**UNIVERSITY INSTITUTE OF ENGINEERING & TECHNOLOGY**

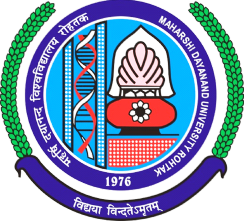
**MAHARASHI DAYANAND UNIVERSITY, ROHTAK**

**PRACTICAL FILE**

ADVANCED JAVA LAB

B.TECH CSE

6TH SEM



**SUBMITTED TO : SUBMITTED BY :**

Mr. DHEERAJ SAHNI PRACHI

ASSISTANT PROFESSOR ROLLNO: 23566

DEPARTMENT OF COMPUTER SCIENCE CSE - 1

MDU, ROHTAK

**Index**

|  |  |  |
| --- | --- | --- |
| S.No. | PROGRAMS | Remarks |
| 1. | Program to print a message using servlet. |  |
| 2. | Simple Calculator using servlet. |  |
| 3. | Program to print the current date and time using JSP. |  |
| 4. | Program to print the specified numbers in words using JSP. |  |
| 5. | Program to print name using Struts. |  |
| 6. | Login form using Struts. |  |
| 7. | Program to send mail using mail API. |  |
| 8. | Program to receive mail using mail API. |  |
| 9. | Java Application using Hibernate. |  |
| 10. | Registration Form using Hibernate. |  |

**1) Program to print a message using servlet.**

**File: simpleHello.java**

import  java.io.\*; import

javax.servlet.\*; import

javax.servlet.http.\*;

public class simpleHello extends HttpServlet{

private String message;

public void init() throws ServletException {

message = "Hello Everyone...!!!"; public void doGet(HttpServletRequest request, HttpServletResponse response)  throws ServletException, IOException {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

out.println("<h1><center>" + message + "<center></h1>");

}public void destroy() { }

}

**File: web.xml**

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

<web-app>

<servlet>

<servlet-name>simpleHello</servlet-name>

<servlet-class>simpleHello</servlet-class>

}

</servlet>

<servlet-mapping>

<servlet-name>simpleHello</servlet-name>

<url-pattern>/simpleHello</url-pattern>

</servlet-mapping>

</web-app>

**Output:**



**2) Simple calculator using servlet.**

**File: index.html**

<html>

<head><title>TODO supply a title</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">  </head>

<body><center>

<h1>My Web Calculator</h1><br><br>

<form action="CalcServlet"

method="POST"> <table>

<tr>

<td>Enter 1st no.</td>

<td><input type="number" name="t1" required="" placeholder="First  no. here" /></td>

</tr>

<tr>

<td>Enter 2nd no.</td>

<td><input type="number" name="t2" required="" placeholder="Second  no. here" /></td>

</tr>

<tr><td colspan="2">

<input type="radio" name="r1" value="+" />+<br>

<input type="radio" name="r1" value="-" />-<br>

<input type="radio" name="r1" value="\*" />\*<br>

<input type="radio" name="r1" value="/" />/<br>

</td></tr>

<tr>

<td></td>

<td><input type="submit" value="SUBMIT"

/></td> </tr>

</table>

</form>

</center></body>

</html>

**File: CalcServlet.java**

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException; import

javax.servlet.http.HttpServlet; import

javax.servlet.http.HttpServletRequest; import

javax.servlet.http.HttpServletResponse;

public class CalcServlet extends HttpServlet {

protected void processRequest(HttpServletRequest request,  HttpServletResponse response)throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

int n1=Integer.parseInt(request.getParameter("t1")); int

n2=Integer.parseInt(request.getParameter("t2"));

String choice = request.getParameter("r1");

PrintWriter out=response.getWriter();

float result=0;

switch(choice){

case "+":

result=n1+n2;

break;

case "-":

result=n1-n2;

break;

case "\*":

result=n1\*n2;

break;

case "/":

result=(float)n1/n2;

break;

}

out.println("<h1><center>Result is :"+result+"</center></h1>"); }

}

**File: web.xml**

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

<web-app>

<servlet>

<servlet-name>CalcServlet</servlet-name>

<servlet-class>CalcServlet</servlet-class>

</servlet>

<servlet-mapping>

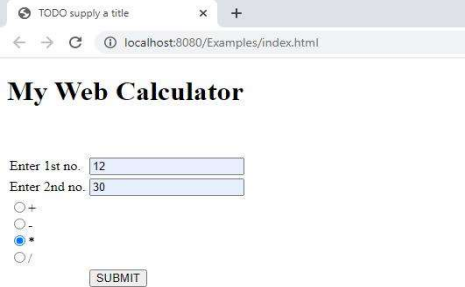
<servlet-name>CalcServlet</servlet-name>

