CONCORD ARTS AND SCIENCE COLLEGE

CONCORD EDUCITY, MUTTANNUR, (PO) PATTANNUR

FIFTH SEMESTER BACHELOR OF COMPUTER APPLICATION

ENTERPRISE JAVA PROGRAMMING

CONCORD ARTS AND SCIENCE COLLEGE

CONCORD EDUCITY, MUTTANNUR, (PO) PATTANNUR

CERTIFICATE

It is certified that this is a bonafide record of the original work done by

Mr./Mrs……………………………………….   Reg.no.....……………………

of Vth semester BCA in the ENTERPRISE JAVA PROGRAMMING lab during the year 2022-2023.

HOD: **Lecturer in charge:**

Submitted for practical examination held on ……………………

External Examiner

1.

2.

INDEX

JDBC

EMPLOYEE TABLE

AIM

JDBC program to insert ,delete and update records into Employee table.

PROGRAM

```
import java.sql.*;
import java.io.*;
public class Employee
{
public static void main(String args[])throws IOException
{
int ch,upc;
int no,sal;
String name;
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
try
{
Class.forName("org.postgresql.Driver");
}
catch(ClassNotFoundException e)
{
System.out.println("Unable to load driver");
}
try
{
Connection con=DriverManager.getConnection("jdbc:postgresql://localhost/empl",
"postgres","concord");
Statement stmt=con.createStatement(); System.out.println("\n.....Current
Records....\n");
System.out.println("ENO\tENAME\tSALARY\n");
ResultSet rs=stmt.executeQuery("select * from employee");
while(rs.next())

{
```

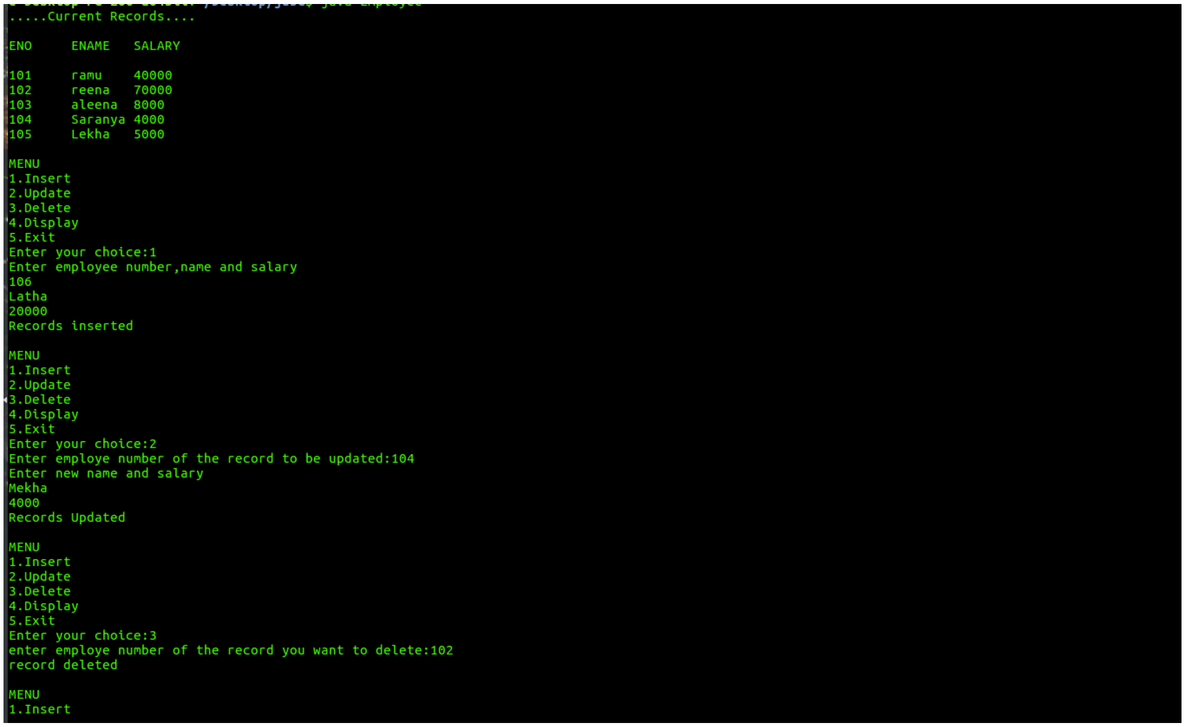
```
System.out.println(rs.getInt("eno")+"\t"+rs.getString("ename")+"\t"+rs.getInt("sala
ry"));
}
do
{
System.out.print("\nMENU\n1.Insert\n2.Update\n3.Delete\n4.Display\n5.Exit\nEn
ter your choice:");
ch=Integer.parseInt(br.readLine());
switch(ch)
{
case 1:
    System.out.println("Enter employee number,name and salary");
    no=Integer.parseInt(br.readLine());
    name=br.readLine();
    sal=Integer.parseInt(br.readLine());
    stmt.executeUpdate("insert into
    employee(eno,ename,salary)values("+no+",'"+name+"',"+sal+")");
    System.out.println("Records inserted");
    break;
case 2:
    System.out.print("Enter employe number of the record to be updated:");
    no=Integer.parseInt(br.readLine());
    System.out.println("Enter new name and salary");
    name=br.readLine();
    sal=Integer.parseInt(br.readLine());
    try
    {
    con.setAutoCommit(false);
    upc=stmt.executeUpdate("update employee set
    ename='"+name+"',salary="+sal+" where eno="+no);
    if(upc!=0)
    {
    con.commit();
    System.out.println("Records Updated");
    }
    else
    {
    System.out.println("No such record exit");
    break;
        }
    }
    catch(SQLException e)
    {
    System.out.println("Exception occured:"+e+"\nRecords not updated\n");
    con.rollback();
    }
    break;
case 3:
    System.out.print("enter employe number of the record you want to delete:");
    no=Integer.parseInt(br.readLine());
    try
    {
    con.setAutoCommit(false);
    upc=stmt.executeUpdate("delete from employee where eno="+no);

    if(upc!=0)
    {
    System.out.println("record deleted");
    con.commit();
    }
    else
    {
    System.out.println("NO such record exist");
    break;
    }
    }
    catch(SQLException e)
    {
    System.out.println("Exception occured:"+e+"\n Record not deleted\n");
    con.rollback();
    }
    break;
case 4:
    ResultSet rsl=stmt.executeQuery("select * from employee");
    System.out.println("ENO\tENAME\tSALARY\n");
    while(rsl.next())
    {

    System.out.println(rsl.getInt("eno")+"\t"+rsl.getString("ename")+"\t"+rsl.getInt
    ("salary"));
```

```
      }
      break;
case 5:
      System.exit(0);
default:
System.out.println("enter a valid choice");
}
}
while(ch!=5);
rs.close();
stmt.close();
con.close();
}
catch(SQLException e)
{
System.out.println("Connection failed:"+e.getMessage());
}
}
}
```

OUTPUT



```
.....Current Records....

ENO     ENAME   SALARY

101     ramu    40000
102     reena   70000
103     aleena  8000
104     Saranya 4000
105     Lekha   5000

MENU
1.Insert
2.Update
3.Delete
4.Display
5.Exit
Enter your choice:1
Enter employee number,name and salary
106
Latha
20000
Records inserted

MENU
1.Insert
2.Update
3.Delete
4.Display
5.Exit
Enter your choice:2
Enter employe number of the record to be updated:104
Enter new name and salary
Mekha
4000
Records Updated

MENU
1.Insert
2.Update
3.Delete
4.Display
5.Exit
Enter your choice:3
enter employe number of the record you want to delete:102
record deleted

MENU
1.Insert
```

STUDENT TABLE

AIM

JDBC program to connect to student tables. Implement record scrolling functions:

first(),next(),previous(),beforeFirst(),alertLast(),absolute() and relative().

