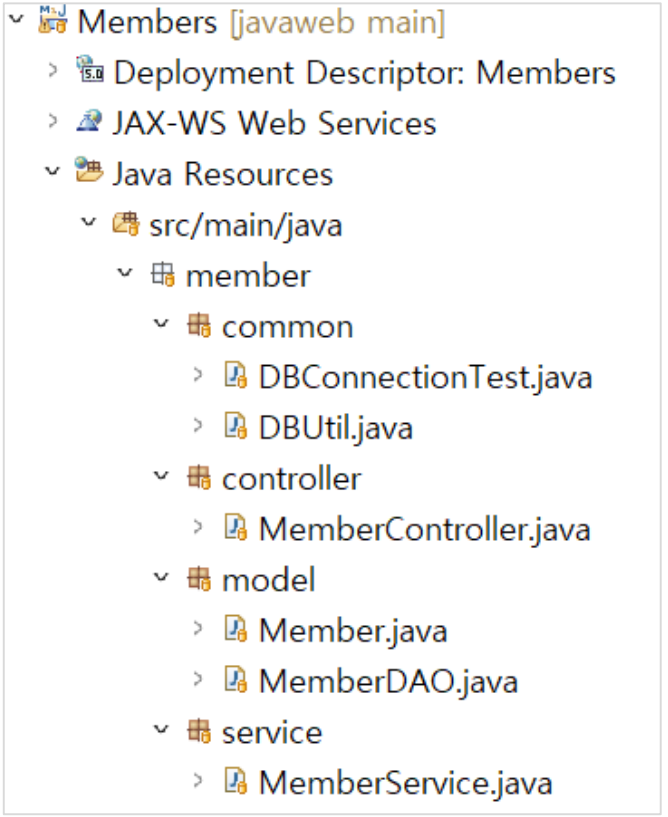
7장. 회원 및 게시판 관리

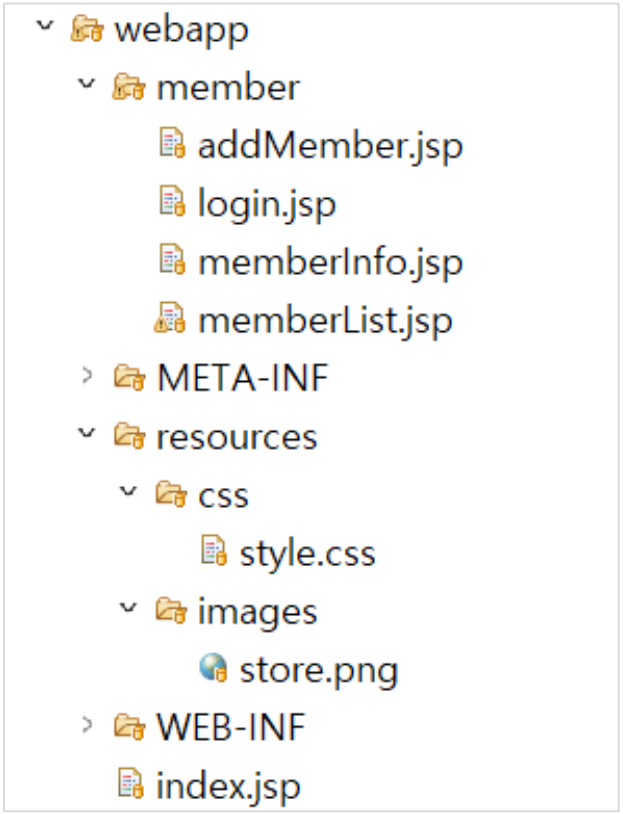
Member & Board

회원 관리 프로젝트

- **회원관리 프로젝트**
  - **프로젝트 이름 : Members**



java



web

Maven 기반 프로젝트

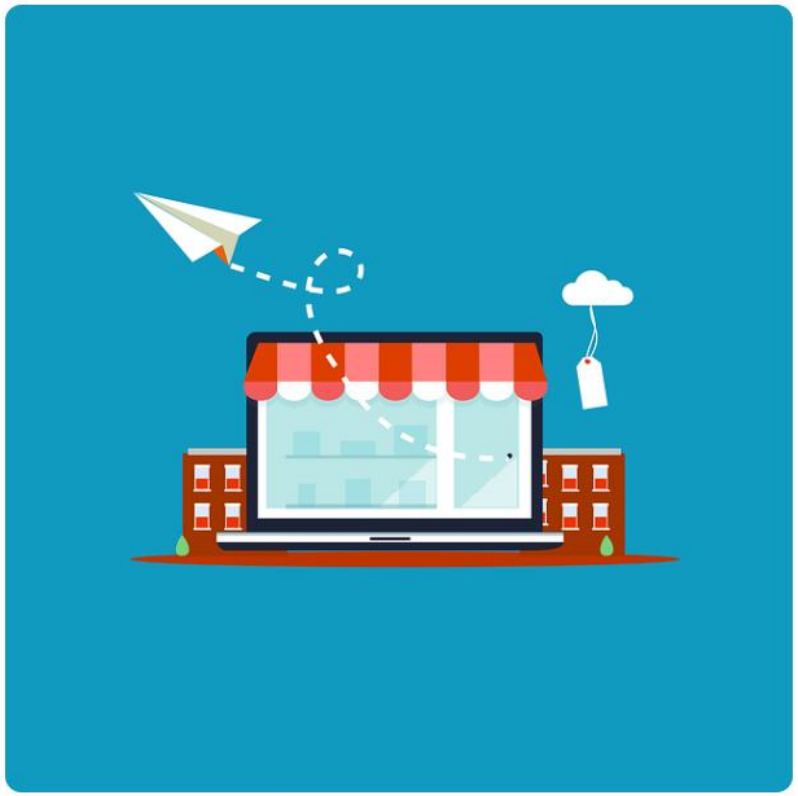
1. 프로젝트 > 우클릭 > configure > Convert to Maven Project
2. pox.xml 에 <dependencies> 의존성 관리 (3개 추가)

```
<dependencies>
  <dependency>
      <groupId>jakarta.servlet.jsp.jstl</groupId>
      <artifactId>jakarta.servlet.jsp.jstl-api</artifactId>
      <version>3.0.0</version>
  </dependency>
  <dependency>
      <groupId>org.glassfish.web</groupId>
      <artifactId>jakarta.servlet.jsp.jstl</artifactId>
      <version>3.0.1</version>
  </dependency>
  <dependency>
      <groupId>com.mysql</groupId>
      <artifactId>mysql-connector-j</artifactId>
      <version>8.4.0</version>
  </dependency>
</dependencies>
```

회원 관리 프로젝트

- 인덱스 페이지

Good Mall에 오신 것을 환영합니다.



회원 가입 | 로그인

회원 관리 프로젝트

- 인덱스 페이지

```
<head>
    <meta charset="UTF-8">
    <title>Welcome~ Good Mall...</title>
    <link rel="stylesheet" href="/resources/css/style.css">
</head>
<body>
    <section id="container">
        <div id="main">
            <h2>Good Mall에 오신 것을 환영합니다.</h2>

            <div class="main-pic">
                <img alt="점포" src="/resources/images/store.png">
