

## NAVER 카페

session 로그인 중복처리의 문제..펌 | └─ &#9658; Ref.....@

찌르릉(chan\*\*\*\*)

<http://cafe.naver.com/planetzone/35>

[ session ]

request.getRequesteSessionId()는 지금 만들어진 세션Id를 보기 위함이라 아니라, 이전에 만들어진 세션의 Id를 보기 위함입니다

세션은 보통 세션만 쓰지는 않습니다. 요것이 life타임이 얻는것이나 셋팅하는 값으로 잘 안되기 때문입니다.

디폴트가 30분인데 30분이 되기 전에 사라지는 일이 종종발생하기 때문입니다.

세션에 대한 메소드를 추가 해서 소스를 올렸습니다.

getCreateTime : 생성된 시간을 반환해주는 메소드입니다.

getLastAccessedTime : 마지막으로 세션을 access한 시간을 반환해줍니다.

getMaxInactiveInterval : 마지막 access로 부터 살아남을 수 있는 시간입니다.

하지만 전 그전에 사라지는 경우를 종종 봐왔기때문에 오히려 살아남을 수 있는 최대한의 시간이라고 말하고 싶습니다.

getId : 서버가 클라이언트별루 주는 세션ID입니다. 인식자인 셈이죠..

아래 소스를 실행하면 아이디를 넣어 달라는 말이 나옵니다. 거기서 아이디를 넣으면 그 아이디를 세션에 넣습니다.

그리고는 소스 하단부분의 if문장을 실행하게 되죠.

```
<%@ page contentType="text/html; charset=euc-kr" session="true" %>
<%@ page import="java.io.*,java.util.*" %>
<html>
<body>
<%
if(session.getValue("id") == null ){
%>
아이디를 입력하세요.<br>
아이디가 없으신 분은 guest를 입력하세요.<br>
<form >
아이디 <input type="text" name="id"><br>
<input type="submit" value="login">
</form>
<%
}
%>
<%
String user="";
if(request.getParameter("id") != null){
user=request.getParameter("id");
session.putValue("id",user);
java.util.Enumeration st = session.getAttributeNames();
%>
<br>
<table border="1" align="center">
<tr>
<td>getCreationTime()</td>
<td><%= (new Date(session.getCreationTime()))%></td>
</tr>
<tr>
<td>getId()</td>
<td><%=session.getId()%></td>
</tr>
<tr>
<td>getMaxInactiveInterval()</td>
<td><%= (session.getMaxInactiveInterval()/60)%>min</td>
</tr>
<tr>
<td>getLastAccessedTime()</td>
<td><%= (new Date(session.getLastAccessedTime()))%></td>
</tr>
</table>
<%
}
%>
```

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

---

일정한 시간이 지나서, 이미 해당 HttpSession 이 invalidate 되었군요...  
 이미 invalidate 된 session을 다시 invalidate 시키려고 하면 IllegalStateException이 발생합니다. 이 경우에 발생하는 IllegalStateException은 무시하시면 되겠습니다.  
 아래처럼 고치세요..

```
> String mem_id = req.getParameter("mem_id");
> HttpSession session = SessionPool.get(mem_id);
>
> if(session!=null) {
>     session.invalidate(); // <--- !!!
>     SessionPool.remove(mem_id);
> }
==>
> String mem_id = req.getParameter("mem_id");
> HttpSession session = SessionPool.get(mem_id);
>
> if(session!=null) {
>     try{session.invalidate();}catch(Exception e){} // <--- !!!
>     SessionPool.remove(mem_id);
> }
```

---

#### 1. 현재 로그인되어 있는 사용자들의 id 와 세션카운트를 하는 방법

```
package session;
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.Enumeration;
public class SessionDisplay extends HttpServlet {
    public void doGet(HttpServletRequest req, HttpServletResponse res)
        throws ServletException, IOException {
        res.setContentType("text/plain");
        PrintWriter out = res.getWriter();
        HttpSession dummySession = req.getSession(true);
        dummySession.putValue("userID", "Watcher");
        HttpSessionContext context = dummySession.getSessionContext();
        int i = 0;
        Enumeration ids = context.getIds();
        out.println("=====");
        out.println("No SessionID UserID");
        out.println("=====");
        while (ids.hasMoreElements()) {
            i++;
            String id = (String) ids.nextElement();
            HttpSession session = context.getSession(id);
            String user_id = (String)session.getValue("userID");
            out.print("(" + i + ") " + id );
            out.println("  userID = [" + user_id + " ]");
        }
        out.println("=====");
        out.println("Total Connect User : " + i );
        out.println("=====");
        out.flush();
    }
}
```

---

1. user들이 로그인시 각각의 session에 아이디등이 저장됩니다.  
 관리자입장에서 모든 유저들의 로그인상태를 알고자 하는데요,  
 모든 유저들의 session에 저장된 값을 읽을수 있는지요...

```
int k = 0;
int j = 0;
HttpSessionContext context = session.getSessionContext();
Enumeration ids = context.getIds();
while(ids.hasMoreElements())
{
    String id = (String) ids.nextElement();
```

```

        out.println("<tr><td>");
        out.println(id);
        HttpSession foreignSession = context.getSession(id);
        String foreignUser = (String) foreignSession.getValue("LoginUser");
        String foreignName = (String) foreignSession.getValue("LoginUserName");
        out.println("<td>" + foreignUser);
        out.println("<td>" + foreignName);
        out.println("<td>" + request.getHeader("REMOTE_ADDR"));
        out.println("<td>" + new Date(foreignSession.getCreationTime()));
        out.println("<td>" + new Date(foreignSession.getLastAccessedTime()));
        out.println("</tr>");
    }

```

위와 같이 하였더니 되더군요...

근데요 브라우저 기준 세션을 보여주더군요...

