1. **/* Write a program to present a set of choice for user to select stationary products and display the price of products after selection from the list. */**

```java
import java.awt.*;

import javax.swing.*;

import javax.swing.event.*;

import java.awt.event.*;

public class Stationary extends JFrame implements ListSelectionListener

{

        JLabel jl1;

        JList list;

        JTextField jtf;

        String name[]={"BOOKS","PEN","PENCIL","SHARPENER","SCALE"};

        public Stationary()

        {

                Container cp=getContentPane();

                cp.setLayout(new BorderLayout());

                jl1=new JLabel("Select a product");

                cp.add(jl1,BorderLayout.NORTH);

                list=new JList(name);

                list.addListSelectionListener(this);

                cp.add(list,BorderLayout.CENTER);

                jtf=new JTextField(15);

                cp.add(jtf,BorderLayout.SOUTH);     }

                public void valueChanged(ListSelectionEvent le)

                {
```
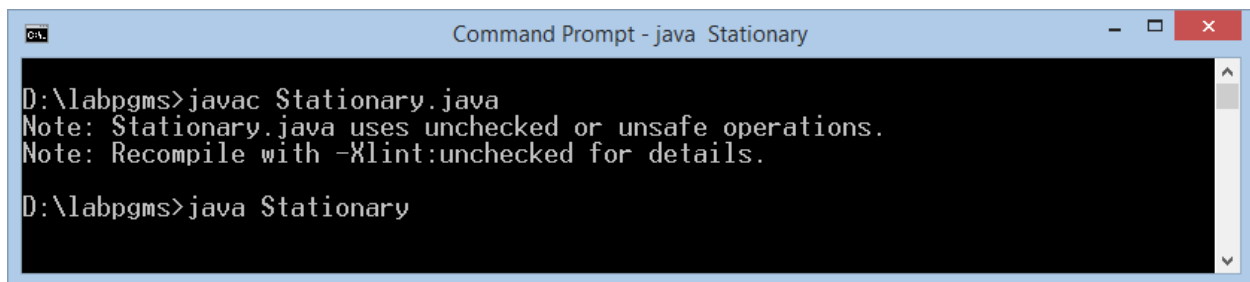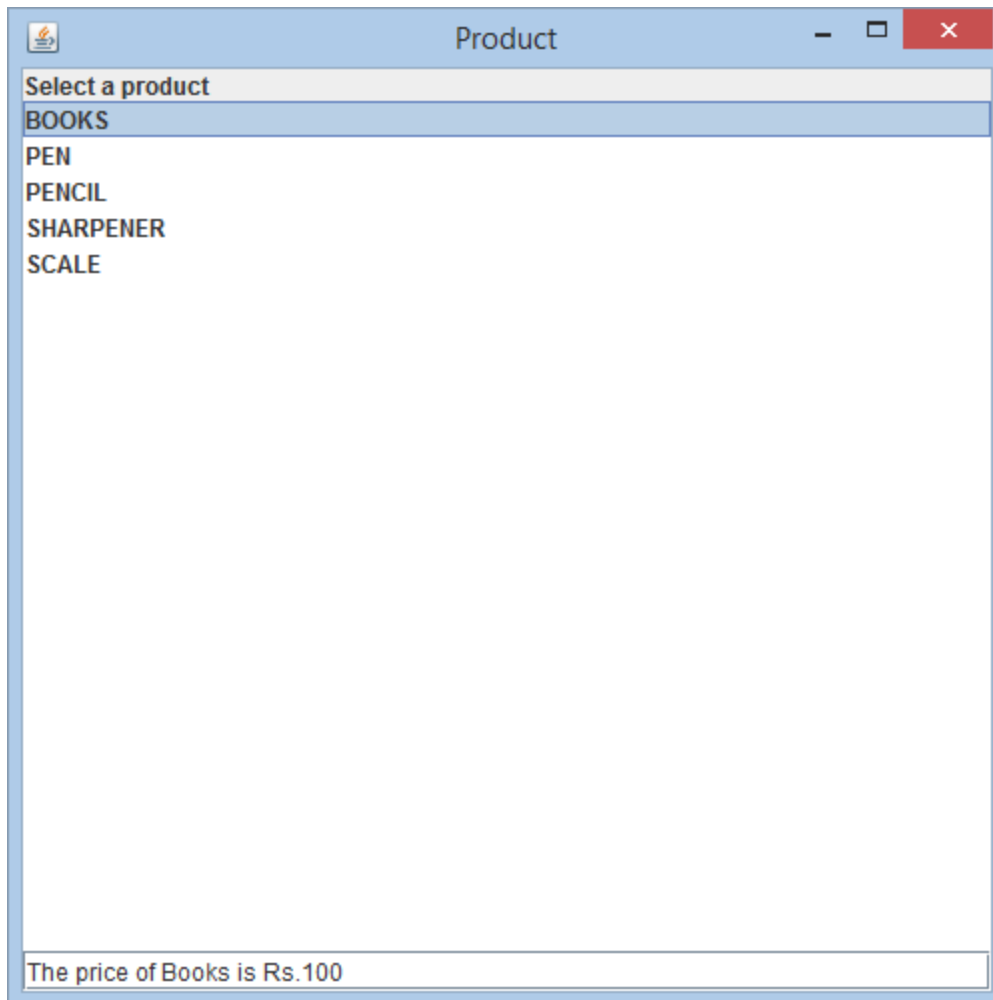
```java
int i=list.getSelectedIndex();
if(i==0)
{
 jtf.setText("The price of Books is Rs.100");
}
else if(i==1)
{
jtf.setText("The price of pen is Rs.10");
}
else if(i==2)
{
jtf.setText("The price of pencil is Rs.5");
}
else if(i==3)
{
jtf.setText("The price of sharpener is Rs.3");
}
else if(i==4)
{
	jtf.setText("The price of scale is Rs.5");
}
else
{
	jtf.setText("Select Item");
```

```java
                }
        }
        public static void main(String args[])
        {
                Stationary p1=new Stationary();

                p1.setSize(500,500);

                p1.setVisible(true);

                p1.setTitle("Product");

                p1.setDefaultCloseOperation(EXIT_ON_CLOSE);
        }
}
```

```
Command Prompt - java  Stationary

D:\labpgms>javac Stationary.java
Note: Stationary.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

D:\labpgms>java Stationary
```

