

Assignment-Hibernate

ตาราง Account

1.หน้าจอ

The screenshot shows a web browser window with the address bar displaying 'localhost:8080/AccountHiber/ListController'. The page title is 'Account List'. The main content area has a light blue background and contains a table with 6 columns: Account ID, Name, Username, Password, Email, and Tel. The table lists 10 accounts. Below the table, there are three links: 'Add new Account', 'Search By ID', and 'Search By Name'.

Account ID	Name	Username	Password	Email	Tel
1	Filmm Tanawat	Film	12345	film@gmail.com	0833919391
2	Da Vachira	Da	12345	Da@gmail.com	0811111111
3	Fang Sutita	Fang	12345	Fang@gmail.com	0822222222
4	Pan Panyapard	Pan	12345	Pan@gmail.com	0833333333
5	Pop Tunyared	Pop	12345	Pop@gmail.com	0844444444
6	Gigi H	Gigi	12345	Gi@gmail.com	0465725224
7	Kendall Jenner	Kendall	12345	K@gmail.com	0986784567
8	Kylie Jenner	Kylie	12345	Kylie@gmail.com	0986784567
9	June K	June	12345	June@hotmail.com	0758093756
10	Jame K	Jame	122345	J@gmail.com	0967483566

[Add new Account](#) [Search By ID](#) [Search By Name](#)

แสดงรายการข้อมูลในตาราง

The screenshot shows a web browser window with the address bar displaying 'localhost:8080/AccountHiber/AccountShow.jsp'. The page title is 'Tain Account'. The main content area has a light blue background and contains a form titled 'Search Account By ID'. The form has a label 'Enter Account ID:' followed by a text input field and a 'Find Account Details' button. Below the form, there are two links: 'Search By Name' and 'Account List'.

Tain Account

Search Account By ID

Enter Account ID:

[Search By Name](#) [Account List](#)

ฟอร์มหน้าค้นหาด้วยID

localhost:8080/AccountHiber/SearchController?searchValue=2&searchButton=Find+Account+Details

Tain Account

Search Account By ID

Enter Account ID

Account ID	2
Name	Da Vachira
Username	Da
Password	12345

[Search By Name](#) [Account List](#)

ผลการค้นหาด้วยID

localhost:8080/AccountHiber/searchName.jsp

Account List

Search Account By Name

Enter Account By Name:

Account ID	Name	Username
1	Filmm Tanawat	Film
2	Da Vachira	Da
3	Fang Sutita	Fang
4	Pan Panyapard	Pan
5	Pop Tunyared	Pop
6	Gigi H	Gigi
7	Kendall Jenner	Kendall
8	Kylie Jenner	Kylie
9	June.K	June
10	Jame.K	Jame

[Search By ID](#) [Account List](#)

ฟอร์มหน้าค้นหาด้วยชื่อ

localhost:8080/AccountHiber/SearchNameController?searchValue=Da&searchButton=Find+Account+Details

Account List

Search Account By Name

Enter Account By Name:

Account ID	Name	Username
2	Da Vachira	Da
7	Kendall Jenner	Kendall

[Search By ID](#) [Account List](#)

ผลการค้นหาด้วยชื่อ

localhost:8080/AccountHiber/addAcc.jsp

Train Account

Add New Account

AccountID:

Name:

Tel:

Email:

Username:

Password:

Account Search

Enter Account ID:

[Account List](#)

ฟอร์มหน้าเพิ่มข้อมูล

localhost:8080/AccountHiber/addAcc.jsp

Train Account

Add New Account

AccountID:

Name:

Tel:

Email:

Username:

Password:

Account Search

Enter Account ID:

[Account List](#)

เพิ่มข้อมูล

localhost:8080/AccountHiber/SearchController?searchValue=11&searchButton=Find+Account+Details

Tain Account

Search Account By ID

Enter Account ID:

Account ID	11
Name	Somsri.G
Username	SomSri
Password	2345

[Search By Name](#) [Account List](#)

