



**ST. ANNE'S  
COLLEGE OF ENGINEERING AND  
TECHNOLOGY**

**CS8383  
OBJECT ORIENTED PROGRAMMING  
LABORATORY**

**NAME** : \_\_\_\_\_

**DEPT** : \_\_\_\_\_

**YEAR** : \_\_\_\_\_

**SEM** : \_\_\_\_\_

**REG.NO** : \_\_\_\_\_

**ANNA UNIVERSITY, CHENNAI**  
**B.E. COMPUTER SCIENCE AND ENGINEERING**  
**REGULATIONS – 2017**

**CS8383 OBJECT ORIENTED PROGRAMMING LABORATORY**  
**LIST OF EXPERIMENTS**

1. Develop a Java application to generate Electricity bill. Create a class with the following members: Consumer no., consumer name, previous month reading, current month reading, type of EB connection (i.e domestic or commercial). Compute the bill amount using the following tariff.

If the type of the EB connection is domestic, calculate the amount to be paid as follows:

- First 100 units - Rs. 1 per unit
- 101-200 units - Rs. 2.50 per unit
- 201 -500 units - Rs. 4 per unit
- > 501 units - Rs. 6 per unit

If the type of the EB connection is commercial, calculate the amount to be paid as follows:

- First 100 units - Rs. 2 per unit
- 101-200 units - Rs. 4.50 per unit
- 201 -500 units - Rs. 6 per unit
- > 501 units - Rs. 7 per unit

2. Develop a java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa) , time converter (hours to minutes, seconds and vice versa) using packages.
3. Develop a java application with Employee class with Emp\_name, Emp\_id, Address, Mail\_id, Mobile\_no as members. Inherit the classes, Programmer, Assistant Professor, Associate Professor and Professor from employee class. Add Basic Pay (BP) as the member of all the inherited classes with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF, 0.1% of BP for staff club fund. Generate pay slips for the employees with their gross and net salary.
4. Design a Java interface for ADT Stack. Implement this interface using array. Provide necessary exception handling in both the implementations.
5. Write a program to perform string operations using ArrayList. Write functions for the following
  - a. Append - add at end
  - b. Insert – add at particular index
  - c. Search
  - d. List all string starts with given letter

6. Write a Java Program to create an abstract class named Shape that contains two integers and an empty method named print Area(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.
7. Write a Java program to implement user defined exception handling.
8. Write a Java program that reads a file name from the user, displays information about whether the file exists, whether the file is readable, or writable, the type of file and the length of the file in bytes.
9. Write a java program that implements a multi-threaded application that has three threads. First thread generates a random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
10. Write a java program to find the maximum value from the given type of elements using a generic function.
11. Design a calculator using event-driven programming paradigm of Java with the following options.
  - a) Decimal manipulations
  - b) Scientific manipulations
12. Develop a mini project for any application using Java concepts.



**ST. ANNE'S**  
**COLLEGE OF ENGINEERING AND TECHNOLOGY**  
ANGUCHETTYPALAYAM, PANRUTI - 607 106

**LIST OF EXPERIMENTS**

1. Develop a Java application to generate Electricity bill
2. Develop a java application to implement currency converter, distance converter, time converter using packages.
3. Develop a java application of Employee payroll system
4. Design a Java interface using array for ADT Stack
5. Develop a Java Program to perform string operations using ArrayList
6. Develop a Java Program to create an abstract class
7. Develop a Java program to implement user defined exception handling.
8. Develop a Java program that reads a file name from the user, displays information about whether the file exists
9. Develop a Java program that implements a multi-threaded application that has three threads.
10. Develop a Java program to find the maximum value from the given type of elements using a generic function.
11. Design a calculator using event-driven programming paradigm of Java
12. Develop a mini project for any application using Java concepts.

<b>EXP NO: 1</b>	<b>DEVELOP A JAVA APPLICATION TO GENERATE ELECTRICITY BILL</b>
<b>DATE:</b>	

**AIM:**

To develop a java application for generate electricity bill

**ALGORITHM:**

**Step 1:** Create a class with members consumer no, consumer name, previous month reading, current month reading, type of EB connection.

**Step 2:** Input data to the class members

**Step 3:** Choose the type of EB connection.

**Step 4:** Compute the bill amount as per the type of connection.

**Step 5:** Calculate the total amount.

**PROGRAM CODE:**

```
import java.util.Scanner;
class ElectBill
{
int ConsumerNo;
String ConsumerName;
int PrevReading;
int CurrReading;
String EBConn;
double Bill;
void input_data()
{
Scanner sc = new Scanner(System.in);
System.out.println("\n Enter Consumer Number: ");
ConsumerNo = sc.nextInt();
System.out.println("\n Enter Consumer Name: ");
ConsumerName = sc.next();
System.out.println("\n Enter Previous Units: ");
PrevReading = sc.nextInt();
System.out.println("Enter Current Units consumed:");
CurrReading = sc.nextInt();
System.out.println("Enter the types of EB Connection(domestic or commercial)");
EBConn=sc.next();
}
double calculate_bill()
{
int choice;
if(EBConn == "domenstic")
choice = 1;
else
choice = 2;
switch(choice)
```

```
{
case 1:
    if(CurrReading>= 0 && CurrReading<= 100)
    Bill=CurrReading*1;
    else if(CurrReading> 100 && CurrReading <= 200)
    Bill=(100*1)+((CurrReading-100)*2.50);
    else if(CurrReading>200 && CurrReading <= 500)
    Bill=(100*1)+(100*2.50)+((CurrReading-200)*4);
    else
    Bill=(100*1)+(100*2.50)+(300*4)+((CurrReading-500)*6);
    break;
case 2:
    if(CurrReading>=0 && CurrReading<= 100)
    Bill=CurrReading*2;
    else if(CurrReading>100 && CurrReading <=200)
    Bill=(100*1)+((CurrReading-100)*4.50);
    else if(CurrReading>200 && CurrReading <= 500)
    Bill=(100*1)+(100*2.50)+((CurrReading-200)*6);
    else
    Bill=(100*1)+(100*2.50)+(300*4)+((CurrReading-500)*7);
    break;
}
return Bill;
}
void display()
{
System.out.println("-----");
System.out.println("ELCTRICITY BILL");
System.out.println("-----");
System.out.println("Consumer Number: "+ConsumerNo);
System.out.println("Consumer Name: "+ConsumerName);
System.out.println("Consumer Previous Units: "+PrevReading);
System.out.println("Consumer Current Units: "+CurrReading);
}
```

```
System.out.println("Type of EBConnection: "+EBConn);
System.out.println("-----");
System.out.println ("Total Amount(Rs.): "+Bill);
}
}
class ElectBillGen
{
public static void main (String[] args)
{
    ElectBill b=new ElectBill();
    b.input_data();
    b.calculate_bill();
    b.display();
}
}
```



**OUTPUT:**

```
C:\MRKIT>javac ElectBillGen.java
C:\MRKIT>java ElectBillGen
Enter Consumer Number:
123
Enter Consumer Name:
MRKIT
Enter Previous Units:
500
Enter Current Units consumed:
1000
Enter the types of EB Connection(domestic or commercial)
commercial
-----
ELCTRICITY BILL
-----
Consumer Number: 123
Consumer Name: MRKIT
Consumer Previous Units: 500
Consumer Current Units: 1000
Type of EBConnection: commercial
-----
Total Amount(Rs.): 5050.0
```

**RESULT:**

Thus the java application was developed for generate electricity bill.

<b>EXP NO: 2</b>	<b>DEVELOP A JAVA APPLICATION TO IMPLEMENT CONVERTERS</b>
<b>DATE:</b>	

**AIM:**

To develop a java application to implement currency, distance and time converters using packages.

**ALGORITHM:**

**Step 1:** Create a folder named ConversionDemo, this is the name of the package.

**Step 2:** Following 3 Java files are stored in this folder(ConversionDemo)

- CurrencyConverter.java
- DistanceCoverter.java
- TimeConverter.java

**Step 3:** Now go to the parent directory of ConversionDemo directory and store following Java program under this parent directory.