2./* Write a java program to demonstrate typical Editable Table, describing employee details for a software company. */

```java
import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
public class ETable extends JApplet
{
    JTable jtb;
    JScrollPane jsp;
    public void init()
    {
        Container c=getContentPane();
        c.setLayout(new BorderLayout());
        String colHeads[]={"Sr.no","Name","Phone No"};
        Object data[][]={{"1","abc","1242642"},
                {"2","def","572656"},
                {"3","ghi","5364535"},
                {"4","jkl","955345"},
                {"5","mno","231212"},
                {"6","pqr","758278"},
                {"7","stu","8532034"},
                {"8","vwx","587255"},
                {"9","yza","899534"},
                {"10","bcd","9259554"}};
        jtb=new JTable(data,colHeads);
```

```
        jsp=new JScrollPane(jtb);

        c.add(jsp);

    }

}
```

/*<applet code="ETable.class" width=300 height=300>

</applet>*/

3./ / Write a java program to create form using swing.

import java.awt.*;

import javax.swing.*;

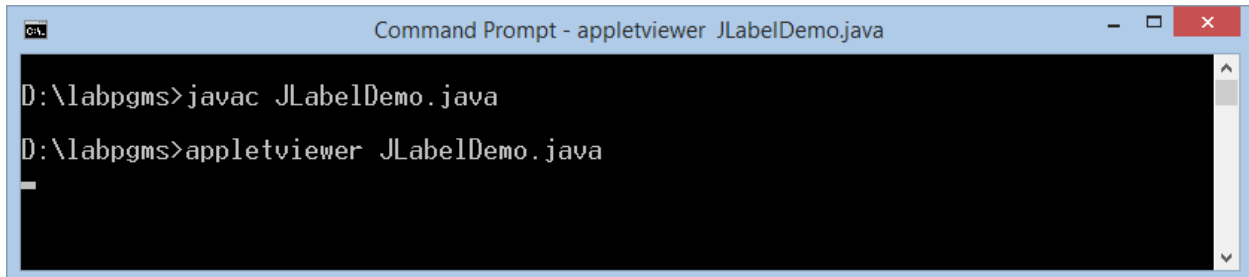public class JLabelDemo extends JApplet

{

        public void init()

```java
{
    Container c=getContentPane();
    c.setLayout(null);
    JLabel L1=new JLabel("REGISTRATION FORM");
    L1.setBounds(50,10,140,20);
    c.add(L1);
    JLabel L2=new JLabel("UserName:");
    L2.setBounds(10,40,140,20);
    c.add(L2);
    JTextField T1=new JTextField(20);
    T1.setBounds(90,40,140,20);
    c.add(T1);
    JLabel L3=new JLabel("Password:");
    L3.setBounds(10,70,140,20);
    c.add(L3);
    JTextField T2=new JTextField(20);
    T2.setBounds(90,70,140,20);
    c.add(T2);
    JLabel L4=new JLabel("CPassword:");
    L4.setBounds(10,100,140,20);
    c.add(L4);
    JTextField T3=new JTextField(20);
    T3.setBounds(90,100,140,20);
    c.add(T3);
```
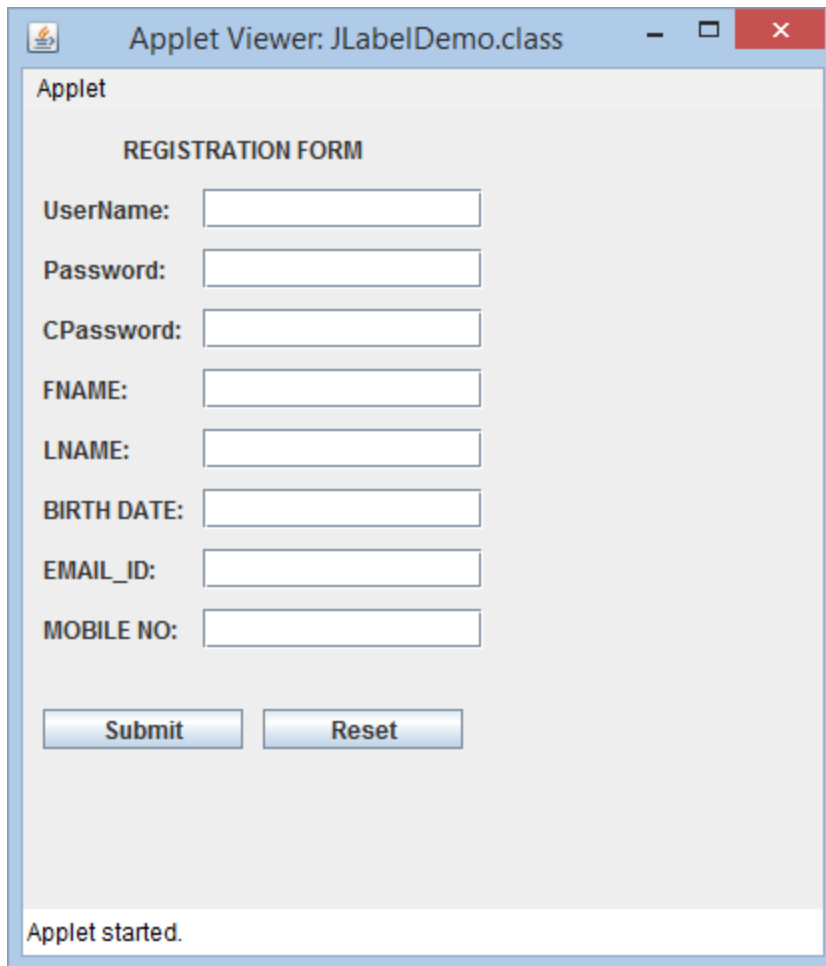
```java
JLabel L5=new JLabel("FNAME:");

L5.setBounds(10,130,140,20);

c.add(L5);

JTextField T4=new JTextField(20);

T4.setBounds(90,130,140,20);

c.add(T4);

JLabel L6=new JLabel("LNAME:");

L6.setBounds(10,160,140,20);

c.add(L6);

JTextField T5=new JTextField(20);

T5.setBounds(90,160,140,20);

c.add(T5);

JLabel L7=new JLabel("BIRTH DATE:");

L7.setBounds(10,190,140,20);

c.add(L7);

JTextField T6=new JTextField(20);

T6.setBounds(90,190,140,20);

c.add(T6);

JLabel L8=new JLabel("EMAIL_ID:");

L8.setBounds(10,220,140,20);

c.add(L8);

JTextField T7=new JTextField(20);

T7.setBounds(90,220,140,20);

c.add(T7);
```

```java
JLabel L9=new JLabel("MOBILE NO:");

L9.setBounds(10,250,140,20);

c.add(L9);

JTextField T8=new JTextField(20);

T8.setBounds(90,250,140,20);

c.add(T8);

JButton jb=new JButton("Submit");

jb.setBounds(10,300,100,20);

c.add(jb);

JButton jb1=new JButton("Reset");

jb1.setBounds(120,300,100,20);

c.add(jb1);   }  }
```