เมื่อเพิ่มข้อมูลแล้วก็สามารถค้นหาข้อมูลที่เพิ่มไว้ได้

1. โครงสร้างตาราง + ตัวอย่างข้อมูลในตาราง

ตาราง Account

ActId	Tel	Email	Username	Name	Password
1	0833919391	film@gmail.com	Film	Filmm Tanawat	12345
2	0811111111	Da@gmail.com	Da	Da Vachira	12345
3	0822222222	Fang@gmail.com	Fang	Fang Sutita	12345
4	0833333333	Pan@gmail.com	Pan	Pan Panyapard	12345
5	0844444444	Pop@gmail.com	Pop	Pop Tunyared	12345

2. ไฟล์ที่เป็น model ของตาราง (ชื่อตาราง.java)

ไฟล์ Account.java

```
package tAccount;
// Generated Oct 30, 2016 9:57:44 PM by Hibernate Tools 4.3.1

/**
 * Account generated by hbm2java
 */
public class Account implements java.io.Serializable {

    private int actId;
    private String username;
    private String password;
    private String name;
    private String tel;
    private String email;

    public Account() {
    }

    public Account(int actId, String username, String password, String name, String tel) {
        this.actId = actId;
        this.username = username;
        this.password = password;
        this.name = name;
        this.tel = tel;
    }

    public Account(int actId, String username, String password, String name, String tel, String email) {
```

```
        this.actId = actId;
        this.username = username;
        this.password = password;
        this.name = name;
        this.tel = tel;
        this.email = email;
    }

    public int getActId() {
        return this.actId;
    }

    public void setActId(int actId) {
        this.actId = actId;
    }
    public String getUsername() {
        return this.username;
    }
    }

    public void setUsername(String username) {
        this.username = username;
    }
    public String getPassword() {
        return this.password;
    }
    }

    public void setPassword(String password) {
        this.password = password;
    }
    public String getName() {
        return this.name;
    }
    }

    public void setName(String name) {
        this.name = name;
    }
    public String getTel() {
        return this.tel;
    }
    }

    public void setTel(String tel) {
        this.tel = tel;
    }
    public String getEmail() {
        return this.email;
    }
    }

    public void setEmail(String email) {
        this.email = email;
    }
    }
}
```

3. code ไฟล์ Hibernate.cfg.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
  <session-factory>
    <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
    <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
    <property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/trainticket?zeroDateTimeBehavior=convertTo
Null</property>
    <property name="hibernate.connection.username">root</property>
    <property name="hibernate.show_sql">true</property>
    <property name="hibernate.current_session_context_class">thread</property>
    <property
name="hibernate.query.factory_class">org.hibernate.hql.internal.classic.ClassicQueryTranslatorFactory</prop
erty>
    <mapping resource="tAccount/Payment.hbm.xml"/>
    <mapping resource="tAccount/Train.hbm.xml"/>
    <mapping resource="tAccount/Timetrain.hbm.xml"/>
    <mapping resource="tAccount/Account.hbm.xml"/>
    <mapping resource="tAccount/Ticket.hbm.xml"/>
  </session-factory>
</hibernate-configuration>
```

4. code ไฟล์ Servlet ทั้งหมด

ไฟล์ **AccountService.java**

ทำหน้าที่จัดการ **Business Logic** สร้าง **Method** การแสดงข้อมูล ค้นหาข้อมูลและเพิ่มข้อมูล

```
package tAccount;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.List;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;
```

```

import javax.servlet.http.HttpServletResponse;

import org.hibernate.Query;

import org.hibernate.Session;

import org.hibernate.Transaction;

/**
 *
 * @author Home
 */

public class AccountService extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */

    public List getAccount(int startID, int endID) {

        List actList = null;

        Session session = null;

        try {

            session = HibernateUtil.getSessionFactory().openSession();

            org.hibernate.Transaction tx = session.beginTransaction();

            Query q = session.createQuery("from Account where ActId between '" + startID + "' and '"
                + endID + "'");

            actList = (List<Account>) q.list();

```



```

    } catch (Exception e) {

        e.printStackTrace();

    }

    return actList;

}

    public Account searchAct(short search_ActID)
{
    String message=null;

    Session session=null;

    Transaction tx = null;

