

## [Session]

비연결형 프로토콜인 HTTP 의 한계를 극복하기 위해 사용  
사용자 로그인 정보나 장바구니 정보 유지 위해 사용

### 서블릿에서 세션 객체 생성

호출방법	설명
request.getSession(true)	request.getSession()와 동일
request.getSession(false)	현재 session 객체가 존재하면 그 session객체를 그대로 리턴. 만약 session 객체가 존재하지 않는다면 null을 리턴.
request.getSession()	우선 session 객체의 존재 여부를 확인하여 기존에 session이 없다면 새롭게 session 객체를 생성하여 리턴해주고, 기존에 존재하는 session객체가 있다면 그대로 그것을 리턴

서블릿 세션 객체 얻기 예)

```
HttpSession session = request.getSession();
```

세션 객체에 정보 저장: `setAttribute( String, Object )`

```
session.setAttribute ("name", "홍길동");
```

세션 객체에서 정보 검색 : `getAttribute` 반환형은 Object

```
Object o = session.getAttribute ("name" );
```

세션 객체에서 정보 삭제

```
session.removeAttribute ("name" );
```

메서드	리턴타입	설명
isNew()	boolean	Session이 처음으로 생성되었다면 true
getId()	String	session 고유의 ID를 문자열 형태 리턴
getCreationTime()	long	session이 생성된 시간을 1970,1,1 GMT부터 밀리세컨드 값
getLastAccessedTime()	Long	현재 session으로 마지막 작업한 시간. 밀리세컨드 값
getMaxInactiveInterval()	int	session의 유효 시간을 초로 반환.
setMaxInactiveInterval(int t)	void	session의 유효 시간을 초로 설정
invalidate()	void	현재 session을 종료

세션 생성 시기: 웹 클라이언트로부터 첫 번째 요청을 처리할 때 생성. 타이머 작동

세션 소멸 시기: 세션 타이머 만료 시 또는 코드상에서 명시적으로 세션 소멸.

세션 타임아웃 설정

- 1) WEB-INF/web.xml 에서  
    <session-config>  
        <session-timeout>100</session-timeout>  
    </session-config>
- 2) 세션 객체에서 session.setMaxInactiveInterval(100 \* 60);

## [JSP 내장 변수 : session ]

### 실습1

WebContent/session/sessionEx1.jsp

```
<%@ page contentType="text/html; charset=UTF-8" %>
<html><head>
<title>Session 연습</title>
</head><body>
<%
    Object value = session.getAttribute("result" ) ;
    if ( value == null ) {
        value = 0 ;
        session.setAttribute("result", 0 ) ;
    }else { value = (Integer)value+10 ;
        session.setAttribute("result", value ) ;
    }
%>
<%= value %>
</body></html>
```

처음 0 출력

여러번 요청하면 값이 10씩 계속 증가한다.

**실습2 로그인 예제**

WebContent/session/login.html

```

<!DOCTYPE html><html><head><meta charset="UTF-8"></head><body>
<form action="loginProcess.jsp" method="post">
    사번 <input type="text" name="empno" />
    이름 <input type="text" name="ename" />
        <input type="submit" value="login" />
</form>
</body></html>

```

src/ sessionEx.EmpVO.java

```

package sessionEx;

public class EmpVO {
    private int empNo;
    private String ename;
    private String job;    // getter setter 메소드
}

```

WebContent/session/loginProcess.jsp

```

<%@ page import="sessionEx.*"%>
<%@ page contentType="text/html; charset=UTF-8" %>
<!DOCTYPE html ><html><body>
<% boolean isSuccess = false;
    String empno = request.getParameter("empno");
    String ename = request.getParameter("ename");

    if ("1111".equals(empno)&& "scott".equals(ename)) //로그인 성공
    {
        EmpVO vo = new EmpVO();
        vo.setEmpNo(Integer.parseInt(empno));
        vo.setEname(ename);
        vo.setJob("SALESMAN");
        session.setAttribute("user", vo) ;
        out.print("<a href=\"main.jsp\">메인 페이지로 이동</a>");
    }else{
        out.print("<a href=\"login.html\">로그인실패.로그인 페이지로
이동 </a>");
    }

```

```
%>
</body></html>
```

#### WebContent/session/main.jsp

```
<%@ page import="sessionEx.*" %>
<%@ page contentType="text/html; charset=UTF-8"%>
<!DOCTYPE html><html><body>

<% String ename = "";
String job = "" ;
int empno = 0 ;
EmpVO vo =(EmpVO)session.getAttribute("user") ;
if (vo != null ){
    ename =vo.getEname();
    job = vo.getJob();
    empno = vo.getEmpNo();

    <%= ename %> 님 환영합니다.      <br/>
    <%= ename %> 님 정보              <br/>
    사번<%= empno %>                 <br/>
    직종 <%= job %>                  <br/>
    <a href="logout.jsp" >로그아웃 </a>

<% }else{    %>
<a href="Login.html" >로그인 하세요 </a>
<% }%>
</body></html>
```

#### WebContent/session/logout.jsp

```
<%@ page contentType="text/html; charset=UTF-8"%>
<!DOCTYPE html><html><body>
<% session.invalidate(); %> 로그 아웃 되었습니다.
</body></html>
```

실행)

<http://localhost:8088/웹컨텍스트패스/login.html> 요청 시 다음과 같은 화면 나옴

사번  이름

로그인 버튼 클릭 시 다음과 같은 화면 나옴

[메인 페이지로 이동](#)

메인 페이지로 이동을 클릭 시 다음과 같은 화면 나옴.

scott 님 환영합니다.  
 scott 님 정보  
 사번1111  
 직종 SALESMAN  
[로그아웃](#)

