Introducing JavaScript and the DOM

Eric D. Wills
University of Oregon
var temp = 98.6;
var beanCounter = 4;
var reallyCool = true;
var motto = “I rule”;  
temp = (temp - 32)*5/9;
motto = motto + “ and so do you!”;
var pos = random();
JavaScript: Do things more than once

while (beanCounter > 0) {
    processBeans();
    beanCounter = beanCounter - 1;
}
JavaScript: Make decisions

If (isReallyCool) {
    invite = "You’re invited!";
} else {
    invite = "Sorry, we’re at capacity…";
}
<html lang="en">
<head>
  <title>Icecream Scoops</title>
  <meta charset="utf-8">
  <script>
    var scoops = 10;
    while (scoops >= 0) {
      if (scoops == 3) {
        alert("Ice cream is running low!");
      } else if (scoops > 9) {
        alert("Eat faster, the ice cream is going to melt!");
      } else if (scoops == 2) {
        alert("Going once!");
      } else if (scoops == 1) {
        alert("Going twice!");
      } else if (scoops == 0) {
        alert("Gone!");
      } else {
        alert("Still lots of ice cream left, come and get it.");
      }
      scoops = scoops - 1;
    }
    alert("life without ice cream isn't the same");
  </script>
</head>
<body>
  <h1>Counting ice cream scoops</h1>
</body>
</html>
<script>
var scoops = 10;
while (scoops >= 0) {
    if (scoops == 3) {
        alert("Ice cream is running low!");
    }
    else if (scoops > 9) {
        alert("Eat faster, the ice cream is going to melt!");
    }
    else if (scoops == 2) {
        alert("Going once!");
    }
    else if (scoops == 1) {
        alert("Going twice!");
    }
    else if (scoops == 0) {
        alert("Gone!");
    }
    else {
        alert("Still lots of ice cream left, come and get it.");
    }
    scoops = scoops - 1;
}
alert("life without ice cream isn't the same");
</script>
<script>
var scoops = 10;
while (scoops >= 0) {
    if (scoops == 3) {
        alert("Ice cream is running low!");
    }
    else if (scoops > 9) {
        alert("Eat faster, the ice cream is going to melt!");
    }
    else if (scoops == 2) {
        alert("Going once!");
    }
    else if (scoops == 1) {
        alert("Going twice!");
    }
    else if (scoops == 0) {
        alert("Gone!");
    }
    else {
        alert("Still lots of ice cream left, come and get it.");
    }
    scoops = scoops - 1;
}
alert("life without ice cream isn't the same");
</script>
Loop through scoops

- var scoops = 10;
  while (scoops >= 0) {
    ...
    scoops = scoops - 1;
  }

  - Loop through all 10 scoops (i.e., for each value of the scoops variable from 10 through 0).
  - Note that this is 11 iterations!
Ice Cream!

<script>
    var scoops = 10;
    while (scoops >= 0) {
        if (scoops == 3) {
            alert("Ice cream is running low!");
        }
        else if (scoops > 9) {
            alert("Eat faster, the ice cream is going to melt!");
        }
        else if (scoops == 2) {
            alert("Going once!");
        }
        else if (scoops == 1) {
            alert("Going twice!");
        }
        else if (scoops == 0) {
            alert("Gone!");
        }
        else {
            alert("Still lots of ice cream left, come and get it.");
        }
        scoops = scoops - 1;
    }
    alert("life without ice cream isn't the same");
</script>
Handle current scoop count

- if (scoops == 3) {
    alert("Ice cream is running low!");
}
else if (scoops > 9) {
    alert("Eat faster, the ice cream is going to melt!");
}
else if (scoops == 2) {
    alert("Going once!");
}
else if (scoops == 1) {
    alert("Going twice!");
}
else if (scoops == 0) {
    alert("Gone!");
}
else {
    alert("Still lots of ice cream left, come and get it.");
}

- Print different alerts for each value of the scoops variable (i.e., == 3, >9, ==2, ==1, ==0, or any other value.
<script>
    var scoops = 10;
    while (scoops >= 0) {
        if (scoops == 3) {
            alert("Ice cream is running low!");
        } else if (scoops > 9) {
            alert("Eat faster, the ice cream is going to melt!");
        } else if (scoops == 2) {
            alert("Going once!");
        } else if (scoops == 1) {
            alert("Going twice!");
        } else if (scoops == 0) {
            alert("Gone!");
        } else {
            alert("Still lots of ice cream left, come and get it.");
        }
        scoops = scoops - 1;
    }
    alert("life without ice cream isn't the same");
</script>
Random Numbers