    Account act = null;

    try {

        session=HibernateUtil.getSessionFactory().openSession();

        tx = session.getTransaction();

        tx.begin();

        Query query = session.createQuery("from Account where ActId =" +search_ActID);

        act = (Account)query.uniqueResult();

        tx.commit();

    } catch (Exception e) {

        if (tx != null) {

            tx.rollback();

        }

        e.printStackTrace();

    } finally {

        session.close();

    }

    return act;

}

```

```

public String addAcc(Account ActId) {

    String message = null;

    Session session = null;

    try {

        session

            = HibernateUtil.getSessionFactory().openSession();

        Transaction tx = session.getTransaction();

        tx.begin();

        session.save(ActId);

        tx.commit();

        message = "Saved Data";

        return message;

    } catch (Exception e) {

        return (e.getMessage());

    } finally {

        session.close();

    }

}

```

```

public List searchAccByName(String searchKeyword) {

    List actList = null;

    Session session = null;

    try {

        session

            = HibernateUtil.getSessionFactory().openSession();

        org.hibernate.Transaction tx

            = session.beginTransaction();

        Query query = session.createQuery(

```

```

        "from Account as account where account.name like :keyword ");

query.setParameter("keyword", "%" + searchKeyword + "%");

        actList = (List<Account>) query.list();

    } catch (Exception e) {

        e.printStackTrace();

    }

    return actList;

}

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

    throws ServletException, IOException {

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

    try (PrintWriter out = response.getWriter()) {

        /* TODO output your page here. You may use following sample code. */

        out.println("<!DOCTYPE html>");

        out.println("<html>");

        out.println("<head>");

        out.println("<title>Servlet AccountService</title>");

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

        out.println("<body>");

        out.println("<h1>Servlet AccountService at " + request.getContextPath() + "</h1>");

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

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

    }

}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/**

* Handles the HTTP `GET` method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

* Handles the HTTP `POST` method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

```

    * Returns a short description of the servlet.

    *

    * @return a String containing servlet description

    */

    @Override

    public String getServletInfo() {

        return "Short description";

    } // </editor-fold>

}

```

ไฟล์ **ListController.java**

ทำหน้าที่เป็น Controller ให้ Method `getAccount` ในการแสดงข้อมูลในหน้าเว็บทั้งหมด

```

package tAccount;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.List;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

/**

 *

 * @author Home

 */

```

```

public class ListController extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */

    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            HttpSession session = request.getSession(true);

            try {

                AccountService acs = new AccountService();

                List ActList = acs.getAccount(1, 100);

                // int num = filmList.size();

                session.setAttribute("listResult", ActList);

            } catch (Exception e) {

                e.printStackTrace();
            }
        }
    }
}

```

```
}
```

```
RequestDispatcher rd = request.getRequestDispatcher("actlist.jsp");
```

```
rd.forward(request, response);
```

```
}
```

```
}
```

```
// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">
```

```
/**
```

```
 * Handles the HTTP <code>GET</code> method.
```

```
 *
```

```
 * @param request servlet request
```

```
 * @param response servlet response
```

```
 * @throws ServletException if a servlet-specific error occurs
```

```
 * @throws IOException if an I/O error occurs
```

```
 */
```

```
@Override
```

```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
```

```
    throws ServletException, IOException {
```

```
    processRequest(request, response);
```

```
}
```

```
/**
```

```
 * Handles the HTTP <code>POST</code> method.
```

```
 *
```

```
 * @param request servlet request
```

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

* Returns a short description of the servlet.

*

* @return a String containing servlet description

*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

ไฟล์ SearchController.java

ทำหน้าที่เป็น Controller ให้ Method searchAct ในการค้นหาข้อมูล โดยID

```
package tAccount;

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

/**
 *
 * @author Home
 */
public class SearchController extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
}
```

```
*/
```

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {
        HttpSession session = request.getSession(true);

        try {
            short search_ActID = Short.parseShort(request.getParameter("searchValue"));

            AccountService acs = new AccountService();
            Account act_obj = acs.searchAct(search_ActID);

            session.setAttribute("searchResult", act_obj);

            RequestDispatcher rd = request.getRequestDispatcher("AccountShow.jsp");
            rd.forward(request, response);