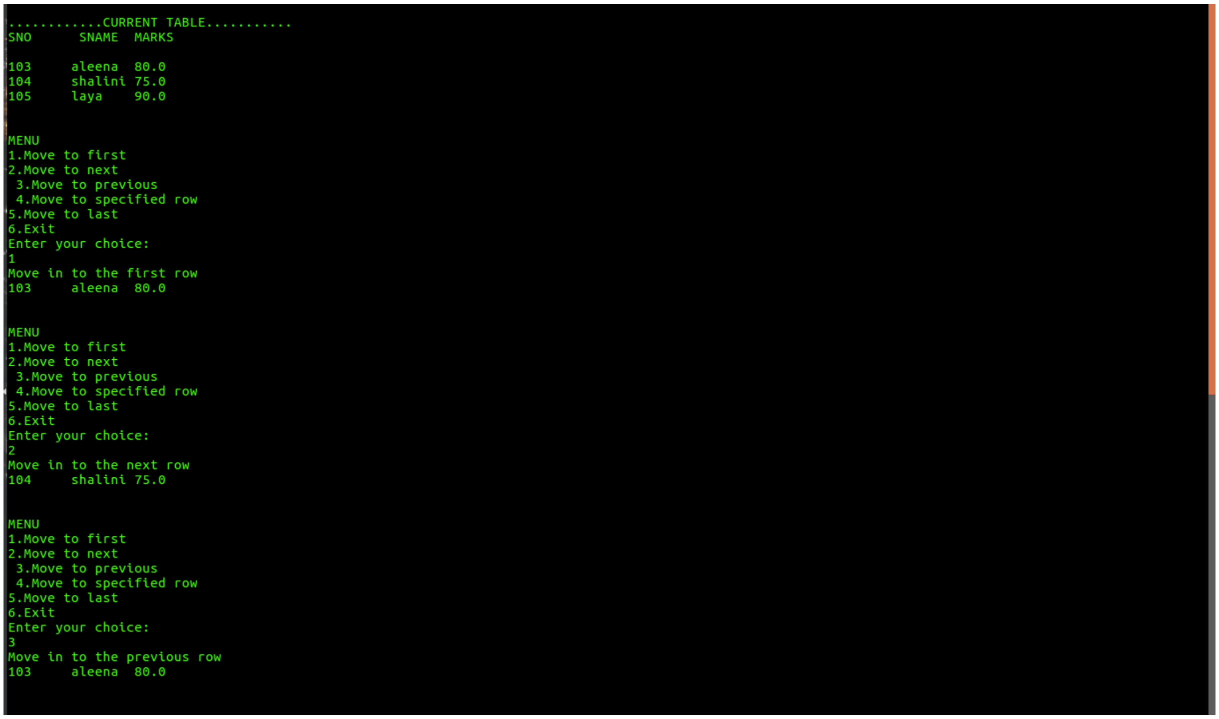
PROGRAM

```
import java.sql.*;
import java.io.*;
public class Student
{
public static void main(String []args)throws IOException
{
int ch,row,rows;
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
try
{
Class.forName("org.postgresql.Driver");
}
catch(ClassNotFoundException e)
{
System.out.println("Unable to load the driver");
}
try
{
Connection
con=DriverManager.getConnection("jdbc:postgresql://localhost/student","postgres
","con
cord");
Statement
stmt=con.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CO
NCUR _UPDATABLE);
ResultSet rs=stmt.executeQuery("select * from student");
System.out.println("\n...........CURRENT TABLE..........");
System.out.println("SNO\tSNAME\tMARKS\n");
```

```
while(rs.next())
{

System.out.println(rs.getInt("sno")+"\t"+rs.getString("sname")+"\t"+rs.getFloat("m
ark"));
}
do
{
System.out.println("\n\nMENU\n1.Move to first\n2.Move to next\n3.Move to
previous\n4.Move to specified row\n5.Move to last\n6.Exit\nEnter your choice:");
ch=Integer.parseInt(br.readLine());
switch(ch)
{
case 1:
    rs.first();
    System.out.println("Move in to the first row");
    System.out.println(rs.getInt("sno")+"\t"+rs.getString("sname")+"\t"+rs.getFloat
    ("mark"));
    break;
case 2:
    rs.next();
    System.out.println("Move in to the next row");
    System.out.println(rs.getInt("sno")+"\t"+rs.getString("sname")+"\t"+rs.getFloat
    ("mark"));
    break;
case 3:
    rs.previous();
    System.out.println("Move in to the previous row");
    System.out.println(rs.getInt("sno")+"\t"+rs.getString("sname")+"\t"+rs.getFloat
    ("mark"));
    break;
case 4:
    System.out.println("Enter the row number:");
    row=Integer.parseInt(br.readLine());
    rs.absolute(row);
    System.out.println("Move in to the specified row");
    System.out.println(rs.getInt("sno")+"\t"+rs.getString("sname")+"\t"+rs.getFloat
    ("mark"));
    break;
case 5:
    rs.last();
    System.out.println("Move to the last row");
    System.out.println(rs.getInt("sno")+"\t"+rs.getString("sname")+"\t"+rs.getFloat
    ("mark"));
    break;
case 6:
    System.out.println
    System.exit(0);
    }
    }
while(ch!=6);
rs.close();
stmt.close();
con.close();
}
catch(SQLException e)
{
System.out.print("Connection failed:"+e.getMessage());
e.printStackTrace(System.out);
}
}
}
```

OUTPUT



```
............CURRENT TABLE...........
SNO      SNAME  MARKS

103     aleena  80.0
104     shalini 75.0
105     laya    90.0


MENU
1.Move to first
2.Move to next
 3.Move to previous
 4.Move to specified row
5.Move to last
6.Exit
Enter your choice:
1
Move in to the first row
103     aleena  80.0


MENU
1.Move to first
2.Move to next
 3.Move to previous
 4.Move to specified row
5.Move to last
6.Exit
Enter your choice:
2
Move in to the next row
104     shalini 75.0


MENU
1.Move to first
2.Move to next
 3.Move to previous
 4.Move to specified row
5.Move to last
6.Exit
Enter your choice:
3
Move in to the previous row
103     aleena  80.0
```

DATABASE METADATA

AIM

JDBC program to display database metadata.