/*<applet code="JLabelDemo.class" height=400 width=400></applet>*/

Applet Viewer: JLabelDemo.class

Applet

REGISTRATION FORM

UserName: [   ]

Password: [   ]

CPassword: [   ]

FNAME: [   ]

LNAME: [   ]

BIRTH DATE: [   ]

EMAIL_ID: [   ]

MOBILE NO: [   ]

[Submit]   [Reset]

Applet started.

4.// Create a jdbc program to create student table and insert two records.

import java.sql.*;

class CreateInsert

{

public static void main(String args[])

{

try

```java
{
Class.forName("com.mysql.jdbc.Driver");

Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/s6","root","mahe
sh");

Statement st=con.createStatement();

st.executeUpdate("create table student(rollno int, sname varchar(20))");

System.out.println("Table Created");

st.executeUpdate("insert into student values(101,'xyz')");

st.executeUpdate("insert into student values(102,'pqr')");

System.out.println("records inserted");

st.close();

con.close();
}
catch(Exception e)
{
e.printStackTrace();
}
}
}
```
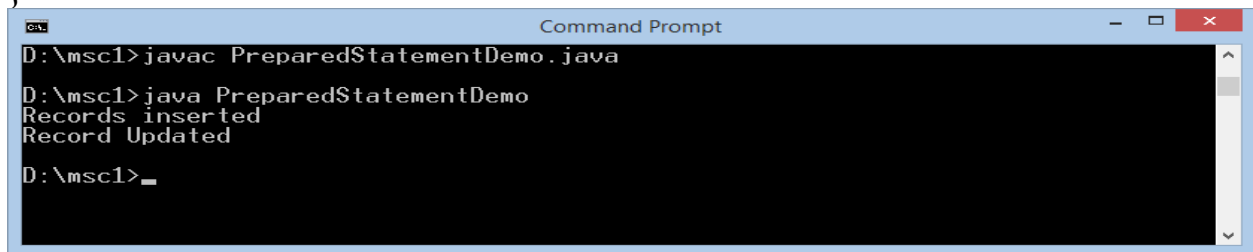
```
D:\msc1>javac CreateInsert.java

D:\msc1>java CreateInsert
Table Created
records inserted

D:\msc1>_
```



```
mysql> create database s6;
Query OK, 1 row affected (1.25 sec)

mysql> use s6
Database changed
mysql> desc student;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| rollno | int(11)     | YES  |     | NULL    |       |
| sname  | varchar(20) | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
2 rows in set (3.08 sec)

mysql> select * from student;
+--------+-------+
| rollno | sname |
+--------+-------+
|    101 | xyz   |
|    102 | pqr   |
+--------+-------+
2 rows in set (0.00 sec)

mysql> _
```

5. // Create a jdbc program to demonstrate PreparedStatement.

import java.sql.*;

class PreparedStatementDemo

{

public static void main(String args[])

```java
{
try
{
Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/s6","root","mahesh");

PreparedStatement pst=con.prepareStatement("insert into student values (?,?)");

pst.setInt(1,106);

pst.setString(2,"x");

pst.executeUpdate();

pst.setInt(1,107);

pst.setString(2,"y");

pst.executeUpdate();

System.out.println("Records inserted");

PreparedStatement pst1=con.prepareStatement("update student set sname=?  where rollno=?");

pst1.setString(1,"z");

pst1.setInt(2,102);

pst1.executeUpdate();

System.out.println("Record Updated");

pst1.close();

pst.close();

con.close();

}
```

catch(Exception e)

{

e.printStackTrace();

}

}

}

6. //Create a jdbc program to demonstrate ScrollableResultSets.

import java.sql.*;

class ScrollableResultSetDemo

{

public static void main(String args[])

{

try

{

Class.forName("com.mysql.jdbc.Driver");

```java
Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/s6","root","mahesh");

Statement st=con.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_UPDATABLE);

ResultSet rs=st.executeQuery("select * from student1");

rs.moveToInsertRow();

rs.updateInt(1,110);

rs.updateString(2,"ajdfksajdf");

rs.insertRow();

rs.updateInt(1,111);

rs.updateString(2,"asdcdc");

rs.insertRow();

System.out.println("Records Inserted");

rs.absolute(2);

rs.updateString(2,"ab");

rs.updateRow();

System.out.println("Record Updated");

rs.absolute(5);

rs.deleteRow();

System.out.println("Record Deleted");

rs.beforeFirst();

System.out.println("Records in forward direction");

while(rs.next())

{
```

```java
int i=rs.getInt(1);

String s=rs.getString(2);

System.out.println("RollNo: "+i+", Name: "+s);

}

rs.afterLast();

System.out.println("Records in reverse direction");

while(rs.previous())

{

int i=rs.getInt(1);

String s=rs.getString(2);

System.out.println("RollNo: "+i+", Name: "+s);

}

rs.close();

st.close();

con.close();

}

catch(Exception e)

{

e.printStackTrace();

}

}

}
```

7. /*Develop Simple Servlet Question Answer Application to demonstrate use of HttpServletRequest and HttpServletResponse interfaces.