            </div>

            <p>
               <a href="/member?action=addForm">회원 가입</a>   |  
               <a href="/member?action=loginForm">로그인</a>  
            </p>
        </div>
    </section>
</body>
```

회원 관리 프로젝트

- **style.css**

```css
#container{
    width: 80%;
    margin: 20px auto;
}
#main{text-align: center;}
a{text-decoration: none;}
a:hover{text-decoration: underline; }
ul li{margin: 10px;}
table{
    border: 1px solid #ccc;
    border-collapse: collapse;
}
h2{
    border-bottom: 2px solid #ccc;
    padding-bottom: 10px;
}
.main-pic img{
    width: 40%;
    border-radius: 10px;
}
```

```css
table th, td{
    border: 1px solid #ccc;
    padding: 5px 10px;
}
form ul li{list-style: none;}
form ul li label{
    width: 80px;
    float: left;
}
/* login 스타일 */
.btn{margin: 20px;}
```

회원 관리 프로젝트

- **MemberDTO**

```java
public class Member {
    private String mid;
    private String passwd;
    private String name;
    private String gender;
    private Timestamp joinDate;

    public Member() {}  //기본 생성자

    public Member(String mid, String passwd, String name,
            String gender, Timestamp joinDate) {
        this.mid = mid;
        this.passwd = passwd;
        this.name = name;
        this.gender = gender;
        this.joinDate = joinDate;
    }
```

