

조직도 TREE 매뉴얼

목차

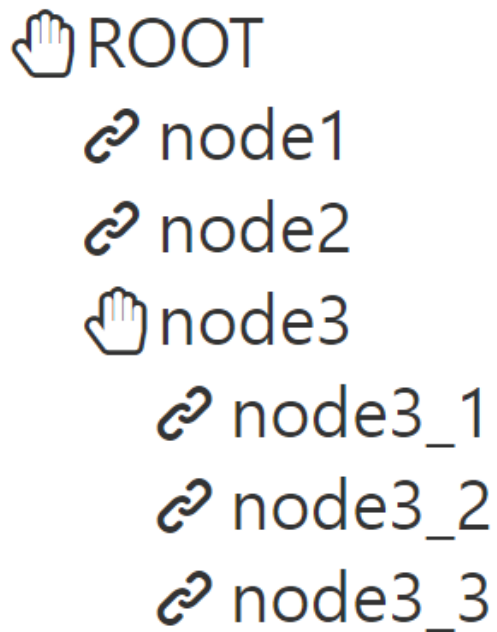
1. 조직도 개념

2. 트리 구현

(ㄱ)DB 데이터 삽입

(ㄴ)JAVA 구현

1. 조직도 개념



정적 파일(HTML, CSS)로 조직도 트리를 구현한 모습

트리는 계층적인 자료구조를 표현하기 위해 사용한다.

데이터가 저장된 부분은 노드(node)라고 하며, 노드와 노드 사이를 잇는 선은 간선(edge)이다.

트리는 활용도가 높은 자료구조로, HTML이나 XML 문서를 다룰 때 사용하는 DOM(Document Object Model)도 트리구조이다. 검색 엔진이나 데이터베이스도 트리 자료구조에 기반해서 구현된다.

2. 트리 구현

트리를 구현하기 위한 주요 연산

- 노드 선언 및 생성
- 자식 노드 연결
- 트리 출력

구현 순서

- ① DB) ORACLE SQL Developer에서 테이블을 생성하기
- ② DB) 데이터 삽입(회사 구조 입력) 하기
- ③ JAVA) DB연결을 위한 추상 클래스 만들기
- ④ JAVA) 추상 클래스를 상속받은 클래스 만들기. 계층형 쿼리 결과를 이용하여 트리 구조를 만든다.
- ⑤ JAVA) DB 행 하나를 담거나 전달할 때 필요한 객체 DTO만들기
- ⑥ JAVA) 계층형 쿼리 결과를 트리 구조로 만드는 클래스를 만들기
- ⑦ HTML) 프레임 구조를 만들기
- ⑧ JSP) TREE를 표현
- ⑨ JSP) 상세조직도
- ⑩ JSP) 기본구조

① DB) ORACLE SQL Developer에서 테이블을 생성하기

```
drop table org_base cascade constraints;
```

```
create table org_base (
    orgcode          VARCHAR2(10)          not null,
    orgstartdate     DATE,
    orgenddate       DATE,
    orgname          VARCHAR2(50)          not null,
    orglevel         VARCHAR2(2)           not null,
    orgstep          NUMBER(2),
    orguppercode     VARCHAR2(10)          not null,
    orgnote          VARCHAR2(100),
    constraint PK_ORG_BASE primary key (orgcode)
)
```

```
create table org_base (
    orgcode          VARCHAR2(10)          not null,
    orgstartdate     DATE,
    orgenddate       DATE,
    orgname          VARCHAR2(50)          not null,
    orglevel         VARCHAR2(2)           not null,
    orgstep          NUMBER(2),
    orguppercode     VARCHAR2(10)          not null,
    orgnote          VARCHAR2(100),
    constraint PK_ORG_BASE primary key (orgcode)
)
```