<url-pattern>/CalcServlet</url-pattern>

</servlet-mapping>

</web-app>

**Output:**



**3) Program to print the current date and time using JSP.**

**File: test.jsp**

<%@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>

<form>

<fieldset style="width:20%; background-color: #ccffeb;">

<legend><b><i>JSP Application<i><b></legend>

<h3>Current Date and Time is :</h3>

<% java.util.Date d = new java.util.Date();

out.println(d.toString()); %>

</fieldset>

</form>

</body>

</html>

**Output:**



**4) Program to print the specified numbers in words using JSP.**

**File: form.html**

<html>

<head>

<title>TODO supply a title</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">  </head>

<body>

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

Enter Any Number : <input type=text name=num><br><br>

<input type=submit value="Display">

</form>

</body>

</html>

**File: NumToWords.jsp**

<%@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>

<font style="color:red; font-family: Monotype corsiva; font-size: 40px;">  <%! int i,n;

String s1;

%>

<% s1= request.getParameter("num");

n=s1.length();

i=0;

do

{ char ch=s1.charAt(i);

switch(ch)

{

case '0': out.println("Zero ");

break;

case '1': out.println("One ");

break;

case '2': out.println("Two ");

break;

case '3': out.println("Three ");

break;

case '4': out.println("Four ");

break;

case '5': out.println("Five ");

break;

case '6': out.println("Six ");

break;

case '7': out.println("Seven ");

break;

case '8': out.println("Eight ");

break;

case '9': out.println("Nine ");

break;

}

i++;

}

while(i<n);

%>

</font>

</body>

</html>

**Output:**



**5) Program to print name using Struts.**

**In NetBeans IDE**

Choose File > New project > Java Web > Web Application > Next. Specify a name as: simpleStruts > click next > again click next > select Struts  Framework > Finish.

Right click on the project name > New > JSP > sample.jsp > Finish. File: sample.jsp

<%@ page language="java" %>

<%@ taglib uri="/WEB-INF/struts-html.tld" prefix="html"

%> <html>

<head>

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

</head>

<body>

<html:form action="name">

<table width="200" border="0">

<tr>

<td>Name:</td>

<td><html:text property="name"/></td>

</tr>

<tr>

<td><html:submit /></td>

</tr>

</table>

</html:form>

</body>

</html>

Right click on project name > New > Other > Struts > Struts ActionForm Beam  > Specify Class Name as: nameForm and select package > Finish.

**File: nameForm.java**

package com.myapp.struts;

import javax.servlet.http.HttpServletRequest;

import org.apache.struts.action.ActionErrors;

import org.apache.struts.action.ActionMapping;

import org.apache.struts.action.ActionMessage;

public class nameForm extends org.apache.struts.action.ActionForm {  private String name = null;

public String getName() {

return (name);

}

public void setName(String name) {

this.name = name;

}

public void reset(ActionMapping mapping, HttpServletRequest request) {  this.name = null;

}

}

Right click on the project name > New > JSP > displayName.jsp >  Finish. File: displayName.jsp

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

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  <title>Sample Struts Display Name</title> </head>

<body>

<H1>Hello <%= request.getAttribute("NAME") %> !!</H1>

</body>

</html>

Right click on project name > New > Other > Struts > Struts Action > Specify Class  Name as: nameAction and select package > Finish.