회원 관리 프로젝트

- **MemberDTO**

```
public String getMid() {
    return mid;
}
public void setMid(String mid) {
    this.mid = mid;
}
public String getPasswd() {
    return passwd;
}
public void setPasswd(String passwd) {
    this.passwd = passwd;
}
public String getName() {
    return name;
}
public void setName(String name) {
    this.name = name;
}
```

회원 관리 프로젝트

- MemberDTO

```
    public String getGender() {
        return gender;
    }
    public void setGender(String gender) {
        this.gender = gender;
    }
    public Timestamp getJoinDate() {
        return joinDate;
    }
    public void setJoinDate(Timestamp joinDate) {
        this.joinDate = joinDate;
    }
}
```

회원 관리 프로젝트

- jspdb > Member 테이블

```sql
-- member 테이블 생성
create table member(
    mid      varchar(20) primary key,
    passwd   varchar(20) not null,
    name     varchar(30) not null,
    gender   varchar(10),
    joindate timestamp default now()
);

insert into member values ('today12', 'm1234', '김기용', '남자', now());
```

mid	passwd	name	gender	joindate
today12	m1234	김기용	남자	2025-10-13 17:04:18

회원 관리 프로젝트

- jspdb > Member 테이블

```sql
-- member 테이블 생성
create table member(
    mid      varchar(20) primary key,
    passwd   varchar(20) not null,
    name     varchar(30) not null,
    gender   varchar(10),
    joindate timestamp default now()
);

insert into member values ('today12', 'm1234', '김기용', '남자', now());
```

mid	passwd	name	gender	joindate
today12	m1234	김기용	남자	2025-10-13 17:04:18

회원 관리 프로젝트

- JDBC 연결 테스트

```
public class DBConnectionTest {
    private static final String URL = "jdbc:mysql://localhost:3306/jspdb";
    private static final String USER = "javauser";
    private static final String PASSWORD = "pwjava";

    public static void main(String[] args) {

        try{
            Connection conn = DriverManager.getConnection(URL, USER, PASSWORD);
            System.out.println(conn + "DB 접속 성공!!");
        }catch(Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
com.mysql.cj.jdbc.ConnectionImpl@58ea606cDB 접속 성공!!
```

회원 관리 프로젝트

- DBUtil 클래스

```
public class DBUtil {
    private static final String URL = "jdbc:mysql://localhost:3306/jspdb";
    private static final String USER = "javauser";
    private static final String PASSWORD = "pwjava";

    public static Connection getConnection() {

        try{
            Class.forName("com.mysql.cj.jdbc.Driver");
            return DriverManager.getConnection(URL, USER, PASSWORD);
        }catch(Exception e) {
            e.printStackTrace();
            return null;
        }
    }
}
```

회원 관리 프로젝트

- **MemberDAO – 회원 가입**

```
public void addMember(Member member) {
    String sql = "insert into member(mid, passwd, name, gender, joindate) "
            + "values (?, ?, ?, ?, now())";

    try(Connection conn = DBUtil.getConnection();
            PreparedStatement pstmt = conn.prepareStatement(sql)){
            pstmt.setString(1, member.getMid());
            pstmt.setString(2, member.getPasswd());
            pstmt.setString(3, member.getName());
            pstmt.setString(4, member.getGender());

            pstmt.executeUpdate();  //실행
    }catch(Exception e) {
        e.printStackTrace();
    }
}
```

회원 관리 프로젝트

- **MemberDAO – 회원 목록**

```
public List<Member> getMemberList(){
    List<Member> list = new ArrayList<>();
    String sql = "select * from member";

    try(Connection conn = DBUtil.getConnection();
        PreparedStatement pstmt = conn.prepareStatement(sql);
        ResultSet rs = pstmt.executeQuery()){

        while(rs.next()) {
            Member member = new Member(
                rs.getString("mid"),
                rs.getString("passwd"),
                rs.getString("name"),
                rs.getString("gender"),
                rs.getTimestamp("joindate")
            );
            list.add(member);
        }
    }catch(Exception e) {
        e.printStackTrace();
    }
    return list;
}
```