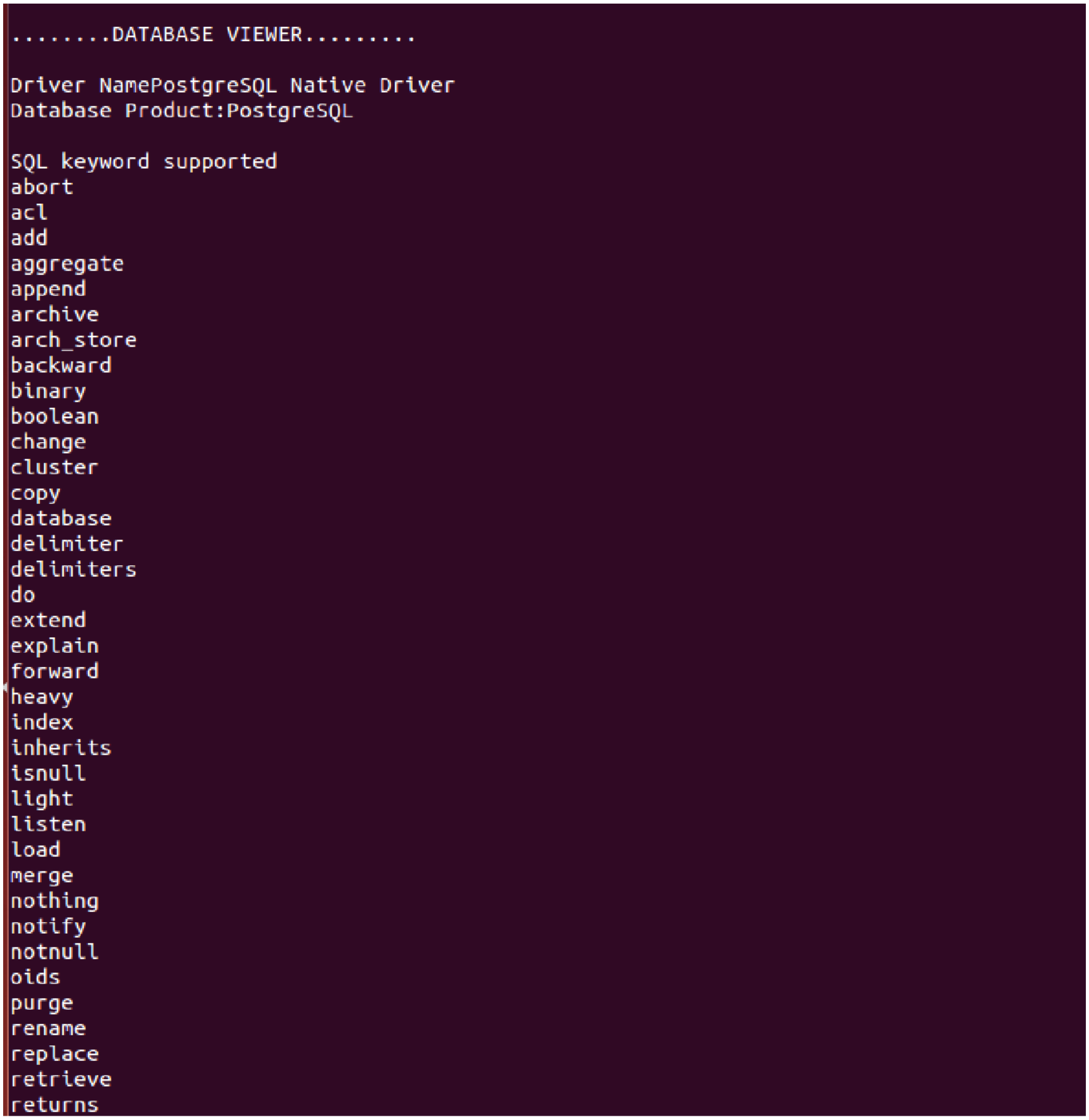
PROGRAM

```
import java.io.*;
import java.sql.*;
import java.util.StringTokenizer;
public class DBViewer
{

final static String jdbcURL="jdbc:postgresql://localhost/metadata";
final static String jdbcdriver="org.postgresql.Driver";
public static void main(String args[])
{
System.out.println("\n........DATABASE VIEWER.........\n");
try
{
Class.forName(jdbcdriver);
Connection con=DriverManager.getConnection(jdbcURL,"postgres","concord");
DatabaseMetaData dbmd=con.getMetaData();
System.out.println("Driver Name"+dbmd.getDriverName());
System.out.println("Database Product:"+dbmd.getDatabaseProductName());
System.out.println("\nSQL keyword supported");
StringTokenizer st=new StringTokenizer(dbmd.getSQLKeywords(),",");
while(st.hasMoreTokens())
System.out.println(""+st.nextToken());
String[] tableTypes={"TABLE"};
ResultSet rs=dbmd.getTables(null,null,null,tableTypes);
while(rs.next())
{
String table_name=rs.getString("TABLE_NAME");
System.out.println("Table Name:"+table_name);
System.out.println("tableType:"+rs.getString("TABLE_TYPE"));
System.out.println("indexes");
ResultSet ilist=dbmd.getIndexInfo(null,null,table_name,false,false);
while(ilist.next())
{
System.out.println("index name:"+ilist.getString("INDEX_NAME"));
System.out.println("Coloumn Name:"+ilist.getString("COLUMN_NAME"));
}
ilist.close();
}
rs.close();
con.close();
}
catch(ClassNotFoundException e)

{
System.out.println("Unable to load driver");
}
catch(SQLException e)
{
System.out.println("SQL Exception:"+e.getMessage());
}
catch(Exception e)
{
System.out.println("Exception:"+e);
}
}
}
```

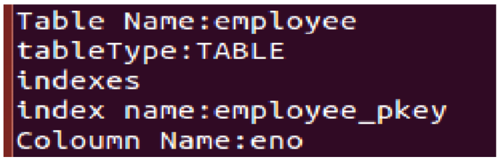
OUTPUT



```
........DATABASE VIEWER.........

Driver NamePostgreSQL Native Driver
Database Product:PostgreSQL

SQL keyword supported
abort
acl
add
aggregate
append
archive
arch_store
backward
binary
boolean
change
cluster
copy
database
delimiter
delimiters
do
extend
explain
forward
heavy
index
inherits
isnull
light
listen
load
merge
nothing
notify
notnull
oids
purge
rename
replace
retrieve
returns
```



```
Table Name:employee
tableType:TABLE
indexes
index name:employee_pkey
Coloumn Name:eno
```

RESULTSET METADATA

AIM

JDBC program to display resultset metadata.

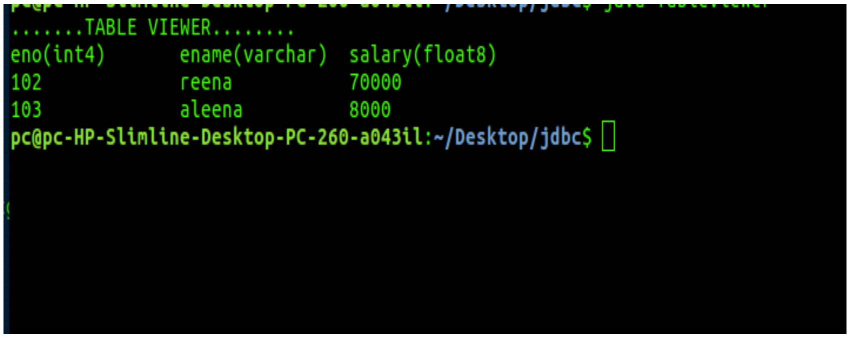
PROGRAM

```
import java.sql.*;
import java.util.StringTokenizer;
public class TableViewer
{

final static String jdbcURL="jdbc:postgresql://localhost/result";
final static String jdbcDriver="org.postgresql.Driver";
public static void main(String []args)
{
System.out.println(".......TABLE VIEWER........");
try
{
Class.forName(jdbcDriver);
Connection con=DriverManager.getConnection(jdbcURL,"postgres","concord");
Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery("select * from employ");
ResultSetMetaData rsmd=rs.getMetaData();
int columncount=rsmd.getColumnCount();
for(int col=1;col<=columncount;col++)
{
System.out.print(rsmd.getColumnLabel(col));
System.out.print("("+rsmd.getColumnTypeName(col)+")");
if(col<columncount)
System.out.print("\t");
}
System.out.println();
while(rs.next())
{
for(int col=1;col<=columncount;col++)
{
```

```
System.out.print(rs.getString(col));
if(col<columncount)
System.out.print("\t\t");
}
System.out.println();
}
rs.close();
stmt.close();
con.close();
}
catch(ClassNotFoundException e)
{
System.out.println("Unable to load database driver class ");
}
catch(SQLException e)
{
System.out.println("SQL Exception"+e.getMessage());
}
}
}
```

OUTPUT



```
pc@pc-HP-Slimline-Desktop-PC-260-a043il:~/Desktop/jdbc$ java TableViewer
.......TABLE VIEWER........
eno(int4)       ename(varchar)  salary(float8)
102             reena           70000
103             aleena          8000
pc@pc-HP-Slimline-Desktop-PC-260-a043il:~/Desktop/jdbc$ 
```

RMI

COMPLEX NUMBER OPERATION

AIM

RMI program to perform complex number operations.