*/

Question.html

```html
<html>
    <head>
        <title>Questions Page</title>
    </head>
    <body>
        <form method="POST" action="http://localhost:8085/h7/ValidateServlet">
        <h1>My paper</h1>
            Q. 1) Which is a valid keyword in java?<br>
            <input type="radio" name="r1" value ="int"/>interface
            <input type="radio" name="r1" value ="F"/>Float
            <input type="radio" name="r1" value ="s"/>string
```

<input type="radio" name="r1" value ="all"/>All of the above

Q. 2) What is sent to the user via HTTP, invoked using the HTTP protocol on the user's computer, and run on the user's computer as an application?

<input type="radio" name="r2" value="app"/>A Java application

<input type="radio" name="r2" value="applet"/>A Java applet

<input type="radio" name="r2" value="servlet"/>A Java servlet

<input type="radio" name="r2" value="none"/>None of the above

Q. 3) JDBC stands for:

<input type="radio" name="r3" value="jdbc"/>Java Database Connectivity

<input type="radio" name="r3" value="jdbc1"/>Java Database Components

<input type="radio" name="r3" value="jdbc2"/>Java Database Control

<input type="radio" name="r3" value="none"/>None of the above

Q. 4) What is bytecode?

<input type="radio" name="r4" value="msc"/>Machine-specific code

<input type="radio" name="r4" value="jcn"/>Java code

<input type="radio" name="r4" value="mic"/>Machine-independent code

<input type="radio" name="r4" value="none"/>None of the above

<input type="submit" value="Display Result"/>

<input type="reset" value="Reset Values"/>

</form>

```
  </body>
</html>
```

ValidateServlet.java

```
import java.io.*;

import javax.servlet.*;

import javax.servlet.http.*;

public class ValidateServlet extends HttpServlet
{
   public void doPost(HttpServletRequest req, HttpServletResponse res)throws
ServletException, IOException {

      res.setContentType("text/html");

      PrintWriter out = res.getWriter();

      try {

         out.println("<html>");

         out.println("<head>");

         out.println("<title>Test Result</title>");

         out.println("</head>");

         out.println("<body>");
if(req.getParameter("r1").equals("int")&&req.getParameter("r2").equals("app")&&
req.getParameter("r3").equals("jdbc")&&req.getParameter("r4").equals("mic"))

            out.println("You are Pass");
```

```java
            else
                out.println("Better luck Next time");


            out.println("</body>");
            out.println("</html>");


    } finally {
        out.close();
    }
}
}
```

web.xml

```xml
<web-app>
<servlet>
<servlet-name>ValidateServlet</servlet-name>
<servlet-class>ValidateServlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>ValidateServlet</servlet-name>
<url-pattern>/ValidateServlet</url-pattern>
</servlet-mapping>
</web-app>
```

```
D:\msc1>jar cvf h7.war WEB-INF Question.html
added manifest
adding: WEB-INF/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/classes/(in = 0) (out= 0)(stored 0%)
adding: WEB-INF/classes/ValidateServlet.class(in = 1345) (out= 779)(deflated 42%)
)
adding: WEB-INF/classes/ValidateServlet.java(in = 997) (out= 374)(deflated 62%)
adding: WEB-INF/web.xml(in = 268) (out= 94)(deflated 64%)
adding: Question.html(in = 1898) (out= 552)(deflated 70%)

D:\msc1>
```

localhost:8085/h7/Question.html

My paper

Q. 1) Which is a valid keyword in java?
◉ interface ○ Float ○ string ○ All of the above

Q. 2) What is sent to the user via HTTP, invoked using the HTTP protocol on the user's computer, and run on the user's computer as an application?
◉ A Java application ○ A Java applet ○ A Java servlet ○ None of the above

Q. 3) JDBC stands for:
◉ Java Database Connectivity ○ Java Database Components ○ Java Database Control ○ None of the above

Q. 4) What is bytecode?
○ Machine-specific code ○ Java code ◉ Machine-independent code ○ None of the above

[Display Result] [Reset Values]

localhost:8085/h7/ValidateServlet

You are Pass

8. /* Develop Servlet application to accept HTNO of a student from client and

display the memorandum of marks from the server */

StudentEntry.html

```html
<html>
<head>
<title>Student Result</title>
</head>
<body>
<form method="POST" action="http://localhost:8085/x24/DisplayResult">
<h1> Student Result</h1><br>
Enter Hallticket No: <input type="text" name="t1"><br>
<input type="submit" value="Submit"><input type="reset" value="Reset">
</form>
</body>
</html>
```