- Converter.java

**Step 4:** Open the command prompt and issue following commands

```
D:\>cd ConversionDemo
```

```
D:\ConversionDemo>javac CurrencyConverter.java
```

```
D:\ConversionDemo>javac DistanceConverter.java
```

```
D:\ConversionDemo>javac TimeConverter.java
```

```
D:\ ConversionDemo>cd..
```

```
D:\>javac Converter.java
```

```
D:\>javac Converter
```

**Step 5:** Run the program.

**PROGRAM CODE:****CurrencyConverter.java**

```
package ConversionDemo;
public class CurrencyConverter
{
double ER=0;
public CurrencyConverter(double CurrentExchange)
{
ER=CurrentExchange;
}
public double DollarToINR(double Dollars)
{
double INR= 0;
INR =Dollars *ER;
return INR;
}
public double INRToDollar(double INR)
{
double Dollars= 0;
Dollars= INR/ER;
return Dollars;
}
public double EuroToINR(double Euros)
{
double INR=0;
INR=Euros*ER;
return INR;
}
public double INRToEuro(double INR)
{
double Euros=0;
Euros=INR*ER;
```

```
return Euros;
}
public double YenToINR(double Yens)
{
double INR=0;
INR=Yens*ER;
return INR;
}
public double INRToYen(double INR)
{
double Yens =0;
Yens=INR/ER;
return Yens;
}
}
```

**DistanceConverter.java**

```
package ConversionDemo;
public class DistanceConverter
{
public double MeterToKM(double Meters)
{
double KM=0;
KM=Meters/1000;
return KM;
}
public double KMToMeter(double KM)
{
double Meters=0;
Meters=KM*1000;
return Meters;
}
public double MileToKM(double Miles)
{
```

```
double KM=0;
KM=Miles/0.621371;
return KM;
}
public double KMToMile(double KM)
{
double Miles=0;
Miles=KM*0.621371;
return Miles;
}
}
```

**TimeConverter.java**

```
package ConversionDemo;
public class TimeConverter
{
public double HrToMin(double Hours)
{
double Minutes=0;
Minutes=Hours*60;
return Minutes;
}
public double MinToHour(double Minutes)
{
double Hours=0;
Hours=Minutes/60;
return Hours;
}
public double HrToSec(double Hours)
{
double Seconds=0;
Seconds=Hours*3600;
return Seconds;
}
}
```

```
public double SecToHour(double Seconds)
{
double Hours=0;
Hours=Seconds/3600;
return Hours;
}
}
```

**Converter.java**

```
import ConversionDemo.CurrencyConverter;
import ConversionDemo.DistanceConverter;
import ConversionDemo.TimeConverter;
import java.util.Scanner;
class Converter
{
public static void main(String[] args)throws NoClassDefFoundError
{
double CurrentExchange;
int choice,choice1,choice2,choice3;
double inr;
double km;
double hr;
char ans='y';
do
{
System.out.println("\n Main Menu");
System.out.println("1.Currency Converter \n2.Distance Converter \n3. Time Converter");
System.out.println("Enter your choice:");
Scanner input=new Scanner(System.in);
choice=input.nextInt();
switch(choice)
{
case 1:System.out.println("\tCurrency Conversion");
do
{
System.out.println("Menu For Currency Conversion");
```

```
System.out.println("1.Dollar to INR");
System.out.println("2.INR to Dollar");
System.out.println("3.Euro to INR");
System.out.println("4.INR to Euro");
System.out.println("5.Yen to INR");
System.out.println("6.INR to Yen");
System.out.println("Enter your choice:");
choice1=input.nextInt();
System.out.println("Please enter the current exchange rate:");
CurrentExchange=input.nextDouble();
CurrencyConverter cc=new CurrencyConverter(CurrentExchange);
switch(choice1)
{
case 1:
System.out.println("Enter Dollars:");
double dollar=input.nextDouble();
System.out.println(dollar+"dollars are Converted to "+cc.DollarToINR(dollar)+"Rs.");
break;
case 2:
System.out.println("Enter INR:");
inr=input.nextDouble();
System.out.println(inr+"Rs.are converted to "+cc.INRToDollar(inr)+"Dollars");
break;
case 3:
System.out.println("Enter Euro:");
double euro=input.nextDouble();
System.out.println(euro+"Euros are converted to "+cc.EuroToINR(euro)+"Rs.");
break;
case 4:
System.out.println("Enter INR:");
inr=input.nextDouble();
System.out.println(inr+"Rs.are converted to "+cc.INRToEuro(inr)+"Euros");
break;
case 5:
System.out.println("Enter Yen:");
double yen=input.nextDouble();
```

```
System.out.println(yen+"Yens are converted to "+cc.YenToINR(yen)+"Rs.");
break;
case 6:
System.out.println("Enter INR:");
inr=input.nextDouble();
System.out.println(inr+"Rs. are converted to "+cc.INRToYen(inr)+"Yens");
break;
}
System.out.println("Do You want to go to Currency Conversion Menu?(y/n)");
ans=input.next().charAt(0);
}
while(ans=='y');
break;
case 2: System.out.println("\t Distance Conversion");
do
{
System.out.println("Menu for Distance Conversion");
System.out.println("1.Meter to Km");
System.out.println("2.Km to Meter");
System.out.println("3.Miles to Km");
System.out.println("4.Km to Miles");
System.out.println("Enter your choice:");
choice2=input.nextInt();
DistanceConverter dc=new DistanceConverter();
switch(choice2)
{
case 1:
System.out.println("Enter meters to convert to Km:");
double meter=input.nextDouble();
System.out.println(meter+"Meters are converted to "+dc.MeterToKM(meter)+"Km.");
break;
case 2:
System.out.println("Enter Km to convert to meters:");
km=input.nextDouble();
System.out.println(km+"Km. are converted meters "+dc.KMToMeter(km)+"Meters");
break;
```



case 3:

```
System.out.println("Enter Miles to convert to Km:");
double miles=input.nextDouble();
System.out.println(miles+"Miles are converted to "+dc.MileToKM(miles)+"Km.");
break;
```

case 4:

```
System.out.println("Enter Km to convert to miles:");
km=input.nextDouble();
System.out.println(km+"Km.are converted miles "+dc.KMToMile(km)+"Miles");
break;
```

```
}
```

```
System.out.println("Do You want to go to Distance Conversion Menu?(y/n)");
```

```
ans=input.next().charAt(0);
```

```
}
```

```
while(ans=='y');
```

```
break;
```

```
case 3:System.out.println("\tTime Conversion");
```

```
do
```

```
{
```

```
System.out.println("Menu For Time Conversion");
```

```
System.out.println("1.Hour to Minutes");
```

```
System.out.println("2.Minutes to Hour");
```

```
System.out.println("3.Hour to Seconds");
```

```
System.out.println("4.Seconds to Hour");
```

```
System.out.println("Enter your choice:");
```

```
choice3=input.nextInt();
```

```
TimeConverter tc=new TimeConverter();
```

```
switch(choice3)
```

```
{
```

case 1:

```
System.out.println("Enter hours to convert to Minutes:");
```

```
hr=input.nextDouble();
```

```
System.out.println(hr+"Hours are converted to "+tc.HrToMin(hr)+"min.");
```

```
break;
```

case 2:

```
System.out.println("Enter Minutes to convert to hours:");
```

```
double minutes=input.nextDouble();
System.out.println(minutes+"Minutes are converted hours "+tc.MinToHour(minutes)+"Hours");
break;
case 3:
System.out.println("Enter Hours to convert to seconds:");
hr=input.nextDouble();
System.out.println(hr+"Hours are converted to "+tc.HrToSec(hr)+"Seconds.");
break;
case 4:
System.out.println("Enter Seconds to convert to hours:");
double seconds=input.nextDouble();
System.out.println(seconds+"Seconds are converted hours "+tc.SecToHour(seconds)+"Hours");
break;
}
System.out.println("Do You want to go to Time Conversion Menu?(y/n)");
ans=input.next().charAt(0);
}
while(ans=='y');
break;
}
System.out.println("Do you want to go back to Main Menu?(y/n)");
ans=input.next().charAt(0);
}while(ans=='y');
}
}
```