② DB) 데이터 삽입(회사 구조 입력) 하기

```

15 INSERT INTO ORG_BASE VALUES('01000000','1997.01.01','2999.12.31','사장','1',0,'00000000','');
16 INSERT INTO ORG_BASE VALUES('02000000','1997.01.01','2999.12.31','전략기획부문','10',1,'01000000','');
17 INSERT INTO ORG_BASE VALUES('02010000','1997.01.01','2999.12.31','경영전략실','20',1,'02000000','');
18 INSERT INTO ORG_BASE VALUES('02010100','1997.01.01','2999.12.31','경영기획팀','4',1,'02010000','');
19 INSERT INTO ORG_BASE VALUES('02010101','1997.01.01','2999.12.31','김경영','4',1,'02010100','');
20 INSERT INTO ORG_BASE VALUES('02010102','1997.01.01','2999.12.31','이경영','4',2,'02010100','');
21 INSERT INTO ORG_BASE VALUES('02010103','1997.01.01','2999.12.31','서경영','4',3,'02010100','');
22 INSERT INTO ORG_BASE VALUES('02020000','1997.01.01','2999.12.31','경영관리실','20',2,'02000000','');
23 INSERT INTO ORG_BASE VALUES('02020100','2003.11.01','2999.12.31','CG팀','4',1,'02020000','');
24 INSERT INTO ORG_BASE VALUES('02020200','1997.01.01','2999.12.31','경영분석팀','4',2,'02020000','');
25 INSERT INTO ORG_BASE VALUES('02030000','2003.11.01','2999.12.31','글로벌사업실','20',3,'02000000','');
26 INSERT INTO ORG_BASE VALUES('02030100','2003.11.01','2999.12.31','글로벌전략팀','4',1,'02030000','');
27 INSERT INTO ORG_BASE VALUES('03000000','1997.01.01','2999.12.31','경영지원부문','10',2,'01000000','');
28 INSERT INTO ORG_BASE VALUES('03010000','1997.01.01','2999.12.31','인력개발실','20',1,'03000000','');
29 INSERT INTO ORG_BASE VALUES('03010100','2003.11.01','2999.12.31','인력혁신팀','4',0,'03010000','');
30 INSERT INTO ORG_BASE VALUES('03010200','1997.01.01','2999.12.31','인사팀','4',1,'03010000','');
31 INSERT INTO ORG_BASE VALUES('03020000','1997.01.01','2999.12.31','사업지원실','4',1,'03010000','');
32 INSERT INTO ORG_BASE VALUES('03020100','1997.01.01','2999.12.31','총무팀','4',1,'03020000','');
33 INSERT INTO ORG_BASE VALUES('03020200','1997.01.01','2999.12.31','구매팀','4',2,'03020000','');

```

행 번호 19~21은 경영기획팀의 하위데이터로써 사원 데이터를 넣어 보기 위해 임의로 작성한 것입니다.

```

INSERT INTO ORG_BASE VALUES('01000000','1997.01.01','2999.12.31','사장','1',0,'00000000','');
INSERT INTO ORG_BASE VALUES('02000000','1997.01.01','2999.12.31','전략기획부문','10',1,'01000000','');
INSERT INTO ORG_BASE VALUES('02010000','1997.01.01','2999.12.31','경영전략실','20',1,'02000000','');
INSERT INTO ORG_BASE VALUES('02010100','1997.01.01','2999.12.31','경영기획팀','4',1,'02010000','');
INSERT INTO ORG_BASE VALUES('02010101','1997.01.01','2999.12.31','김경영','4',1,'02010100','');
INSERT INTO ORG_BASE VALUES('02010102','1997.01.01','2999.12.31','이경영','4',2,'02010100','');
INSERT INTO ORG_BASE VALUES('02010103','1997.01.01','2999.12.31','서경영','4',3,'02010100','');
INSERT INTO ORG_BASE VALUES('02020000','1997.01.01','2999.12.31','경영관리실','20',2,'02000000','');
INSERT INTO ORG_BASE VALUES('02020100','2003.11.01','2999.12.31','CG 팀','4',1,'02020000','');
INSERT INTO ORG_BASE VALUES('02020200','1997.01.01','2999.12.31','경영분석팀','4',2,'02020000','');
INSERT INTO ORG_BASE VALUES('02030000','2003.11.01','2999.12.31','글로벌사업실','20',3,'02000000','');
INSERT INTO ORG_BASE VALUES('02030100','2003.11.01','2999.12.31','글로벌전략팀','4',1,'02030000','');
INSERT INTO ORG_BASE VALUES('03000000','1997.01.01','2999.12.31','경영지원부문','10',2,'01000000','');