회원 관리 프로젝트

- **MemberDAO – 회원 상세(정보)**

```
public Member getMember(String mid) {
    Member member = new Member();
    String sql = "select * from member where mid = ?";

    try(Connection conn = DBUtil.getConnection();
            PreparedStatement pstmt = conn.prepareStatement(sql)){
        pstmt.setString(1, mid);

        ResultSet rs = pstmt.executeQuery();
        if(rs.next()) {
            member.setMid(rs.getString("mid"));
            member.setPasswd(rs.getString("passwd"));
            member.setName(rs.getString("name"));
            member.setGender(rs.getString("gender"));
            member.setJoinDate(rs.getTimestamp("joindate"));
        }
    }catch(Exception e) {
        e.printStackTrace();
    }
    return member;
}
```

회원 관리 프로젝트

- **MemberDAO – 로그인**

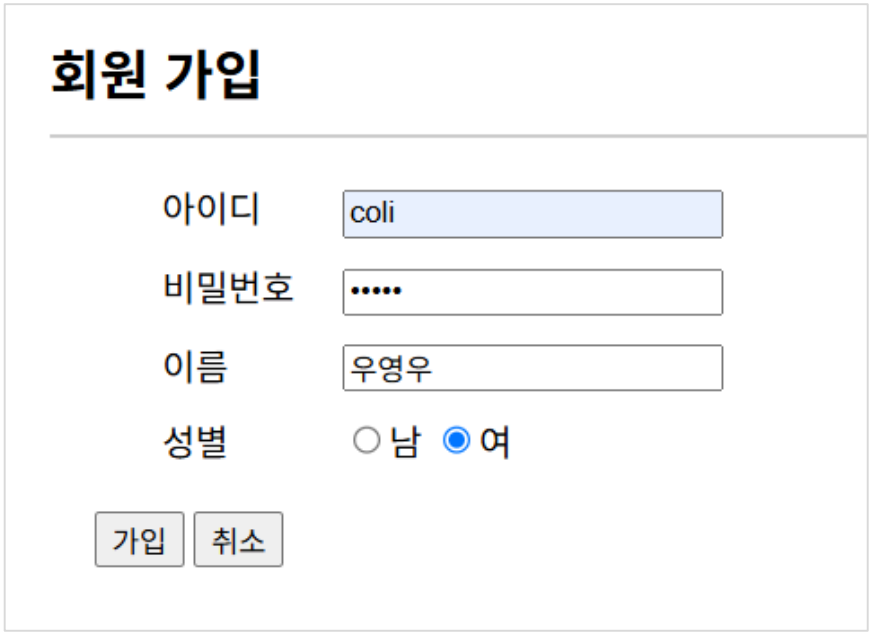
```
public boolean checkLogin(Member member) {

    String sql = "select * from member where mid=? and passwd=?";
    try(Connection conn = DBUtil.getConnection();
            PreparedStatement pstmt = conn.prepareStatement(sql)){
        pstmt.setString(1, member.getMid());
        pstmt.setString(2, member.getPasswd());

        ResultSet rs = pstmt.executeQuery();
        if(rs.next()) {
            return true;
        }
    }catch(Exception e) {
        e.printStackTrace();
    }
    return false;
}
```

회원 관리 프로젝트

- 회원 가입



회원 관리 프로젝트

- addMember.jsp

```
<section id="container">
    <h2>회원 가입</h2>
    <form action="/member?action=add" method="post">
        <ul>
            <li>
                <label>아이디</label>
                <input type="text" name="mid">
            </li>
            <li>
                <label>비밀번호</label>
                <input type="password" name="passwd">
            </li>
            <li>
                <label>이름</label>
                <input type="text" name="name">
            </li>
```

회원 관리 프로젝트

- addMember.jsp

```
            <li>
                <label>성별</label>
                <input type="radio" name="gender" value="남자" checked>남
                <input type="radio" name="gender" value="여자">여
            </li>
        </ul>
        <div class="btn">
            <input type="submit" value="가입">
            <input type="reset" value="취소">
        </div>
</form>
```

회원 관리 프로젝트

- **회원 목록**

회원 목록

[coli]님 로그아웃

번호	아이디	이름	성별	가입일
1	coli	우영우	여자	2025-10-15 16:54:43.0
2	today12	김기용	남자	2025-10-13 17:04:18.0