**OUTPUT:**

```
C:\MRKIT>cd ConversionDemo
C:\ MRKIT \ConversionDemo>javac CurrencyConverter.java
C:\ MRKIT \ConversionDemo>javac DistanceConverter.java
C:\ MRKIT \ConversionDemo>javac TimeConverter.java
C:\ MRKIT \ ConversionDemo>cd..
C:\MRKIT>javac Converter.java
C:\MRKIT>java Converter
Main Menu
1. Currency Converter
2. Distance Converter
3. Time Converter
Enter your choice:
1
    Currency Conversion
Menu for Currency Conversion
1. Dollar to INR
2. INR to Dollar
3. Euro to INR
4. INR to Euro
5. Yen to INR
6. INR to Yen
Enter your choice:
1
Please enter the current exchange rate:
66.21
Enter Dollars:
25
25.0dollars are Converted to1655.2499999999998Rs.
Do You want to go to Currency Conversion Menu?(y/n)
n
Do you want to go back to Main Menu?(y/n)
y
Main Menu
```

1. Currency Converter
2. Distance Converter
3. Time Converter

Enter your choice:

2

Distance Conversion

Menu For Distance Conversion

1. Meter to Km
2. Km to Meter
3. Miles to Km
4. Km to Miles

Enter your choice:

1

Enter meters to convert to Km:

5000

5000.0Meters are converted to 5.0Km.

Do You want to go to Distance Conversion Menu?(y/n)

n

Do you want to go back to Main Menu?(y/n)

**RESULT:**

Thus the Java application of currency, distance and time converters are implemented using packages and executed successfully.

<b>EXP NO: 3</b>	<b>DEVELOP A JAVA APPLICATION FOR EMPLOYEE PAYROLL SYSTEM</b>
<b>DATE:</b>	

**AIM:**

To develop a java application to generate pay slips for the employees with their gross and net salary.

**ALGORITHM:**

**Step 1:** Create a class employee with emp\_name, Emp\_id, Address, Mail\_id, Mobile\_no as members

**Step 2:** Inherit the classes, Programmer, Asst\_Professor, Professor from employee class.

**Step 3:** Adding BP, DA, HRA, PF and Fund as the member of inherited classes.

**Step 4:** Compute the pay with 97% of BP as DA, 10% of BP as HRA, 12% of BP as PF and 01% of BP for club fund

**Step 5:** Finally generate pay slips for the employees with their gross and net salary.

**PROGRAM CODE:**

```
import java.util.Scanner;
class Employee
{
int Emp_id;
String Emp_name;
String Address;
String Mail_Id;
int Mobile_no;
Employee(int id, String name,String addr,String mail,int mob)
{
this.Emp_id=id;
this.Emp_name=name;
this.Address=addr;
this.Mail_Id=mail;
this.Mobile_no=mob;
}
}
class Programmer extends Employee
{
double BP,Gross_salary,Net_salary;
public Programmer(int id,String name,String addr,String mail,int mob)
{
super(id,name,addr,mail,mob);
}
void computePay()
{
System.out.print("Enter Basic Pay:");
Scanner input=new Scanner(System.in);
BP=input.nextDouble();
double DA,HRA,PF,Fund;
DA=(BP*97/100);
HRA=(BP*10/100);
```

```
PF=(BP*12/100);
Fund=(BP*0.1/100);
Gross_salary=BP+DA+HRA;
Net_salary=BP+DA+HRA-(PF+Fund);
System.out.println("Emp_ID:"+Emp_id);
System.out.println("Emp_Name:"+Emp_name);
System.out.println("Address:"+Address);
System.out.println("Mail_ID:"+Mail_Id);
System.out.println("Mobile Number:"+Mobile_no);
System.out.println("Gross Pay:"+Gross_salary);
System.out.println("Net Pay:"+Net_salary);
}
}
class Asst_professor extends Employee
{
double BP,Gross_salary,Net_salary;
public Asst_professor(int id,String name,String addr,String mail,int mob)
{
super(id,name,addr,mail,mob);
}
void computePay()
{
System.out.print("Enter Basic Pay:");
Scanner input=new Scanner(System.in);
BP=input.nextDouble();
Gross_salary=BP;
double DA,HRA,PF,Fund;
DA=(BP*97/100);
HRA=(BP*97/100);
PF=(BP*12/100);
Fund=(BP*0.1/100);
Net_salary=BP+DA+HRA-(PF+Fund);
System.out.println("Emp_ID:"+Emp_id);
```

```
System.out.println("Emp_Name:"+Emp_name);
System.out.println("Address:"+Address);
System.out.println("Mail_ID:"+Mail_Id);
System.out.println("Mobile Number:"+Mobile_no);
System.out.println("Gross Pay:"+Gross_salary);
System.out.println("Net Pay:"+Net_salary);
}
}
class Associate_professor extends Employee
{
double BP,Gross_salary,Net_salary;
public Associate_professor(int id,String name,String addr,String mail,int mob)
{
super(id,name,addr,mail,mob);
}
void computePay()
{
System.out.print("Enter Basic Pay:");
Scanner input=new Scanner(System.in);
BP=input.nextDouble();
Gross_salary=BP;
double DA,HRA,PF,Fund;
DA=(BP*97/100);
HRA=(BP*97/100);
PF=(BP*12/100);
Fund=(BP*0.1/100);
Net_salary=BP+DA+HRA-(PF+Fund);
System.out.println("Emp_ID:"+Emp_id);
System.out.println("Emp_Name:"+Emp_name);
System.out.println("Address:"+Address);
System.out.println("Mail_ID:"+Mail_Id);
System.out.println("Mobile Number:"+Mobile_no);
System.out.println("Gross Pay:"+Gross_salary);
```



```
System.out.println("Net Pay:"+Net_salary);
}
}
class Professor extends Employee
{
double BP,Gross_salary,Net_salary;
public Professor (int id,String name,String addr,String mail,int mob)
{
super(id,name,addr,mail,mob);
}
void computePay()
{
System.out.print("Enter Basic Pay:");
Scanner input=new Scanner(System.in);
BP=input.nextDouble();
Gross_salary=BP;
double DA,HRA,PF,Fund;
DA=(BP*97/100);
HRA=(BP*12/100);
PF=(BP*12/100);
Fund=(BP*0.1/100);
Net_salary=BP+DA+HRA-(PF+Fund);
System.out.println("Emp_Id:"+Emp_id);
System.out.println("Emp_Name:"+Emp_name);
System.out.println("Address:"+Address);
System.out.println("Mail_ID:"+Mail_Id);
System.out.println("Mobil Number:"+Mobile_no);
System.out.println("Gross Pay:"+Gross_salary);
System.out.println("Net Pay:"+Net_salary);
}
}
public class PaySlip
{
```

```
public static void main(String[] args)
{
Programmer p=new Programmer(10,"AAA","XXX","aaa_xxx@gail.com",11111111);
System.out.println("-----Programmer-----");
p.computePay();
Asst_professor Ap=new Asst_professor(20,"BBB","yyy","bbb_yyy@gmail.com",22222222);
System.out.println("-----Assistant Professor-----");
Ap.computePay();
Associate_professor As=new
Associate_professor(30,"CCC","zzz","ccc_zzz@gmail.com",33333333);
System.out.println("-----Associate Professor-----");
As.computePay();
Professor pf=new Professor(40,"DDD","WWW","ddd_www@gmail.com",44444444);
System.out.println("-----Professor-----");
pf.computePay();
}
}
```

**OUTPUT:**

```
C:\MRKIT>javac PaySlip.java
```

```
C:\MRKIT>java PaySlip
```

```
-----Programmer-----
```

```
Enter Basic Pay:8000
```

```
Emp_ID:10
```

```
Emp_Name:AAA
```

```
Address:XXX
```

```
Mail_ID;aaa_xxx@gail.com
```

```
Mobile Number:11111111
```

```
Gross Pay:16560.0
```

```
Net Pay:15592.0
```

```
-----Assistant Professor-----
```

```
Enter Basic Pay:10000
```

```
Emp_ID:20
```

```
Emp_Name:BBB
```

```
Address:yyy
```

```
Mail_ID:bbb_yyy@gmail.com
```

```
Mobile Number:22222222
```

```
Gross Pay:10000.0
```

```
Net Pay:28190.0
```

```
-----Associate Professor-----
```

```
Enter Basic Pay:15000
```

```
Emp_ID:30
```

```
Emp_Name:CCC
```

```
Address:zzz
```

```
Mail_ID:ccc_zzz@gmail.com
```

```
Mobile Number:33333333
```

```
Gross Pay:15000.0
```

```
Net Pay:42285.0
```

```
-----Professor-----
```

```
Enter Basic Pay:20000
```

```
Emp_Id:40
```

Emp\_Name:DDD

Address:WWW

Mail\_ID:ddd\_www@gmail.com

Mobil Number:44444444

Gross Pay:20000.0

Net Pay:39380.0

**RESULT:**

Thus the Java application of employee payroll system implemented and generated pay slip successfully.