세션이 null로 나온것은 세션저장안된 브라우저랍니다.

#### 1. 세션 ID획득하는 방법임다.

// 이건 세션의 컨텍스트를 얻기위한겁니다...(컨텍스트를 이용해서 세션의 ID를 알려고...)

```
HttpSessionContext context = session.getSessionContext();
```

// 위의 컨텍스트를 이용해서...

```
Enumeration ids = context.getIds();
```

// 최신의 상태가 아닌 세션을 검색하기위해 세션ID를 살펴봄..

```

while (ids.hasMoreElements()) {
    String id = (String)ids.nextElement();
    out.println("Checking " + id + "..."); // 세션의 ID가 나오겠죠...
    HttpSession session = context.getSession(id);
    ...
    String foreignUser = (String)foreignSession.getValue("user_id");
}

```

#### 1. jsp에서 회원로그인에서 session을 받아 처리하는 부분하고 이 session을 가지고 정보수정이랑 하고 싶은데 어떻게 처리를 해 줘야 하나요?

로그인할때 회원아이디값을 넘겨줍니다.

값을 받은 페이지에서는 그 값을 세션에 넣습니다.

이렇게요..

```
session.putValue("id",userid);
```

여기서 userid는 회원아이디값을 받은 String입니다.

그러면 세션에 id가 들어가겠지요.

다음으로 정보수정하는 페이지에 세션을 가져오는 문장을 넣지요.

```
session.getValue("id");
```

그리고 그 값을 String형으로 변환해서 임의의 변수에 넣습니다.

그리고 그 임의의 변수를 이용해서 해당회원의 정보를 수정하면 되겠지요...

#### 1. >client별 세션객체를 얻어서 client들의 세션 저장내용을 지우는 부분입니다...

>근데 맨날도 아니구 가끔 아주 가끔 foreignSession객체에서 NullPointerException이

>발생하는데요... 왜 그런여러가 나는지 모르겠네요...

```
> HttpSessionContext context = session.getSessionContext();
```

```
> Enumeration ids = context.getIds();
```

```
> while(ids.hasMoreElements())
```

```
> {
```

```
> id = (String) ids.nextElement();
```

```
> HttpSession foreignSession = context.getSession(id);
```

```
> String[] session_valuenames = foreignSession.getValueNames();
```

```
> for(int i=0; i<session_valuenames.length; i++)
```

```
> {
```

```
> foreignSession.removeValue(session_valuenames[i]);
```

```
> }
```

```
> }
```

일종의..동기화 문제로 보여집니다.

세션목록을 가져온 후,

그리고 세션저장내용을 지우기 전에

세션타임아웃 등의 이유로 세션이 없어지면

해당 id값이 없으니 null이 들어갈것이구..

모 일단 그렇게 보여지네요 ^^

## 또한 참고하세요..

☞ 로그인하고 로그아웃할때 데이터베이스 테이블에 값을 설정해서 비교하는 방법이 있습니다.

☞ 세션정보에 로그인한 사용자의 데이터도 넣을수 있을수 있다면,  
세션을 체크해서 주기적으로 데이터베이스를 체크한다면 가능할거 같기도 한데요...

☞ javax.servlet.http.\* 패키지에 보면 HttpSessionBindingListener 인터페이스가 있습니다.  
이 인터페이스에는 몇개의 메소드가 있는데 그 중 한개가 아래에 있는것입니다.

```
// HttpSessionBindingListener 구현
public void valueUnbound(HttpSessionBindingEvent http) {
    logout();
}
```

위 메소드는 서블릿에서 사용자의 세션이 끊어졌을때 생기는 이벤트입니다.

그러니깐 사용자가 로그인 했을때 데이터 베이스에 세션 정보가 있다면 중복 로그인한 사용자이고, 없다면 세션 정보를 넣어두고, 위 메소드에서 세션 종료시 DB에 저장된 세션정보를 지우던지 필드를 셸팅하던지 작업을 하시면 될것같습니다.

### ‡ 관련 메소드 정리

Object getAttribute(String name)  
- 주어진 이름으로 세션에 저장된 객체를 리턴한다. ※ Object getValue(String name)

Enumeration getAttributeNames()  
- 세션에 저장된 모든 객체들의 이름을 Enumeration 타입으로 리턴한다. ※ String

getValueNames()  
long getCreationTime()  
- 1970년 1월 1일 0시 0분 0초를 기준으로 세션이 생성된 시간을 리턴한다.(단위 ms)

String getId()  
- 세션의 ID를 리턴한다.

long getLastAccessedTime()  
- 1970년 1월 1일 0시 0분 0초를 기준으로 클라이언트가 이 세션에서 요청을 보낸 마지막 시간을 리턴한다.

int getMaxInactiveInterval()  
- 서버가 클라이언트의 요청없이 세션을 유지하는 최대 시간을 리턴한다. (단위 초)

void invalidate()  
- 세션을 종료한다.

boolean isNew()  
- 세션이 만들어지고 클라이언트가 아직 세션에 조인되지 않았으면 true를 리턴한다.

void setAttribute(String name, Object value)  
- 세션에 이름을 이용해서 객체를 저장시킨다. ※ void putValue(String name, Object value)

void removeAttribute(String name)  
- 주어진 이름의 객체를 세션에서 삭제한다. ※ void removeValue(String name)

void setMaxInactiveInterval(int interval)  
- 클라이언트의 요청없이 세션을 유지할 수 있는 최대 시간을 설정한다. (단위 초)

### 참고...

<http://www.sys-dev.info/web/servlet05.htm> 01....

<http://www.devworld.co.kr/java/etc/session1.html>

<http://home.skuniv.ac.kr/~kingka/java/servlet01.html>

<http://java.sun.com/j2ee/1.4/docs/api/>