

Essentials of Information Technology

PC-CS-305

Session Management

Topic & Structure of the lesson



- Http Session
- How Session Works
- How Container know Who Client Is
- Ways of Getting Session
- URL Rewriting
- HttpSession Methods
- Ways Session Times Out
- Cookie

gauravgambhir.cse@piet.co.in

Topic & Structure of the lesson



- Servlet Creating & Setting Cookie
- Servlet Getting the Cookie
- Key MileStones of HttpSession
- Session Migration

Http Session



- HttpSession object hold the conversational state across multiple requests from the same client.
- HttpSession persist for an entire session with a specific client.

How Session Works

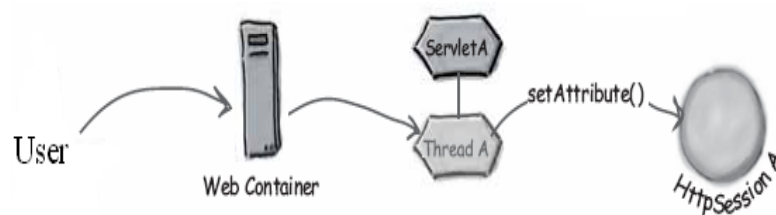


User select an option “Dark” and hit the submit button.

The container sends the request to a new thread of the servlet.

The thread finds the session associated with user and store his choice “Dark” in the session as an attribute.

How Session Works

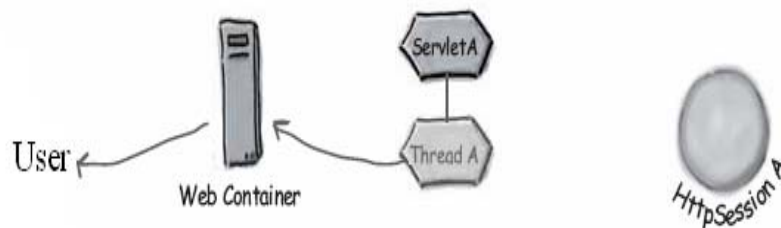


gauravgambhir.cse@piet.co.in

How Session Works



The servlet runs its business logic (including calls to the model) and returns a response In this case another question “ What’s price range ?”



PC-CS-305: Essentials of Information Technology

How Session Works



User considers the new question on the page.
Select “Expensive” and hits the submit button.

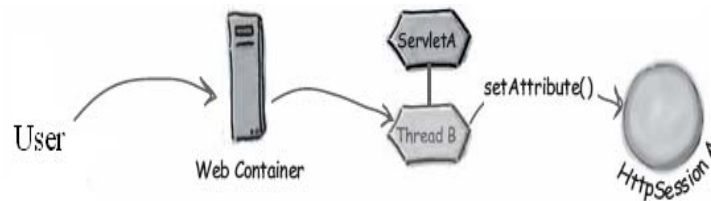
The container sends the request to a new thread of the BeerApp servlet.

The BeerApp thread finds the session associated with user and stores his new choice (“Expensive”) in the session as an attribute.

gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

How Session Works

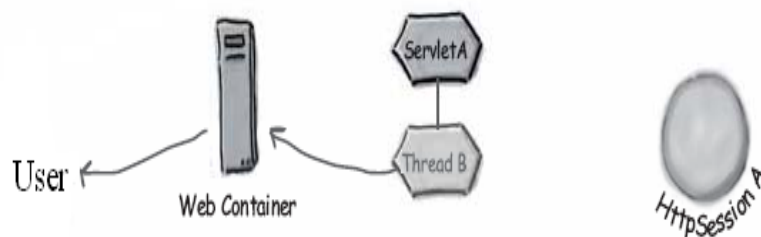


PC-CS-305: Essentials of Information Technology

How Session Works



The servlet runs its business logic (including calls to the model) and returns a response, in this case another question.



gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

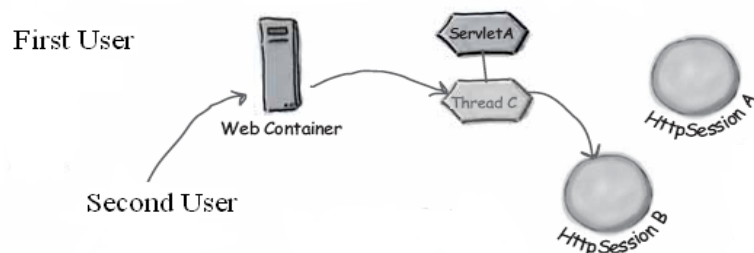
How Session Works



- First user session is still active, but meanwhile second user select “Pale” and hits the submit button.
- The container sends the second user request to a new thread of the BeerApp servlet.
- The BeerApp thread starts a new session for second user and call `setAttribute()` to store his choice (“Pale”).