<b>EXP NO: 4</b>	<b>DESIGN A JAVA INTERFACE USING ARRAY FOR ADT STACK</b>
<b>DATE:</b>	

**AIM:**

To design a java interface using array for ADT stack with provide necessary exception handling.

**ALGORITHM:**

**Step 1:** Create a one interface1.java file inside that declare two methods push(int) and pop.

**Step 2:** Implements an interface1 to class name ArrStack.

**Step 3:** Use push operation  $st[++top]=item$  to insert elements.

**Step 4:** Use pop operation to remove the elements from stack.

**Step 5:** Using exception handling mechanism to catch the exception if certain condition meets.

**Step 6:** Run the program.

**Step 7:** Stop.

**PROGRAM CODE:****interface1 .java:**

```
interface interface1
{
void push(int item);
int pop();
}
```

**StackDemo.java:**

```
import java.io.*;
import java.util.*;
class ArrStack implements interface1
{
int size;
int[] st;
private int top;
private int items_count;
public ArrStack(int Max)
{
```

```
size=Max;
st=new int[size];
top=-1;
//items_count=0;
}
public void push(int item)
{
try{
st[++top]=item;
//items_count=0;
}
catch(Exception e){
System.out.println("\n Stack Full!! Cannot insert element");
}
}
public int pop()
{
int item;
try{
item=st[top];
top--;
//items_count--;
return item;
}
catch(Exception e)
{
System.out.println("\n Stack Empty!!Cannot pop the element");
return -1;
}
}
public void display()
{
System.out.println("\n Stack contains:");
```

```
for(int i=top;i>=0;i--)  
{  
if(top==0)  
System.out.println("Stack is empty");  
else if(size==top)  
System.out.println("Stack is Full!!");  
System.out.println(" "+st[i]);  
}  
}  
}  
class StackDemo  
{  
public static void main(String[]args)  
{  
ArrStack obj=new ArrStack(10);  
System.out.println("\n-----");  
System.out.println("\nStack using Array");  
System.out.println("\n-----");  
obj.push(10);  
obj.push(20);  
obj.push(30);  
obj.display();  
System.out.println("\n The Popped item is "+obj.pop());  
obj.display();  
}  
}
```

**OUTPUT:**

```
C:\MRKIT>javac interface1.java
```

```
C:\MRKIT>javac StackDemo.java
```

```
C:\MRKIT>java StackDemo
```

```
-----
```

```
Stack using Array
```

```
-----
```

```
Stack contains:
```

```
30
```

```
20
```

```
10
```

```
The Popped item is 30
```

```
Stack contains:
```

```
20
```

```
10
```

**RESULT:**

Thus the ADT stack were implemented using interface with arrays successfully.



<b>EXP NO: 5</b>	<b>DEVELOP A JAVA PROGRAM TO PERFORM STRING OPERATIONS USING ARRAYLIST</b>
<b>DATE:</b>	

**AIM:**

To develop a java program to perform string operations using array list.

**ALGORITHM:**

**Step 1:** Start the program.

**Step 2:** Create a class as name Lst inside the class declared Scanner class object as sc, string variable str.

**Step 3:** Create a function append, insert and for list all strings starts with given letter.

**Step 4:** In append() function add string at end of the string ans=sc.next().charAt(0)

**Step 5:** In insert() function add string at beginning of the string index=sc.nextInt()

**Step 6:** In find() function search the string from all the strings starts with given letter.

**Step 7:** Run the program

**Step 8:** Stop.

**PROGRAM CODE:**

```
import java.util.*;
class Lst
{
Scanner sc;
String str;
public void append(ArrayList AL)
{
char ans;
do
{
System.out.println("Enter string:");
sc=new Scanner(System.in);
str=sc.next();
AL.add(str);
System.out.println("Do you want append more string:");
```

```
ans=sc.next().charAt(0);
}
while(ans=='y');
System.out.println("The array elements are..." +AL);
System.out.println("The array size is..." +AL.size());
}
public void insert(ArrayList AL)
{
System.out.println("Enter string:");
sc=new Scanner(System.in);
str=sc.next();
System.out.println("Enter index at which the string is to be inserted:");
sc=new Scanner(System.in);
int index=sc.nextInt();
AL.add(index,str);
System.out.println("They array elements are..." +AL);
}
public void search(ArrayList AL)
{
/*append(AL);*/
String searchStr;
System.out.println("Enter string for searching");
sc=new Scanner(System.in);
searchStr=sc.next();
boolean found=false;
Iterator<String> iter=AL.iterator();
String curItem="";
int pos=0;
while(iter.hasNext() == true)
{
pos=pos+1;
curItem=(String)iter.next();
if(curItem.equals(searchStr))
```

```
{
found=true;
break;
}
}
if(found==false)
{
pos=0;
}
if(pos!=0)
{
System.out.println(searchStr+"String Found in position:"+pos);
}
else
{
System.out.println(searchStr+"String not found");
}
}
public void find(ArrayList AL)
{
/*append(AL);*/
/*ArrayList Array of objects Conversion*/
Object[] objs=AL.toArray(new String[0]);
/*Array of objects to Array of String Conversion*/
String[]str_list=new String[objs.length];
System.arraycopy(objs,0,str_list,0,objs.length);
/*Enter the Starting letter*/
System.out.println("Enter starting letter:");
sc=new Scanner(System.in);
String searchChar=sc.next();
System.out.println("The strings starting with letter" +searchChar+" are...");
for(int i=0;i<str_list.length;i++)
{
```

```
if(str_list[i].startsWith(searchChar))
System.out.println(str_list[i]);
}
}
}
class ArrayListProg
{
public static void main(String[]args)
{
System.out.println("\n\t\t Program for Implementating Array List for of Strings");
ArrayList AL=new ArrayList<String>();
Lst obj=new Lst();
char ans;
do
{
System.out.println("Main Menu");
System.out.println("1.Append\n2.Insert at particular Index\n3.Search\n4.List strings");
System.out.print("Enter your choice:");
Scanner sc=new Scanner(System.in);
int choice=sc.nextInt();
switch(choice)
{
case 1:obj.append(AL);
break;
case 2:obj.insert(AL);
break;
case 3:obj.search(AL);
break;
case 4:obj.find(AL);
break;
}
System.out.println("Do you want to go to Main Menu?(y/n):");
ans=sc.next().charAt(0);
```

```
}  
while(ans=='y');  
}  
}
```

**RESULT:**

Thus the string operations append, insert and find were performed using array list successfully.

<b>EXP NO: 6</b>	<b>DEVELOP A JAVA PROGRAM TO CREATE AN ABSTRACT CLASS</b>
<b>DATE:</b>	

**AIM:**

To develop a java program to create an abstract class.

