
Java and J2EE Lab

Course Code: ISL67
Contact Hours: 28

Credit: 0:0:1

1. Write Java program to
Create a collection of contact lists (contact number and name)
Accept missed calls from the user and store them in another list. If the caller details exist in the contact list, retrieve the caller details else store as "private caller". Store them in the order.
User can trace the missed call list and display them if required.
Delete the number if user specifies a number to delete.
Write main program to test the class.
2. Write a Java program using user-defined storage classes to create a book database and store it in a Collection List. Books collection should include title, author, publisher and price.
Write method to sort the books in ascending order of price and store it in another List. Maintain the book details with respect to a unique book id.
Prompt for an author name and list all the books with the same author's name.
Create a new list holding all the book details with price greater than a user specified price.
For a given value by the user, find all the books that match either the whole or a part of the book title.
Identify a publisher and print books from a particular publisher. Update the publisher details based on a title.
3. Create a desktop java application using swings to enable an user to enter student information such as name, usn, age, address, sgpa of 8 semesters, category.
Perform validations on age and sgpa. Display appropriate messages in pop up boxes to indicate wrong entries.
On clicking of the "compute" button, find the cgpa based on the obtained sgpa's.
On clicking of the "done" button, place the student details in a collection.
A click on the "display" button should display the collection in a textarea.
4. Write a java program using Swings to validate user login information using dialog boxes.
 - a. Once validated, allow the user to enter the customer id, if the person is a new customer, else check whether the customer exists in a collection and obtain the customer id.
 - b. The customer id can be obtained given a mobile number. Allow the user to enter the item purchased by giving the item id and quantity purchased.
 - c. On clicking of a button, the item name and the total cost should appear in the corresponding GUI components.

- d. Using option dialog box, indicate the types of discount available for the customer. On clicking on the print button, print the details in information dialog box.
5. Write a program that uses Java Swing and JDBC to create a stand-alone application:
 - i. Create a table namely,
 - a. Customer (CustNo, CustName, State, Credit_Limit, RepNo) in MySQL database. CustNo is the primary key
 - ii. Use appropriate Swing components to insert values in a form.
 - iii. Display Customer information when Credit_Limit is above 15,000.

Steps:

1. Open mysql , select/create a database data1

Create a table in the database called customer by using the command
 create table customer(custNo int, custName varchar(25), state varchar(25), credit_limit
 varchar(25) , repno double);

2. In a java project, in package CustomerProg create class Customer .
3. Do a right click to the project, select build path, click on libraries, click on classpath, then on right side of the window, click external libraries and browse to include mysql-connector-java-8.0.19.jar (Note: another version of mysql connector jar file can be used)

package CustomerProg;

```
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.Statement;
```

```
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JTextField;
```

```
public class Customer implements ActionListener {
    JFrame jf1;
    JLabel jl1 = new JLabel("Enter customer number");
    JTextField cno = new JTextField(10);
    JLabel jl2 = new JLabel("Enter Customer Name ");
    JTextField cname = new JTextField(20);
    JLabel jl3 = new JLabel("Enter State");
```

```

    JTextField st = new JTextField(20);
    JLabel jl4 = new JLabel("Enter Credit Limit");
    JTextField credit_l = new JTextField(10);
    JLabel jl5 = new JLabel("Enter Representative No");
    JTextField repno = new JTextField(10);
    JLabel jl6 = new JLabel("Click to insert into database");
    JButton b1 = new JButton("insert");

```

```

Customer(){
    jf1 = new JFrame("Read Customer Information");
    jf1.add(jl1);jf1.add(cno);
    jf1.add(jl2); jf1.add(cname);
    jf1.add(jl3); jf1.add(st);
    jf1.add(jl4); jf1.add(credit_l);
    jf1.add(jl5);jf1.add(repno);
    jf1.add(jl6);jf1.add(b1);

    jf1.setVisible(true);
    jf1.setSize(400, 400);
    jf1.setLayout(new GridLayout(8,1));
    b1.addActionListener(this);

}