• Useful functions:
  – Math.random()
    • Returns a pseudorandom real number between 0.0 (inclusive) and 1.0 (exclusive)
    • For example:
      – alert(Math.random())
  – Math.floor(number)
    • Returns the specified number rounded down to the nearest integer (i.e., whole number)
    • For example:
      – alert(Math.floor(10.5))
  – Use these together to create pseudorandom integers:
    • For example, print an integer between 0 (inclusive) and 9 (inclusive)
      – alert(Math.floor(Math.random()*10) )
Random Example

- alert(Math.floor(Math.random() * 10 + 1))
Random Example

• `alert(Math.floor(Math.random() * 10 + 1))`
  – This will print a random number between 1 (inclusive) and 10 (inclusive).
For loops

- for (var scoops = 0; scoops < 10; scoops++) {
  alert("There's more ice cream");
}

For loops

- `for (var scoops = 0; scoops < 10; scoops++) {
    alert("There's more ice cream!");
}

  – This will print 10 times. Why? An alert is printed for each scoops value (i.e., 0, 1, 2, 3, 4, 5, 6, 7, 8, 9).
Another For loop

• for (var berries = 5; berries > 0; berries--) {
    alert("Eating a berry");
}
Another For loop

• for (var berries = 5; berries > 0; berries--) {
    alert("Eating a berry");
}

  – This will print 5 times. Why? An alert is printed for each berries value (i.e., 5, 4, 3, 2, 1).
Yet another For loop

• var count = 0;
  
  for (var i = 0; i < 5; i++) {
    count = count + i;
  }

  alert("count is " + count);
Yet another For loop

- var count = 0;
  for (var i = 0; i < 5; i++) {
    count = count + i;
  }
  alert("count is " + count);
  - This will print 10. Why? The variable i has the values 0 to 4 (i.e., <5), and 0 + 1 + 2 + 3 + 4 = 10.
Nested loops

• var tops = 5;
  while (tops > 0) {
    for (var spins = 0; spins < 3; spins++) {
      alert("Top " + tops + " is spinning!");
    }
    tops = tops - 1;
  }
Nested loops

• var tops = 5;
  while (tops > 0) {
    for (var spins = 0; spins < 3; spins++) {
      alert("Top " + tops + " is spinning!");
    }
    tops = tops - 1;
  }

  – This will print 15 times. Why? The inner loop prints 3 times (i.e., for spins values of 0, 1, 2), and the outer loop executes the inner loop 5 times (i.e., for tops values of 5, 4, 3, 2, 1).
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Movies</title>
  <meta charset="utf-8">
</head>
<body>
<h1>Movie Showtimes</h1>
<h2 id="movie1">Plan 9 from Outer Space</h2>
<p>
  Playing at 3:00pm, 7:00pm.
</p>
<span>
  Special showing tonight at <em>midnight</em>!
</span>
<p>
  Playing at 3:00pm, 7:00pm.
</p>
<h2 id="movie2">Forbidden Planet</h2>
<p>
  Playing at 5:00pm, 9:00pm.
</p>
</body>
</html>
The DOM

• `<h2 id="movie1">Plan 9 from Outer Space</h2>`
  – Assign ID “movie1” to this header element so that it can be accessed from JavaScript.

• `<h2 id="movie2">Forbidden Planet</h2>`
  – Same idea but with ID “movie2”.
<!doctype html>
<html lang="en">
<head>
    <title>Movies</title>
    <meta charset="utf-8">
</head>
<body>
    <h1>Movie Showtimes</h1>
    <h2 id="movie1">Plan 9 from Outer Space</h2>
    <p>
        Playing at 3:00pm, 7:00pm.
        <span>
            Special showing tonight at <em>midnight</em>!
        </span>
    </p>
    <h2 id="movie2">Forbidden Planet</h2>
    <p>
        Playing at 5:00pm, 9:00pm.
    </p>
</body>
</html>
Modifying the DOM

