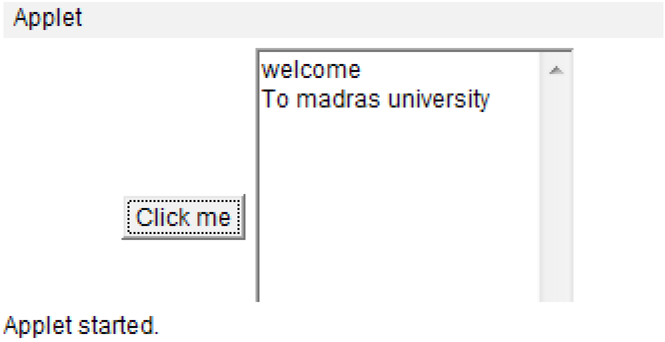
```
public class applet1 extends Applet implements
ActionListener {
TextArea result=new TextArea(10,20);
Button b1=new Button(“Click me”);

public void init()
{
add(b1);
add(result);
b1.addActionListener(this);
setVisible(true);
}
public void actionPerformed(ActionEvent e)
{
try
{
URL url=new URL(getCodeBase(),”http://localhost:8080/
WebApplication1/NewServlet1");
URLConnection con=url.openConnection();
con.setUseCaches(false);
DataInputStream in=new DataInputStream(con.getInputStream());
String line;
while((line=in.readLine())!=null)
result.append(line+”\n”);
System.out.println(“action performed”);
}
catch(Exception e1)
{
}
}
}
```

Servlet:

****

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet(urlPatterns = {“/NewServlet1”})
public class NewServlet1 extends HttpServlet {
protected void processRequest(HttpServletRequest request,
HttpServletResponse response)
throws ServletException, IOException {
response.setContentType(“text/html;charset=UTF-8”);
try (PrintWriter out = response.getWriter()) {
/* TODO output your page here. You may use following sample code. */
out.println(“welcome”);
out.println(“To madras university”);
out.close();
}

}
```

Applet to Servlet:



```
welcome
To madras university
```

Applet started.

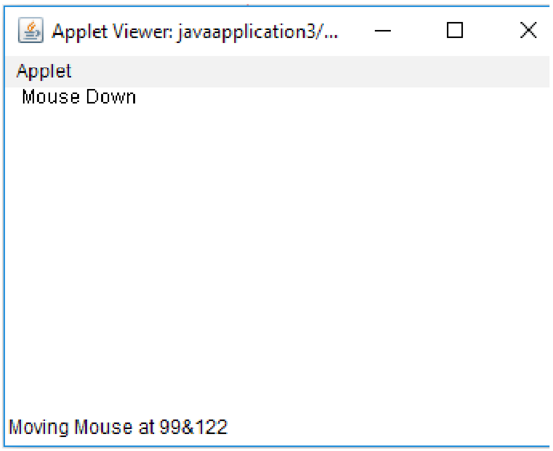
14. Write a java program to demonstrate Mouse Events

```
package javaapplication3;


importjava.applet.*;

importjava.awt.event.*;

importjava.awt.*;

/*<applet code=”NewApplet” width=300 height=200>

</applet>*/

public class NewApplet extends Applet implements MouseListener, MouseMotionListener

{

String msg=””;

int x=0,y=0;
```

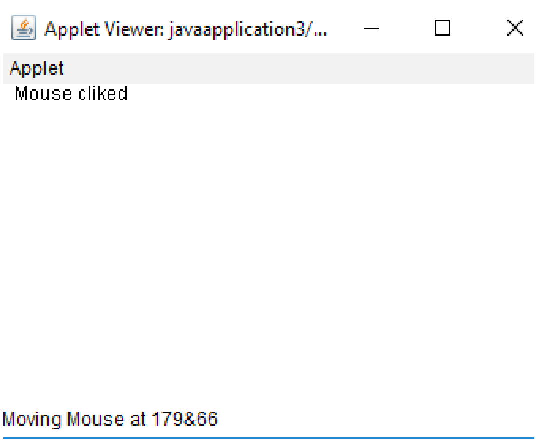
```
public void init()
{
addMouseListener(this);
addMouseMotionListener(this);
}
public void mouseClicked(MouseEvent m)
{
x=10;
y=10;
msg=”Mouse cliked”;
repaint();
}
public void mouseEntered(MouseEvent m)
{
x=10;
y=10;
msg=”Mouse Entered”;
repaint();
}
public void mouseExited(MouseEvent m)
{
x=10;
y=10;
msg=”Mouse Exit”;
repaint();
}
```