DisplayResult.java

```java
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class DisplayResult extends HttpServlet
{
Connection con;
Statement st;
ResultSet rs;
```

```java
PrintWriter out;
public void init(ServletConfig sc) throws ServletException
{
try
{
super.init(sc);
Class.forName("com.mysql.jdbc.Driver");
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/s3","root","mahe
sh");
}
catch(Exception e)
{
e.printStackTrace();
}
}
public void doPost(HttpServletRequest req, HttpServletResponse res)throws
ServletException, IOException
{
try
{
int rno=Integer.parseInt(req.getParameter("t1"));
res.setContentType("text/html");
 out=res.getWriter();
st=con.createStatement();
rs=st.executeQuery("select * from studentdetails where rollno="+rno);
```

```java
if(rs.next())
{
out.println("<html><body><table border=4 bordercolor='green'><caption>Student
Result</caption>");
out.println("<tr><th>Roll No:</th><td>"+rs.getInt(1)+"</td></tr>");
out.println("<tr><th>Student Name:</th><td>"+rs.getString(2)+"</td></tr>");
out.println("<tr><th>Subject   1:</th><td>"+rs.getString(3)+"</td><th>Subject   1
Marks</th><td>"+rs.getInt(6)+"</td></tr>");
out.println("<tr><th>Subject   2:</th><td>"+rs.getString(4)+"</td><th>Subject   2
Marks</th><td>"+rs.getInt(7)+"</td></tr>");
out.println("<tr><th>Subject   3:</th><td>"+rs.getString(5)+"</td><th>Subject   3
Marks</th><td>"+rs.getInt(8)+"</td></tr>");
out.println("</table></body></html>");
}
else
{
out.println("Rollno. not found");
}
}
catch(Exception e)
{
out.println("Roll No. Not Found");
e.printStackTrace();
}
}
public void destroy()
```

```
{
try
{
rs.close();
st.close();
con.close();
}
catch(Exception e)
{
e.printStackTrace();
}
}
}
```
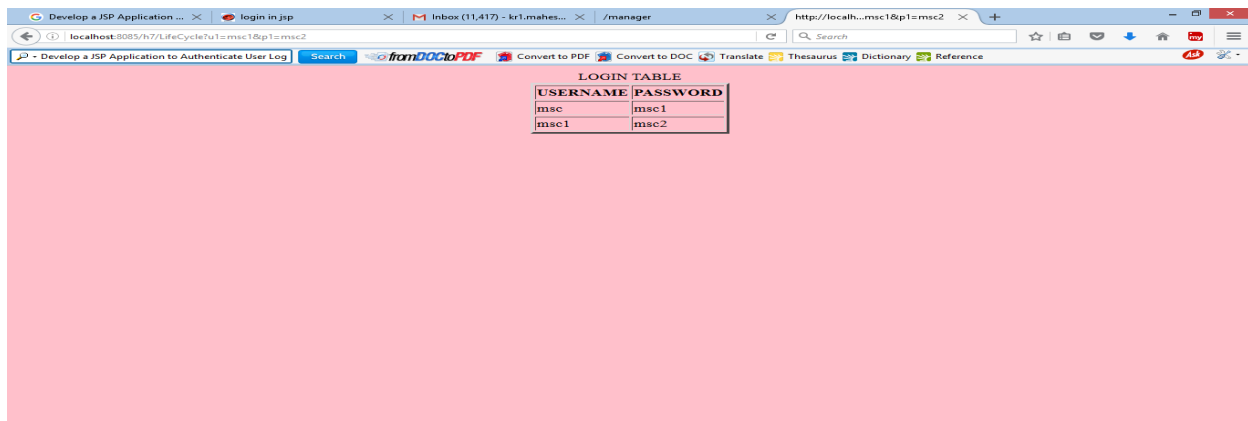
web.xml

```xml
<web-app>
<servlet>
<servlet-name>DisplayResult</servlet-name>
<servlet-class>DisplayResult</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>DisplayResult</servlet-name>
<url-pattern>/DisplayResult</url-pattern>
</servlet-mapping> </web-app>
```

Student Result

Enter Hallticket No: 101

Submit | Reset
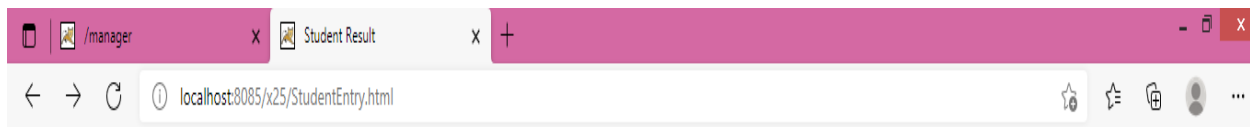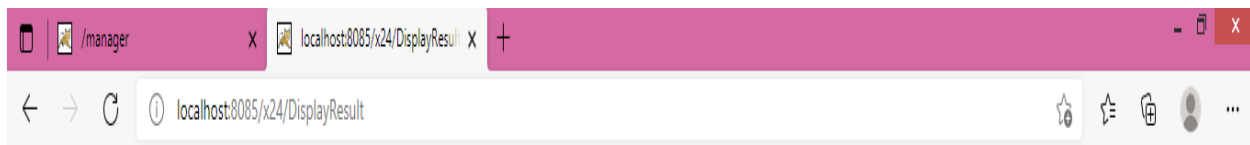
Student Result

Roll No:	101		
Student Name:	zyz		
Subject 1:	Advanced Java	Subject 1 Marks	56
Subject 2:	Operating Systems	Subject 2 Marks	67
Subject 3:	Discrete Mathematics	Subject 3 Marks	84

9. // Create 3-tier application using servlets.

LoginServlet.html

<html>

<head>

<title>User Login</title>

</head>

<body>

<center><h1>USER LOGIN FORM</h1><hr>

<form method="GET" action="http://localhost:8085/h7/LifeCycle"> <pre>

   <table border=2><tr><td><b>USER-ID</td><td> <input type=text name=u1 size=15 ><td></tr><tr><td>

   <b>PASSWORD </td><td >   <input type=password name=p1 size=15 ></tr><tr><td align=center>

   <input type=submit value=Login>

</td><td align=center><input type=Reset value=Reset></td></tr></table>

</pre> </form>

</body> </html>

LifeCycle.java

import javax.servlet.*;

import javax.servlet.http.*;

import java.io.*;

import java.sql.*;