**File: nameAction.java**

package com.myapp.struts;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

public class nameAction extends org.apache.struts.action.Action {

@Override

public ActionForward execute(ActionMapping mapping, ActionForm form,  HttpServletRequest request, HttpServletResponse response)

throws Exception {

String target = new String("success");

String name=null;

if ( form != null ) {

nameForm nameForm = (nameForm)form;

name = nameForm.getName();

}

if ( name == null || name.equals("")) {

target = new String("failure");

}else {

request.setAttribute("NAME", name);

}

return (mapping.findForward(target));

}}

**File: struts-config.xml**

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

<struts-config>

<form-beans>

<form-bean name="nameForm"

type="com.myapp.struts.nameForm"/> </form-beans>

<global-forwards>

<forward name="welcome" path="/Welcome.do"/>

</global-forwards>

<action-mappings>

<action name="nameForm" path="/name"

scope="request" type="com.myapp.struts.nameAction"

validate="false"> <forward name="success"

path="/displayName.jsp"/> <forward name="failure"

path="/sample.jsp"/> </action>

<action path="/Welcome"

forward="/welcomeStruts.jsp"/> </action-mappings>

<controller

processorClass="org.apache.struts.tiles.TilesRequestProcessor"/>

<message-resources parameter="com/myapp/struts/ApplicationResource"/> <plug-in className="org.apache.struts.tiles.TilesPlugin" >

<set-property property="definitions-config" value="/WEB-INF/tiles defs.xml" />

<set-property property="moduleAware" value="true" />

</plug-in>

<plug-in className="org.apache.struts.validator.ValidatorPlugIn">  <set-property

property="pathnames"

value="/WEB-INF/validator-rules.xml,/WEB-INF/validation.xml"/> </plug-in>

</struts-config>

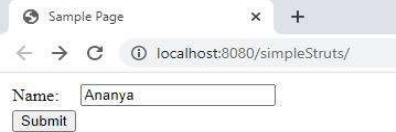
File: index.jsp

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

<%@page pageEncoding="UTF-8"%>

<jsp:forward page="sample.jsp"/>

**Output:**



**6) Login form using Struts.**

In NetBeans IDE

Choose File > New project > Java Web > Web Applications > Next. Specify a name > click next > again click next > select Struts Framework > Finish.

Right click on the project name > New > JSP > login.jsp >

Finish.

**File: login.jsp**

<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean"  %> <%@ taglib uri="http://struts.apache.org/tags-html"

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

pageEncoding="UTF-8"%> <!DOCTYPE html>

<html>

<head>

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

</head>

<body>

<h1>Login Form</h1>

<html:form

action="/login"> <table

border="0"> <tbody>

<tr>

<td colspan="2">

<bean:write name="LoginForm" property="error"

filter="false"/> &nbsp;</td>

</tr>

<tr>

<td>Enter your name:</td>

<td><html:text property="name"

/></td> </tr>

<tr>

<td>Enter your email:</td>

<td><html:text property="email"

/></td> </tr>

<tr>

<td></td>

<td><html:submit value="Login" /></td>

</tr>

</tbody>

</table>

</html:form>

</body>

</html>

Right click on WEB-INF > New > JSP > success.jsp > Finish.

**File: success.jsp**

<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>  <%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

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

</head>

<body>

<h1>Congratulations!</h1>

<p>You have successfully logged in.</p>

<p>Your name is:<bean:write name="LoginForm" property="name" /> .  </p>

<p>Your email address is:<bean:write name="LoginForm"

property="email" /> .</p>

</body>

</html>

Right click on project name > New > Other > Struts > Struts ActionForm Beam  > Specify Class Name as: LoginForm and select package > Finish.

**File: LoginForm.java**

package com.myapp.struts;

import javax.servlet.http.HttpServletRequest;

import org.apache.struts.action.ActionErrors;

import org.apache.struts.action.ActionMapping;

import org.apache.struts.action.ActionMessage;

public class LoginForm extends org.apache.struts.action.ActionForm {

private String name;

private String email;

private String error;

public String getError() {

return error;

}

public void setError(String error) {

this.error = "<span style='color:red'>Please provide valid entries for  both fields</span>";

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getName() {

return name;

}

public void setName(String string) {

name = string;

}

public LoginForm() {

super();

}

public ActionErrors validate(ActionMapping mapping, HttpServletRequest  request) {

ActionErrors errors = new ActionErrors();

if (getName() == null || getName().length() < 1) {

errors.add("name", new ActionMessage("error.name.required"));

}

return errors;

}

}

Right click on project name > New > Other > Struts > Struts Action > Specify Class  Name as: LoginAction and select package > in Action Path: /login > Next > make  appropriate changes > Finish.