<!doctype html>
<html lang="en">
<head>
    <title>Planets</title>
    <meta charset="utf-8">
    <script>
        function init() {
            var planet = document.getElementById("greenplanet");
            planet.innerHTML = "Red Alert: hit by phaser fire!";
        }
        window.onload = init;
    </script>
</head>
<body>
    <h1>Green Planet</h1>
    <p id="greenplanet">All is well</p>
    <h1>Red Planet</h1>
    <p id="redplanet">Nothing to report</p>
    <h1>Blue Planet</h1>
    <p id="blueplanet">All systems A-OK</p>
</body>
</html>
Modifying the DOM

<script>
    function init() {
        var planet = document.getElementById("greenplanet");
        planet.innerHTML = "Red Alert: hit by phaser fire!";
    }
    window.onload = init;
</script>
Modifying the DOM

<script>
    function init() {
        var planet = document.getElementById("greenplanet");
        planet.innerHTML = "Red Alert: hit by phaser fire!";
    }
    window.onload = init;
</script>
Modifying the DOM

• function init() {
  ...
}
  – Define a function which will be executed when the page is loaded.

• window.onload = init;
  – Specify that the window will call the init function when it has finished loading.
Modifying the DOM

```html
<script>
    function init() {
        var planet = document.getElementById("greenplanet");
        planet.innerHTML = "Red Alert: hit by phaser fire!";
    }
    window.onload = init;
</script>
```
Modifying the DOM

- var planet =
  document.getElementById("greenplanet");
  - Set the variable planet to the element with ID “greenplanet”.
- planet.innerHTML = "Red Alert: hit by phaser fire!";
  - Set the inner HTML of the planet element to the specified literal text.
Modifying the DOM

```javascript
function init() {
    var planet = document.getElementById("greenplanet");
    planet.innerHTML = "Red Alert: hit by phaser fire!";
}
window.onload = init;
</script>
Another DOM example

```html
<!doctype html>
<html lang="en">
<head>
  <title>My Playlist</title>
  <meta charset="utf-8">
  <script charset="utf-8">
    function addSongs() {
      var song1 = document.getElementById("song1");
      var song2 = document.getElementById("song2");
      var song3 = document.getElementById("song3");
      song1.innerHTML = "Blue Suede Strings, by Elvis Pagely";
      song2.innerHTML = "Great Objects on Fire, by Jerry JSON Lewis";
      song3.innerHTML = "I Code the Line, by Johnny JavaScript";
    }
    window.onload = addSongs;
  </script>
</head>
<body>
  <h1>My awesome playlist</h1>
  <ul id="playlist">
    <li id="song1"></li>
    <li id="song2"></li>
    <li id="song3"></li>
  </ul>
</body>
</html>
```
Another DOM example