PROGRAM

File: Complex1.java

```
import java.io.Serializable;
public class Complex1 implements Serializable
{
      public int r,i;
      public Complex1()
      {
         r=0;
         i=0;
      }

      public Complex1(int x,int y)
      {
         r=x;
         i=y;
      }
}
```

File: ComplexInter.java

```
import java.rmi.Remote;
import java.rmi.RemoteException;
public interface ComplexInter extends java.rmi.Remote
{
public Complex1 add(Complex1 x,Complex1 y)throws RemoteException;
public Complex1 sub(Complex1 x,Complex1 y)throws RemoteException;
public Complex1 mul(Complex1 x,Complex1 y)throws RemoteException;
}
```

File: ComplexImpl.java

```
import java.rmi.server.UnicastRemoteObject;
import java.rmi.RemoteException;
public class ComplexImpl extends UnicastRemoteObject implements
ComplexInter
{
public Complex1 c;
public ComplexImpl(int a,int b) throws RemoteException
{
c=new Complex1(a,b);
}
public ComplexImpl()throws RemoteException{}
public Complex1 add(Complex1 x,Complex1 y)throws RemoteException
{
Complex1 z=new Complex1();
z.r=x.r+y.r;
z.i=x.i+y.i;
return z;
}
public Complex1 sub(Complex1 x,Complex1 y)throws RemoteException
{
Complex1 z=new Complex1();
z.r=x.r-y.r;
z.i=x.i-y.i;
return z;
}
public Complex1 mul(Complex1 x,Complex1 y)throws RemoteException
{
Complex1 z=new Complex1();
z.r=((x.r)*(y.r))-((x.i)*(y.i));
z.i=((x.r)*(y.i))+((y.r)*(x.i));
return z;
}
}
```

File: ComplexReg.java

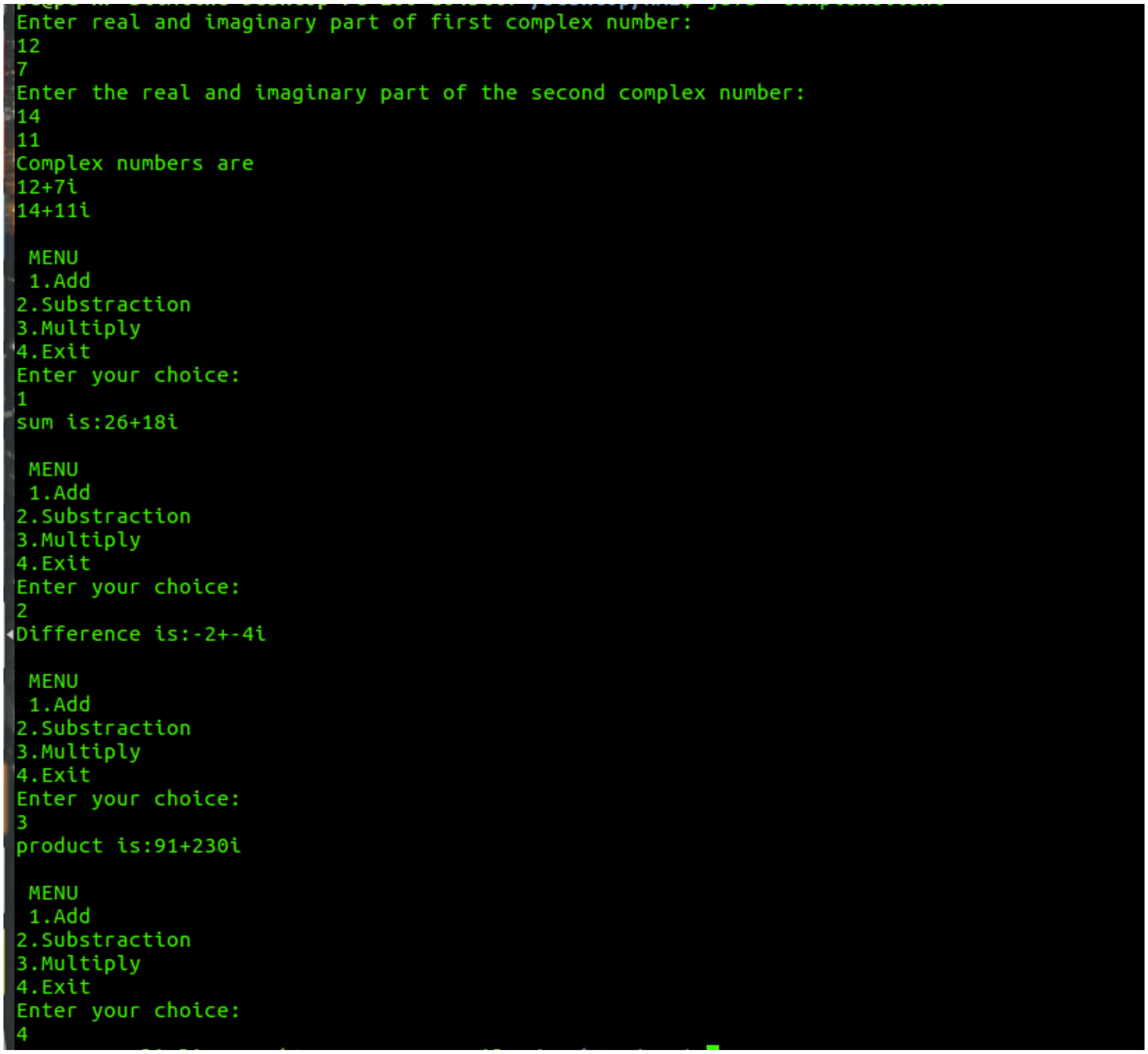
```
import java.rmi.Naming;
```

```
public class ComplexReg
{
public static void main(String args[])
{
try
{
ComplexImpl comp=new ComplexImpl();
Naming.rebind("com",comp);
System.out.println("Object Registered");
}
catch(Exception e)
{
e.printStackTrace();
}
}
}
```

File: ComplexClient.java

```
import java.rmi.Naming;
import java.io.*;
public class ComplexClient
{
public static void main(String args[])throws IOException
{
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
int x1,y1,x2,y2;
int ch;
System.out.println("Enter real and imaginary part of first complex number:");
x1=Integer.parseInt(br.readLine());
y1=Integer.parseInt(br.readLine());
System.out.println("Enter the real and imaginary part of the second complex
number:");
x2=Integer.parseInt(br.readLine());
y2=Integer.parseInt(br.readLine());
Complex1 a1=new Complex1(x1,y1);
Complex1 a2=new Complex1(x2,y2);
Complex1 a3;
try
{
ComplexInter obj=(ComplexInter)Naming.lookup("rmi://localhost/com");
System.out.println("Complex numbers are");
System.out.println(a1.r+"+"+a1.i+"i");
System.out.println(a2.r+"+"+a2.i+"i");
do
{
System.out.println("\n MENU\n 1.Add\n2.Substraction\n3.Multiply\n4.Exit\nEnter
your choice:");
ch=Integer.parseInt(br.readLine());
switch(ch)
{
case 1:
    a3=obj.add(a1,a2);
    System.out.println("sum is:"+a3.r+"+"+a3.i+"i");
    break;
case 2:
    a3=obj.sub(a1,a2);
    System.out.println("Difference is:"+a3.r+"+"+a3.i+"i");
    break;
case 3:
    a3=obj.mul(a1,a2);
    System.out.println("product is:"+a3.r+"+"+a3.i+"i");
    break;
case 4:
    System.exit(0);
default:
    System.out.println("Enter a valid choice");
}
}
while(ch!=4);
}
catch(Exception e)
{
System.out.println("Error:"+e);
}
}
}
```

OUTPUT



```
Enter real and imaginary part of first complex number:
12
7
Enter the real and imaginary part of the second complex number:
14
11
Complex numbers are
12+7i
14+11i

 MENU
 1.Add
2.Substraction
3.Multiply
4.Exit
Enter your choice:
1
sum is:26+18i

 MENU
 1.Add
2.Substraction
3.Multiply
4.Exit
Enter your choice:
2
Difference is:-2+-4i

 MENU
 1.Add
2.Substraction
3.Multiply
4.Exit
Enter your choice:
3
product is:91+230i

 MENU
 1.Add
2.Substraction
3.Multiply
4.Exit
Enter your choice:
4
```

BANK OPERATION

AIM

RMI program to perform bank operations.

