Model2를 이용한 회원관리

안 화 수

회원관리

❖ 주요 기능

1. Connection Pool

2. request, session 객체 공유 설정

3. Controller 클래스 : Java Servlet

4. Model = Service + DAO
   Service, DTO, DAO 클래스

5. View (화면 인터페이스) : EL , JSTL 사용

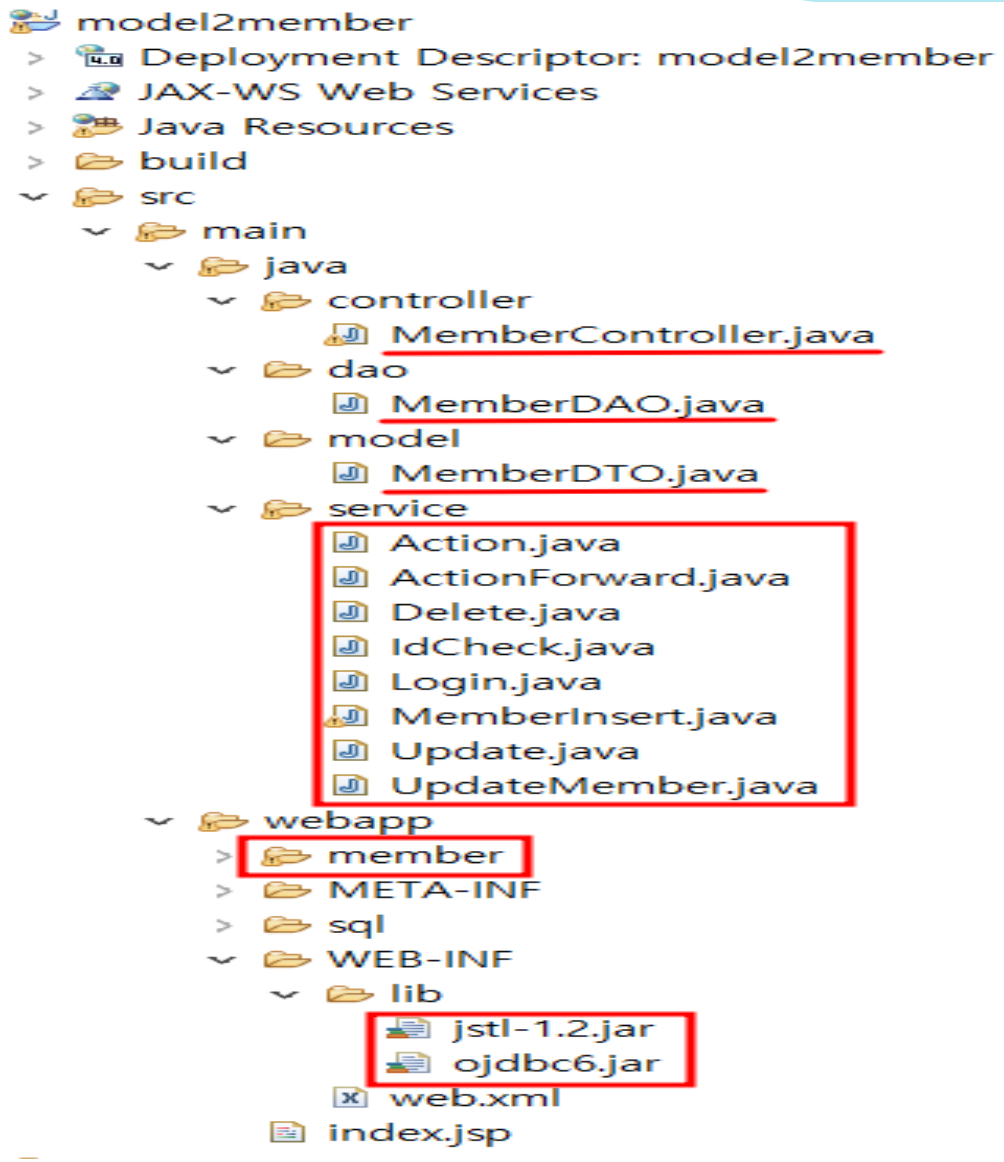
회원관리

- 회원 테이블 생성

```
create  table  model2member(
          id  varchar2(20) primary key,
          passwd  varchar2(20) not null,
          name varchar2(20) not null,
          jumin1 varchar2(6) not null,
          jumin2 varchar2(7) not null,
          mailid  varchar2(30),
          domain  varchar2(30),
          tel1 varchar2(5),
          tel2 varchar2(5),
          tel3 varchar2(5),
          phone1 varchar2(5),
          phone2 varchar2(5),
          phone3 varchar2(5),
          post varchar2(10),
          address varchar2(200),
          gender varchar2(20),
          hobby varchar2(50),
          intro varchar2(2000),
          register timestamp );
```

회원관리

❖ 프로젝트 구조


model2member
Deployment Descriptor: model2member
JAX-WS Web Services
Java Resources
build
src
main
java
controller
MemberController.java
dao
MemberDAO.java
model
MemberDTO.java
service
Action.java
ActionForward.java
Delete.java
IdCheck.java
Login.java
MemberInsert.java
Update.java
UpdateMember.java
webapp
member
META-INF
sql
WEB-INF
lib
jstl-1.2.jar
ojdbc6.jar
web.xml
index.jsp

회원관리

프로그램 주요 파일

Controller클래스 : controller - MemberController.java

DTO클래스 : model - MemberDTO.java (DTO 클래스)
DAO클래스 : dao - MemberDAO.java (DAO 클래스)

Action 인터페이스 : service - Action.java

ActionForward클래스 : service - ActionForward.java

Service 클래스 : service - MemberInsert.java (회원가입)
IdCheck.java (ID중복검사)
Login.java (로그인)
Logout.java (로그아웃)
UpdateMember.java (정보수정폼)
Update.java (정보수정)
DeleteMember.java (회원탈퇴폼)
Delete.java (회원탈퇴)