```

```

@Override
public void actionPerformed(ActionEvent e) {
    // TODO Auto-generated method stub
    if(e.getSource()==b1) {
        //insert into database
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection
connection=DriverManager.getConnection("jdbc:mysql://localhost:3306/data1"
,"root","root");

            PreparedStatement statement;

            String query = "INSERT INTO
customer(custNo,custName,state,credit_limit,repno)" + "VALUES (?, ?, ?,?,?)";
            statement = connection.prepareStatement(query);
            statement.setInt(1, Integer.parseInt(cno.getText()));
            statement.setString(2,cname.getText());
            statement.setString(3, st.getText());

```

```

statement.setDouble(4, Double.parseDouble(credit_l.getText()));
statement.setInt(5, Integer.parseInt(repno.getText()));
int status = statement.executeUpdate();
if(status>0) {
    System.out.println("Insertion successful");
}
ResultSet resultSet;
resultSet = statement.executeQuery("select * from customer where
credit_limit > 15000;");
while(resultSet.next()) {
    System.out.println(resultSet.getInt(1)+"          "+resultSet.getString(2)+"
"+resultSet.getString(3)+" "+resultSet.getDouble(4)+" "+resultSet.getInt(5));
}

        } catch (Exception e1) {
            // TODO Auto-generated catch block
            e1.printStackTrace();
        }
    }
}

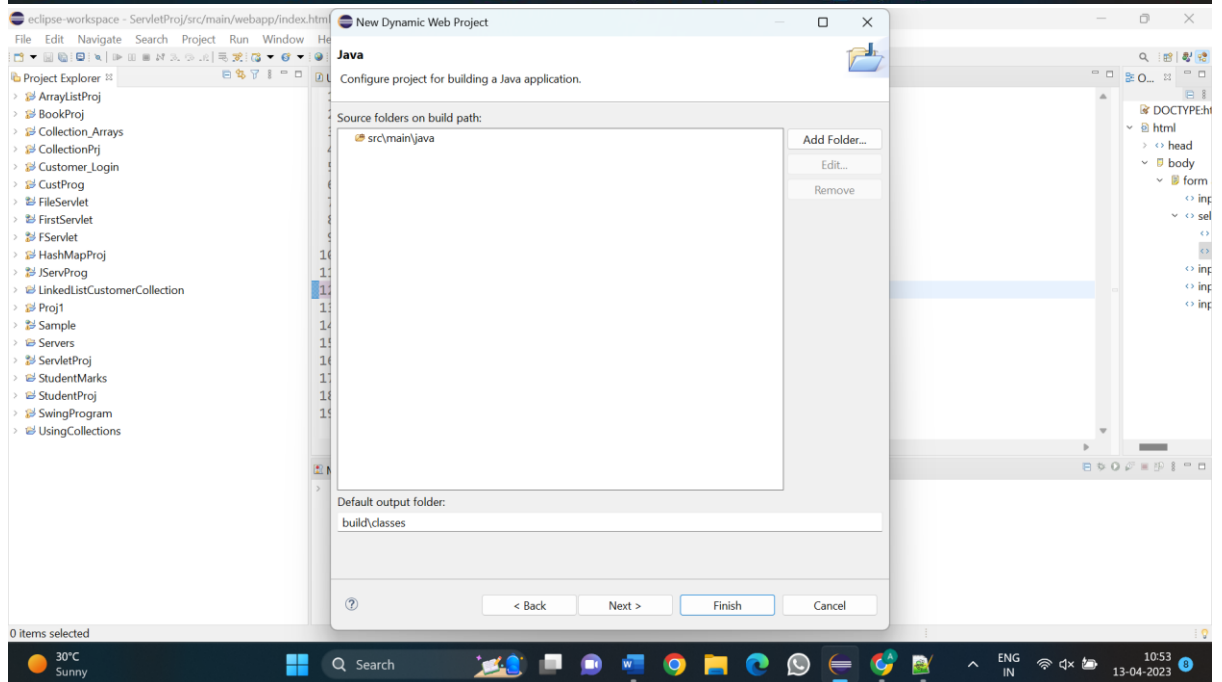
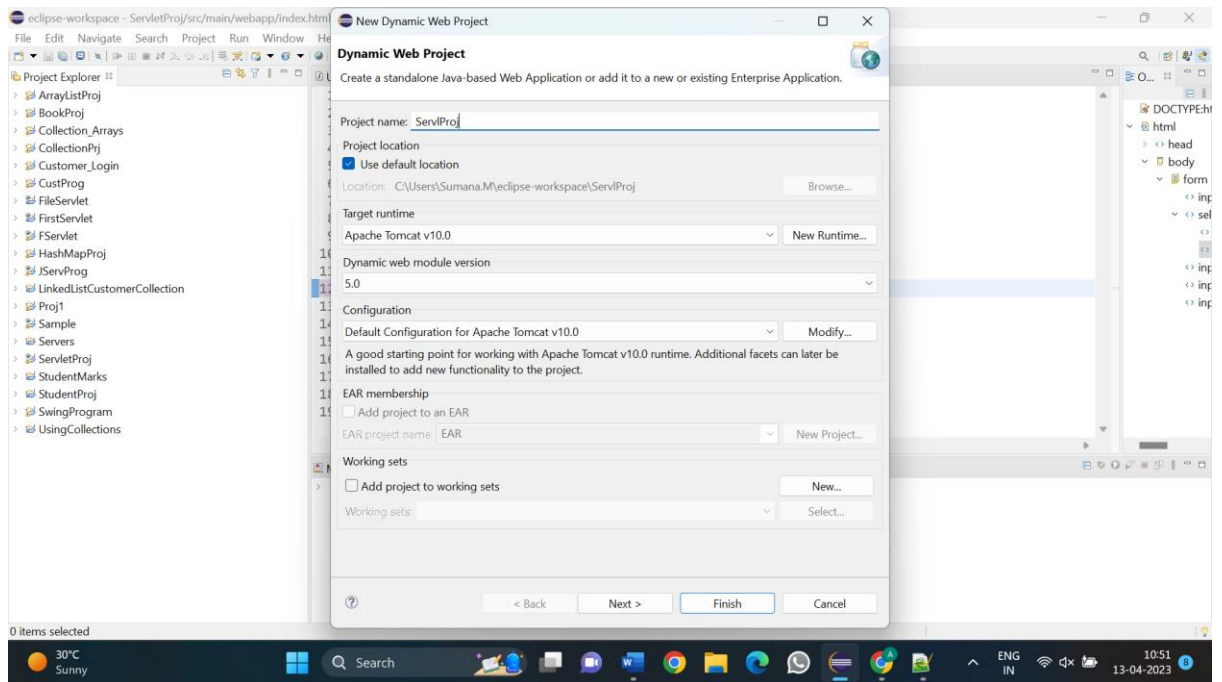
public static void main(String arg[]) {

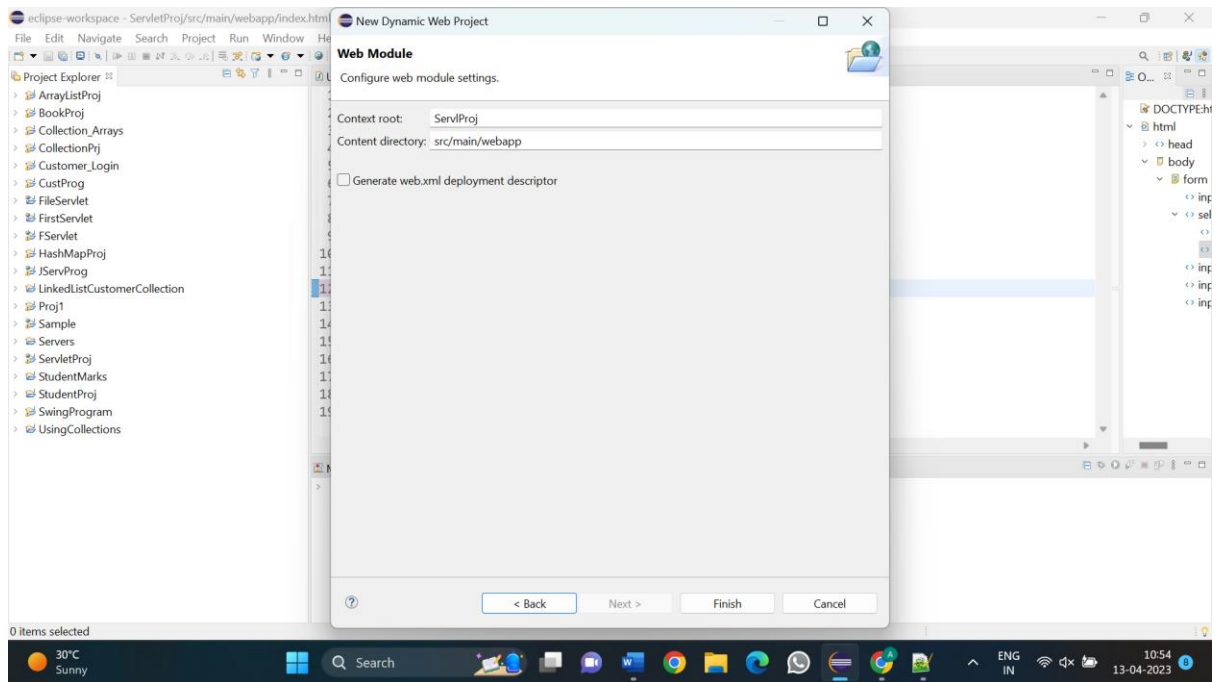
    Customer cust = new Customer();

}
}