PROGRAM

File:Account.java

```
import java.rmi.Remote;
import java.rmi.RemoteException;
public interface Account extends java.rmi.Remote
{
public String getName()throws RemoteException;
public float getBal()throws RemoteException;
public float withdraw(float amt)throws RemoteException;
public float deposit(float amt)throws RemoteException;
}
```

File:AccountImpl.java

```
import java.rmi.server.UnicastRemoteObject;
import java.rmi.RemoteException;
public class AccountImpl extends UnicastRemoteObject implements Account
{
private float mbal;
private String mname="";
public AccountImpl(String name)throws RemoteException
{
mname=name;
mbal=1000;
}
public String getName() throws RemoteException
{
return mname;
}
public float getBal()throws RemoteException

{
return mbal;
}
public float withdraw(float amt) throws RemoteException
{
if(mbal-amt>=1000)
{
mbal=mbal-amt;
return 0;
}
else
{
return -1;
}
}
public float deposit(float amt)throws RemoteException
{
mbal=mbal+amt;
return mbal;
}
}
```

File:AccountReg.java

```
import java.rmi.Naming;
public class AccountReg
{
public static void main(String args[])
{
try
{
AccountImpl ob1=new AccountImpl("jack");
Naming.rebind("jack",ob1);
System.out.println("Registered Account");
}
catch(Exception e)
{
e.printStackTrace();
}
}
}
```

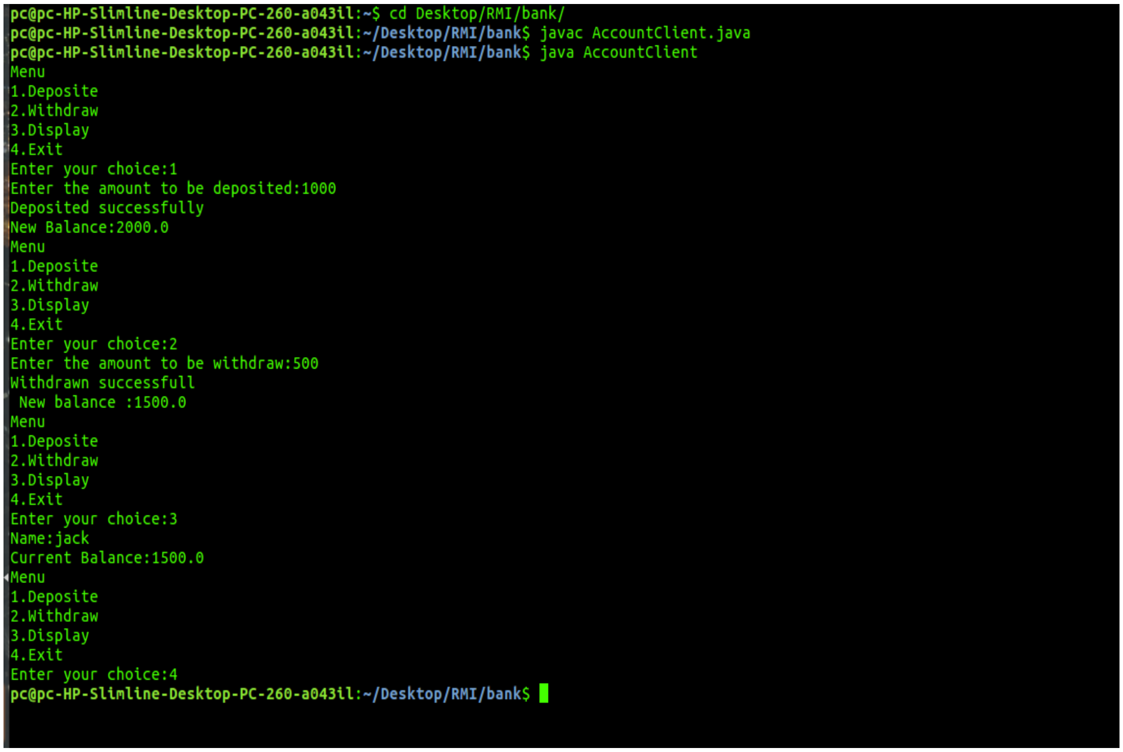
File:AccountClient.java

```
import java.io.*;
import java.rmi.Naming;
public class AccountClient
{
public static void main(String args[]) throws IOException
{
String name;
int ch;
float amount,bal,bal1;
try
{
Account obj=(Account)Naming.lookup("rmi://localhost/jack");
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
do
{

System.out.print("Menu\n1.Deposite\n2.Withdraw\n3.Display\n4.Exit\nEnter your
choice:");
ch=Integer.parseInt(br.readLine());
switch(ch)
{
case 1:
        System.out.print("Enter the amount to be deposited:");
        amount=Float.parseFloat(br.readLine());
        bal=obj.deposit(amount);
        System.out.println("Deposited successfully");
        System.out.println("New Balance:"+bal);
        break;
case 2:
        System.out.print("Enter the amount to be withdraw:");
```

```
         amount=Float.parseFloat(br.readLine());
         bal=obj.withdraw(amount);
         if(bal!=-1)
         {
            bal1=obj.getBal();
            System.out.println("Withdrawn successfull\n New balance :"+bal1);
         }
        else
        System.out.println("No sufficient balance");
        break;
case 3:
        System.out.println("Name:"+obj.getName());
        System.out.println("Current Balance:"+obj.getBal());
        break;
case 4:
       System.exit(0);
       default:System.out.println("Enter a valid choice");
}
}
while(ch!=4);
}
catch(Exception e)
{
System.out.print("Error:"+e);
}
}
}
```

OUTPUT



```
pc@pc-HP-Slimline-Desktop-PC-260-a043il:~$ cd Desktop/RMI/bank/
pc@pc-HP-Slimline-Desktop-PC-260-a043il:~/Desktop/RMI/bank$ javac AccountClient.java
pc@pc-HP-Slimline-Desktop-PC-260-a043il:~/Desktop/RMI/bank$ java AccountClient
Menu
1.Deposite
2.Withdraw
3.Display
4.Exit
Enter your choice:1
Enter the amount to be deposited:1000
Deposited successfully
New Balance:2000.0
Menu
1.Deposite
2.Withdraw
3.Display
4.Exit
Enter your choice:2
Enter the amount to be withdraw:500
Withdrawn successfull
 New balance :1500.0
Menu
1.Deposite
2.Withdraw
3.Display
4.Exit
Enter your choice:3
Name:jack
Current Balance:1500.0
Menu
1.Deposite
2.Withdraw
3.Display
4.Exit
Enter your choice:4
pc@pc-HP-Slimline-Desktop-PC-260-a043il:~/Desktop/RMI/bank$
```

SERVLET

STUDENT DETAILS

AIM

Create an html form to read student details such as rollno, name, age, sex, qualification and percentage of mark .Write a servlet program that display the same details.