회원 관리 프로젝트

- **memberList.jsp**

```
<%@ taglib uri="jakarta.tags.core" prefix="c" %>
<section id="container">
    <h2>회원 목록</h2>
    <p>
        [${sessionId}]님 <a href="/member?action=logout">로그아웃</a>
    </p>
    <table>
        <thead>
            <tr>
                <th>번호</th>
                <th>아이디</th>
                <th>이름</th>
                <th>성별</th>
                <th>가입일</th>
            </tr>
        </thead>
```

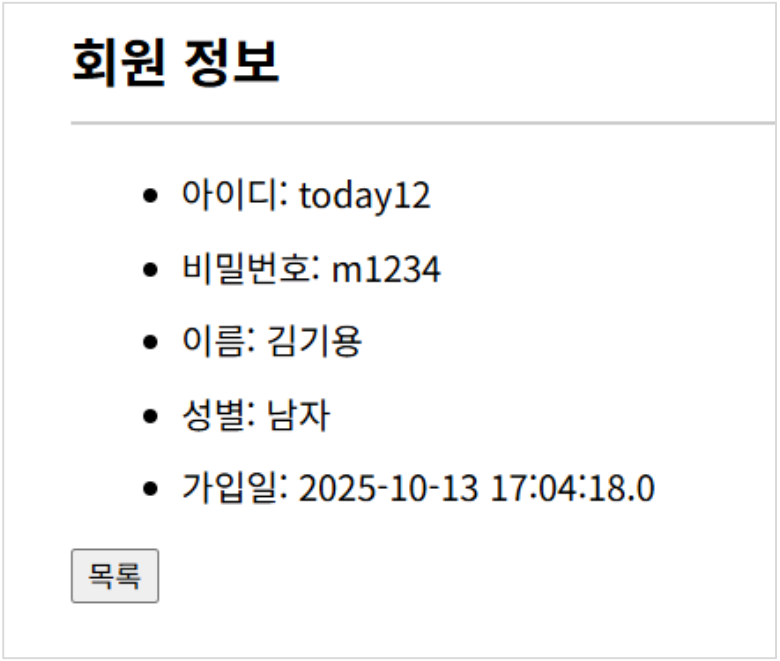
회원 관리 프로젝트

- **memberList.jsp**

```
    <tbody>
        <c:forEach var="member" items="#{memberList}" varStatus="i">
        <tr>
            <td>${i.count}</td>
            <td>
                <a href="/member?action=info&mid=${member.mid}">${member.mid}</a>
            </td>
            <td>${member.name}</td>
            <td>${member.gender}</td>
            <td>${member.joinDate}</td>
        </tr>
        </c:forEach>
    </tbody>
</table>
```

회원 관리 프로젝트

- **회원 상세 보기**



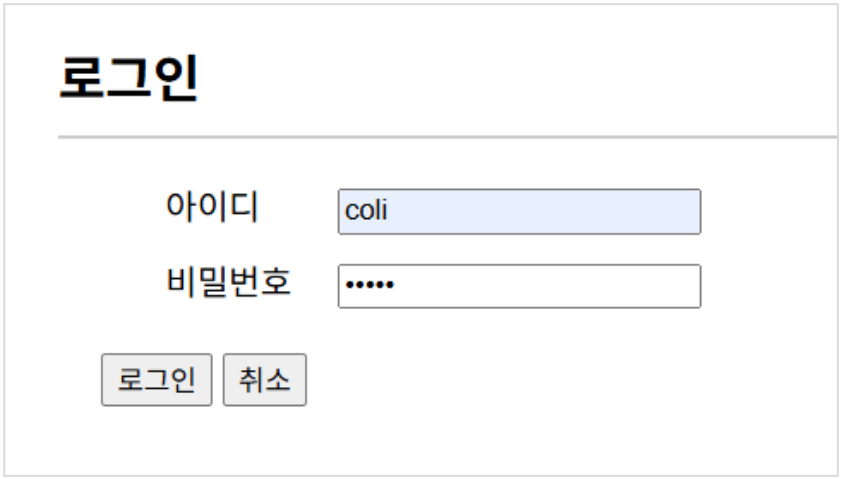
회원 관리 프로젝트

- **memberInfo.jsp**

```
<link rel="stylesheet" href="../resources/css/style.css">
</head>
<body>
    <section id="container">
        <h2>회원 정보</h2>
        <ul>
            <li>아이디: ${member.mid}</li>
            <li>비밀번호: ${member.passwd}</li>
            <li>이름: ${member.name}</li>
            <li>성별: ${member.gender}</li>
            <li>가입일: ${member.joinDate}</li>
        </ul>
        <a href="/member?action=list">
            <button type="button">목록</button>
        </a>
    </section>
</body>
```