```

6. Create a Servlet to file IT returns that accepts personal information, salary information and Tax deduction details from the user. Write the information into a file. Accept the name of the person and display in on the page.





1. Create a project ServlProj (new-> dynamic Web Project) and attach the Apache Tomcat v10.0 . Front page should be as shown above. Click next to create project.
2. Right click on the project ,ServlProj (new->html file) create index.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<form action="ITReturns" method="get" >
name:<input type="text" name="name"/>
<select name="gender">
<option>male</option>
<option>female</option>
</select>
sal:<input type="text" name="salary"/>
tax:<input type="text" name="tax"/>
<input type="submit"/>
</form>
</body>
</html>
```

3. Right click on the project, ServlProj (new->servlet) create ITReturns.java

```
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.PrintWriter;
```

```

import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import java.io.IOException;
/**
 * Servlet implementation class ITReturns
 */
public class ITReturns extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public ITReturns() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        String name=request.getParameter("name");
        String gender=request.getParameter("gender");
        String salary=request.getParameter("salary");
        String tax=request.getParameter("tax");
        PrintWriter out=response.getWriter();
        File file = new File("C:\\Users\\Sumana.M\\eclipse-
workspace\\ServletProj\\src\\main\\java\\1.txt"); // Place the location of the txt file
        file.createNewFile();
        FileOutputStream fout = new FileOutputStream(file);
        out.println(""+name+gender+salary+tax);
        fout.write(("hello"+name+gender+salary+tax).getBytes());
        fout.close();

        response.getWriter().append("Served at:
").append(request.getContextPath());
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
     */

```

```

        protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
            // TODO Auto-generated method stub
            doGet(request, response);
        }
    }
}

```

4. In the project explorer, right click on the index.html file -> run on server.

7. Create a servlet that accepts patient information in a hospital such as patient id, patient name, and age, date of admission, cause of admission, doctor diagnosed, and treatment proposed. Use HTML for front end.
 - a. Place the details into a database. Allow options to insert and view the contents in the database.

