

INFM 603: Information Technology and Organizational Context

Session 3: JavaScript – DOM and Events



Jimmy Lin
The iSchool
University of Maryland

Thursday, September 27, 2012



Programming... is a lot like cooking!

Arrays

- An array holds a collection of values
 - Each value is referenced with an index, starting from 0
- Creating an array:

```
var arr = new Array();  
arr[0] = 0;  
arr[1] = 3;  
arr[2] = 2;  
arr[3] = 4;
```

What happens if you don't specify
value for a particular index?

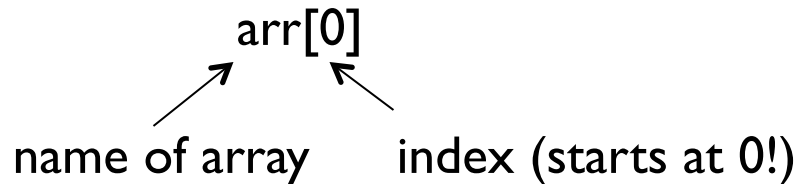
- Or, alternatively:

```
var arr = [0, 3, 2, 4];
```

- Note, arrays automatically grow in size

Using Arrays

- Referencing values in an array:



- Array values can be used in other expressions and statements:
`var f = 5 + arr[0] + arr[2];`
- Find out the length of an array: `arr.length`
- Arrays and *for* loops go hand in glove:

```
var arr = [0, 3, 2, 4];  
var sum = 0;  
for (var i=0; i<arr.length; i++) {  
    sum += arr[i];  
}  
console.log(sum);
```



Cooking analogy?

A large, leafy green tree stands prominently in the center of a field of bright yellow flowers. The tree's canopy is wide and dense, with many small green leaves. The field of flowers stretches out to the horizon, creating a vibrant yellow foreground. The sky above is blue with scattered white and grey clouds. The overall scene is peaceful and natural.

The Document Object Model (DOM)



The Document Object Model (DOM)

document

head

body

h1

p

p

ul

li

li

li

Asking the DOM to “do stuff”

the *method* is what you want the document “to do”, usually a verb phrase



```
document.method(“argument”);
```



document represents the entire page and contains the DOM



arguments are additional details that you specify

More on the dot notation later...

DOM: Selecting Nodes

- Selecting a DOM node by id:

 - `document.getElementById("id");`

 - Note, returns a DOM node

- Selecting DOM nodes by tag:

 - `document.getElementsByTagName("p");`

 - Note, returns a collection (treat as an array)

- Once you select a DOM node:

 - Get a node's children: `list.childNodes`
 - Get a node's number of children: `list.childNodes.length`
 - Natural to iterate over child nodes using for loops

BTW, `<div>` tags are very useful for grouping elements together.

DOM: Manipulating Nodes

- Simplest way to manipulate DOM nodes: select the node and modifying its innerHTML property:

```
var p = document.getElementById("para");
```

```
p.innerHTML = "some text";
```

- innerHTML can be *any* HTML!

- Modify a child node using innerHTML:

```
document.getElementById("list").childNodes[1].innerHTML = "new item";
```

DOM: Building Nodes

- Building DOM nodes programmatically:

```
var p = document.createElement("p");  
p.innerHTML = "here is some new text.";  
document.getElementById("div1").appendChild(p);
```

```
var newItem = document.createElement("li");  
newItem.innerHTML = "new list item";  
document.getElementById("list").appendChild(newItem);
```

- Set `setAttribute` method to set attributes

DOM: Removing Nodes

- Select the node to remove, then use the `removeChild` method in its parent:

```
var list = document.getElementById("list");  
var listItem = list.childNodes[1];  
list.removeChild(listItem);
```

Let's build a table!




```
var t = document.createElement("table");
t.setAttribute("border", 1);
var row1 = document.createElement("tr");
var row1col1 = document.createElement("td");
row1col1.innerHTML = "A";
var row1col2 = document.createElement("td");
row1col2.innerHTML = "B";

row1.appendChild(row1col1);
row1.appendChild(row1col2);

t.appendChild(row1);

document.getElementById("div1").appendChild(t);
```

Events

- GUI are driven by events
- When an event happens, an event handler is called to “handle” the event
- Easier to show in an example...

Note, what I’m showing is slightly easier than what’s in the book...

A wide-angle photograph of a massive concrete dam under construction. The dam's structure is composed of several large, light-colored concrete panels, some of which are being hoisted into place by a system of cables and cranes. A tall, dark metal lattice tower stands on the left side of the dam. In the background, a river flows through a valley, and distant mountains are visible under a clear blue sky. The foreground shows some greenery and a small waterfall on the left.

Putting everything together...