

## 6장 요청과 응답-http 프로토콜 약속 언어

요청 방식 – get / post (서버에 데이터 전달)

get방식요청처리 / post방식요청처리

get	<ol style="list-style-type: none"> <li>1. 입력 데이터를 주소 노출</li> <li>2. url?변수명1=값1&amp;변수명2=값2.....</li> <li>3. 웹서버 url 길이 제한</li> <li>4. 파일 업로드 불가능</li> <li>5. 암호 보안 민감 데이터 전송 부적절</li> <li>6. url 공백 불가 -&gt; '+' 변경</li> <li>7. url전송데이터 암호화</li> </ol> <p>공백- + 영문 숫자 그대로 전송 한글 다국어 %16진수 인코딩</p>
post	<ol style="list-style-type: none"> <li>1. 입력 데이터를 주소 노출하지 않고 전송</li> <li>2. 따로 전송 - 변수명1=값1&amp;변수명2=값2.....</li> <li>3. 데이터길이 무제한</li> <li>4. 파일 업로드 가능</li> <li>5. 암호 보안 민감 데이터 전송 적절</li> <li>6. url 공백 불가 -&gt; '+' 변경</li> <li>7. url전송데이터 암호화</li> </ol> <p>공백- + 영문 숫자 그대로 전송 한글 다국어 %16진수 인코딩</p>

-http 프로토콜 정하는 규칙 (=spec.) 요청 = 서블릿 HttpServletRequest 객체

1. 시작url /insert?한글기본자동인코딩..... http버전 요청방식-get/post
2. 요청헤더
3. 요청바디 변수명=한글값&... request.setCharacterEncoding("utf-8");

-http 전송방식 규정

<form method=get> <form > <a href="/insert?id=aaa&pw=123.."> 회원가입하러 가 기</a> 주소 입력	<pre>doGet(){   request.getParameter("") }</pre>
<form method=post>	<pre>doPost(){   request.setCharacterEncoding("utf-8");   request.getParameter("") }</pre>

요청 - 처리(db연동필요) - 응답

-필요한 db정보

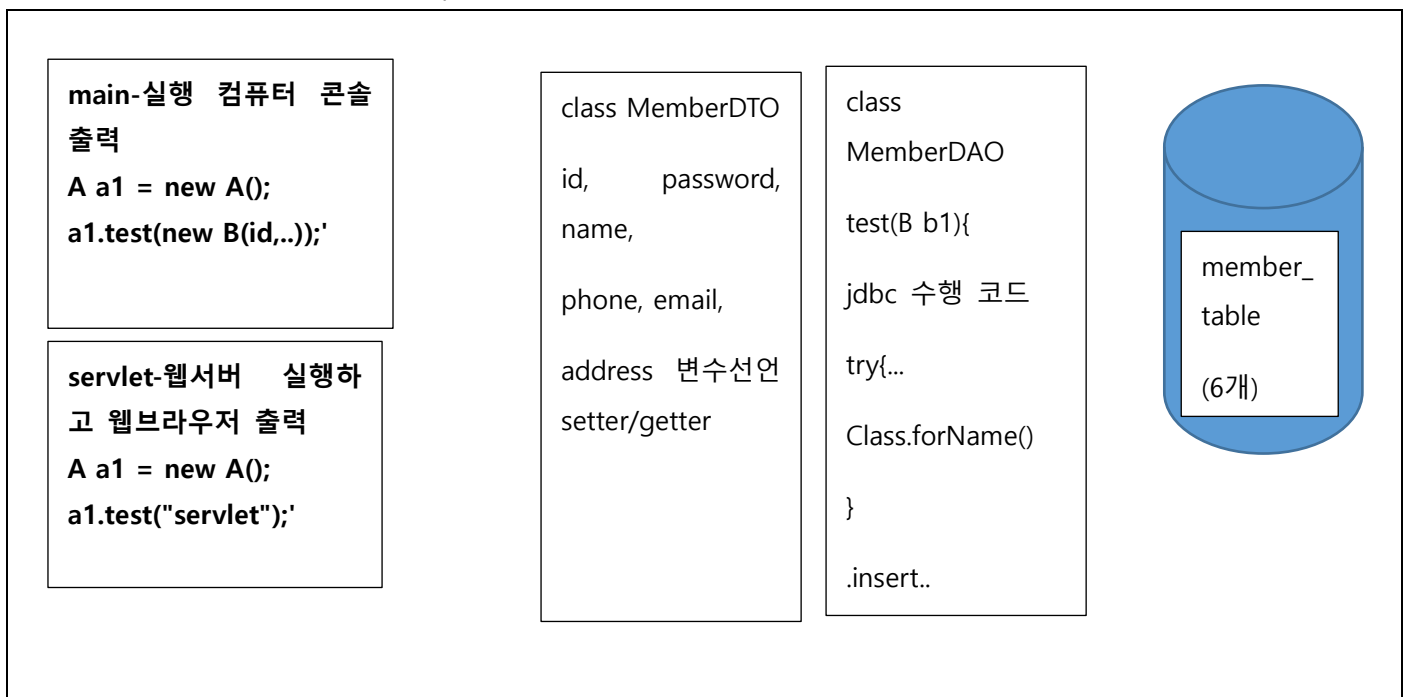
db이름 : mysql
db위치 : localhost:3306
db이름 : employeesdb
계정/암호 : emp/emp
테이블이름 : member_table