**File: LoginAction.java**

package com.myapp.struts;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.ActionForm;

import org.apache.struts.action.ActionForward;

import org.apache.struts.action.ActionMapping;

public class LoginAction extends org.apache.struts.action.Action {  private static final String SUCCESS = "success";

private static final String FAILURE = "failure";

@Override

public ActionForward execute(ActionMapping mapping, ActionForm form,  HttpServletRequest request, HttpServletResponse response) throws  Exception {

// extract user data

LoginForm formBean = (LoginForm) form;

String name = formBean.getName();

String email = formBean.getEmail();

// perform validation

if ((name == null) || // name parameter does not

exist email == null || // email parameter does not

exist name.equals("") || // name parameter is empty

email.indexOf("@") == -1) { // email lacks '@'

formBean.setError("Some error");

return mapping.findForward(FAILURE);

}

return mapping.findForward(SUCCESS);

}

}

**File: struts-config.xml**

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

<!DOCTYPE struts-config PUBLIC

"-//Apache Software Foundation//DTD Struts Configuration 1.3//EN" "http://jakarta.apache.org/struts/dtds/struts-config\_1\_3.dtd">

<struts-config>

<form-beans>

<form-bean name="LoginForm" type="com.myapp.struts.LoginForm"/> </form-beans>

<global-exceptions> </global-exceptions>

<global-forwards>

<forward name="welcome" path="/Welcome.do"/>

</global-forwards>

<action-mappings>

<action name="LoginForm" path="/login" scope="request"

type="com.myapp.struts.LoginAction" validate="false">

<forward name="success" path="/WEB-INF/success.jsp"/>

<forward name="failure" path="/login.jsp"/> </action>

<action path="/Welcome"

forward="/welcomeStruts.jsp"/> </action-mappings>

<controller processorClass="org.apache.struts.tiles.TilesRequestProcessor"/> <message-resources parameter="com/myapp/struts/ApplicationResource"/>

<plug-in className="org.apache.struts.tiles.TilesPlugin" > <set

property property="definitions-config" value="/WEB-INF/tiles

defs.xml" />

<set-property property="moduleAware" value="true"

/> </plug-in>

<plug-in className="org.apache.struts.validator.ValidatorPlugIn">

<set-property

property="pathnames"

value="/WEB-INF/validator-rules.xml,/WEB-INF/validation.xml"/> </plug-in>

</struts-config>

**File: web.xml**

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

<web-app>

<servlet>

<servlet-name>action</servlet-name> <servlet

class>org.apache.struts.action.ActionServlet</servlet-class> <init param>

<param-name>config</param-name> <param

value>/WEB-INF/struts-config.xml</param-value>

</init-param>

<init-param>

<param-name>debug</param-name>

<param-value>2</param-value>

</init-param>

<init-param>

<param-name>detail</param-name>

<param-value>2</param-value>

</init-param>

Aakash 21501

<load-on-startup>2</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>action</servlet-name>

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

</servlet-mapping>

<session-config>

<session-timeout>

30

</session-timeout>

</session-config>

<welcome-file-list>

<welcome-file>login.jsp</welcome-file>

</welcome-file-list>

</web-app>

**Output:**



O Lc•gin Success x -+

~~.i.~~ lOcalhOSt:B0B0/MyStrutsApp/login.do:jsessionid = 75f47038af269043d7a2fb7af7’c1 Congratulations!

You have successfully logged

Your name is.Ananya Ayr .

Your email address is.ananya ,gmail.coin .

Lc•g in Form

https://lh4.googleusercontent.com/dfrL07jGn9RerYDjMuUXAjd9nNNUF-e4SkTAhV1mMpVkZJHYjNBj_wLgGKkh3tAYxABLhMq_3c7Y20vln8SJ0SxLqJUxmevoXV1qkSv7bRwwnIQVAJbE2dH2KuyVWQQ7oP0HT1SFYb9bpQqFnw

localhost:B0B0/MyStrutsApp/

https://lh6.googleusercontent.com/hsziw8jQ1rxNeoKpMmdtOtqNdeRup40isxBXMsxoWg5xUfMl6_9ZQtek65Q8fSSQmA9a2liwuwfDCnT9epxZGAMnjwIhzRIz2-oEKX1Rg-cLUaiqQblmrh9LkwCr5aCG3-hoCm5YDXFBkIRBSQnazi: 

https://lh5.googleusercontent.com/NRs4O_KEXa8gBUNw77a_rIQwBivG56yCY_P46Qx9RQ4-ulxFU8tigYrTCVnxwZPCx-oqF1Qgivd6iey4ZR05Iqmr3M9hDH24bx6ZRQq0qdD-QikX-R-2f8sSZusvRM6hpwAGUtovM5-No14GkA