PC-CS-305: Essentials of Information Technology

How Session Works



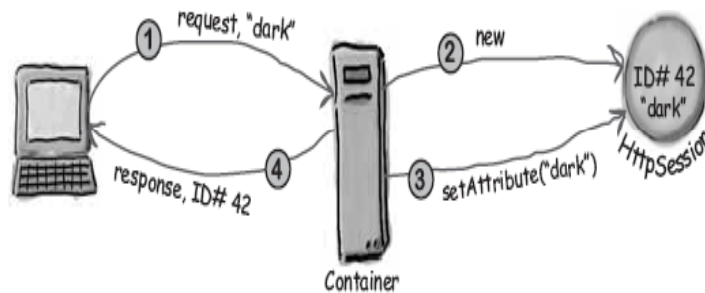
gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

How Container know Who Client Is



- On the client first request the container generate a unique session ID and gives back to the client with the response.

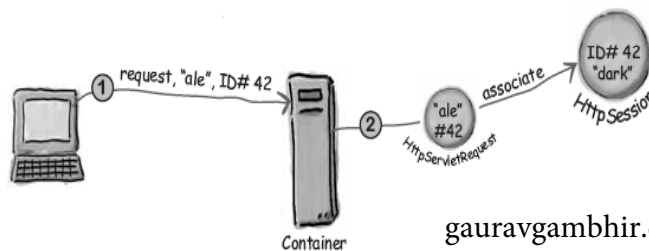


PC-CS-305: Essentials of Information Technology

How Container know Who Client Is



- The client sends back the session ID with each subsequent request
- The container sees the session ID find the matching session and associate the session with the request.



gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

How Client & Container Exchange Session ID



- Container has to get the session id to the client as a part of the response and client has to send back the session id as part of request.
- The simplest way to exchange the info is through cookies.

PC-CS-305: Essentials of Information Technology

HTTP Response



```
HTTP/1.1 200 OK
Set-Cookie: JSESSIONID=0AAB6C8DE415
Content-type: text/html
Content-length:397
Date: Wed 19, Nov 2003 03:25:40 GMT
Server:Apache-Coyote/1.1
Connection: Close
<html>
</html>
```

gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

HTTP Request



POST /select/selectBeerTaste2.do HTTP/1.1
Host: www.wickedlysmart.com
User-Agent: Mozilla/5.0
Cookie: JSESSIONID=0AAB6C8DE415
Accept: text/xml, application/xml, image/jpeg
Accept-Language: en-us
Accept-Encoding: gzip, deflate

PC-CS-305: Essentials of Information Technology

Note



- Container is responsible for generating the session ID, creating a new cookie object, stuffing the session id into the cookie and setting the cookie as a part of response.

gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

Remember



Sending the session cookie in the response

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

Getting the session ID from the request

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

Method for getting a session id cookie is the same as sending a session id cookie.

PC-CS-305: Essentials of Information Technology

Question



How do we know whether the session already exist or was just created ?

session.isNew() returns true if the client has not yet responded with this session id.

gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

Ways of Getting Session



1. From request object

Request is made from a particular client so the session is for the client associated with request.

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

Ways of Getting Session



2. From session event object

Event handling methods defined by listener interface related to sessions take an argument of type HttpSessionEvent or its subclass

HttpSessionBindingEvent that has got getSession() method.

```
public void sessionCreated(HttpSessionEvent event){  
    HttpSession session=event.getSession();  
}
```

gauravgambhir.cse@piet.co.in

Question



What in case if you want an existing session and do not want to create a new session?

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

Passing false means the method returns a pre-existing session or null if there was no session associated with this client.

URL Rewriting



- If cookies disabled, URL rewriting takes the session ID in cookie onto end of every URL that comes in to app.
- The extra work that needs to be done here is to encode the URL.

URL + ;sessionId=1234567

gauravgambhir.cse@piet.co.in

URL Rewriting



- Each web page link has a little bit of extra info tracked onto end of the URL.
- When the user clicks that “enhanced” link, the request goes to the container with that extra bit on the end.