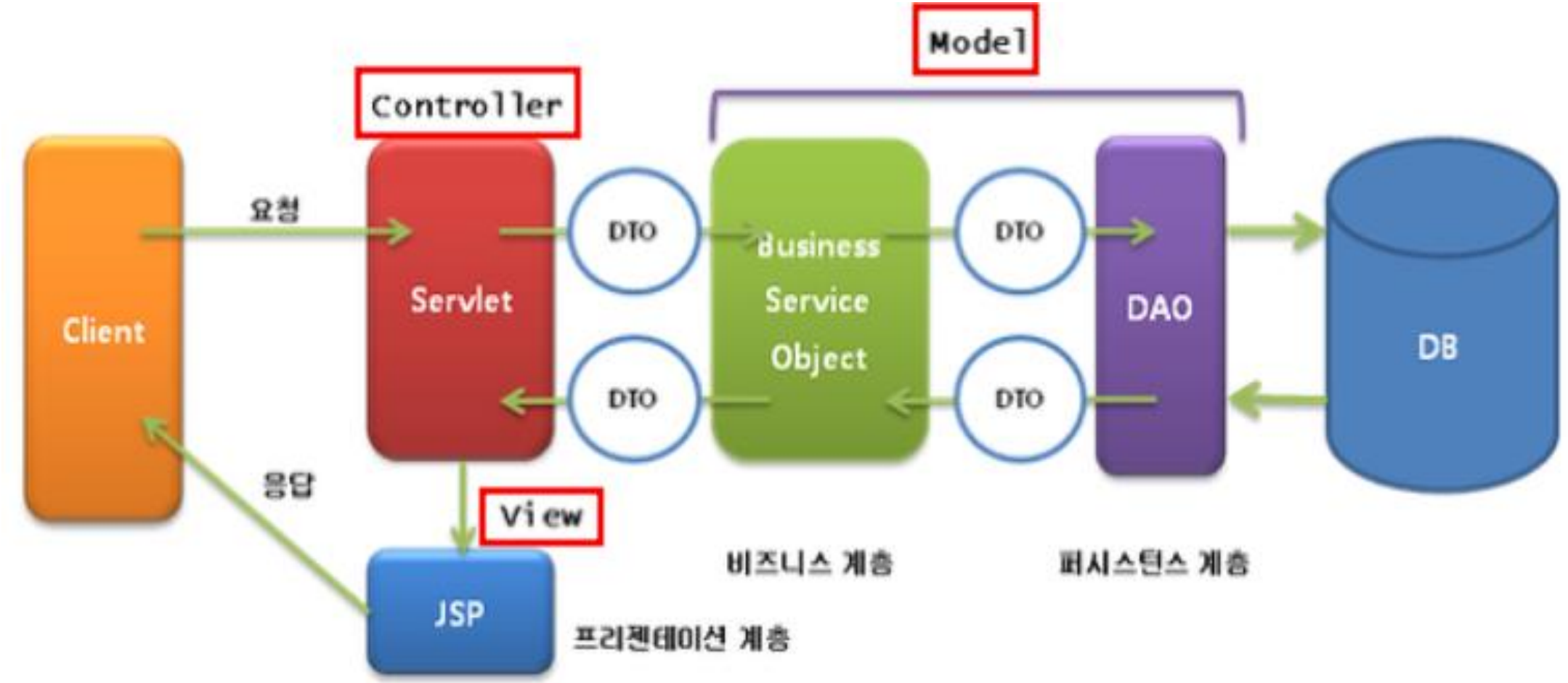
회원관리

- 프로그램 주요 파일

회원가입폼 : memberform.jsp --> member.jsp
ID중복검사 : idcheck.jsp
로그인 폼 : loginform.jsp --> login.jsp --> main.jsp
수정 폼 : updateform.jsp --> update.jsp
삭제 폼 : deleteform.jsp --> delete.jsp
로그아웃 : logout.jsp

모델2 설계

- MVC Architecture



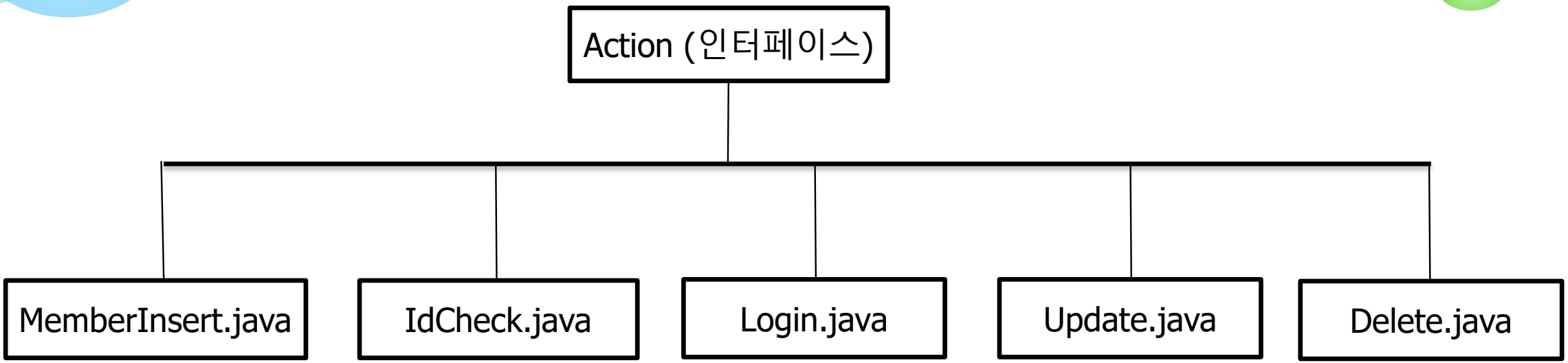
controller 클래스

```
package controller;

import java.io.IOException;

@WebServlet("*.do")
public class MemberController extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // doGet(), doPost() 메소드에서 공통적인 작업을 처리하는 메소드
    private void doProcess(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

        String requestURI = request.getRequestURI();
        String contextPath = request.getContextPath();
        String command = requestURI.substring(contextPath.length());

        System.out.println("requestURI:"+requestURI);   // /model2member/MemberInsert.do
        System.out.println("contextPath:"+contextPath); // /model2member
        System.out.println("command:"+command);         // /MemberInsert.do

        Action  action = null;
        ActionForward forward = null;

        // 회원 가입
        if(command.equals("/MemberInsert.do")) {
            try {
                action = new MemberInsert();
                forward = action.execute(request, response);
            }catch(Exception e) {
                e.printStackTrace();
            }
```

