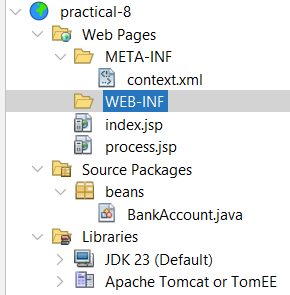
Practical – 8 BankAccount.Java

package beans;



import java.io.Serializable;

public class BankAccount implements Serializable {

private String accountHolder;

private double balance;

public BankAccount() {

this.accountHolder = "Tanmay";

this.balance = 0.0;

}

public String getAccountHolder() {

return accountHolder;

}

public void setAccountHolder(String accountHolder) {

this.accountHolder = accountHolder;

}

public double getBalance() {

return balance;

}

public void deposit(double amount) {

if (amount > 0) {

balance += amount;

}

}

public boolean withdraw(double amount) {

if (amount > 0 && amount <= balance) {

balance -= amount;

return true;

}

return false;

}

}

Index.jsp

<%@ page language="java" import="beans.BankAccount" %>

<jsp:useBean id="account" class="beans.BankAccount" scope="session" />

<jsp:setProperty name="account" property="accountHolder" param="name" />

<html>

<head>

<title>Bank Account</title>

</head>

<body>

<h2>Welcome, <jsp:getProperty name="account" property="accountHolder" /></h2>

<p>Current Balance: <jsp:getProperty name="account" property="balance" /></p>

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

<label>Deposit Amount:</label>

<input type="text" name="depositAmount">

<input type="submit" name="action" value="Deposit">

</form>

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

<label>Withdraw Amount:</label>

<input type="text" name="withdrawAmount">

<input type="submit" name="action" value="Withdraw">

</form>

</body>

</html>

Process.jsp

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

<%@ page language="java" import="beans.BankAccount" %>

<jsp:useBean id="account" class="beans.BankAccount" scope="session" />

<%

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

if (action != null) {

try {

if ("Deposit".equals(action)) {

String depositAmountStr = request.getParameter("depositAmount");

if (depositAmountStr != null && !depositAmountStr.isEmpty()) {

double depositAmount = Double.parseDouble(depositAmountStr);

account.deposit(depositAmount);

out.println("<script>alert('Deposit successful!'); window.location='index.jsp';</script>");

} else {

out.println("<script>alert('Please enter a valid deposit amount.'); window.location='index.jsp';</script>");

}

} else if ("Withdraw".equals(action)) {

String withdrawAmountStr = request.getParameter("withdrawAmount");

if (withdrawAmountStr != null && !withdrawAmountStr.isEmpty()) {

double withdrawAmount = Double.parseDouble(withdrawAmountStr);

if (!account.withdraw(withdrawAmount)) {

out.println("<script>alert('Insufficient Balance!'); window.location='index.jsp';</script>");

} else {

out.println("<script>alert('Withdrawal successful!'); window.location='index.jsp';</script>");

}

} else {

out.println("<script>alert('Please enter a valid withdrawal amount.'); window.location='index.jsp';</script>");

}

}

} catch (NumberFormatException e) {

out.println("<script>alert('Invalid amount format! Please enter a valid number.'); window.location='index.jsp';</script>");

}

} else {

response.sendRedirect("index.jsp");

}

%>