```
mysql> create database hospitaldata;
```

```
mysql> use hospitaldata;
```

Database changed

```
mysql> create table hospital(
```

```
    -> name varchar(20),
```

```
    -> age int,
```

```
    -> data varchar(20),
```

```
    -> cause varchar(20),
```

```
    -> doc varchar(20),
```

```
    -> treat varchar(20));
```

PatientServlet.java //Servlet Program

```

import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;

```

```
/**
```

```
 * Servlet implementation class PatientServlet
```

```
 */
```

```

public class PatientServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

```

```
/**
```

```
 * @see HttpServlet#HttpServlet()
```



```

*/
public PatientServlet() {
    super();
    // TODO Auto-generated constructor stub
}

/**
 * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
    // TODO Auto-generated method stub
    String name = request.getParameter("name");
    String age = request.getParameter("age");
    String date = request.getParameter("date");
    String cause = request.getParameter("cause");
    String doctor = request.getParameter("doc");
    String treat = request.getParameter("treat");
    String s=null;
    try {
        Class.forName("com.mysql.cj.jdbc.Driver");
        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/hospitaldata", "root",
"root");
        String updateDB = "insert into hospital
values(?,?,?, ?,?, ?)";
        PreparedStatement ps =
con.prepareStatement(updateDB);
        ps.setString(1, name);
        ps.setString(2, age);
        ps.setString(3, date);
        ps.setString(4, cause);
        ps.setString(5, doctor);
        ps.setString(6, treat);
        int i = ps.executeUpdate();
        System.out.println("Rows inserted: "+i);
        s = name + "\n" + age + "\n" + date + "\n" + cause + "\n"
+ doctor + "\n" + treat + "\n" + i;
    }
    catch(Exception e)
    {
        System.out.println(e);
    }
}

```

```

        request.setAttribute("data", s);

        request.getRequestDispatcher("/view.jsp").forward(request,response);

        response.getWriter().append("Served
    ").append(request.getContextPath());
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
    response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }
}

```

Insert.jsp // for inserting into database

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<form action='PatientServlet' method='post'>
Name: <input type='text' name='name' /><br>
<br> Age: <input type='number' name='age' />
<br><br> Date of Admission: <input type='date' name='date' />
<br><br> Cause of Admission: <input type='text' name='cause' />
<br><br> Doctor Diagnosed: <input type='text' name='doc' />
<br><br> Treatment Proposed: <input type='text' name='treat' />
<br><br> <input type="submit" value='update' /> </form> <br>

</body>
</html>

```

GetServlet.java // for viewing

```

import jakarta.servlet.ServletException;

```

```

import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;

/**
 * Servlet implementation class GetServlet
 */
public class GetServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public GetServlet() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request,
    HttpServletResponse response)
     */
    protected void doGet(HttpServletRequest request,
    HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub

        try {
            PrintWriter out=response.getWriter();
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/hospitaldata", "root", "root");
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("select * from
hospital;");
            while(rs.next()) {
                out.println(rs.getString(1)+" "+rs.getInt(2)+"
"+rs.getString(3)+" ");
                out.println(rs.getString(4)+" "+rs.getInt(5)+"
"+rs.getString(6)+" ");
            }
        }
    }

```

```

        } catch (Exception e1) {
            // TODO Auto-generated catch block
            e1.printStackTrace();
        }
        response.getWriter().append("Served at:
").append(request.getContextPath());
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request,
HttpServletResponse response)
     */
    protected void doPost(HttpServletRequest request,
HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        doGet(request, response);
    }
}

```

view.jsp //for viewing

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<form action='GetServlet' method='post'>
Get Current Database data: <input type='submit' value='Get Data'/>
</form> </div>
</body>
</html>

```

8. Create the following application with JSP and Servlet.
Create a Telephone Directory database that searches the database based on phone number or name. Inserting values to the table obtaining the data from the front end. Directory Table should include attributes: Phone_Number, Name, Address, Company, Pin_Code.
9. Write a Java Program that creates two threads object of Thread class. Where one thread asks the user to enter a number not less than four digits. Split the digits of the number and display in words the value of the number. Ex: 1 – One. Second thread finding the number of vowels in a word. Ex: JAVA – Vowel - A, Count – 2.

