1. **Write a program to implement reverse string server using network programming.**

**Client.java**

import java.io.\*;

import java.net.Socket;

import java.util.Scanner;

public class ReverseString {

public static void main(String[] args) throws Exception {

try {

Scanner sc = new Scanner(System.in);

System.out.print("Enter your port : ");

int port = sc.nextInt();

sc.nextLine();

Socket s = new Socket("localhost", port);

if (s.isConnected()) {

System.out.println("Connected to Server....");

}

while (true) {

System.out.print("Enter String to reverse : ");

String inputStr = sc.nextLine();

DataOutputStream dataOut = new DataOutputStream(s.getOutputStream());

dataOut.writeUTF(inputStr);

DataInputStream dataIn = new DataInputStream(s.getInputStream());

String revStr = dataIn.readUTF();

System.out.println("Reversed String : " + revStr);

System.out.println();

}

} catch (Exception e) {

System.out.println("Client System Error!\n" + e);

}

sc.close();

}

}

**Server.java**

import java.io.DataInputStream;

import java.io.DataOutputStream;

import java.net.ServerSocket;

import java.net.Socket;

import java.util.Scanner;

import static java.lang.Integer.parseInt;

public class Server {

public static void main(String[] args) throws Exception {

try {

Scanner sc = new Scanner(System.in);

System.out.print("Enter Port Number : ");

int port = sc.nextInt();

sc.nextLine();

System.out.println("Server is running ... !");

ServerSocket ss = new ServerSocket(port);

new ChatServers(ss.accept());

} catch (Exception e) {

System.out.println("Server Error : " + e);

}

}

}

class ChatServers extends Thread {

Socket s;

public ChatServers(Socket socket) {

s = socket;

Thread t = new Thread(this);

t.start();

}

@Override

public void run() {

receive();

}

public void receive() {

try {

while (true) {

DataInputStream dataIn = new DataInputStream(s.getInputStream());

String str = dataIn.readUTF();

StringBuffer rev = new StringBuffer();

rev.append(str);

rev = rev.reverse();

String revStr = new String(rev);

DataOutputStream dataOut = new DataOutputStream(s.getOutputStream());

dataOut.writeUTF(revStr);

}

} catch (Exception e) {

System.out.println("Server Error \n" + e);

}

}

}

1. **Write a filter to count user for all requests.**

**Servlet File**

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

import java.util.\*;

public class SiteHitCounter implements Filter {

private int hitCount;

public void init(FilterConfig config) throws ServletException {

// Reset hit counter.

hitCount = 0;

}

public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)

throws java.io.IOException, ServletException {

// increase counter by one

hitCount++;

// Print the counter.

System.out.println("Site visits count :"+ hitCount );

// Pass request back down the filter chain

chain.doFilter(request,response);

}

public void destroy() {

// This is optional step but if you like you can write hitCount value in your database.

}

}

**web.xml**

....

<filter>

<filter-name>SiteHitCounter</filter-name>

<filter-class>SiteHitCounter</filter-class>

</filter>

<filter-mapping>

<filter-name>SiteHitCounter</filter-name>

<url-pattern>/\*</url-pattern>

</filter-mapping>

....

1. **Implement JSP file to process user registration data (username,password,email,phone) using useBean, setProperty and getProperty JSP actions.**

**Details.java**

public class Details {

private String username;

private String email;

private String phone;

private String password;

public Details(String username, String email, String phone, String password) {

this.username = username;

this.email = email;

this.phone = phone;

this.password = password;

}

public String getUsername() {

return username;

}

public void setUsername(String username) {

this.username = username;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getPhone() {

return phone;

}

public void setPhone(String phone) {

this.phone = phone;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

}

**Index.jsp**

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>useBean, getProperty and setProperty example</title>

</head>

<body>

<form action="userdetails.jsp" method="post">

User Name: <input type="text" name="username"><br>

User Password: <input type="password" name="password"><br>

User Email: <input type="email" name="email"><br>

User Phone Number: <input type="text" name="phone\_number"><br>

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

</form>

</body>

</html>

**userdetails.jsp**

<jsp:useBean id="userinfo" class="com.vvp.web.Details"></jsp:useBean>

<jsp:setProperty property="\*" name="userinfo"/>

You have enterted below details:<br>

<jsp:getProperty property="username" name="userinfo"/><br>

<jsp:getProperty property="password" name="userinfo"/><br>

<jsp:getProperty property="email" name="userinfo" /><br>

<jsp:getProperty property="phone\_number" name="userinfo" /><br>

1. **Implement public ArrayList select(String[] columnnames,String tableName,String condition) to retrieve data from database.**

**addProduct.jsp**

<%@page import="webservlet.Products"%>

<%@ page import="webservlet.databasecon, java.sql.\*, java.util.ArrayList" %>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<div class="main2">

<form method="post">

Search By:&nbsp;<select name="searchby">

<option value="pid">Product Id</option>

<option value="pname">Product Name</option>

<option value="pstock">Stock Items</option>

<option value="price">Price</option>

<option value="pcat">Category</option>

</select>&nbsp;

