

우편번호 검색기능 만들기

1.테이블 생성

```
CREATE TABLE zipcode (
zipcode          char(7)          NOT NULL ,
area1            char(10)         NULL    ,
area2            char(20)         NULL    ,
area3            char(40)         NULL    ,
area4            char(20)         NULL
);
```

2.테이블 수정

```
ALTER TABLE member
add (zipcode varchar2(7), address varchar2(60));
```

3.ZipcodeBean 생성

```
package ch11.logon;
```

```
public class ZipcodeBean {
```

```
private String zipcode;
private String area1;
private String area2;
private String area3;
private String area4;
```

```
public void setZipcode(String zipcode){
this.zipcode=zipcode;
}
```

```
public void setArea1(String area1){
this.area1=area1;
}
```

```
public void setArea2(String area2){
this.area2=area2;
}
```

```
public void setArea3(String area3){
this.area3=area3;
}
```

```
public void setArea4(String area4){
this.area4=area4;
}
```

```
public String getZipcode(){
return zipcode;
}
```

```
public String getArea1(){
return area1;
}
```

```
public String getArea2(){
return area2;
}
```

```
public String getArea3(){
return area3;
}
```

```

}
public String getArea4(){
return area4;
}
}

```

7.LogonDataBean 수정

```

private String zipcode;
private String address;

```

```

public String getZipcode() {
return zipcode;
}
public void setZipcode(String zipcode) {
this.zipcode = zipcode;
}

```

8.LogonDBBean에 zipcodeRead()추가 및 insert()/getMember()수정

```

public void insertMember(LogonDataBean member) throws Exception {
    Connection conn = null;
    PreparedStatement pstmt = null;

    try {
        conn = getConnection();

        pstmt = conn.prepareStatement(
            "insert into MEMBER values (?,?,?,?,?,?,?,?,?)");
        pstmt.setString(1, member.getId());
        pstmt.setString(2, member.getPasswd());
        pstmt.setString(3, member.getName());
        pstmt.setString(4, member.getJumin1());
        pstmt.setString(5, member.getJumin2());
        pstmt.setString(6, member.getEmail());
        pstmt.setString(7, member.getBlog());
        pstmt.setTimestamp(8, member.getReg_date());
        pstmt.setString(9, member.getZipcode());
        pstmt.setString(10, member.getAddress());
        pstmt.executeUpdate();
    } catch (Exception ex) {
        ex.printStackTrace();
    } finally {
        if (pstmt != null) try { pstmt.close(); } catch (SQLException ex) {}
        if (conn != null) try { conn.close(); } catch (SQLException ex) {}
    }
}

```

public LogonDataBean getMember(String id) throws Exception {

```

    Connection conn = null;
    PreparedStatement pstmt = null;
    ResultSet rs = null;
    LogonDataBean member=null;
    try {
        conn = getConnection();

        pstmt = conn.prepareStatement(
            "select * from MEMBER where id = ?");
        pstmt.setString(1, id);
        rs = pstmt.executeQuery();

```

```

        if (rs.next()) {
            member = new LogonDataBean();
            member.setId(rs.getString("id"));
            member.setPasswd(rs.getString("passwd"));
            member.setName(rs.getString("name"));
            member.setJumin1(rs.getString("jumin1"));
            member.setJumin2(rs.getString("jumin2"));
            member.setEmail(rs.getString("email"));
            member.setBlog(rs.getString("blog"));
            member.setReg_date(rs.getTimestamp("reg_date"));
            member.setZipcode(rs.getString("zipcode"));
            member.setAddress(rs.getString("address"));
        }
        } catch(Exception ex) {
            ex.printStackTrace();
        } finally {
            if (rs != null) try { rs.close(); } catch(SQLException ex) {}
            if (pstmt != null) try { pstmt.close(); } catch(SQLException ex) {}
            if (conn != null) try { conn.close(); } catch(SQLException ex) {}
        }
        return member;
    }

    public Vector zipcodeRead(String area3) {
        Connection con = null;
        PreparedStatement pstmt = null;
        ResultSet rs = null;
        Vector vecList = new Vector();
        try {
            con = getConnection();
            String strQuery = "select * from zipcode where area3 like '"+area3+"%";
            pstmt = con.prepareStatement(strQuery);
            rs = pstmt.executeQuery();
            while(rs.next()){
                ZipcodeBean tempZipcode = new ZipcodeBean();
                tempZipcode.setZipcode(rs.getString("zipcode"));
                tempZipcode.setArea1(rs.getString("area1"));
                tempZipcode.setArea2(rs.getString("area2"));
                tempZipcode.setArea3(rs.getString("area3"));
                tempZipcode.setArea4(rs.getString("area4"));
                vecList.addElement(tempZipcode);
            }

        } catch(Exception ex) {
            System.out.println("Exception" + ex);
        } finally{
            if (rs != null) try { rs.close(); } catch(SQLException ex) {}
            if (pstmt != null) try { pstmt.close(); } catch(SQLException ex) {}
            if (con != null) try { con.close(); } catch(SQLException ex) {}
        }
        return vecList;
    }
}