- 자바 구현 jdbc 패턴

XXXDAO 클래스 - "약속" - DATA ACCESS OBJECT = DB 접근 코드 = JDBC=crud 메소드 기능

XXXDTO 클래스 - "약속" - DATA TRANSFER OBJECT =

XXXVO = XXXXDO VALUE OBJECT / DATA OBJECT



http 오류

404	url매핑서블릿 못찾는 경우 해당 파일 없다
405	get-->doPost구현 post-->doGet구현
500	servlet 예외상황들

1. insert into old\_member select \* from member\_table where id='aaa';

2. delete from member\_table where id='aaa';

탈퇴회원 관리/1개월간 재가입금지/회원가입금지

## 실습

logindbservlet.html	LoginDBServlet	MemberDAO
<pre>&lt;form action="logindb"  id pw 로그인</pre>	<pre>doGet request.getParameter(); .... MemberDAO dao = new MemberDAO(); int result = dao.selectMember(id, pw); if(result == 0) 정상 사용자입니다. 출력 else if(result == 1) logindbservlet.html로 링크 태그 출 력 else if(result == 2) insertservlet.html 로 링크 태그 출 력  response.setCon...()</pre>	<pre>public int selectMember(String id, String pw){ id member_table에 존재하는지 확 인 존재한다면 해당id의 db암호와 pw암호 같은지 비교 같다 return 0 같지 않다 return 1 id 존재하지 않는다면 return 2 }</pre>

- data source

1.

```
class xxxdao {
```

```
.....
```

```
method(){
```

```
con = DriverManager.getConnection()
```

```
.....
```

```
.....
```

```
con.close();//자바와 db 연동 해제
```

```
}
```

1. 웹서버는 동시 여러 클라이언트 접속 유지
2. 웹클라이언트 db 연결 요구 연결 생성 가능해야 한다
3. db 최대허용치 제한
4. con.close()--> 즉각 db 연결해제하는 것이 아니고 con 소멸 가능성

5. db 연결 (sql 전송 - 처리 ) 작업보다 시간 소요된다

--->

"미리 CON 생성- 일정 갯수 생성- 소멸하지 않고 공유 재사용"

- > WEB SERVER DB연결
- > **CONNECTION POOLING** 개념
- > 개발자 직적 구현
- > TOMCAT 서버 구현 방법
- > **javax.sql.DataSource api** 사용 방식

## 7장 db 연결

7.1 ~ 7.2 crud

7.3 ~ connection pool/ TOMCAT 구현 / DATA SOURCE

### 1. db 연결 – DriverManager 이용

Class.forName("jdbc driver클래스명")

DriverManager.getConnection("db명 포트 ip", "계정", "암호");//필요시 con 생성

### 2. tomcat DataSource 이용

2-1. tomcat 서버 시작시 미리 con 생성

2-2. 관리 – tomcat (con 빌려오고 반납한다- 재사용)

2-3. con 생성 많이 시간 소요 절약

-구현

2-4. context.xml