```

```
'10',2,'01000000','');  
INSERT INTO ORG_BASE VALUES('03010000','1997.01.01','2999.12.31','인력개발실'  
'20',1,'03000000','');  
INSERT INTO ORG_BASE VALUES('03010100','2003.11.01','2999.12.31','인력혁신팀'  
'4',0,'03010000','');  
INSERT INTO ORG_BASE VALUES('03010200','1997.01.01','2999.12.31','인사팀','4',1,'03010000','');  
INSERT INTO ORG_BASE VALUES('03020000','1997.01.01','2999.12.31','사업지원실'  
'4',1,'03010000','');  
INSERT INTO ORG_BASE VALUES('03020100','1997.01.01','2999.12.31','총무팀','4',1,'03020000','');  
INSERT INTO ORG_BASE VALUES('03020200','1997.01.01','2999.12.31','구매팀','4',2,'03020000','');
```

③ JAVA) DB연결을 위한 추상 클래스 만들기



AbstractDataBase.java

←더블클릭하세요!

```
// DB 연결을 위한 추상 클래스
public abstract class AbstractDataBase {

    public AbstractDataBase(){

        try {
            Class.forName("oracle.jdbc.driver.OracleDriver");
            System.out.println("-----1/6 ");
        } catch (Exception e) {
            System.out.println("-----1/6 실패");
        }
    }

    public Connection getConnection() throws SQLException {

        String url="jdbc:oracle:thin:@localhost:1521:xe";
        String user="pc17";
        String passwd="java";
        Connection conn=DriverManager.getConnection(url,user,passwd);
        System.out.println("-----2/6 ");
        return conn;

    }//getConnection

    public void close(Connection conn, PreparedStatement psmt, ResultSet select) {
        if(select!=null){
            try {
                select.close();
            } catch (SQLException e) {}
        }

        if(psmt!=null){
            try {
                psmt.close();
            } catch (SQLException e) {}
        }

        if(conn!=null){
            try {
                conn.close();
            } catch (SQLException e) {}
        }
    }//close
}
```


- ④ JAVA) 추상 클래스를 상속받은 클래스 만들기. 계층형 쿼리 결과를 이용하여 트리 구조를 만든다.



Tree_arch.java



icon.zip

←더블클릭하세요! (zip 파일: 이미지)