회원 관리 프로젝트

- 로그인



회원 관리 프로젝트

- **login.jsp**

```
<h2>로그인</h2>
<form action="/member?action=login" method="post">
    <ul>
        <li>
            <label>아이디</label>
            <input type="text" name="mid">
        </li>
        <li>
            <label>비밀번호</label>
            <input type="password" name="passwd">
        </li>
    </ul>
    <div style="color: red">
        <c:if test="${not empty error}">
          ${error}
        </c:if>
    </div>
    <div class="btn">
        <input type="submit" value="로그인">
        <input type="reset" value="취소">
    </div>
</form>
```

회원 관리 프로젝트

- **MemberService – 서비스 계층**

```
public class MemberService {

    MemberDAO dao = new MemberDAO();

    //회원 가입
    public void addMember(Member member) {
        dao.addMember(member);
    }

    //회원 목록
    public List<Member> getMemberList(){
        return dao.getMemberList();
    }

    //로그인
    public boolean checkLogin(Member member) {
        return dao.checkLogin(member);
    }

    //회원 정보
    public Member getMember(String mid) {
        return dao.getMember(mid);
    }
}
```

회원 관리 프로젝트

- **MemberController – 서블릿**

```
@WebServlet("/member")
public class MemberController extends HttpServlet {
    private static final long serialVersionUID = 1L;

    MemberService service; //인스턴스 선언

    public MemberController() { //인스턴스 생성
        service = new MemberService();
    }

    protected void doPost(HttpServletRequest request, HttpServletResp
        String action = request.getParameter("action");
        String nextPage = "";
        HttpSession session = request.getSession();

        if(action.equals("list")) {
            List<Member> memberList = service.getMemberList();
            request.setAttribute("memberList", memberList);
            nextPage = "/member/memberList.jsp";
        }else if(action.equals("addForm")) {
            nextPage = "/member/addMember.jsp";
```

회원 관리 프로젝트

- **MemberController – 서블릿**

```
}else if(action.equals("add")) {
    String mid = request.getParameter("mid");
    String pwd = request.getParameter("passwd");
    String name = request.getParameter("name");
    String gender = request.getParameter("gender");

    Member member = new Member();
    member.setMid(mid);
    member.setPasswd(pwd);
    member.setName(name);
    member.setGender(gender);

    service.addMember(member);

    //등록 후 목록 페이지로 이동
    response.sendRedirect("/member?action=list");
    return;  //즉시 종료
}else if(action.equals("loginForm")) {
    nextPage = "/member/login.jsp";
```

회원 관리 프로젝트

- **MemberController – 서블릿**

```
}else if(action.equals("login")) {
    String id = request.getParameter("mid");
    String pw = request.getParameter("passwd");

    Member member = new Member();
    member.setMid(id);
    member.setPasswd(pw);

    boolean result = service.checkLogin(member);
    if(result) {
        session.setAttribute("sessionId", id); //세션 발급
        response.sendRedirect("/member?action=list");
        return;
    }else {
        String error = "아이디나 비밀번호가 일치하지 않습니다.";
        request.setAttribute("error", error);
        nextPage = "/member/login.jsp";
    }
```

회원 관리 프로젝트

- **MemberController – 서블릿**

```
    }else if(action.equals("logout")) {
        session.invalidate();
        nextPage = "index.jsp";
    }else if(action.equals("info")) {
        String mid = request.getParameter("mid");

        Member member = service.getMember(mid);
        request.setAttribute("member", member);
        nextPage = "/member/memberInfo.jsp";
    }

    //포워딩
    RequestDispatcher rd =
            request.getRequestDispatcher(nextPage);
    rd.forward(request, response);
}
```