```
public void mousePressed(MouseEvent m)
{
x=10;
y=10;
msg=”Mouse Down”;
repaint();
}
public void mouseReleased(MouseEvent m)
{
x=10;
y=10;
msg=”Mouse Up”;
repaint();
}
public void mouseDragged(MouseEvent m)
{
x=10;
y=10;
msg=”*”;
showStatus(“Dragging mouse at”+x+”&”+y);
repaint();
}

public void mouseMoved(MouseEvent m)
{
```

```
x=10;
y=10;
showStatus(“Moving Mouse at “+m.getX()+”&”+m.getY());
repaint();
}
public void paint(Graphics g)
{
g.drawString(msg,x,y);
}
```



```
Mouse Down
Moving Mouse at 99&122
```



```
Mouse cliked
Moving Mouse at 179&66
```

```
}
```

Output

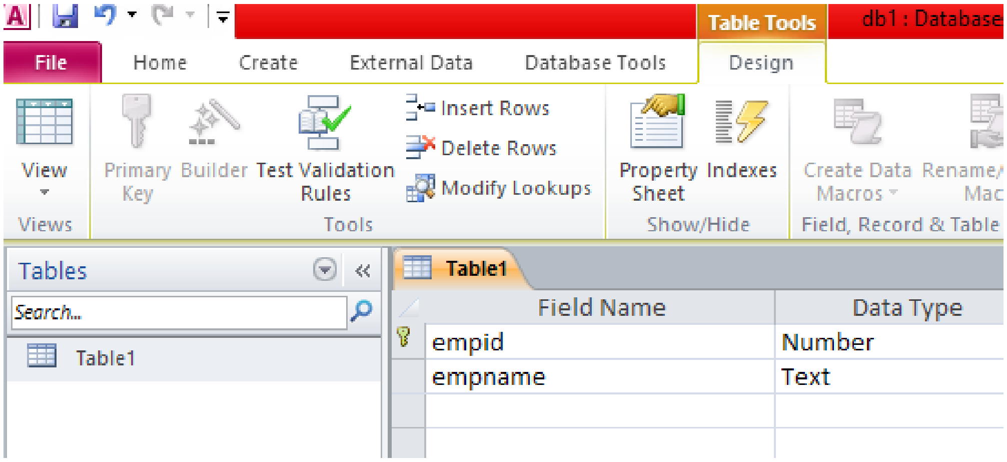
15. Application using JDBC Connectivity to develop employee System.

```
importjava.sql.*;
public class JDBCemp1
```

```
{
public static void main(String args[])
{
try
{
Connection con;
Class.forName(“sun.jdbc.odbc.JdbcOdbcDriver”);
con=DriverManager.getConnection(“jdbc:odbc:data1”);
try{
System.out.println(“getting all Rows from a table:”);
Statement st=con.createStatement();
ResultSet res=st.executeQuery(“SELECT * FROM college”);
System.out.println(“EmpID:”+”\t+EMPName:”);
while(res.next())
{
int i=res.getInt(“empid”);
String s=res.getString(“empname”);
System.out.println(i+”\t\t”+s);
}
con.close();
}
catch(SQLException s){
System.out.println(“sql code does not execute”);
}
}
catch(Exception e){
```

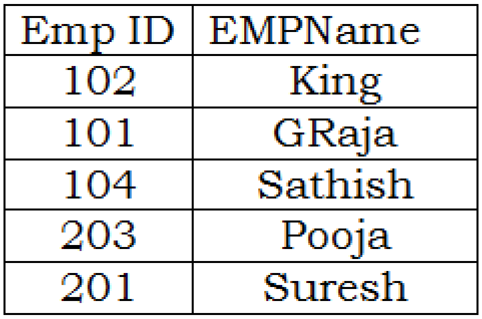
```
System.out.println(“Error:connection not created”);
}


}
}
```

Create a database using MS-Access with the following Fields.



Field Name	Data Type
empid	Number
empname	Text

Output



Emp ID	EMPName
102	King
101	GRaja
104	Sathish
203	Pooja
201	Suresh