controller 클래스

```
        // 포워딩 처리
        if(forward != null) {
            if(forward.isRedirect()) {  // redirect 방식으로 포워딩
                response.sendRedirect(forward.getPath());
            }else {                     // dispatcher 방식으로 포워딩
                RequestDispatcher dispatcher = request.getRequestDispatcher(forward.getPath());
                dispatcher.forward(request, response);
            }
        }

    } // doProcess() end

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        System.out.println("get");

        doProcess(request, response);
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        System.out.println("post");

        doProcess(request, response);
    }

}
```

Service 클래스



- 구현 클래스들은 Action 인터페이스를 implements로 상속을 받고, 추상 메소드(execute())를 메소드 오버라이딩 해서 통일성 있게 구현한다.

Service 클래스

❖ Action 인터페이스

```
package service;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public interface Action {

// 추상 메소드
public ActionForward execute(HttpServletRequest request, HttpServletResponse response)
  throws Exception;

}
```

Service 클래스

```
package service;

import javax.servlet.http.HttpServletRequest;

public class MemberInsert implements Action{

    @Override
    public ActionForward execute(HttpServletRequest request, HttpServletResponse response) throws Exception {
        // TODO Auto-generated method stub
        System.out.println("MemberInsert");

        request.setCharacterEncoding("utf-8");

        MemberDTO member = new MemberDTO();
        member.setId(request.getParameter("id"));
        member.setPasswd(request.getParameter("passwd"));
        member.setName(request.getParameter("name"));
        member.setJumin1(request.getParameter("jumin1"));
        member.setJumin2(request.getParameter("jumin2"));
        member.setMailid(request.getParameter("mailid"));
        member.setDomain(request.getParameter("domain"));
        member.setTel1(request.getParameter("tel1"));
        member.setTel2(request.getParameter("tel2"));
        member.setTel3(request.getParameter("tel3"));
        member.setPhone1(request.getParameter("phone1"));
        member.setPhone2(request.getParameter("phone2"));
        member.setPhone3(request.getParameter("phone3"));
        member.setPost(request.getParameter("post"));
        member.setAddress(request.getParameter("address"));
        member.setGender(request.getParameter("gender"));
```

Service 클래스

```
        String [] hobby = request.getParameterValues("hobby");
        String h = "";
        for(String h1 : hobby) {
            h += h1 + "-";              // h = "공부-게임-"
        }
        member.setHobby(h);
        member.setIntro(request.getParameter("intro"));

        MemberDAO dao = MemberDAO.getInstance();
        int result = dao.insert(member);
        if(result == 1) {
            System.out.println("회원 가입 성공");
        }

        ActionForward forward = new ActionForward();
        forward.setRedirect(false);
        forward.setPath("./member/loginform.jsp");

        return forward;
    }

}
```

Service 클래스

❖ ActionForward 클래스

```
package service;

public class ActionForward {

        private boolean redirect;                    // 포워딩 방식 설정
        private String path;                         // 포워딩 페이지명 설정

        public boolean isRedirect() {
                return redirect;
        }
        public void setRedirect(boolean redirect) {
                this.redirect = redirect;
        }
        public String getPath() {
                return path;
        }
        public void setPath(String path) {
                this.path = path;
        }
}
```