PROGRAM

File: StudentServlet.java

```
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
public class StudentServlet  extends HttpServlet
{
  public void doGet(HttpServletRequest req,HttpServletResponse resp)
  throws ServletException,IOException
 {
     resp.setContentType("text/html");
     PrintWriter out=resp.getWriter();
     out.println("<html>");
     out.println("<head><title>Student Details</title></head>");
     out.println("<body><h3>ROLL NUMBER : " +
req.getParameter("rno")+"</h3>");
     out.println("<h3>NAME:"+req.getParameter("name")+"</h3>");
     out.println("<h3>AGE:"+req.getParameter("age")+"</h3>");
     out.println("<h3>GENDER:"+req.getParameter("gen")+"</h3>");
     out.println("<h3>QUALIFICATION:"+req.getParameter("ql")+"</h3>");
     out.println("<h3>PERCENTAGE:"+req.getParameter("mark")+"</h3>");
     out.println("</body></html>");
  }
}
```

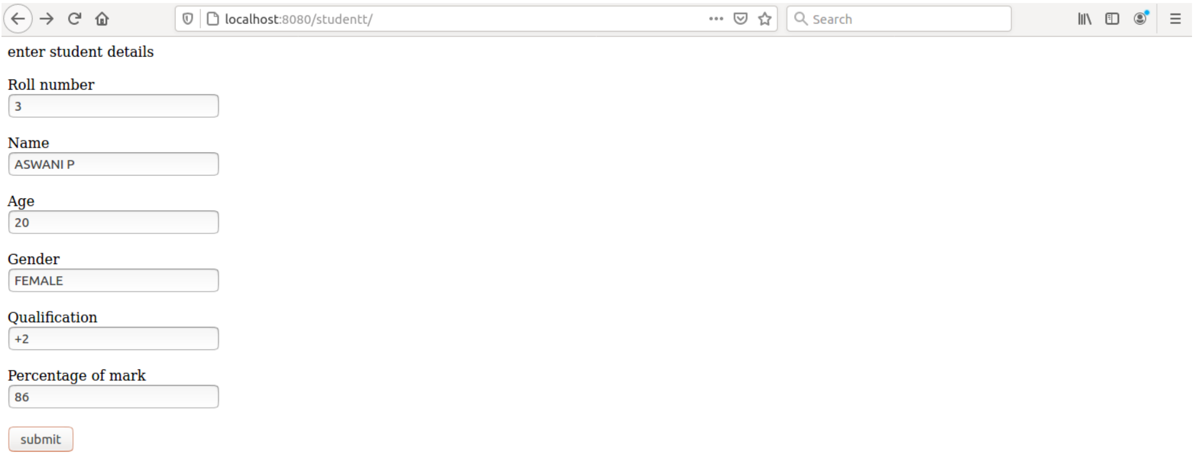
File: index.html

```
<html>
<head>
<title>STUDENT</title>
</head>
<body>
<form method="GET" action="stu">enter student details<br>
<br>Roll number<br><input type="text" name="rno" size="20"><br>
<br> Name<br><input type="text" name="name" size="20"><br>
<br> Age<br><input type="text" name="age" size="20"><br>
<br> Gender<br><input type="text" name="gen" size="20"><br>
<br> Qualification<br><input type="text" name="ql" size="20"><br>
<br> Percentage of mark<br><input type="text" name="mark"
size="20"><br><br>
 <input type=submit value="submit">
</form>
</body>
</html>
```

File:web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app>
    <servlet>
        <servlet-name>StudentServlet</servlet-name>
        <servlet-class>StudentServlet</servlet-class>
    </servlet>
 <servlet-mapping>
        <servlet-name>StudentServlet</servlet-name>
        <url-pattern>/stu</url-pattern>
    </servlet-mapping>
</web-app>
```

OUTPUT



enter student details

Roll number
3

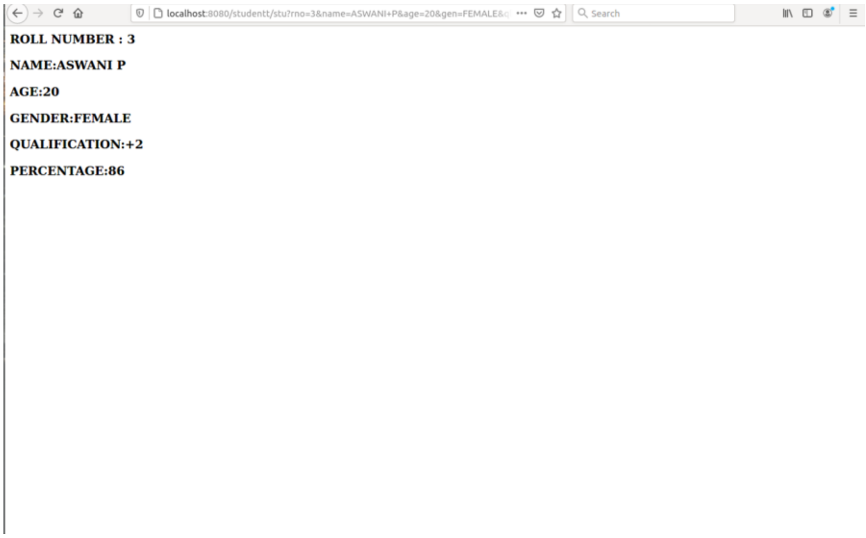
Name
ASWANI P

Age
20

Gender
FEMALE

Qualification
+2

Percentage of mark
86


localhost:8080/studentt/stu?rno=3&name=ASWANI+P&age=20&gen=FEMALE&q
Search

ROLL NUMBER : 3

NAME:ASWANI P

AGE:20

GENDER:FEMALE

QUALIFICATION:+2

PERCENTAGE:86

SESSION HANDLING

AIM

Session handling servlet that displays total number of visits to that page.

PROGRAM

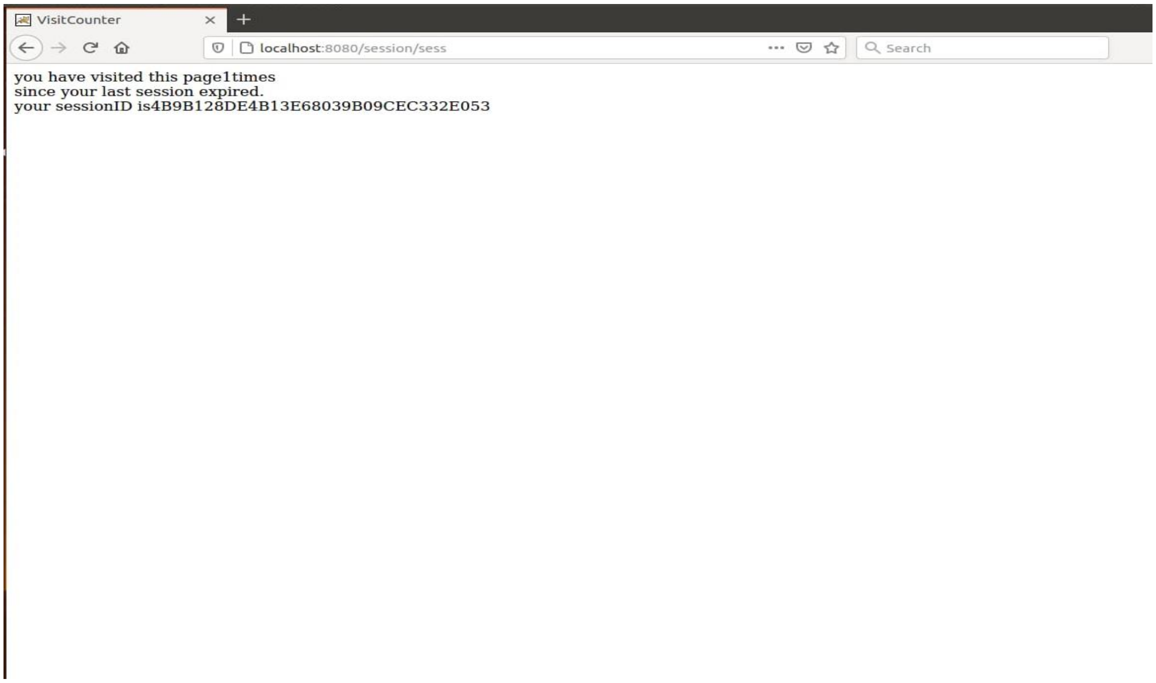
File:Session.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class session extends HttpServlet
{
public void doGet(HttpServletRequest req,HttpServletResponse resp)throws
ServletException,IOException
{
 resp.setContentType("text/html");
PrintWriter out=resp.getWriter();
HttpSession thisUser=req.getSession(true);
Integer visits;
if(!thisUser.isNew())
{
visits=(Integer)thisUser.getAttribute("VisitCounter.visits");
if(visits==null)
visits=new Integer(1);
else
visits=new Integer(visits.intValue()+1);
}
Else
{
visits=new Integer(1);
thisUser.setAttribute("VisitCounter.visits",visits);
out.println("<html><head><title>VisitCounter</title></head>");
out.println("<body>you have visited this page"+visits+"times");
out.println("<br>since your last session expired.");
out.println("<br>your sessionID is"+thisUser.getId());
out.println("</body></html>");
}
}
```