<select name="searchopt">

<option value=">">></option>

<option value="=">=</option>

<option value="<"><</option>

<option value="LIKE">LIKE</option>

</select>&nbsp;

<input type="text" name="searchproduct" placeholder="Search ur item..">&nbsp;

<input type="submit" name="searchbt" value="Search">

<input type="submit" name="clear" value="clear">

</form>

<br>

<%

String condition = "1=1";

String searchbt = request.getParameter("searchbt");

String clear = request.getParameter("clear");

if (clear != null) {

condition = "1=1";

}

if (searchbt != null) {

String searchby = request.getParameter("searchby");

String searchopt = request.getParameter("searchopt");

String val = request.getParameter("searchproduct");

if (searchopt.equals("LIKE")) {

val = "%" + val + "%";

}

condition = searchby + " " + searchopt + " '" + val + "'";

System.out.println("where " + condition);

}

databasecon ob=new databasecon();

String columnnames[]={"pid","pname","pstock","pimage","price","pcat"};

String tablename="products";

ArrayList<Products> ar=ob.select(columnnames,tablename,condition);

%>

<table class="table">

<thead class="table-dark">

<td>Product ID</td>

<td>Product Name</td>

<td>Product Category</td>

<td>Product Stock</td>

<td>Product Image</td>

<td>Product Price</td>

</thead>

<tbody>

<%

for(Products pro:ar){

%>

<tr>

<td align='center'><%=pro.getPid()%></td>

<td align='center'><%=pro.getPname()%></td>

<td align='center'><%=pro.getPcat()%></td>

<td align='center'><%=pro.getPstock()%></td>

<td align='center'><img src="<%=pro.getPimage()%>" class="img-fluid" height="100px" width="100px" alt=""></td>

<td align='center'><%=pro.getPrice()%></td>

</tr>

<%

}

%>

</tbody>

</table>

</div>

</body>

</html>

**Java File**

import java.util.\*;

import java.sql.\*;

public class databasecon {

public ArrayList select(String[] columnnames, String tablename, String condition) throws ClassNotFoundException, SQLException {

Connection myconn = null;

Statement mystmt = null;

ResultSet myres = null;

ArrayList<Products> arr=new ArrayList();

try {

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

myconn = DriverManager.getConnection("jdbc:mysql://localhost:3360/ecommerce21", "root", "");

mystmt = myconn.createStatement();

myres = mystmt.executeQuery("select \* from "+tablename+" where " + condition);

while(myres.next())

{

int pid=myres.getInt(columnnames[0]);

String pname = myres.getString(columnnames[1]);

int pstock=myres.getInt(columnnames[2]);

String pimage=myres.getString(columnnames[3]);

double price = myres.getDouble(columnnames[4]);

String pcat=myres.getString(columnnames[5]);

Products pd = new Products(pid,pname,pstock,pimage,price,pcat);

arr.add(pd);

}

}

catch (ClassNotFoundException | NumberFormatException | SQLException e) {

e.printStackTrace();

}

return arr;

}

}

**POJO File**

package webservlet;

public class Product {

public int qty;

public double price;

public String pname;

public String pimage,category;

public String getPimage() {

return pimage;

}

public void setPimage(String pimage) {

this.pimage = pimage;

}

public Product(int qty, double price, String pname,String category, String pimage) {

this.qty = qty;

this.price = price;

this.pname = pname;

this.pimage = pimage;

this.category=category;

}

public String getCategory() {

return category;

}

public void setCategory(String category) {

this.category = category;

}

}

1. **Write a mapping file for product\_master(pid,pname,cat\_id,price,stock,img), product\_category(cat\_id,cat\_name,parent\_id)**

For **ProductMaster**

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<!-- Generated 16 Apr, 2021 1:32:14 PM by Hibernate Tools 4.3.1 -->

<hibernate-mapping>

<class name="com.vvp.web.ProductMaster" table="product\_master" catalog="ecommerce21" optimistic-lock="version">

<id name="pid" type="java.lang.Integer">

<column name="pid" />

<generator class="identity" />

</id>

<property name="pname" type="string">

<column name="pname" length="100" not-null="true" />

</property>

<property name="cat\_id" type="int">

<column name="cat\_id" not-null="true" />

</property>

<property name="price" type="double">

<column name="price" precision="22" scale="0" not-null="true" />

</property>

<property name="stock" type="int">

<column name="stock" not-null="true" />

</property>

<property name="image" type="string">

<column name="image" length="300" not-null="true" />

</property>

</class>

</hibernate-mapping>

For **ProductCategories**

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<!-- Generated 16 Apr, 2021 1:32:14 PM by Hibernate Tools 4.3.1 -->

<hibernate-mapping>

<class name="com.vvp.web.ProductCategories" table="product\_categories" catalog="ecommerce21" optimistic-lock="version">

<id name="cat\_id" type="java.lang.Integer">

<column name="cat\_id" />

<generator class="identity" />

</id>

<property name="cat\_name" type="string">

<column name="cat\_name" length="100" not-null="true" />

</property>

<property name="parent\_id" type="int">

<column name="parent\_id" not-null="true" />

</property>

</class>

</hibernate-mapping>