O Lc•gin Form

Lai

https://lh6.googleusercontent.com/3mV76s_nAPL5K0sTmeeqcajk6W9fQqNJnGhAOkOnP3jRvxK0R6Up6spham_7115fPsHB0LRaOL8BDEHwgDbEwtOPiKv5y9vi6bSZ7YufPUpl0jqNKzufQGYLFAvm8TnBjbeeeh1bNGzv8k9nXw

@ localhost:8080/MyStrutsApp/login.do:jsessionid =75f47038af269043d7a2fb7af7c1 

**Login Form**

Please proc ide slid entries far both

6etds Water youz oazae: Nis

Water youz ezaad. dj j liegh Login

**7) Program to send mail using mail API.**

Libraries > Add JAR/Folder > Import

mail\_1.4.jar File: sendEmail.java

import java.util.Properties;

import javax.mail.\*; import

javax.mail.internet.\*;

class Mailer{

public static void send(String from,String password,String to,String sub,String msg){

Properties props = new Properties();

props.put("mail.smtp.host", "smtp.gmail.com");

props.put("mail.smtp.socketFactory.port", "465");

props.put("mail.smtp.socketFactory.class",

"javax.net.ssl.SSLSocketFactory");

props.put("mail.smtp.auth", "true");

props.put("mail.smtp.port", "465");

Session session = Session.getDefaultInstance(props,

new javax.mail.Authenticator() {

protected PasswordAuthentication getPasswordAuthentication()

{ return new PasswordAuthentication(from,password); }

});

try {

MimeMessage message = new MimeMessage(session);

message.addRecipient(Message.RecipientType.TO,new InternetAddress(to)); message.setSubject(sub);

message.setText(msg);

Transport.send(message);

System.out.println("message sent successfully");

} catch (MessagingException e) {throw new RuntimeException(e);} }

}

public class sendEmail {

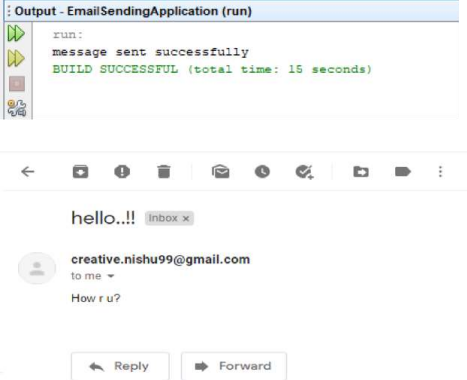
public static void main(String[] args) {

Mailer.send("creative.nishu99@gmail.com","987654nishu123","creative.nishu99@gmail .com","hello..!!","How r u?");

}

}

**Output:**



**8) Program to receive mail using mail API.**

Libraries > Add JAR/Folder > Import

mail\_1.4.jar

**File: receiveMail.java**

import java.util.\*;

import javax.mail.\*;

public class receiveEmail {

public static void receiveEmail(String pop3Host, String storeType,

String user, String password){

Properties props = new Properties();

props.put("mail.pop3.host", pop3Host);

props.put("mail.pop3.port", "995");

props.put("mail.pop3.starttls.enable", "true");

props.put("mail.store.protocol", "pop3");

Session session = Session.getInstance(props);

try {

Store mailStore = session.getStore(storeType);

mailStore.connect(pop3Host, user, password);

Folder folder = mailStore.getFolder("INBOX");

folder.open(Folder.READ\_ONLY);

Message[] emailMessages = folder.getMessages();

System.out.println("Total Message - "

+ emailMessages.length);

//Iterate the messages

for (int i = 0; i < emailMessages.length; i++) {

Message message = emailMessages[i];

Address[] toAddress =

message.getRecipients(Message.RecipientType.TO);

System.out.println();

System.out.println("Email " + (i+1) + "-");

System.out.println("Subject - " + message.getSubject());

System.out.println("From - " + message.getFrom()[0]);

System.out.println("To - ");

for(int j = 0; j < toAddress.length; j++){

System.out.println(toAddress[j].toString());

}

System.out.println("Text - " +

message.getContent().toString());

}

folder.close(false);

mailStore.close();

}catch (Exception e) {

e.printStackTrace();

System.err.println("Error in receiving email.");