public class LifeCycle extends HttpServlet

{

```java
Connection con;

Statement st;

ResultSet rs;

public void init(ServletConfig sc) throws  ServletException

{

try

{

super.init(sc);

Class.forName("com.mysql.jdbc.Driver");

con=DriverManager.getConnection("jdbc:mysql://localhost:3306/s6","root","mahesh");

}

catch(Exception e)

{

e.printStackTrace();

}

}

public void doGet(HttpServletRequest req, HttpServletResponse res) throws
ServletException, IOException

{

try

{

res.setContentType("text/html");

PrintWriter pw=res.getWriter();

String u=req.getParameter("u1");
```
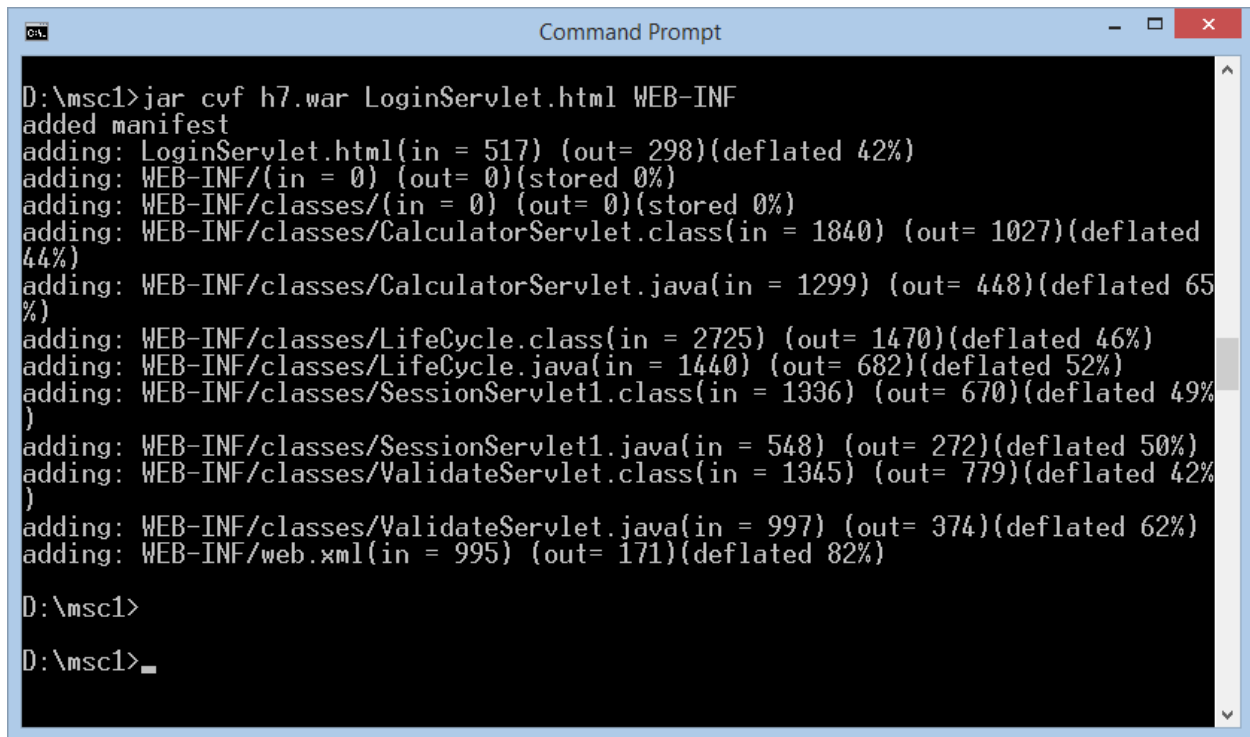
```
String p=req.getParameter("p1");

st=con.createStatement();

st.executeUpdate("insert into user values(' "+u+" ',' "+p+" ')");

pw.println("<html><body bgcolor=pink><center><table  border=3><caption>
LOGIN
TABLE</caption><tr><th>USERNAME</th><th>PASSWORD</th></tr>");

rs=st.executeQuery("select * from user");

while(rs.next())

{

pw.println("<tr><td>"+rs.getString(1)+"<td>"+rs.getString(2)+"</tr>");

}

pw.println("</table></center></body></html>");

}

catch(Exception e)

{

e.printStackTrace();

}

}

public void destroy()

{

try

{

rs.close();

st.close();

con.close();   }
```

catch(Exception e)

{   e.printStackTrace(); } } }

10. // Develop a servlet program to demonstration Http Session.

SessionServlet1.java

import javax.servlet.*;

import javax.servlet.http.*;

import java.io.*;

import java.util.*;

public class SessionServlet1 extends HttpServlet

{

```java
public void doGet(HttpServletRequest req, HttpServletResponse res)throws
ServletException, IOException
{
HttpSession ses=req.getSession(true);
Date d=(Date)ses.getValue("date");
PrintWriter pw=res.getWriter();
if(d==null)
{
pw.println("First Request");
d=new Date();
ses.putValue("date",d);
}
else
{
pw.println("Last Request Time:"+d);
d=new Date();
ses.putValue("date",d);
}
}
}
```

web.xml

```xml
<web-app>
<servlet>
<servlet-name>SessionServlet1</servlet-name>
<servlet-class>SessionServlet1</servlet-class>
```

\</servlet>

\<servlet-mapping>

\<servlet-name>SessionServlet1\</servlet-name>

\<url-pattern>/SessionServlet1\</url-pattern>

\</servlet-mapping>

\</web-app>

11. /* Develop a JSP Application to accept Registration Details from the user and store database. */

Login.jsp

```
<%@page contentType="text/html"%>

<html>

  <head>

        <title>Registration</title>

  </head>

  <body>

    <h1>Registration</h1>

    <form  action="http://localhost:8085/h7/Registration.jsp">

       username: <input type="text" name="t1"><br>

       password: <input type="password" name="t2"><br>

       email:       <input
     type="text" name="t3"><br>

       <input type="submit" value="login">

       <input type="reset" value="reset">

    </form>     </body> </html>
```
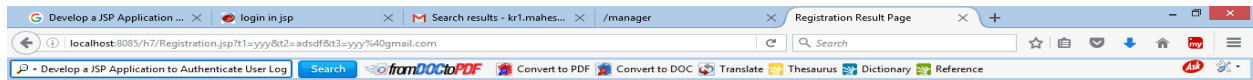
Registration.jsp

```
<html>
  <head>
    <title>Registration Result Page</title>
  </head>
  <body>
    <%@page import="java.sql.*" contentType="text/html" %>
    <%
    try
        {
        Class.forName("com.mysql.jdbc.Driver");
Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/s6","root","mahesh");
String s=request.getParameter("t1");
String s1=request.getParameter("t2");
String s2=request.getParameter("t3");
 PreparedStatement ps=con.prepareStatement("insert into login values(?,?,?)");
        ps.setString(1,s);
        ps.setString(2,s1);
        ps.setString(3,s2);
```
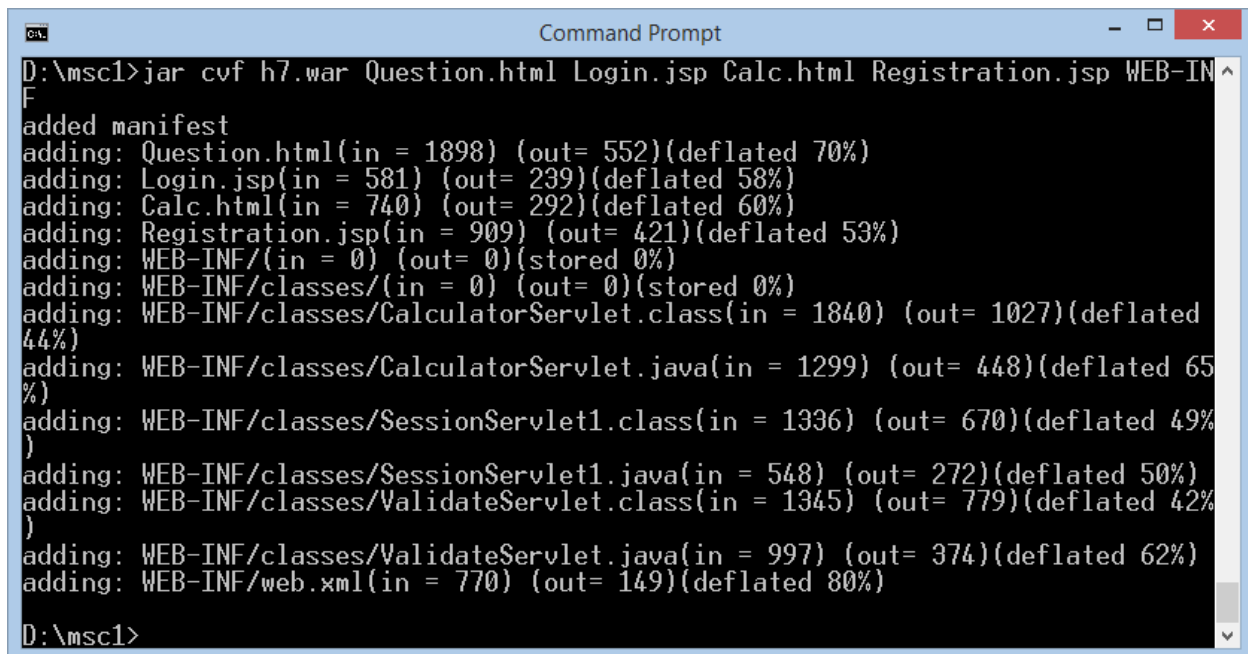
```
        ps.executeUpdate();

            }


catch(Exception e)

            {

        e.printStackTrace();

            }

    %>

    Record inserted successfully!!!!!!!!!!!!    </body></html>
```