PC-CS-305: Essentials of Information Technology

URL Rewriting



- The container simply strip off the extra part of the request URL and use it to find the matching session .
- The container will first attempt to use cookies for session for session management and fall back to URL rewriting only if the cookies approach fails.

gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

URL Rewriting



```
public void doGet(HttpServletRequest request, HttpServletResponse response) throws
IOException{
    response.setContentType("text/html");
    PrintWriter out=response.getWriter();
    HttpSession session=request.getSession();
    out.println("<html><body>");
    out.println("<a href='"+response.encodeURL("/BeerTest.do")+"'>Click me</a>");
    out.println("</body></html>");
}
```

Question



How the container knows that cookies are not working, At what point container decide to use URL rewriting ?

- Container sees a call to getSession()
- Container didn't get a sessionID with the client's request.
- The container start a new session with the client.

gauravgambhir.cse@piet.co.in

Question (Continued)



- The container doesn't know if cookies will work so with this first response back to client, it tries both cookies and URL rewriting.
- Container sends the response with both a "Set-Cookie" header for the session ID and the session ID appended to the URL,

Client makes a request with session ID appended to request URL but if client accepts cookies the request will also have a session ID cookie.

Question (Continued)



Servlet call `request.getSession()` the container read the session ID from the request find the session, now since the container know that the client accepts cookies so it will ignore the `response.encodeURL()`.

gauravgambhir.cse@piet.co.in

URL Rewriting



- URL rewriting is automatic but only if you encode your URL.
- Run all your URLs through a method of response object `encodeURL()` or `encodeRedirectURL()` and the container does everything else .

PC-CS-305: Essentials of Information Technology

Question



How does the container know when its safe to destroy the session ?

The container checks the state of session and find that no request have come in with that session ID for long time so, it finally destroy the session.

gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

HttpSession Methods



	What it does	What you'd use it for
getCreationTime()	Returns the time the session was first created	To find out how old the session is or to restrict certain session to a fixed length of time.
getLastAccessedTime()	Return the last time the container got a request with this session ID.	To find out when a client last accessed this session
setMaxInactiveInterval()	Specifies the maximum time, in seconds that you want to allow between client requests for this session	To cause a session to be destroyed after a certain amount of time has passed without a client making any request for this session.

PC-CS-305: Essentials of Information Technology

HttpSession Methods



	What it does	What you'd use it for
getMaxInactiveInterval()	Returns the max time in seconds that is allowed between client requests for this session.	To find out how long this session can be inactive and still be alive
invalidate()	Ends the session. This includes unbinding all session attributes currently stored in this session	To kill a session if the client has been inactive or if you know the session is over.

gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

Ways Session Can Die



Three ways session can die

- It times out
- You call invalidate() on the session object.
- The application goes down.

Ways Session Times Out



Configuring session timeout in the DD

```
<web-app>
  <servlet>
  </servlet>
  <session-config>
    <session-timeout>15</session-timeout>
  </session-config>
</web-app>
```

gauravgambhir.cse@piet.co.in

Note



session-timeout value is specified in minutes.

session-timeout value of 0 or less means that the session will never expire

Ways Session Times Out



Setting session timeout for a specific session.

```
session.setMaxInactiveInterval(20*60);
```

Note: `setMaxInactiveInterval()` method accept seconds.

negative value means session will never expire.

gauravgambhir.cse@piet.co.in

Cookie



- Cookie is a piece of data exchanged between the client and server.
- The server send the cookie to the client, and the client returns the cookie when the clients makes another request.

PC-CS-305: Essentials of Information Technology

Cookie



Creating a new cookie

```
Cookie cookie=new Cookie("username",name);
```

Setting how long a cookie will live on the client

```
cookie.setMaxAge(30*60);
```

Sending the cookie to the client

```
response.addCookie(cookie);
```

gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

Cookie



Getting the cookie from the client request

```
Cookie[ ] cookies = request.getCookies();
for(int i=0;i<cookies.length;i++){
    Cookie cookie=cookies[i];
    If(cookie.getName().equals("username")){
        String username=cookie.getValue();
        out.println("Hello"+userName);
        break;
    }
}
```