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

```
// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit
the code.">
```

```
/**
```

* Handles the HTTP `GET` method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

* Handles the HTTP `POST` method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

```

    * Returns a short description of the servlet.

    *

    * @return a String containing servlet description

    */

@Override

public String getServletInfo() {

    return "Short description";

} // </editor-fold>

}

```

ไฟล์ **SearchNameController.java**

ทำหน้าที่เป็น Controller ให้ Method searchAccByName ในการค้นหาข้อมูลด้วยชื่อ

```

package tAccount;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.List;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

/**

 *

 * @author Home

 */

```

```

public class SearchNameController extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */

    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            // Create a session object if it is already not created.

            HttpSession session = request.getSession(true);

            try {
                String searchValue

                = request.getParameter("searchValue");

                AccountService acs = new AccountService();

                List actList

                = acs.searchAccByName(searchValue);

                int num = actList.size();

                session.setAttribute("listResult",actList);
            }
        }
    }
}

```

```

    } catch (Exception e) {

        e.printStackTrace();

    }

    RequestDispatcher rd

        = request.getRequestDispatcher("searchName.jsp");

    rd.forward(request, response);

}

}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```
/**
```

```
 * Handles the HTTP <code>GET</code> method.
```

```
 *
```

```
 * @param request servlet request
```

```
 * @param response servlet response
```

```
 * @throws ServletException if a servlet-specific error occurs
```

```
 * @throws IOException if an I/O error occurs
```

```
 */
```

```
@Override
```

```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
```

```
    throws ServletException, IOException {
```

```
        processRequest(request, response);
```

```
    }
```

```
/**
```

```
 * Handles the HTTP <code>POST</code> method.
```

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

* Returns a short description of the servlet.

*

* @return a String containing servlet description

*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

ไฟล์ AddAccController.java

ทำหน้าที่เป็น Controller ให้ Method addAcc ในการเพิ่มข้อมูลในดาต้าเบส

```
package tAccount;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.Date;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

/**
 *
 * @author Home
 */
public class AddAccController extends HttpServlet {

    /**
     * Processes requests for both HTTP GET and POST
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            out.println("add new account");
            HttpSession session = request.getSession(true);

            int ActId = Integer.parseInt(request.getParameter("ActId"));
            String Username = request.getParameter("Username");
            String Password = request.getParameter("Password");
            String Name = request.getParameter("Name");
            String Tel = request.getParameter("Tel");
            String Email = request.getParameter("Email");

            Date now = new Date();
            Account ac = new Account(ActId, Username, Password, Name, Tel, Email);
```



```

        //ac.setFirstName(firstname);
        //ac.setLastName(lastname);

        AccountService acs = new AccountService();
        String resultMessage = acs.addAcc(ac);

        RequestDispatcher rd
            = request.getRequestDispatcher("AccountShow.jsp?message=" + resultMessage
        );
        request.setAttribute("searchResult", null);
        rd.forward(request, response);
    }
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```

/**
 * Handles the HTTP <code>GET</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

```

```

/**
 * Handles the HTTP <code>POST</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

```

```

/**
 * Returns a short description of the servlet.
 *
 * @return a String containing servlet description
 */
@Override

```

```
public String getServletInfo() {  
    return "Short description";  
} // </editor-fold>
```

```
}
```