File: web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app>
    <servlet>
        <servlet-name>Session</servlet-name>
        <servlet-class>Session</servlet-class>
    </servlet>
<servlet-mapping>
    <servlet-name>Session</servlet-name>
    <url-pattern>/sess</url-pattern>
</servlet-mapping>
</web-app>
```

OUTPUT



```
you have visited this page1times
since your last session expired.
your sessionID is4B9B128DE4B13E68039B09CEC332E053
```

FILE SERVLET

AIM

Create an HTML form that read a name from the user. Write a servlet program that displays the contents of the file specified by the user.

PROGRAM

```
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
public class FileServlet extends HttpServlet
{
     public void doPost(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException
   {
        File r;
        FileReader fr;
        BufferedReader br;
            try
           {
                r = new File(req.getParameter("filename"));
        fr = new FileReader(r);
                br = new BufferedReader(fr);
        if (!r.isFile())
              {
                 resp.sendError(resp.SC_NOT_FOUND);
                 return;
              }
            }
            catch (FileNotFoundException e)
           {
               resp.sendError(resp.SC_NOT_FOUND);
            return;
           }
           catch (SecurityException se)
          {
              throw (new UnavailableException(this, "Servlet lacks appropriate
        priviledge"));
          }
           resp.setContentType("text/html");
           PrintWriter out = resp.getWriter();
           String text;
           while ((text = br.readLine()) != null)
           out.println(text);
           br.close();
        }
}
```

File: index.html

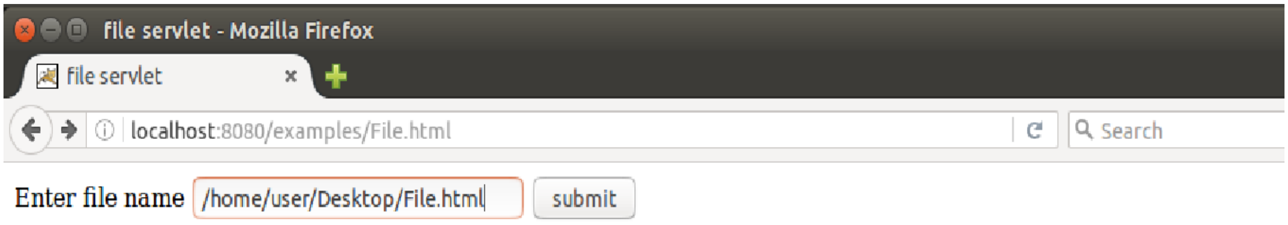
```
<html>
<head>
<title>File Servlet</title>
</head>
<body>
<form method="POST" action="http://localhost:8080/file/fileservlet">Enter
file name
<input type="text" name="filename">
<input type="submit" value="submit">
</form>
</body>
</html>
```

File: web.xml

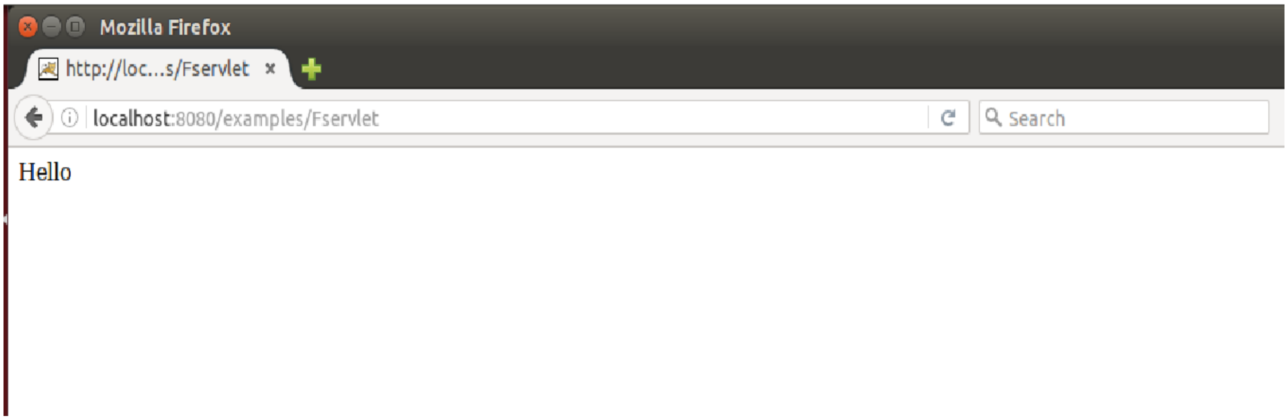
```
<?xml version="1.0" encoding="UTF-8"?>
<web-app>
```

```
        <servlet>
                <servlet-name>FileServlet</servlet-name>
                <servlet-class>FileServlet</servlet-class>
        </servlet>
        <servlet-mapping>
                <servlet-name>FileServlet</servlet-name>
                <url-pattern>/fileservlet</url-pattern>
        </servlet-mapping>
</web-app>
```

OUTPUT


File servlet - Mozilla Firefox
File servlet
localhost:8080/examples/File.html
Search
submit

Enter file name /home/user/Desktop/File.html



Hello

CORBA

ARITHMETIC OPERATION

AIM

CORBA program for arithmetic operations.

PROGRAM

File:Arith.idl

```
module ArithApp
{
        interface Arith
        {
                float sum(in float a,in float b);
                float sub(in float a,in float b);
                float mul(in float a,in float b);
                float div(in float a,in float b);
        };
};
```

File: ArithImpl.java

```
import ArithApp.*;
import org.omg.CORBA.*;
import org.omg.CosNaming.*;
import org.omg.PortableServer.POA;
public class ArithImpl extends ArithPOA
{
 public ArithImpl()
 {
 }
public float sum(float a, float b)
```