//추상클래스를 상속받은 클래스를 만든다. 계층형 쿼리결과를 이용하여 트리구조를 만든다.

```
public class Tree_arch extends AbstractDataBase{

    public Tree_arch() throws Exception {

        super();
    }

    //Timestamp time
    public StringBuffer getMenuTreeString(String time) throws Exception {

        StringBuffer menuString = new StringBuffer(2048);

        String[] sel_menu_id = null;

        String menu_id = "";           //조직코드
        String menu_nm = "";           //조직명
        String []menu_type = null;      //조직Type

        long menu_level = 0;           // 조직Level이 아닌 조회된 계층의 Level

        int div_cnt = 0;               // <div> Tag Count

        String javascript_1 = "";
        String javascript_2 = "";
        String javascript_3 = "expandsub('1'); \r\n";

        try{
            Vector select = getTree(time);//time
            System.out.println("-----">select.size());
            sel_menu_id = new String(select.size());

            int h = select.size();

            for(int i=0;ich;i++){
                sel_menu_id[i] = ((TreeDTO)select.get(i)).getOrgcode();
            }
            div_cnt++;

            menuString.append("<div id=PG8 nowrap> \r\n");

            int i=0;
            for(i=0;ich;i++){

                if (i!=0) {
                    javascript_1 = "javascript:togglesub('"+i+"')";
                    javascript_2 = "javascript:expandsub('"+i+"');gothere('"+menu_id+"')";

                    if( ((TreeDTO)select.get(i)).getOrglevel() == menu_level ) {
                        menuString.append("<cspan id=SG"+i+"><IMG SRC='image/minus.png' BORDER=0 ID=IMG"+i+"></cspan> \r\n");
                        menuString.append("<a HREF='"+javascript_2+"' ID=A"+i+" class=fldblue> "+ menu_nm +"</a></span><br> \r\n");
                    } else if ( ((TreeDTO)select.get(i)).getOrglevel() > menu_level ) {
                        menuString.append("<cspan id=SG"+i+"><a HREF='"+javascript_1+"'><IMG SRC='image/plus.png' BORDER=0 ID=IMG"+i+"></a>"+
                            "<a HREF='"+javascript_2+"' ID=A"+i+" class=fldblue> "+ menu_nm +"</a></span><br> \r\n");
                        menuString.append("<div id=PG"+i+" style=display:none;margin-left=15' nowrap> \r\n");

                        div_cnt++;
                    } else if ( ((TreeDTO)select.get(i)).getOrglevel() < menu_level ) {
                        menuString.append("<cspan id=SG"+i+"><IMG SRC='image/minus.png' BORDER=0 ID=IMG"+i+"><a HREF='"+javascript_2+"' ID=A"+i+" class=fldblue> "+ menu_nm +"</a></span><br> \r\n");

                        for (int j=0; j < (menu_level - ((TreeDTO)select.get(i)).getOrglevel()); j++) {
                            menuString.append("</div> \r\n");
                        }
                        div_cnt--;
                    }
                }
            }

            menu_id = ((TreeDTO)select.get(i)).getOrgcode();
            menu_nm = ((TreeDTO)select.get(i)).getOrgname();
            menu_level = ((TreeDTO)select.get(i)).getOrglevel();
            menu_type = ((TreeDTO)select.get(i)).getOrder2();

        }

        javascript_1 = "javascript:togglesub('"+i+"')";
        javascript_2 = "javascript:expandsub('"+i+"');gothere('"+menu_id+"')";

        menuString.append("<cspan id=SG"+i+"><IMG SRC='image/minus.png' BORDER=0 ID=IMG"+i+"><a HREF='"+javascript_2+"' ID=A"+i+" class=fldblue> "+ menu_nm +"</a></span><br> \r\n");

        for (int j=0; j < div_cnt; j++) {
            menuString.append("</div> \r\n");
        }

        return menuString;

    }catch(Exception e){
        e.printStackTrace();
    }
    return null;
}
//getMenuTreeString @
```

```

> public Vector getTree(String time){
    //limestan time

    Connection conn = null;
    PreparedStatement pstmt = null;
    ResultSet rs = null;
    Vector v = new Vector();
    StringBuffer sb=new StringBuffer();
    sb.append("select A.orgcode, A.orguppercode,")
        .append(" A.orgname , A.orglevel, A.order2 ")
        .append(" from (select orgcode, ")
        .append(" orguppercode, ")
        .append(" orgname, ")
        .append(" level orglevel, ")
        .append(" SYS_CONNECT_BY_PATH(to_char(orglevel,'fm000')||orgcode, '/') order2 ")
        .append(" from org_base ")
        .append(" start with orguppercode = '00000000' ")
        .append(" connect by prior orgcode = orguppercode ")
        .append(" and to_date(?, 'yyyy-mm-dd') between orgstartdate and orgenddate) A ")
        .append(" where a.orgcode <> '00000000' ")
        .append(" order by A.order2 ");

    System.out.println(sb.toString());

    try {

        conn= getConnection();
        pstmt = conn.prepareStatement(sb.toString());
        pstmt.setString(1,time);
        rs=pstmt.executeQuery();

        while(rs.next()){
            int i = 1;
            String orgcode = rs.getString(i++);
            String orguppercode = rs.getString(i++);
            String orgname = rs.getString(i++);
            int orglevel = rs.getInt(i++);
            String order2 = rs.getString(i++);

            TreeDTO tree = new TreeDTO();

            tree.setOrgcode(orgcode);
            tree.setOrguppercode(orguppercode);
            tree.setOrgname(orgname);
            tree.setOrglevel(orglevel);
            tree.setOrder2(OrgCode2Token.split(order2));

            v.add(tree);
        }

    } catch (SQLException e) {
        System.out.println("SQL Exception "+e);
    }finally{
        close(conn,pstmt,rs);
    }

    return v;
}
}

```

⑤ JAVA) DB 행 하나를 담거나 전달할 때 필요한 객체 DTO만들기



TreeDTO.java

←더블클릭하세요!

