SY306 Web and Databases for Cyber Operations

Cookies: JavaScript and Python


Cookies Example

Hello, Paul.

Click here if you are not Paul

Welcome to cookies!
JavaScript: Using Cookies

• Cookie
  – Data stored on _____________ to maintain information about client during and between browser sessions
  – A string: identifier=value pairs separated by ;
  – Can be accessed through `document.cookie` property
  – Set expiration date using `expires` keyword
    – Use `escape` or `encodeURI` function to convert non-alphanumeric characters to hexadecimal escape sequences
    – `unescape` or `decodeURI` function converts hexadecimal escape sequences back to English characters

Storing Cookies – Simple Version

```javascript
document.writeln("<br/>Cookie is: "+document.cookie);

document.cookie = "name=" + escape("J Smith");
document.writeln("<br/>Cookie is: "+document.cookie);

document.cookie = "rank=" + escape("Captain");
document.writeln("<br/>Cookie is: "+document.cookie);
```
Reading Cookies – Simple Version

myCookies = document.cookie;

cookieElements = myCookies.split("=");

document.writeln("<br/>Identifier stored is: "+ cookieElements[0] +
"<br/>Value stored is: " + cookieElements[1]);

Simple Cookie Example

// reset the document's cookie if wrong person
function wrongPerson() {
  // reset the cookie
  document.cookie= "name=null;" + " expires=Thu, 01-Jan-95 00:00:01 GMT"
  // after removing the cookie reload the page to get a new name
  location.reload();
}

// determine whether there is a cookie
if ( document.cookie ) {
  var cookie = document.cookie;
  var cookieTokens = cookie.split("=");

  // set name to the part of the cookie that follows the = sign
  name = cookieTokens[1];
  name = unescape(name);
}
else { // if there was no cookie then ask the user to input a name
  name = window.prompt("Please enter your name", "Paul");
  document.cookie = "name=" + escape(name);
}
document.writeln("<h1>Hello, " + name + ". </h1> ");
document.writeln("<p><a href= 'javascript:wrongPerson()' > Click here if you are not " + name + "</a></p> ");
Exercise #1: JS:
Ask user for favorite quote using a window prompt.
Save quote in a cookie identified by “favQuote”.
Display quote on the page.

Storing Cookies – More Realistic

• By default, cookies expire when session ends
• Set “expires” attribute to make stick around longer

```
function createCookie(identifier,value,days) {
    if (days) {
        var date = new Date();
        date.setTime(date.getTime()+(days*24*60*60*1000));
        var expires = "; expires="+date.toGMTString();
    } else
        var expires = "";
    document.cookie = identifier=""+escape(value)+expires;
}
```

```
function eraseCookie(identifier) {
    createCookie(identifier,"",-1);
}
```

(modified from http://www.quirksmode.org/js/cookies.html)
Reading Cookies – More Realistic

// Return the 'value' of the cookie with identifier 'desiredId'
// returns null if no match found.
function readCookie(desiredId) {
    // First split the pairs apart on ';' '
    var pairs = document.cookie.split("; ");

    // Now split each pair on '='. Check if have a match
    for (var i=0; i < pairs.length; i++) {
        var aPair = pairs[i];

        // split into desired parts and check for match
        var cookieTokens = aPair.split("=");
        var id = cookieTokens[0];
        var value = cookieTokens[1];
        if (id == desiredId) {
            // found desired cookie -- return value
            return unescape(value);
        }
    }
    return null;  // no match
}

Exercise #2: JS: Read the value of cookie identified by “favQuote” and display it in a pop-up msg if it exists, otherwise display “no quotes”
Exercise #3: Re-write Simple Cookie Example using the helper functions

Cookies – Java Script and Python

- Cookies with JavaScript
  - Create cookie
    - `document.cookie = “color=red”;`
  - Read cookie (from JavaScript)
    - Read and parse `document.cookie`
    - Use `readCookie()` function to help with this
  - Where are cookies stored??

- Cookies with Python
  - Create cookie with `print()` BEFORE printing “Content-type …”
    - Sent to browser
  - Browser **always** sends appropriate cookies back to server with request
  - Read cookie
    - Access “HTTP_COOKIE” environment variable (from `os import environ`)
    - Use `SimpleCookie` class (from `http import cookies`)
  - Where are cookies stored??

- Cookies created with Python **can** be read via JavaScript and vice versa
Create Cookies with Python

(Assume this file invoked from a HTML form with fields name, and color)

```python
#!/usr/bin/env python3
from http import cookies
import urllib.parse, cgi, cgitb

# get parameters
params = cgi.FieldStorage()
name = params.getvalue("name")
height = params.getvalue("height")
color = params.getvalue("color")

# set cookies
# set expiration time in 1 hour
expires = 60*60;
mycookie = cookies.SimpleCookie()
mycookie["Name"] = name
mycookie["Color"] = color
mycookie["Color"]['expires'] = expires
print (mycookie) # BEFORE content-type line
print( "Content-type:text/html\n");

print ("
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>Storing cookies</title>
</head>
<body>
<h1>2 cookies were stored!</h1>
***
\n"+ name + "<br />
Color: "+color + ";
print("</body></html">
```

set10_createCookies.py – part1

set10_createCookies.py – part2
Read Cookies With Python

#!/usr/bin/env python3

from http import cookies
import os

# read cookies
name = ""
color = ""
if 'HTTP_COOKIE' in os.environ:
    cookie_string=os.environ.get('HTTP_COOKIE')
    mycookie=cookies.SimpleCookie()  
    mycookie.load(cookie_string)
    try:
        name=mycookie['Name'].value
        color = mycookie['Color'].value
    except KeyError:
        name=""
        color=""

print ("Content-type: text/html\n")

print ("\n"
<!DOCTYPE html>
<html>
<head>
<meta charset = "utf-8">
<title>Reading cookies</title>
</head>
<body>
<h1>Could not find cookies for Name or Color</h1>
</body>
</html>\n"

print ("\n")

Uses for Cookies

• Most common:
  – User logs in using secure page (https)
  – Server checks password. If good, creates cookie
    • E.g. “login=m168987&auth=356af12cd124552”
  – User redirected to other pages. These pages don’t ask for password – instead just check that have valid login cookie
  
  – Why do we need the auth field?

Exercise #4: Python: a) Create a cookie with identifier “favQuote” and content “DTT/FSA”
b) change your program to store the quote provided by user (not hardcoded) through CGI – param name “quote”
Remember

• Relevant cookies always sent by browser to the server

• Can create with JavaScript and read with Python

• Or create with Python and read with JavaScript