```
 {
  float c = a + b;
  return c;
 }
public float sub(float a, float b)
 {
  float c = a - b;
  return c;
 }
 public float mul(float a, float b)
 {
  float c = a * b;
  return c;
 }
 public float div(float a, float b)
 {
  float c = a / b;
  return c;
 }
}
```

File: ArithInitPOA.java

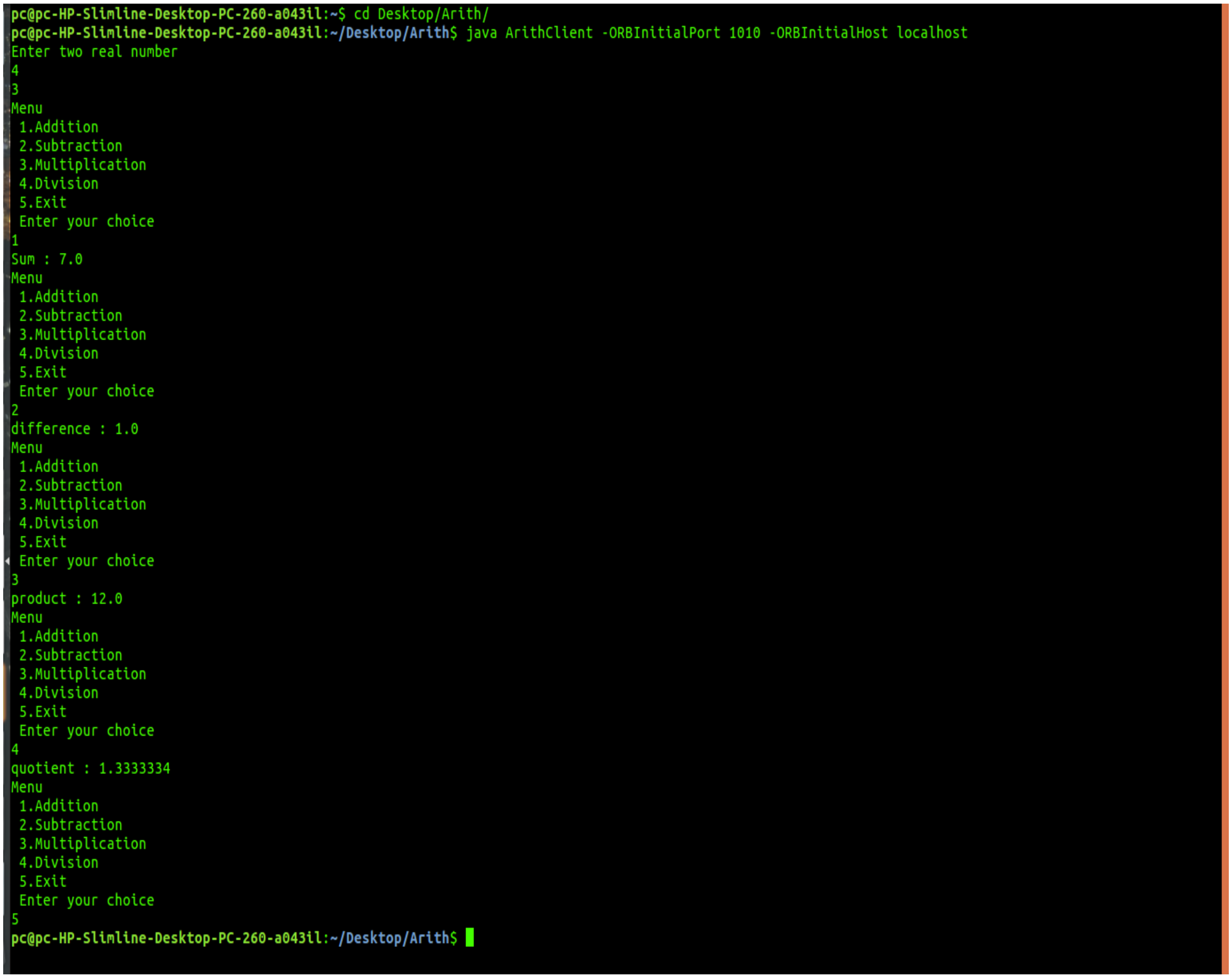
```
import ArithApp.*;
import org.omg.CORBA.*;
import org.omg.CosNaming.*;
import org.omg.PortableServer.POA;
public class ArithInitPOA
{
 public static void main(String[] args)
 {
  try
  {
   ORB myOrb = ORB.init(args, null);
   ArithImpl ob = new ArithImpl();
 POA rootPOA = (POA) myOrb.resolve_initial_references("RootPOA");
 rootPOA.the_POAManager().activate();
 org.omg.CORBA.Object obj = rootPOA.servant_to_reference(ob);
 Arith acct = (Arith) ArithHelper.narrow(obj);
 org.omg.CORBA.Object objref =
myOrb.resolve_initial_references("NameService");
 NamingContextExt nc = NamingContextExtHelper.narrow(objref);
 NameComponent[] name = nc.to_name(args[0]);
 nc.rebind(name, acct);
 System.out.println("registerd obj under name" + args[0]);
 myOrb.run();
 }
 catch (Exception e)
{
   System.out.println("Error");
   e.printStackTrace();
}
}
}
```

File:ArithClient.java

```
import ArithApp.*;
import org.omg.CORBA.*;
import org.omg.CORBA.ORBPackage.*;
import org.omg.CosNaming.*;
import java.util.*;
import java.io.*;
public class ArithClient
{
 public static void main(String[] args) throws IOException
  {
  ORB orb = ORB.init(args, null);
  org.omg.CORBA.Object ref = null;
  Arith obj;
```

```
BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
  try
  {
   ref = orb.resolve_initial_references("NameService");
  }
catch (InvalidName e)
  {
   System.out.println("could not locate name");
   System.exit(0);
  }
  NamingContext nc = NamingContextHelper.narrow(ref);
  NameComponent comp = new NameComponent(args[0], "");
  NameComponent[] path = {
   comp
  };
  try
  {
   ref = nc.resolve(path);
   obj = ArithHelper.narrow(ref);
   int ch;
   float n1, n2, res;
   System.out.println("Enter two real number");
   n1 = Float.parseFloat(br.readLine());
   n2 = Float.parseFloat(br.readLine());
   do
   {
    System.out.println("Menu \n 1.Addition \n 2.Subtraction \n
3.Multiplication \n 4.Division \n 5.Exit \n Enter your choice ");
     ch = Integer.parseInt(br.readLine());
     switch (ch)
     {
      case 1:
           res = obj.sum(n1, n2);
           System.out.println("Sum : " + res);
           break;
     case 2:
          res = obj.sub(n1, n2);
          System.out.println("difference : " + res);
          break;
     case 3:
          res = obj.mul(n1, n2);
          System.out.println("product : " + res);
          break;
     case 4:
          res = obj.div(n1, n2);
          System.out.println("quotient : " + res);
          break;
     case 5:
         System.exit(0);
     default:
         System.out.println("enter an invalid choice\n");
    }
   }
   while (ch != 5);
  }
  catch (Exception e)
  {
   System.out.println("error");
   e.printStackTrace();
  }
 }
}
```

OUTPUT



```
pc@pc-HP-Slimline-Desktop-PC-260-a043il:~$ cd Desktop/Arith/
pc@pc-HP-Slimline-Desktop-PC-260-a043il:~/Desktop/Arith$ java ArithClient -ORBInitialPort 1010 -ORBInitialHost localhost
Enter two real number
4
3
Menu
 1.Addition
 2.Subtraction
 3.Multiplication
 4.Division
 5.Exit
 Enter your choice
1
Sum : 7.0
Menu
 1.Addition
 2.Subtraction
 3.Multiplication
 4.Division
 5.Exit
 Enter your choice
2
difference : 1.0
Menu
 1.Addition
 2.Subtraction
 3.Multiplication
 4.Division
 5.Exit
 Enter your choice
3
product : 12.0
Menu
 1.Addition
 2.Subtraction
 3.Multiplication
 4.Division
 5.Exit
 Enter your choice
4
quotient : 1.3333334
Menu
 1.Addition
 2.Subtraction
 3.Multiplication
 4.Division
 5.Exit
 Enter your choice
5
pc@pc-HP-Slimline-Desktop-PC-260-a043il:~/Desktop/Arith$
```