}}

public static void main(String[] args) {

String pop3Host = "pop.gmail.com";

String mailStoreType = "pop3s";

final String userName = "creative.nishu99@gmail.com";

final String password = "987654nishu123";

//call receiveEmail

receiveEmail(pop3Host, mailStoreType, userName, password);

}}

**Output:**



**9) Java Application using Hibernate.**

**File: Employee.java**

public class Employee

{ private int id;

private String firstName;

private String lastName;

private int salary;

public Employee() {}

public Employee(String fname, String lname, int salary)

{ this.firstName = fname;

this.lastName = lname;

this.salary = salary;

}

public int getId() { return id; }

public void setId( int id ) { this.id = id; }

public String getFirstName() { return firstName; }

public void setFirstName( String first\_name ) { this.firstName =  first\_name; }

public String getLastName() { return lastName; }

public void setLastName( String last\_name ) { this.lastName =  last\_name; }

public int getSalary() { return salary; }

public void setSalary( int salary ) { this.salary = salary; }

}

Create a database table:

create table EMPLOYEE (

id INT NOT NULL auto\_increment,

first\_name VARCHAR(20) default NULL,

last\_name VARCHAR(20) default NULL,

salary INT default NULL, PRIMARY

KEY (id)

);

**File: Employee.hbm.xml**

<?xml version = "1.0" encoding = "utf-8"?> <!DOCTYPE

hibernate-mapping PUBLIC "-//Hibernate/Hibernate

Mapping DTD//EN"

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

<hibernate-mapping>

<class name = "Employee" table = "EMPLOYEE">

<meta attribute = "class-description">

This class contains the employee detail.

</meta>

<id name = "id" type = "int" column =

"id"> <generator class="native"/>

</id>

<property name = "firstName" column = "first\_name" type = "string"/>  <property name = "lastName" column = "last\_name" type = "string"/>  <property name = "salary" column = "salary" type = "int"/> </class>

</hibernate-mapping>

**File: ManageEmployee.java**

import java.util.List;

import java.util.Date;

import java.util.Iterator;

import org.hibernate.HibernateException;

import org.hibernate.Session;

import org.hibernate.Transaction;

import org.hibernate.SessionFactory;

import org.hibernate.cfg.Configuration;

public class ManageEmployee {

private static SessionFactory factory;

public static void main(String[] args) {

try {

factory = new Configuration().configure().buildSessionFactory();  } catch (Throwable ex) {

System.err.println("Failed to create sessionFactory object." +

ex); throw new ExceptionInInitializerError(ex);

}

ManageEmployee ME = new ManageEmployee();

/\* Add few employee records in database \*/

Integer empID1 = ME.addEmployee("Zara", "Ali", 1000);

Integer empID2 = ME.addEmployee("Daisy", "Das", 5000);

Integer empID3 = ME.addEmployee("John", "Paul", 10000);

/\* List down all the employees \*/ ME.listEmployees();

/\* Update employee's records \*/

ME.updateEmployee(empID1, 5000);

/\* Delete an employee from the database \*/

ME.deleteEmployee(empID2);

/\* List down new list of the employees \*/

ME.listEmployees();

}

/\* Method to CREATE an employee in the database \*/

public Integer addEmployee(String fname, String lname, int salary){  Session session = factory.openSession(); Transaction tx = null;

Integer employeeID = null;

try {

tx = session.beginTransaction();

Employee employee = new Employee(fname, lname, salary);

employeeID = (Integer) session.save(employee);

tx.commit();

} catch (HibernateException e) {

if (tx!=null) tx.rollback();

e.printStackTrace();

} finally {

session.close();

}return employeeID;

}

/\* Method to READ all the employees \*/

public void listEmployees( ){

Session session = factory.openSession();

Transaction tx = null;

try {

tx = session.beginTransaction();

List employees = session.createQuery("FROM Employee").list();  for (Iterator iterator = employees.iterator(); iterator.hasNext();){  Employee employee = (Employee) iterator.next();  System.out.print("First Name: " + employee.getFirstName());  System.out.print(" Last Name: " + employee.getLastName());  System.out.println(" Salary: " + employee.getSalary());  }tx.commit();

} catch (HibernateException e) {

if (tx!=null) tx.rollback();

e.printStackTrace();

} finally {

session.close();