**ALGORITHM:**

**Step 1:** Start the program.

**Step 2:** Create a abstract class as shapes inside that declared abstract method as printArea().

**Step 3:** Create three classes named Rectangle, Triangle and Circle extends these with shapes.

**Step 4:** Initialized the method printArea() in each classes that prints the area of the given shape.

**Step 5:** Run the program

**Step 6:** Stop.

**PROGRAM CODE:**

```
import java.util.*;
abstract class Shapes
{
double a,b;
abstract void printArea();
}
class Rectangle extends Shapes
{
void printArea()
{
System.out.println("\t\t Calculating Area of Rectangle");
Scanner input=new Scanner(System.in);
System.out.println("Enter length:");
a=input.nextDouble();
System.out.println("Enter breadth:");
b=input.nextDouble();
double area=a*b;
System.out.println("Area of Rectangle:"+area);
```

```
}  
}  
class Triangle extends Shapes  
{  
void printArea()  
{  
System.out.println("\t \t Calculating Area of Triangle");  
Scanner input=new Scanner(System.in);  
System.out.println("Enter height:");  
a=input.nextDouble();  
System.out.print("Enter breath:");  
b=input.nextDouble();  
double area=0.5*a*b;  
System.out.println("Area of Triangle:"+area);  
}  
}  
class Circle extends Shapes  
{  
void printArea()  
{  
System.out.println("\t \t Calculating Area of Circle");  
Scanner input=new Scanner(System.in);  
System.out.print("Enter radius:");  
a=input.nextDouble();  
double area=3.14*a*a;  
System.out.println("Area of Circle:"+area);  
}  
}  
class AbstractClassDemo  
{  
public static void main(String[] args)  
{  
Shapes obj;
```

```
obj=new Rectangle();  
obj.printArea();  
obj=new Triangle();  
obj.printArea();  
obj=new Circle();  
obj.printArea();  
}  
}
```

**RESULT:**

Thus the java program were developed with an abstract class and executed successfully.



<b>EXP NO: 7</b>	<b>DEVELOP A JAVA PROGRAM TO IMPLEMENT USER DEFINED EXCEPTION HANDLING</b>
<b>DATE:</b>	

**AIM:**

To implement user defined exception handling using java program.

**ALGORITHM:**

**Step 1:** Start the program.

**Step 2:** Create a class as MyOwnException extends with Exception.

**Step 3:** Create a constructor as same as class name in parent class

**Step 4:** Create another class as MyExceptDemo inside that throws own created exception.

**Step 5:** Run the program.

**Step 6:** Stop.

**PROGRAM CODE:**

```
import java.lang.Exception;
class MyOwnException extends Exception
{
MyOwnException(String msg)
{
super(msg);
}
}
class MyExceptDemo
{
public static void main(String ags[])
{
int age;
age=15;
try
{
if(age<21)
throw new MyOwnException ("Your age is very less than the condition");
}
}
```

```
catch(MyOwnException e)
{
System.out.println("This is My Exception block");
System.out.println(e.getMessage());
}
finally
{
System.out.println("Finally block:End of the program");
}
}
}
```

**RESULT:**

Thus the user defined exception handling were implemented and executed successfully.

<b>EXP NO: 8</b>	<b>DEVELOP A JAVA PROGRAM THAT READS A FILE NAME FROM THE USER AND DISPLAYS INFORMATION ABOUT WHETHER THE FILE EXISTS</b>
<b>DATE:</b>	

**AIM:**

To develop a java program that reads a file name from the user and displays information about whether the file exists.

**ALGORITHM:**

**Step 1:** Start the program

**Step 2:** Create class name as FileDemo inside that class implement main() method.

**Step 3:** Create object for scanner class and also File class.

**Step 4:** Get the input through input.nextLine() method.

**Step 5:** Using some File class methods to get name, getpath etc..

**Step 6:** Run the program

**Step 7:** Stop

**PROGRAM CODE:**

```
import java.util.Scanner;
import java.io.File;
class FileDemo
{
public static void main(String[] args)
{
System.out.println("Enter the name of the file");
Scanner input=new Scanner(System.in);
String s=input.nextLine();
File f1=new File(s);
System.out.println("-----");
System.out.println("File Name:"+f1.getName());
System.out.println("Path:"+f1.getPath());
System.out.println("Abs Path:"+f1.getAbsolutePath());
System.out.println("This file is:"+(f1.exists()?"Exists":"Does not exists"));
System.out.println("Is file:"+f1.isFile());
System.out.println("Is Directory:"+f1.isDirectory());
System.out.println("Is Readable:"+f1.canRead());
```

```
System.out.println("Is Writable:"+f1.canWrite());  
System.out.println("Is Absolute:"+f1.isAbsolute());  
System.out.println("File Size:"+f1.length()+"bytes");  
System.out.println("Is Hidden:"+f1.isHidden());  
}  
}
```

**OUTPUT:**

```
C:\MRKITCSE>javac FileDemo.java
```

```
C:\MRKITCSE>java FileDemo
```

```
Enter the name of the file
```

```
FileDemo.java
```

```
-----
```

```
File Name:FileDemo.java
```

```
Path:FileDemo.java
```

```
Abs Path:C:\MRKITCSE\FileDemo.java
```

```
This file is:Exists
```

```
Is file:true
```

```
Is Directory:false
```

```
Is Readable:true
```

```
Is Writable:true
```

```
Is Absolute:false
```

```
File Size:890bytes
```

```
Is Hidden:false
```

**RESULT:**

Thus the java program for reads the filename from user and display information about whether the file exists were developed and executed successfully.

<b>EXP NO: 9</b>	<b>DEVELOP A JAVA PROGRAM THAT IMPLEMENTS A MULTI-THREADED APPLICATION THAT HAS THREE THREADS.</b>
<b>DATE:</b>	

**AIM:**

To implement a multithreaded application that has three threads using java program.

**ALGORITHM:**

**Step 1:** Start the program

**Step 2:** Create a class as NumberGenerate inside that class declared private variable and public method name as put().

**Step 3:** Create a thread by extending thread class

**Step 4:** Using some methods such as getName(), getPriority(), run(), sleep() and start().

**Step 5:** Run the program

**Step 6:** Stop

**PROGRAM CODE:**

```
import java.util.*;
class NumberGenerate
{
private int value;
private boolean flag;
public synchronized void put()
{
while (flag)
{
try
{
wait();
}
catch(InterruptedException e) {}
}
flag=true;
Random random=new Random();
```

```
this.value=random.nextInt(10);
System.out.println("The generated Number is:"+value);
notifyAll();
}
public synchronized void get1()
{
while (!flag){
try{
wait();
}
catch(InterruptedException e) {}
}
if(value%2==0)
{
System.out.println("Second is executing now...");
int ans=value*value;
System.out.println(value+"is Even Number and its square

is:"+ans);
}
flag=false;
notifyAll();
}
public synchronized void get2()
{
while (!flag){
try{
wait();
}
catch(InterruptedException e) {}
}
if(value%2==0)
{
```

```
System.out.println("Third Thread is executing now...");
int ans=value*value*value;
System.out.println(value+"is Odd Number and its cube is:"+ans);
}
flag=false;
notifyAll();
}
}
public class TestNumber
{
public static void main(String[]args)
{
final NumberGenerate obj=new NumberGenerate();
Thread producerThread=new Thread()
{
public void run()
{
for(int i=1;i<=6;i++)
{
System.out.println("Main thread started...");
obj.put();
try
{
Thread.sleep(1000);
}
catch(InterruptedException e){}
}
}
};
Thread consumerThread1=new Thread()
{
public void run()
{
```



```
for(int i=1;i<=3;i++)
{
obj.get1();
try
{
Thread.sleep(2000);
}
catch(InterruptedException e){}
}
};
Thread consumerThread2=new Thread()
{
public void run()
{
for(int i=1;i<=3;i++)
{
obj.get2();
try
{
Thread.sleep(3000);
}
catch(InterruptedException e){}
}
}
};
producerThread.start();
consumerThread1.start();
consumerThread2.start();
}
}
```

**OUTPUT:**

```
C:\MRKITCSE>javac TestNumber.java
```

```
C:\MRKITCSE>java TestNumber
```

```
Main thread started...
```

```
The generated Number is:7
```

```
Third Thread is executing now...
```

```
7is Odd Number and its cube is:343
```

```
Main thread started...
```

```
The generated Number is:1
```

```
Main thread started...
```

```
The generated Number is:8
```

```
Main thread started...
```

```
The generated Number is:6
```

```
Second is executing now...
```

```
6is Even Number and its square is:36
```

```
Main thread started...
```

```
The generated Number is:3
```

```
Main thread started...
```

```
The generated Number is:9
```

```
Third Thread is executing now...
```

```
9is Odd Number and its cube is:729
```

**RESULT:**

Thus the multithreaded application that has three threads were implemented and executed successfully.