//DB 행 하나를 담거나 전달할 때 필요한 객체 DTO를 만든다.

```
public class TreeDTO implements Serializable{

    private String orgcode;
    private String orguppercode;
    private String orgname;
    private int orglevel;
    private String []order2;

    public TreeDTO() {

    }

    /**
     * @param orgcode
     * @param orguppercode
     * @param orgname
     * @param orglevel
     * @param order2
     */
    public TreeDTO(String orgcode, String orguppercode, String orgname,
        int orglevel, String []order2) {

        this.orgcode = orgcode;
        this.orguppercode = orguppercode;
        this.orgname = orgname;
        this.orglevel = orglevel;
        this.order2 = order2;
    }
}
```

⑥ JAVA) 계층형 쿼리 결과를 트리 구조로 만드는 클래스를 만들기



OrgCode2Token.java

←더블클릭하세요!

```
// 계층형 쿼리 결과를 트리 구조로 만드는 클래스가 필요하다.
public class OrgCode2Token {

    private static final String SEP="/";
    private String test="/0010100000000/0020200000000/0030300000000";
    public static int count(String str){
        String [] split=str.split(SEP);
        return split.length-1;
    }
    public static String[] split(String str){
        System.out.println("str----->" +str);
        String [] split=str.split(SEP);
        String[] sp=new String[split.length-1];
        for(int i=0;i<sp.length;i++){
            sp[i]=split[i+1];
        }
        return sp;
    }
    public static void makeTee(Vector vv,String trees){

        if(count(trees)==1){
            Vector v=new Vector();
            if(!vv.contains((split(trees))[0])){
                v.add((split(trees))[0]);
                return ;
            }
            return ;
        }
        else{
            System.out.println("trees----->" +trees);

            if(!vv.contains((split(trees))[1])){
                makeTee(vv,trees.substring(11));
            }
            else{
                return ;
            }
        }
    }
    public static void print(Vector v){
        if(v.size()==0){
            return ;
        }
        else if(v.size()==1){
            return ;
        }
        else{
            print((Vector)v.get(1));
        }
    }
}
```

⑦ HTML) 프레임 구조를 만들기



org_frame.zip

← 더블클릭하세요!

```
<!--
  Frameset(프레임셋)을 사용하기 위해 "XHTML Frameset DTD" 선언
-->
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<meta charset="UTF-8">
<title>조직조회</title>
</head>
<!--
  <frameset> 태그
  HTML5에서 지원하지 않는다고 합니다.
  body태그를 사용하지 않음
-->
  <frameset rows="*" cols="192,*" frameborder="Yes" border="1" >
    <frame src='tree.jsp' name='menu' scrolling="AUTO" noresize>

    <frameset rows="30%,*" cols="*" frameborder="Yes" border="0">
      <frame src='org_base_inform.jsp' name='top' scrolling="AUTO" noresize>
      <frame src='org_code.jsp' name='bottom'>
    </frameset>

  </frameset>

  <noframes>
<body>
  </noframes>
</body>

</html>
```

⑧ JSP) TREE를 표현



tree.jsp

←더블클릭하세요!

```
<%
String sdate="";
sdate = request.getParameter("ttime");
if(sdate ==null || sdate.equals("")){
    Calendar cal = Calendar.getInstance();
    int year = cal.get(Calendar.YEAR);
    int month = cal.get(Calendar.MONTH)+1;
    int date = cal.get(Calendar.DATE);
    sdate = (year+"-"+month+"-"+date);
}
%>
<form name='times' method="post" action="tree.jsp">
    <table width="164" border="0">
        <tr>
            <td width="50">날짜</td>
            <td width="75"><input name="ttime" type="text" size="10" value="<%=sdate%>"></td>
            <td width="39"><input type="submit" value="조회"></td>
        </tr>
    </table>
</form>

<form method='post' name='codeform'>
    <input name='code' />

    <%
        Tree_arch tree= new Tree_arch();
        String sb = tree.getMenuTreeString(sdate).toString();
    %>
</form>

<%=sb%>
```

⑨ JSP) 상세조직도



org_code.jsp

조직도에서 부서를 클릭 시 경로

⑩ JSP) 기본구조



org_base_inform.js
p

조직도에서 부서를 클릭 시 경로

날짜

2022-7-27

조회

02010103

- 사장
 - 전략기획부문
 - 경영전략실
 - 경영기획팀
 - 김경영
 - 이경영
 - 이경영
 - 경영관리실
 - CG팀
 - 경영분석팀
 - 글로벌사업실
 - 글로벌전략팀
 - 경영지원부문
 - 인력개발실
 - 인력혁신팀
 - 인사팀
 - 사업지원실
 - 총무팀
 - 구매팀

기본구조

상세 조직도

org_frame.html에서 실행한 후 보여지는 처리결과

클릭이벤트

부서를 클릭했을 때 alert창과 상단 input태그로 orgcode(조직 코드)를 확인할 수 있다.