SERVLET TECHNOLOGY

SEE THE INDEX

≡

Using Cookies for Session Management

Cookies are small pieces of information that are sent in response from the web server to the client. **Cookies** are the simplest technique used for storing client state.

Cookies are stored on client's computer. They have a lifespan and are destroyed by the client browser at the end of that lifespan.

Using Cookies for storing client state has one shortcoming though, if the client has turned of COokie saving settings in his browser then, client state can never be saved because the browser will not allow the application to store cookies.

Cookies API

Cookies are created using **Cookie** class present in Servlet API. Cookies are added to **response** object using the `addCookie()` method. This method sends cookie information over the HTTP response stream. `getCookies()` method is used to access the cookies that are added to response object.

Creating a new Cookie

```
Cookie ck = new Cookie("username",name);
```
← ——————— creating a new cookie object

Setting up lifespan for a cookie

```
ck.setMaxAge(30*60);
```
← ——— setting maximum age of cookie

Sending the cookie to the client
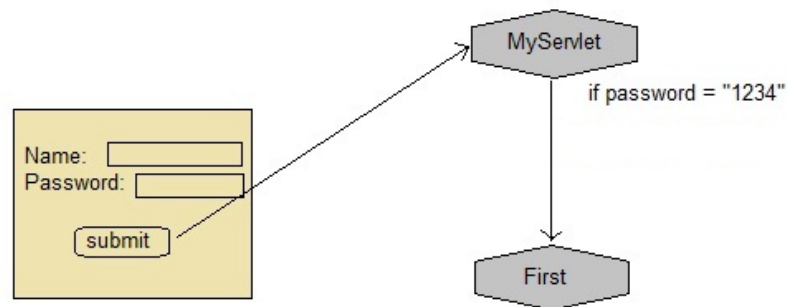
```
response.addCookie(ck);
```
← adding cookie to **response** object

Getting cookies from client request

```
Cookie[] cks = request.getCookies();
```
← getting the cookie for **request** object

Example demonstrating usage of Cookies

Below mentioned files are required for the example:

index.html

```html
<form method="post" action="validate">
  Name:<input type="text" name="user" /><br/>
  Password:<input type="text" name="pass" ><br/>
  <input type="submit" value="submit">
</form>
```

web.xml

```xml
<web-app...>

    <servlet>
        <servlet-name>validate</servlet-name>
        <servlet-class>MyServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>validate</servlet-name>
        <url-pattern>/validate</url-pattern>
    </servlet-mapping>

    <servlet>
        <servlet-name>First</servlet-name>
        <servlet-class>First</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>First</servlet-name>
        <url-pattern>/First</url-pattern>
    </servlet-mapping>

    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
    </welcome-file-list>

</web-app>
```

MyServlet.java

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

public class MyServlet extends HttpServlet {

  protected void doPost(HttpServletRequest request, HttpServletResponse response)
          throws ServletException, IOException {
      response.setContentType("text/html;charset=UTF-8");
      String name = request.getParameter("user");
      String pass = request.getParameter("pass");

      if(pass.equals("1234"))
      {
          Cookie ck = new Cookie("username",name);
          response.addCookie(ck);
          response.sendRedirect("First");
      }
    }
}
```

First.java

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

public class First extends HttpServlet {

  protected void doGet(HttpServletRequest request, HttpServletResponse response)
          throws ServletException, IOException {
      response.setContentType("text/html;charset=UTF-8");
      PrintWriter out = response.getWriter();
      Cookie[] cks = request.getCookies();
      out.println("Welcome "+cks[0].getValue());
    }
}
```