6. code ไฟล์ JSP ทั้งหมด

ไฟล์ **actlist.jsp**

ทำหน้าที่เป็น View หรือหน้าฟอร์มที่ใช้แสดงผลของ ListController

```
<%@page import="java.util.List"%>  
  
<%@page import="tAccount.Account"%>  
  
<%@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>  
  
        <center>  
  
            <h1>Account List</h1>  
  
            <div style="background-color:#CC99FF; width:100%;"><br>  
  
            <%  
  
                try {  
  
                    List<Account> actList = (List<Account>) session.getAttribute("listResult");  
  
                    if (actList != null) {  
  
                        // out.println("not null");
```

```
// out.println(filmList.size());
```

```
%>
```

```
<table border="1">
```

```
<tbody>
```

```
<tr>
```

```
<td>Account ID</td>
```

```
<td>Name</td>
```

```
<td>Username</td>
```

```
<td>Password</td>
```

```
<td>Email</td>
```

```
<td>Tel</td>
```

```
</tr>
```

```
<%           for (Account acc : actList) {
```

```
%>
```

```
<tr>
```

```
<td><% out.println(acc.getActId()); %></td>
```

```
<td><% out.println(acc.getName()); %></td>
```

```
<td><% out.println(acc.getUsername()); %></td>
```

```
<td><% out.println(acc.getPassword()); %></td>
```

```
<td><% out.println(acc.getEmail()); %></td>
```

```
<td><% out.println(acc.getTel()); %></td>
```

```
</tr>
```

```
<% } %>
```

```
</tbody>
```

```
</table>
```

```
<%
```



```

</head>

<body>

<center>

    <h1>Tain Account </h1>

    <div style="background-color: #33FFFF; width:100%;">

        <br />

        <h2>Search Account By ID</h2>

        <form name="searchForm" action="SearchController">

            Enter Account ID:<input type="text" name="searchValue" value="" >

            <input type="submit" value="Find Account Details" name="searchButton" /><br><br>

        </form>

        <%

            try {

                // Create a session object if it is already not created.

                Account acc = (Account) session.getAttribute("searchResult");

                if (acc != null) {

                    %>

                    <table border="1">

                        <tbody>

                            <tr>

                                <td>Account ID</td>

                                <td><% out.println(acc.getActId()); %></td>

                            </tr>

                            <tr>

                                <td>Name</td>

                                <td><% out.println(acc.getName()); %></td>

                            </tr>

                            <tr>

```

[illegible]

ไฟล์ SearchName.jsp

ทำหน้าที่เป็น View หรือหน้าฟอร์มที่ใช้แสดงผลของ SearchNameController.java (การค้นหาคำด้วยชื่อ)

```
<%@page import="tAccount.Account"%>

<%@page import="java.util.List"%>

<%@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>

        <center>

            <h1>Account List</h1>

            <div style="background-color: #FFFF66; width:100%;">

                <br />

                <h2>Search Account By Name</h2>

                <form name="searchForm" action="SearchNameController">

                    Enter Account By Name:<input type="text" name="searchValue" value="" >

                    <input type="submit" value="Find Account Details" name="searchButton" /><br><br>

                </form>

            <%

                try {

                    List<Account> actList = (List<Account>) session.getAttribute("listResult");

                    if (actList != null) {

                        //out.println("not null");
```

```
//out.println(actList.size());
```

```
%>
```

```
<table border="1">
```

```
<tbody>
```

```
<tr>
```

```
<td>Account ID</td>
```

```
<td>Name</td>
```

```
<td>Username</td>
```

```
</tr>
```

```
<% for (Account acc
```

```
: actList) {
```

```
%>
```

```
<tr>
```

```
<td><% out.println(acc.getActId()); %></td>
```

```
<td><% out.println(acc.getName()); %></td>
```

```
<td><% out.println(acc.getUsername()); %></td>
```

```
</tr>
```

```
<% } %>
```

```
</tbody>
```

```
</table>
```

```
<%
```

```
} else {
```

```
out.println("return null");
```

```
}
```

```
} catch (Exception e) {
```

```
e.printStackTrace();
```

```
out.println("error");
```


Add New Account

```
value="" /><br><br>
```

```
value="" /><br><br>
```

```
value="" /><br><br>
```

```
value="" /><br><br>
```

```
value="" /><br><br>
```

```
value="" /><br><br>
```

```
name="AddAcc" />
```


Account Search


```
try {
```

```
// Create a session object if it is already not created.
```

```
Account acc = (Account) session.getAttribute("searchResult");
```

```
if (acc != null) {
```

```
%>
```

```
<%
```

```
}
```

```
} catch (Exception e) {
```

```
out.println(e.getMessage());
```

```
}
```

```
%>
```

```
<br><br>
```

```
<a href="ListController">Account List</a> <br><br>
```

```
<br><br>
```

```
</center>
```

```
</body>
```

```
</html>
```