PC-CS-305: Essentials of Information Technology

Servlet Creating & Setting Cookie



```
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
public class CookieTest extends HttpServlet{
    public void doPost(HttpServletRequest req,HttpServletResponse res)
        throws IOException,ServletException{

        res.setContentType("text/html");
        String name=req.getParameter("username");
        Cookie cookie=new Cookie("username",name);
        cookie.setMaxAge(30*60);
        res.addCookie(cookie);

        RequestDispatcher view=req.getRequestDispatcher("cookiesresult.jsp");
        view.forward(req,res);
    }
}
```

gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

Servlet Getting the Cookie



```
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;

public class CheckCookie extends HttpServlet{
    public void doPost(HttpServletRequest req,HttpServletResponse res)
        throws IOException,ServletException{

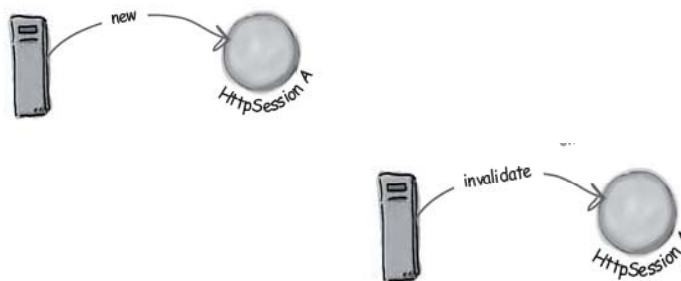
        res.setContentType("text/html");
        PrintWriter out=response.getWriter();
        Cookie[ ] cookies=req.getCookies();
        for(int i=0;i<cookies.length;i++){
            Cookie cookie=cookies[i];
            if(cookie.getName().equals("username")){
                String userName=cookie.getValue();
                out.println("Hello"+userName);
                break;
            }
        }
    }
}
```

PC-CS-305: Essentials of Information Technology

Key MileStones of HttpSession



The session is created or destroyed



gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

Key MileStones of HttpSession



The session attributes are added, removed or replaced

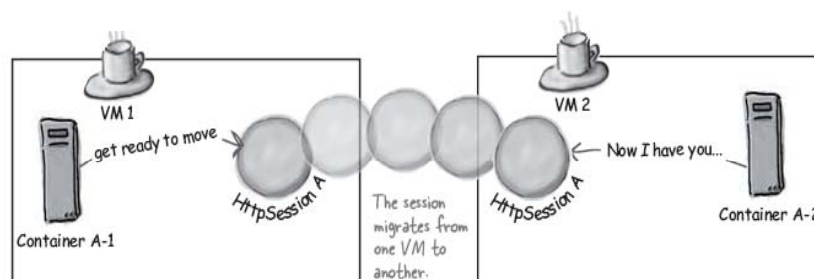


PC-CS-305: Essentials of Information Technology

Key MileStones of HttpSession



The session is passivated in one VM and activated in another



gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

Life Cycle Events



Milestone *Lifecycle*

Event and Listener Type

The session was created

HttpSessionEvent

When the container first creates a session, at this point the session is still considered new.

The session was destroyed

HttpSessionListener

When the container invalidates a session.

Life Cycle Events



Milestone *Attributes*

Event and Listener Type

An Attribute was added

HttpSessionBindingEvent

When some part of the application call setAttribute() on session.

The session was removed

When some part of the application call removeAttribute() on a session.

gauravgambhir.cse@piet.co.in

Life Cycle Events



Milestone *Attributes*

Event and Listener Type

An Attribute was replaced

HttpSessionAttributeListener

When some part of the application call setAttribute() on session.
and name of the attribute has already been bound to session.

Life Cycle Events



Milestone *Migration*

Event and Listener Type

The Session is about to be passivated

HttpSessionEvent

When the container is about to migrate the session into a different
VM. Called before the session moves so that attributes have a
chance to prepare themselves for migration

gauravgambhir.cse@piet.co.in

Life Cycle Events



Milestone
Migration

Event and Listener Type

The Session has been activated

HttpSessionActivationListener

When the container has just migrated the session into a different VM. Called before any other part of app can call `getAttribute()` on the session

PC-CS-305: Essentials of Information Technology

Examples



<http://localhost:8080/UsingSessionsServlet/UsingSessions>

<http://localhost:8080/SessionAttributes/Session>

<http://localhost:8080/SessionDisplay/SessionDisplay>

gauravgambhir.cse@piet.co.in

PC-CS-305: Essentials of Information Technology

Question and Answer Session



Q & A

gauravgambhir.cse@piet.co.in