}}

/\* Method to UPDATE salary for an employee \*/

public void updateEmployee(Integer EmployeeID, int salary ){  Session session = factory.openSession(); Transaction tx = null;

try {

tx = session.beginTransaction();

Employee employee = (Employee)session.get(Employee.class, EmployeeID);

employee.setSalary( salary );

session.update(employee);

tx.commit();

} catch (HibernateException e) {

if (tx!=null) tx.rollback();

e.printStackTrace();

} finally {

session.close();

}}

/\* Method to DELETE an employee from the records \*/

public void deleteEmployee(Integer EmployeeID){

Session session = factory.openSession();

Transaction tx = null;

try {

tx = session.beginTransaction();

Employee employee =

(Employee)session.get(Employee.class, EmployeeID);

session.delete(employee);

tx.commit();

} catch (HibernateException e) {

if (tx!=null) tx.rollback();

e.printStackTrace();

} finally {

session.close();

}}

}

**Output:**

$java ManageEmployee

.......VARIOUS LOG MESSAGES WILL DISPLAY HERE........

First Name: Zara Last Name: Ali Salary: 1000

First Name: Daisy Last Name: Das Salary: 5000

First Name: John Last Name: Paul Salary: 10000

First Name: Zara Last Name: Ali Salary: 5000

First Name: John Last Name: Paul Salary: 10000

If you check your EMPLOYEE table, it should have the following records −

mysql> select \* from EMPLOYEE;

+ -+ -+- +--- -+

| id | first\_name | last\_name | salary |

+ -+ -+- +--- -+

| 29 | Zara | Ali | 5000 |

| 31 | John | Paul | 10000 |

+ -+ -+- +- --+

2 rows in set (0.00 sec)

mysql>

**10) Registration form using Hibernate.**

**File: index.jsp**

<form action="register.jsp" method="post"> Name:<input

type="text" name="name"/><br><br/> Password:<input

type="password" name="password"/><br><br/> Email ID:<input  type="text" name="email"/><br><br/>

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

</form>

**File: register.jsp**

<%@page import="com.mypack.UserDao"%>

<jsp:useBean id="obj" class="com.mypack.User">

</jsp:useBean>

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

<%

int i=userDao.register(obj);

if(i>0)

out.print("You are successfully

registered"); %>

**File: user.java**

package com.javatpoint.mypack;

public class user {

private int id;

private String name,password,email;

//getters and setters

}

**File: user.hbm.xml**

<?xml version='1.0' encoding='UTF-8'?> <!DOCTYPE

hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping

DTD 5.3//EN" "http://hibernate.sourceforge.net/hibernate

mapping-5.3.dtd">

<hibernate-mapping>

<class name="com.mypack.user" table="u400">

<id name="id"><generator class="increment"></generator></id>  <property name="name"></property> <property

name="password"></property>

<property name="email"></property>

</class>

</hibernate-mapping>

**File: userDao.java**

package com.mypack;

import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.Transaction;

import org.hibernate.boot.Metadata;

import org.hibernate.boot.MetadataSources;

import org.hibernate.boot.registry.StandardServiceRegistry; import

org.hibernate.boot.registry.StandardServiceRegistryBuilder;

public class userDao {

public static int register(user u){

int i=0;

StandardServiceRegistry ssr = new StandardServiceRegistryBuilder().configure("h  ibernate.cfg.xml").build();

Metadata meta = new MetadataSources(ssr).getMetadataBuilder().build(); SessionFactory factory = meta.getSessionFactoryBuilder().build();

Session session = factory.openSession();

Transaction t = session.beginTransaction();

i=(Integer)session.save(u);

t.commit();

session.close();

return i;

} }

**File: hibernate.cfg.xml**

<?xml version='1.0' encoding='UTF-8'?>

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD 5.3//EN"

"http://hibernate.sourceforge.net/hibernate-configuration-5.3.dtd">

<hibernate-configuration>

<session-factory>

<property name="hbm2ddl.auto">create</property>

<property name="dialect">org.hibernate.dialect.Oracle9Dialect</property>  <property name="connection.url">jdbc:oracle:thin:@localhost:1521:xe</pr  operty>

<property name="connection.username">system</property>

<property name="connection.password">jtp</property>

<property name="connection.driver\_class">oracle.jdbc.driver.OracleDriver  </property> 

<mapping resource="user.hbm.xml"/>

</session-factory>

</hibernate-configuration>