<b>EXP NO: 10</b>	<b>DEVELOP A JAVA PROGRAM TO FIND THE MAXIMUM VALUE FROM THE GIVEN TYPE OF ELEMENTS USING A GENERIC FUNCTION.</b>
<b>DATE:</b>	

**AIM:**

To find the maximum value from the given type of elements using a generic function.

**ALGORITHM:**

**Step 1:** Start the program

**Step 2:** Create a class as Test inside that class defined generic functions

**Step 3:** Create a list and add instances to the list and invoke the method.

**Step 4:** Create ArrayList for integers (Arrays.asList(1,55,4,3,23,12,25,9)) and also character (Arrays.asList('a','e','i','o','u')).

**Step 5:** Using max() functions to find the maximum values.

**Step 6:** Run the program.

**Step 7:** Stop

**PROGRAM CODE:**

```
import java.util.*;
public class Test
{
public static <T extends Object & Comparable<?super T>> T max(Collection<? extends T> coll)
{
Iterator<? extends T> MyList=coll.iterator();
T element=MyList.next();
while(MyList.hasNext())
{
T next_element=MyList.next();
if(next_element.compareTo(element)>0)
element=next_element;
}
return element;
}
public static void main(String[] args)
```

```
{  
List<Integer> ints=new ArrayList<Integer>(Arrays.asList(1,55,4,3,23,12,25,9));  
int max=Collections.max(ints);  
System.out.println(ints);  
System.out.println("Maximum Value is " + max);  
List<Character> chars=new ArrayList<Character>(Arrays.asList('a','e','i','o','u'));  
char maxc=Collections.max(chars);  
System.out.println(chars);  
System.out.println("Maximum value is " + maxc);  
}  
}
```

**OUTPUT:**

```
C:\MRKITCSE>javac Test.java
```

```
C:\MRKITCSE>java Test
```

```
[1, 55, 4, 3, 23, 12, 25, 9]
```

```
Maximum Value is 55
```

```
[a, e, i, o, u]
```

```
Maximum value is u
```

**RESULT:**

Thus the maximum values were found from the given type of elements using a generic function.

<b>EXP NO: 11</b>	<b>DESIGN A CALCULATOR USING EVENT-DRIVEN PROGRAMMING PARADIGM OF JAVA</b>
<b>DATE:</b>	

**AIM:**

To design a calculator using event driven programming paradigm of java.

**ALGORITHM:**

**Step 1:** Start the program.

**Step 2:** Create a class as calci and extends with JFrame class

**Step 3:** Using calci constructor create buttons that are in the calculator

**Step 4:** Using actionPerformed() method define the function that has to be done when the corresponding buttons are pressed.

**Step 5:** In the main function create the objects for the class scientific calculator.

**Step 6:** Run the program

**Step 7:** Stop.

**PROGRAM CODE:**

```
import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import javax.swing.event.*;
public class calci extends JFrame implements ActionListener
{
    Container contentpane;
    JPanel DisplayPanel,ControlPanel;
    JTextField txt;
    JButton one,two,three,four,five,six,seven,eight,nine,zero;
    JButton plus,min,mul,div,log,CLR,exp;
    JButton eq,addSub,dot,memread,memcancel,memplus;
    JButton sqrt,sin,cos,tan,onebyx;
    double tempnum,tempnextnum,result,ans;
    static double ValueInMem;
    int num1=0,num2=0;
```

```
int MemPlusFlag=1,RepeatFlag=0;
char ch;
calci()
{
contentpane=getContentPane();
contentpane.setLayout(new BorderLayout());
JPanel DisplayPanel=new JPanel();
txt=new JTextField(30);
txt.setHorizontalAlignment(SwingConstants.RIGHT);
txt.addKeyListener(
new KeyAdapter()
{
public void keyTyped(KeyEvent keyevent)
{
char ch=keyevent.getKeyChar();
if(ch>='0' && ch<='9')
{ }
else
{
keyevent.consume();
}
}
});
DisplayPanel.add(txt);
ControlPanel=new JPanel();
contentpane.add("Center",ControlPanel);
contentpane.add("North",DisplayPanel);
ControlPanel.setLayout(new GridLayout(7,4,5,5));
memcancel=new JButton("MC");
ControlPanel.add(memcancel);
memcancel.addActionListener(this);
one=new JButton("1");
ControlPanel.add(one);
```

```
one.addActionListener(this);
two=new JButton("2");
ControlPanel.add(two);
two.addActionListener(this);
three=new JButton("3");
ControlPanel.add(three);
three.addActionListener(this);
memread=new JButton("MR");
ControlPanel.add(memread);
memread.addActionListener(this);
four=new JButton("4");
ControlPanel.add(four);
four.addActionListener(this);
five=new JButton("5");
ControlPanel.add(five);
five.addActionListener(this);
six=new JButton("6");
ControlPanel.add(six);
six.addActionListener(this);
memplus=new JButton("M+");
ControlPanel.add(memplus);
memplus.addActionListener(this);
seven=new JButton("7");
ControlPanel.add(seven);
seven.addActionListener(this);
eight=new JButton("8");
ControlPanel.add(eight);
eight.addActionListener(this);
nine=new JButton("9");
ControlPanel.add(nine);
nine.addActionListener(this);
zero=new JButton("0");
ControlPanel.add(two);
```



```
zero.addActionListener(this);

addSub=new JButton("/-");
ControlPanel.add(addSub);
addSub.addActionListener(this);
dot=new JButton(".");
ControlPanel.add(dot);
dot.addActionListener(this);
eq=new JButton("=");
ControlPanel.add(eq);
eq.addActionListener(this);
plus=new JButton("+");
ControlPanel.add(plus);
plus.addActionListener(this);
min=new JButton("-");
ControlPanel.add(min);
min.addActionListener(this);
mul=new JButton("*");
ControlPanel.add(mul);
mul.addActionListener(this);
div=new JButton("/");
ControlPanel.add(div);
div.addActionListener(this);
sqrt=new JButton("Sqrt");
ControlPanel.add(sqrt);
sqrt.addActionListener(this);
log=new JButton("LOG");
ControlPanel.add(log);
log.addActionListener(this);
sin=new JButton("SIN");
ControlPanel.add(sin);
sin.addActionListener(this);
cos=new JButton("COS");
```

```
ControlPanel.add(cos);
cos.addActionListener(this);
tan=new JButton("TAN");
ControlPanel.add(tan);
tan.addActionListener(this);
exp=new JButton("Exp");
ControlPanel.add(exp);
exp.addActionListener(this);
onebyx=new JButton("1/x");
ControlPanel.add(onebyx);
onebyx.addActionListener(this);
CLR=new JButton("AC");
ControlPanel.add(CLR);
CLR.addActionListener(this);

setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
public void actionPerformed(ActionEvent e)
{
String s=e.getActionCommand();
if(s.equals("1"))
{
if(RepeatFlag==0)
{
txt.setText(txt.getText() +"1");
}
else
{
txt.setText("");
txt.setText(txt.getText() +"1");
RepeatFlag=0;
}
}
}
```

```
if(s.equals("2"))
{
if(RepeatFlag==0)
{
txt.setText(txt.getText() +"2");
}
else
{
txt.setText("");
txt.setText(txt.getText() +"2");
RepeatFlag=0;
}
}
if(s.equals("3"))
{
if(RepeatFlag==0)
{
txt.setText(txt.getText() +"3");
}
else
{
txt.setText("");
txt.setText(txt.getText() +"3");
RepeatFlag=0;
}
}
if(s.equals("4"))
{
if(RepeatFlag==0)
{
txt.setText(txt.getText() +"4");
}
else
```

```
{
txt.setText("");
txt.setText(txt.getText() + "4");
RepeatFlag=0;
}
}
if(s.equals("5"))
{
if(RepeatFlag==0)
{
txt.setText(txt.getText() + "5");
}
else
{
txt.setText("");
txt.setText(txt.getText() + "5");
RepeatFlag=0;
}
}
if(s.equals("6"))
{
if(RepeatFlag==0)
{
txt.setText(txt.getText() + "6");
}
else
{
txt.setText("");
txt.setText(txt.getText() + "6");
RepeatFlag=0;
}
}
if(s.equals("7"))
```

```
{
if(RepeatFlag==0)
{
txt.setText(txt.getText() +"7");
}
else
{
txt.setText("");
txt.setText(txt.getText() +"7");
RepeatFlag=0;
}
}
if(s.equals("8"))
{
if(RepeatFlag==0)
{
txt.setText(txt.getText() +"8");
}
else
{
txt.setText("");
txt.setText(txt.getText() +"8");
RepeatFlag=0;
}
}
if(s.equals("9"))
{
if(RepeatFlag==0)
{
txt.setText(txt.getText() +"9");
}
else
{
```

```
txt.setText("");
txt.setText(txt.getText() + "9");
RepeatFlag=0;
}
}
if(s.equals("0"))
{
if(RepeatFlag==0)
{
txt.setText(txt.getText() + "0");
}
else
{
txt.setText("");
txt.setText(txt.getText() + "0");
RepeatFlag=0;
}
}
if(s.equals("AC"))
{
txt.setText("");
num1=0;
num2=0;
RepeatFlag=0;
}
if(s.equals("/-"))
{
if(num1==0)
{
txt.setText("-"+txt.getText());
num1=1;
}
else
```

```
{
txt.setText(txt.getText());
}
}
if(s.equals("."))
{
if(num2==0)
{
txt.setText(txt.getText()+".");
num2=1;
}
else
{
txt.setText(txt.getText());
}
}
if(s.equals("+"))
{
if(txt.getText().equals(""))
{
txt.setText("");
tempnum=0;
ch='+';
}
else
{
tempnum=Double.parseDouble(txt.getText());
txt.setText("");
ch='+';
num2=0;
num1=0;
}
txt.requestFocus();
```

```
}  
if(s.equals("-"))  
{  
if(txt.getText().equals(""))  
{  
txt.setText("");  
tempnum=0;  
ch='-';  
}  
else  
{  
num1=0;  
num2=0;  
tempnum=Double.parseDouble(txt.getText());  
txt.setText("");  
ch='-';  
}  
txt.requestFocus();  
}  
if(s.equals("/"))  
{  
if(txt.getText().equals(""))  
{  
txt.setText("");  
tempnum=1;  
ch='/';  
}  
else  
{  
num1=0;  
num2=0;  
tempnum=Double.parseDouble(txt.getText());  
ch='/';  
}
```



```
txt.setText("");
}
txt.requestFocus();
}
if(s.equals("*"))
{
if(txt.getText().equals(""))
{
txt.setText("");
tempnum=1;
ch='*';
}
else
{
num1=0;
num2=0;
tempnum=Double.parseDouble(txt.getText());
ch='*';
txt.setText("");
}
txt.requestFocus();
}
if(s.equals("MC"))
{
ValueInMem=0;
txt.setText("");
}
if(s.equals("MR"))
{
txt.setText("");
txt.setText(txt.getText() + ValueInMem);
}
if(s.equals("M+"))
```