```

9.inputForm.jsp 수정

- 자바스크립트 함수 추가

```
function zipCheck(){
```

```
url="ZipCheck.jsp?check=y";
open(url,"post","toolbar=no ,width=500 ,height=300
,directories=no,status=yes,scrollbars=yes,menubar=no");
}
```

- 폼 추가

```
<tr>
  <td width="200">우편번호</td>
  <td width="400"> <input type="text" name="zipcode" size="7">
    <input type="button" value="우편번호찾기" onClick="zipCheck()"></td>
</tr>
<tr>
  <td width="200">주소</td>
  <td width="400"><input type="text" name="address" size="70"></td>
</tr>
```

10.ZipCheck.jsp 추가

```
<%@ page contentType="text/html; charset=EUC-KR" %>
<%@ page import="java.util.*,ch11.logon.*" %>
<%@ include file="../view/color.jsp"%>
```

```
<%
```

```
request.setCharacterEncoding("euc-kr");
```

```
String check = request.getParameter("check");
```

```
String area3 = request.getParameter("area3");
```

```
LogonDBBean memMgr = LogonDBBean.getInstance();
```

```
Vector zipcodeList = memMgr.zipcodeRead(area3);
```

```
int totalList = zipcodeList.size();
```

```
%>
```

```
<html>
```

```
<head>
```

```
<title>우편번호검색</title>
```

```
<link href="style.css" rel="stylesheet" type="text/css">
```

```
<script language="JavaScript" src="script.js"></script>
```

```
<script>
```

```
function dongCheck(){
```

```
if (document.zipForm.area3.value == ""){
```

```
alert("동 이름을 입력하세요");
```

```
document.zipForm.area3.focus();
```

```
return;
```

```
}
```

```
document.zipForm.submit();
```

```
}
```

```
function sendAddress(zipcode,area1,area2,area3,area4){
```

```
var address =area1+" "+area2+" "+area3+" "+area4;
```

```
opener.document.userinput.zipcode.value=zipcode;
```

```
opener.document.userinput.address.value=address;
```

```
self.close();
```

```
}
```

```
</script>
```

```
</head>
```

```
<body bgcolor="#FFFFCC">
```

```
<center>
```

```
<b>우편번호 찾기</b>
```

```
<table>
<form name="zipForm" method="post" action="ZipCheck.jsp">
  <tr>
    <td><br>
      동이름 입력 : <input name="area3" type="text">
      <input type="button" value="검색" onclick= "dongCheck();">
    </td>
  </tr>
  <input type="hidden" name="check" value="n">
</form>
<%
if(check.equals("n")){
%>
<%
  if (zipcodeList.isEmpty()) {
%>
    <tr><td align="center"><br>검색된 결과가 없습니다.</td></tr>
<% }
else {
%>
<tr><td align="center"><br>
  ※검색 후, 아래 우편번호를 클릭하면 자동으로 입력됩니다.</td></tr>
<%
for (int i=0;i<totalList;i++) {
ZipcodeBean zipBean = (ZipcodeBean)zipcodeList.elementAt(i);
String tempZipcode = zipBean.getZipcode();
String temptArea1 =zipBean.getArea1().trim();
String temptArea2 =zipBean.getArea2().trim();
String temptArea3 =zipBean.getArea3().trim();
String temptArea4 =zipBean.getArea4().trim();
%>
<tr><td>
<a
href="javascript:sendAddress('<%=tempZipcode%>','<%=temptArea1%>','<%=temptArea2%>','<%=
<%=tempZipcode%>&nbsp;<%=temptArea1%>&nbsp;<%=temptArea2%>&nbsp;<%=temptArea3%>&nbsp;<%=temptArea4%>'>
<%
} //for
}
%>
<%}%>
</td></tr>
<tr><td align="center"><br><a href="javascript:this.close();">닫기 </a></td>
</table>
</center>
</body>
</html>
```