Record inserted successfully!!!!!!!!!!!!

12. /*Develop a JSP Application to Authenticate User Login as per the Registration Details. If Login Success then forward User to Index Page otherwise show Login failure Message. */

UserLogin.jsp

```html
<html>

<head>

<title>User Login</title>

</head>

<body>

<center><h1>USER LOGIN FORM</h1><hr>

<form method="POST" action="http://localhost:8085/h7/ValidateUser.jsp">
<pre>   <table border=2><tr><td><b>USER-ID</td><td> <input type=text
name=u1 size=15 ><td></tr>

<tr><td> <b>PASSWORD</td><td > <input type=password name=p1
size=15></tr><tr>

<td align=center> <input type=submit value=Login></td>
<td align=center><input type=Reset value=Reset></td></tr>

</table>

</pre>  </form>

</body> </html>
```
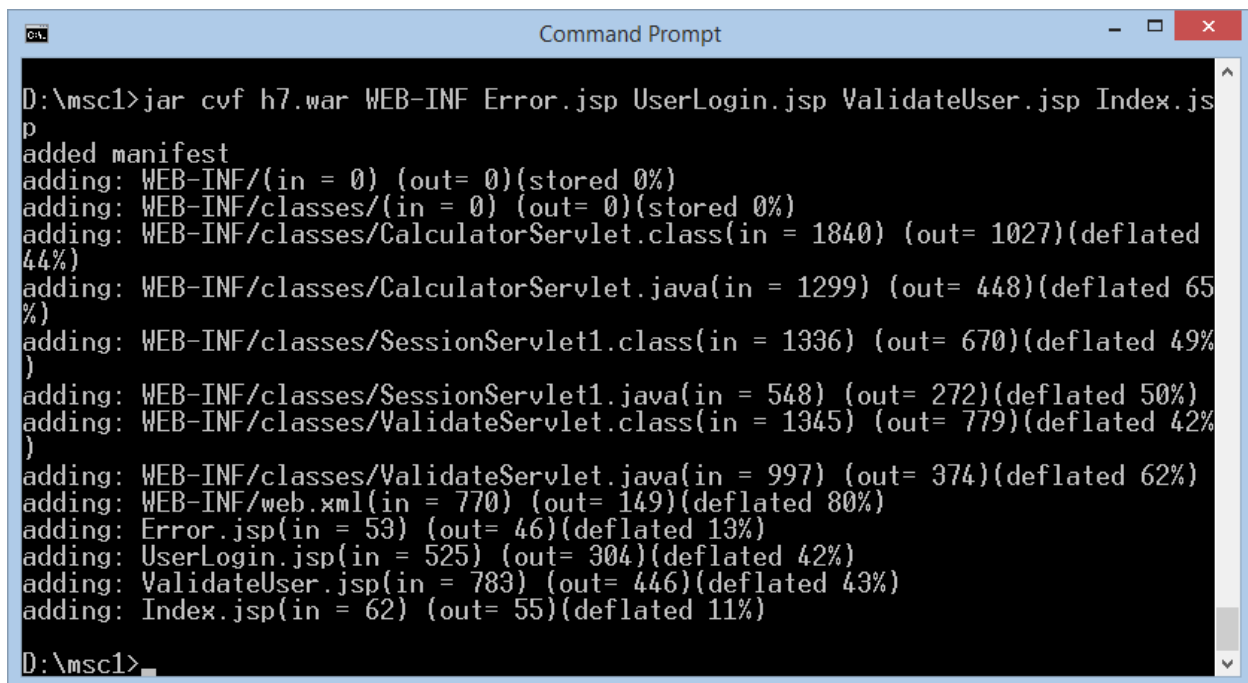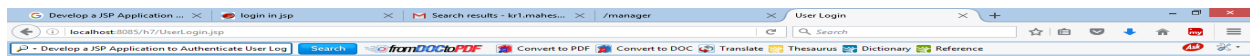
ValidateUser.jsp

```html
<html>
```

```jsp
<head><title>User Validation</title></head>

<body>

<%@page import="java.sql.*" %>

<%   try

        {

            String s1=request.getParameter("u1");

            String s2=request.getParameter("p1");

Class.forName("com.mysql.jdbc.Driver");

Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/s6","root","mahe
sh");

Statement st=con.createStatement();

ResultSet rs=st.executeQuery("select * from login where uname='"+s1+"' and
password='"+s2+"'");

if(rs.next())

{

 response.sendRedirect("http://localhost:8085/h7/Index.jsp");

}

else

  {

response.sendRedirect("http://localhost:8085/h7/Error.jsp");

  }

 st.close();

 con.close();

 }
```

catch(Exception e)