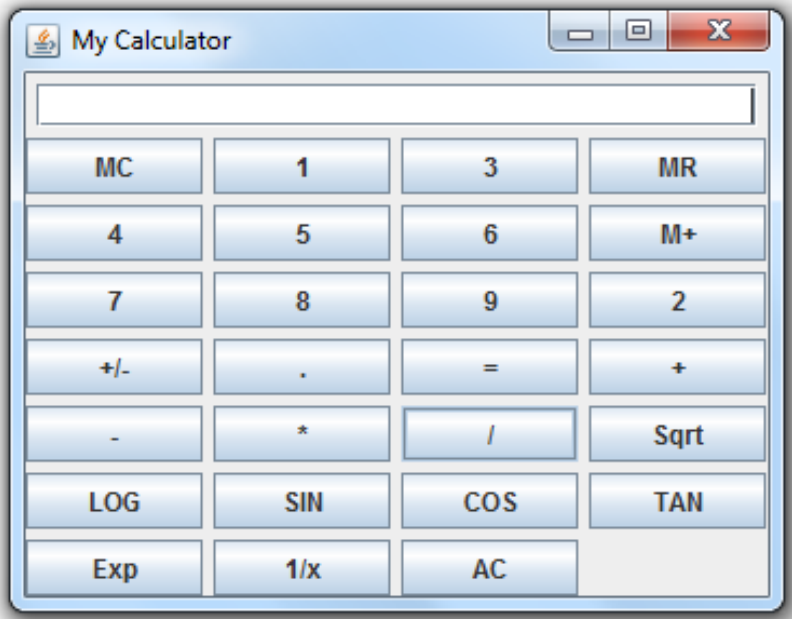
```
{
if(MemPlusFlag==1)
{
ValueInMem=Double.parseDouble(txt.getText());
MemPlusFlag++;
}
else
{
ValueInMem+=Double.parseDouble(txt.getText());
txt.setText(""+ValueInMem);
}
}
if(s.equals("LOG"))
{
if(txt.getText().equals(""))
{
txt.setText("");
}
else
{
ans=Math.log(Double.parseDouble(txt.getText()));
txt.setText("");
txt.setText(txt.getText() + ans);
}
}
if(s.equals("1/x"))
{
if(txt.getText().equals(""))
{
txt.setText("");
}
else
{
```

```
ans=1/Double.parseDouble(txt.getText());
txt.setText("");
txt.setText(txt.getText() + ans);
}
}
if(s.equals("Exp"))
{
if(txt.getText().equals(""))
{
txt.setText("");
}
else
{
ans=Math.exp(Double.parseDouble(txt.getText()));
txt.setText("");
txt.setText(txt.getText() + ans);
}
}
if(s.equals("Sqrt"))
{
if(txt.getText().equals(""))
{
txt.setText("");
}
else
{
ans=Math.sqrt(Double.parseDouble(txt.getText()));
txt.setText("");
txt.setText(txt.getText() + ans);
}
}
if(s.equals("SIN"))
{
```

```
if(txt.getText().equals(""))
{
txt.setText("");
}
else
{
ans=Math.sin(Double.parseDouble(txt.getText()));
txt.setText("");
txt.setText(txt.getText() + ans);
}
}
if(s.equals("COS"))
{
if(txt.getText().equals(""))
{
txt.setText("");
}
else
{
ans=Math.cos(Double.parseDouble(txt.getText()));
txt.setText("");
txt.setText(txt.getText() + ans);
}
}
if(s.equals("TAN"))
{
if(txt.getText().equals(""))
{
txt.setText("");
}
else
{
ans=Math.tan(Double.parseDouble(txt.getText()));
```

```
txt.setText("");
txt.setText(txt.getText() + ans);
}
}
if(s.equals("="))
{
if(txt.getText().equals(""))
{
txt.setText("");
}
else
{
tempnextnum=Double.parseDouble(txt.getText());
switch(ch)
{
case'+':
result=tempnum + tempnextnum;
break;
case'-':
result=tempnum - tempnextnum;
break;
case'/':
result=tempnum / tempnextnum;
break;
case'*':
result=tempnum * tempnextnum;
break;
}
txt.setText("");
txt.setText(txt.getText() + result);
RepeatFlag=1;
}
}
```

```
txt.requestFocus();  
}  
public static void main(String args[])  
{  
    calci c=new calci();  
    c.setTitle("My Calculator");  
    c.pack();  
    c.setVisible(true);  
}  
}
```

**OUTPUT:****RESULT:**

Thus the calculator was designed using event driven programming paradigm of java and executed successfully.

<b>EXP NO: 12</b>	<b>ONLINE EXAMINATION SYSTEM</b>
<b>DATE:</b>	

**AIM**

To design an Online Examination System using Java concepts.

