SY306 Web and Databases for Cyber Operations

Set #10: Cookies in JavaScript and Python

Cookies Example

https://www.w3schools.com/js/js_cookies.asp
https://docs.python.org/3/library/http.cookies.html
http://www.tutorialspoint.com/python/python cgi_programming.htm
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

Why Cookies
Cookie Attributes

- Expires
- Path
- Domain
- Secure
- Http-only

- Identifier (name)
- Value

JavaScript: Using cookies

Accessing a cookie:
```javascript
var cookies = document.cookie.split(";");
for( i = 0; i < cookies.length; i++ )
    var cookie = cookies[i].split("=");
```

Setting a cookie:
```javascript
document.cookie = "name=" + escape("J Smith");
document.cookie = "name=" + escape("Bob K");
document.cookie = "rank=" + escape("Captain");
```
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 it stick around longer

```javascript
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”
Cookies – Java Script and Python

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

- Cookies with Python
  - Ask browser to create cookie by printing “Set-cookie…” BEFORE printing “Content-type …”
  - 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

HTTP Protocol – HTTP Response

HTTP/1.0 200 OK

Set-Cookie: theme=light
Set-Cookie: session=5gd7324dx; Expires=Wed, 11 Oct 2018 12:27:03 GMT
Content-type: text/html

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
  ...
</html>
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("\n""
<!DOCTYPE html>
<html>
<head>
<meta charset = "utf-8">
<title>Storing cookies</title>
</head>
<body>
<h1>2 cookies were stored!</h1>
""
print(""""+ name + "br /" Color: "+color + ")
print("""");
```

set10_createCookies.py – part1

set10_createCookies.py – part2
# Read Cookies With Python

```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=collections.SimpleCookie()
    mycookie.load(cookie_string)
    try:
        name=mycookie['Name'].value
        color = mycookie['Color'].value
    except KeyError:
        name=""
print ("Content-type: text/html\n")

print ("\n")
</DOCTYPE html>
<html>
<head>
<meta charset = "utf-8">
<title>Reading cookies</title>
</head>
<body>
""
if name or color:
    print ("<h1>Cookies found!</h1>"
print ("<h2>Name: \"+name+\"</h2>"
print ("<h2>Color: \" + color +\"</h2>"
else:
    print ("<h1>Could not find cookies for Name or Color</h1>"
)
print("</body></html>")
```

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