{   out.println(e.toString());   }%> </body> </html>

Index.jsp

<html><body>

<h1>Welcome to index page</h1>

</body></html>

Error.jsp

<html><body>

<h1>Invalid USER</h1>

</body></html>

USER LOGIN FORM

USER–ID: xyz
PASSWORD: •••
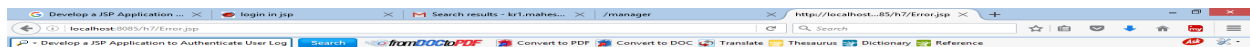Login   Reset
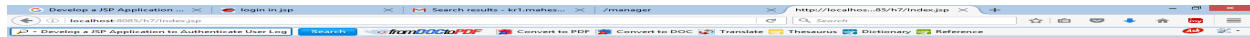
Welcome to index page
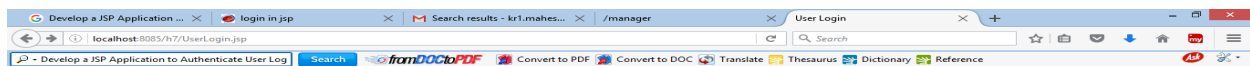
USER LOGIN FORM

USER–ID: xyz2
PASSWORD: •••
Login   Reset

Invalid USER

13. a) /*Create a JSP page that prints temperature conversion (from Celsius to Fahrenheit) */

Celsius.jsp

```
<html>
<body>
<%
float Fahrenheit, Celsius;
Celsius= 13;
Fahrenheit = ((Celsius*9)/5)+32;
out.println("Temperature in Fahrenheit is: "+Fahrenheit);
%>
</body>
</html>
```
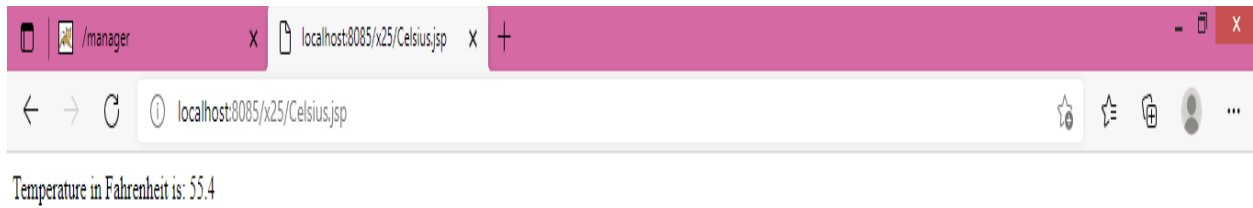
Temperature in Fahrenheit is: 55.4
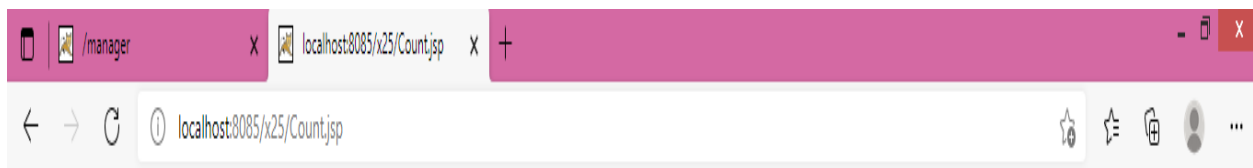
b) /*Create a JSP page to print current Date and Time */

```
<html>
<body>
<h2>The current system data and time is: <%=new java.util.Date()%> </h2>
</body>
</html>
```


The current system data and time is: Fri Jul 30 23:10:09 IST 2021

c) /*Create a JSP page to print number of times page is referred after the page is loaded.*/

```
<html>
<body>
<%! int count=0;%>
<% count++; %>
<h2>The page is requested by <%=count%> number of users till now.</h2>
</body>
</html>
```


The page is requested by 5 number of users till now.

14. /* Write a program to demonstrate RMI application */

Calculator.java (Remote Interface Program)

```java
import java.rmi.*;
public interface Calculator extends Remote
{
public int add(int a, int b) throws RemoteException;
}
```

CalculatorImpl.java (Implementation Program)

```java
import java.rmi.*;
import java.rmi.server.*;
public class CalculatorImpl extends UnicastRemoteObject implements Calculator
{
public CalculatorImpl() throws RemoteException
{
super();
}
```

```java
public int add(int a, int b) throws RemoteException
{
  return a+b;
}
}
```

CalculatorServer.java (Server Program)

```java
import java.rmi.*;
public class CalculatorServer
{
public static void main(String args[])
{
try
{
CalculatorImpl c1=new CalculatorImpl();
Naming.rebind("msc",c1);
System.out.println("Object is binded and Server is ready");
}
catch(Exception e)
{
e.printStackTrace();
}
}
}
```

CalculatorClient.java (Client Program)

```java
import java.rmi.*;
public class CalculatorClient
{
public static void main(String args[])
{
try
{
Calculator c2=(Calculator)Naming.lookup("rmi://localhost:1099/msc");
int result=c2.add(100,200);
System.out.println("Result ="+result);
}
catch(Exception e)
{
e.printStackTrace();
}
}
}
```

I:\BCA ONLINE CLASS MATERIALS\MSC CS I SEM AJP 2020-2021\rmi>javac *.java

I:\BCA ONLINE CLASS MATERIALS\MSC CS I SEM AJP 2020-2021\rmi>start rmiregistry

I:\BCA ONLINE CLASS MATERIALS\MSC CS I SEM AJP 2020-2021\rmi>

I:\BCA ONLINE CLASS MATERIALS\MSC CS I SEM AJP 2020-2021\rmi>start java CalculatorServer

I:\BCA ONLINE CLASS MATERIALS\MSC CS I SEM AJP 2020-2021\rmi>

Object is binded and Server is ready

Command Prompt

```
I:\BCA ONLINE CLASS MATERIALS\MSC CS I SEM AJP 2020-2021\rmi>java CalculatorClient
Result =300

I:\BCA ONLINE CLASS MATERIALS\MSC CS I SEM AJP 2020-2021\rmi>
```