**HARDWARE REQUIREMENTS:**

Pentium IV with 2 GB RAM

160 GB HARD Disk

Monitor 1024 x 768 colour

**SOFTWARE REQUIREMENTS:**

Microsoft Windows XP and above operating system

JDK 1.6 and above

Notepad

Internet Explorer 4.0

**PROCEDURE:**

**Step 1:** Start the program

**Step 2:** Using the swing components design the necessary layout.

**Step 3:** Using here Java array to store the questions, options and answers not database.

**Step 4:** Bookmark any question for the reconsideration while going to result.

**Step 5:** Show the result

**Step 6:** Stop the program

**PROGRAM CODE:**

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class OnlineTest extends JFrame implements ActionListener
{
    JLabel l;
    JRadioButton jb[]=new JRadioButton[5];
    JButton b1,b2;
    ButtonGroup bg;
    int count=0,current=0,x=1,y=1,now=0;
    int m[]=new int[10];
```



```
OnlineTest(String s)
{
super(s);
l=new JLabel();
add(l);
bg=new ButtonGroup();
for(int i=0;i<5;i++)
{
jb[i]=new JRadioButton();
add(jb[i]);
bg.add(jb[i]);
}
b1=new JButton("Next");
b2=new JButton("Bookmark");
b1.addActionListener(this);
b2.addActionListener(this);
add(b1);add(b2);
set();
l.setBounds(30,40,450,20);
jb[0].setBounds(50,80,100,20);
jb[1].setBounds(50,110,100,20);
jb[2].setBounds(50,140,100,20);
jb[3].setBounds(50,170,100,20);
b1.setBounds(100,240,100,30);
b2.setBounds(270,240,100,30);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setLayout(null);
setLocation(250,100);
setVisible(true);
setSize(600,350);
}
public void actionPerformed(ActionEvent e)
{
```

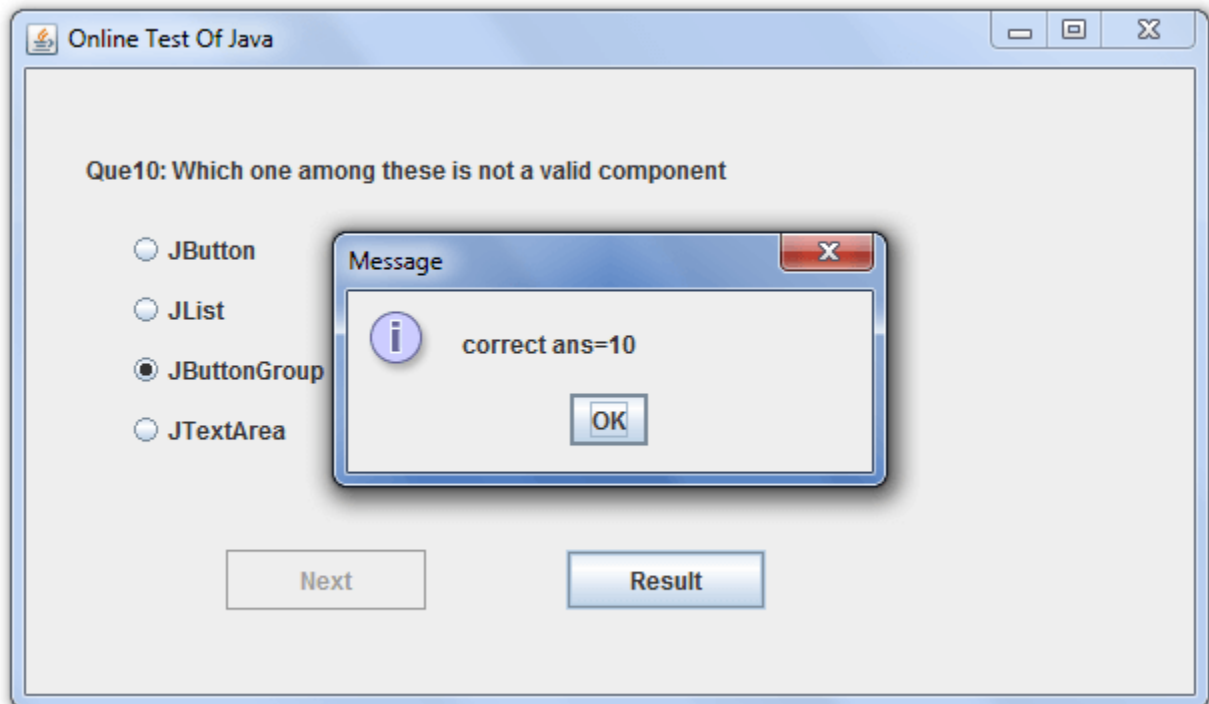
```
if(e.getSource()==b1)
{
if(check())
count=count+1;
current++;
set();
if(current==9)
{
b1.setEnabled(false);
b2.setText("Result");
}
}
if(e.getActionCommand().equals("Bookmark"))
{
JButton bk=new JButton("Bookmark"+x);
bk.setBounds(480,20+30*x,100,30);
add(bk);
bk.addActionListener(this);
m[x]=current;
x++;
current++;
set();
if(current==9)
b2.setText("Result");
setVisible(false);
setVisible(true);
}
for(int i=0,y=1;i<x;i++,y++)
{
if(e.getActionCommand().equals("Bookmark"+y))
{
if(check())
count=count+1;
```

```
now=current;
current=m[y];
set();
((JButton)e.getSource()).setEnabled(false);
current=now;
}
}
if(e.getActionCommand().equals("Result"))
{
if(check())
count=count+1;
current++;
//System.out.println("correct ans="+count);
JOptionPane.showMessageDialog(this,"correct ans="+count);
System.exit(0);
}
}
void set()
{
jb[4].setSelected(true);
if(current==0)
{
l.setText("Que1: Which one among these is not a datatype");
jb[0].setText("int");jb[1].setText("Float");jb[2].setText("boolean");jb[3].setText("char");
}
if(current==1)
{
l.setText("Que2: Which class is available to all the class automatically");
jb[0].setText("Swing");jb[1].setText("Applet");jb[2].setText("Object");jb[3].setText("ActionEvent");
}
if(current==2)
{
```

```
l.setText("Que3: Which package is directly available to our class without importing it");
jb[0].setText("swing");jb[1].setText("applet");jb[2].setText("net");jb[3].setText("lang");
}
if(current==3)
{
l.setText("Que4: String class is defined in which package");
jb[0].setText("lang");jb[1].setText("Swing");jb[2].setText("Applet");jb[3].setText("awt");
}
if(current==4)
{
l.setText("Que5: Which institute is best for java coaching");
jb[0].setText("Utek");jb[1].setText("Aptech");jb[2].setText("SSS IT");jb[3].setText("jtek");
}
if(current==5)
{
l.setText("Que6: Which one among these is not a keyword");
jb[0].setText("class");jb[1].setText("int");jb[2].setText("get");jb[3].setText("if");
}
if(current==6)
{
l.setText("Que7: Which one among these is not a class ");
jb[0].setText("Swing");jb[1].setText("Actionperformed");jb[2].setText("ActionEvent");jb[3].setText("Button");
}
if(current==7)
{
l.setText("Que8: which one among these is not a function of Object class");
jb[0].setText("toString");jb[1].setText("finalize");jb[2].setText("equals");jb[3].setText("getDocumentBase");
}
if(current==8)
{
l.setText("Que9: which function is not present in Applet class");
```

```
jb[0].setText("init");jb[1].setText("main");jb[2].setText("start");jb[3].setText("destroy");
}
if(current==9)
{
l.setText("Que10: Which one among these is not a valid component");
jb[0].setText("JButton");jb[1].setText("JList");jb[2].setText("JButtonGroup");jb[3].setText("JText
Area");
}
l.setBounds(30,40,450,20);
for(int i=0,j=0;i<=90;i+=30,j++)
jb[j].setBounds(50,80+i,200,20);
}
boolean check()
{
if(current==0)
return(jb[1].isSelected());
if(current==1)
return(jb[2].isSelected());
if(current==2)
return(jb[3].isSelected());
if(current==3)
return(jb[0].isSelected());
if(current==4)
return(jb[2].isSelected());
if(current==5)
return(jb[2].isSelected());
if(current==6)
return(jb[1].isSelected());
if(current==7)
return(jb[3].isSelected());
if(current==8)
return(jb[1].isSelected());
if(current==9)
```

```
return(jb[2].isSelected());  
return false;  
}  
public static void main(String s[])  
{  
new OnlineTest("Online Test Of Java");  
}  
}
```

**OUTPUT:****RESULT:**

Thus the design of an online examination system using java concepts has been successfully executed.