```javascript
function addSongs() {
    var song1 = document.getElementById("song1");
    var song2 = document.getElementById("song2");
    var song3 = document.getElementById("song3");
    song1.innerHTML = "Blue Suede Strings, by Elvis Pagely";
    song2.innerHTML = "Great Objects on Fire, by Jerry JSON Lewis";
    song3.innerHTML = "I Code the Line, by Johnny JavaScript";
}
window.onload = addSongs;
</script>
```
Arrays

- Store multiple values in a variable using:
  - `var myArray = [1, 3, 5, “a”]`
- Access array values using square brackets:
  - `myArray[0], myArray[1], etc.`
  - For example:
    - `alert(myArray[0])`
- The number of items in the array is accessed as:
  - `myArray.length`
  - For example:
    - `alert(myArray.length)`
Phrase-o-matic

<html lang="en">
<head>
    <title>Phrase-o-matic</title>
    <meta charset="utf-8">
    <script>
    function makePhrases() {
        var words1 = ["24/7", "multi-Tier", "30,000 foot", "B-to-B", "win-win"];  
        var words2 = ["empowered", "value-added", "oriented", "focused", "aligned"];  
        var words3 = ["process", "solution", "tipping-point", "strategy", "vision"];  

        var rand1 = Math.floor(Math.random())*words1.length;  
        var rand2 = Math.floor(Math.random())*words2.length;  
        var rand3 = Math.floor(Math.random())*words3.length;  

        var phrase = words1[rand1] + " " + words2[rand2] + " " + words3[rand3];  
        var phraseElement = document.getElementById("phrase");  

        phraseElement.innerHTML = phrase;
    }
    window.onload = makePhrases;
    </script>
</head>
<body>
    <h1>Phrase-o-Matic says:</h1>
    <p id="phrase"></p>
</body>
</html>
Phrase-o-matic

<script>
  function makePhrases() {
    var words1 = ["24/7", "multi-Tier", "30,000 foot", "B-to-B", "win-win"];
    var words2 = ["empowered", "value-added", "oriented", "focused", "aligned"];
    var words3 = ["process", "solution", "tipping-point", "strategy", "vision"];
    var rand1 = Math.floor(Math.random()*words1.length);
    var rand2 = Math.floor(Math.random()*words2.length);
    var rand3 = Math.floor(Math.random()*words3.length);
    var phrase = words1[rand1] + " " + words2[rand2] + " " + words3[rand3];
    var phraseElement = document.getElementById("phrase");
    phraseElement.innerHTML = phrase;
  }
  window.onload = makePhrases;
</script>
Phrase-o-matic

<script>
  function makePhrases() {
    var words1 = ["24/7", "multi-Tier", "30,000 foot", "B-to-B", "win-win"];  
    var words2 = ["empowered", "value-added", "oriented", "focused", "aligned"];    
    var words3 = ["process", "solution", "tipping-point", "strategy", "vision"];   

    var rand1 = Math.floor(Math.random()*words1.length); 
    var rand2 = Math.floor(Math.random()*words2.length); 
    var rand3 = Math.floor(Math.random()*words3.length); 

    var phrase = words1[rand1] + " " + words2[rand2] + " " + words3[rand3]; 
    var phraseElement = document.getElementById("phrase"); 

    phraseElement.innerHTML = phrase; 
  }

  window.onload = makePhrases;
</script>
Phrase-o-matic

• var words1 = ["24/7", "multi-Tier", "30,000 foot", "B-to-B", "win-win"];

  – Creates an array containing the literal text values “24/7”, “multi-Tier”, “30,000 foot”, “B-to-B”, and “win-win” and assigns the array to the variable words1.
Phrase-o-matic

```javascript
function makePhrases() {
    var words1 = ["24/7", "multi-Tier", "30,000 foot", "B-to-B", "win-win"];
    var words2 = ["empowered", "value-added", "oriented", "focused", "aligned"];
    var words3 = ["process", "solution", "tipping-point", "strategy", "vision"];

    var rand1 = Math.floor(Math.random()*words1.length);
    var rand2 = Math.floor(Math.random()*words2.length);
    var rand3 = Math.floor(Math.random()*words3.length);

    var phrase = words1[rand1] + " " + words2[rand2] + " " + words3[rand3];
    var phraseElement = document.getElementById("phrase");

    phraseElement.innerHTML = phrase;
}

window.onload = makePhrases;
</script>
Phrase-o-matic

• var rand1 =
  Math.floor(Math.random()*words1.length);
  – Creates a random integer between 0 (inclusive) and the length of the words1 array (exclusive) and assigns it to the rand1 variable.
Phrase-o-matic

<script>
    function makePhrases() {
        var words1 = ["24/7", "multi-Tier", "30,000 foot", "B-to-B", "win-win"];
        var words2 = ["empowered", "value-added", "oriented", "focused", "aligned"];
        var words3 = ["process", "solution", "tipping-point", "strategy", "vision"];

        var rand1 = Math.floor(Math.random() * words1.length);
        var rand2 = Math.floor(Math.random() * words2.length);
        var rand3 = Math.floor(Math.random() * words3.length);

        var phrase = words1[rand1] + " " + words2[rand2] + " " + words3[rand3];
        var phraseElement = document.getElementById("phrase");

        phraseElement.innerHTML = phrase;
    }
    window.onload = makePhrases;
</script>
Phrase-o-matic

• var phrase = words1[rand1] + " " + words2[rand2] + " " + words3[rand3];
  – Looks up the value in words1 at the index specified by the rand1 variable and similar for words2, rand2 and words3, rand3; then assigns the phrase variable to the concatenation of these text values with a space between each word.
Phrase-o-matic

<script>
function makePhrases() {
    var words1 = ["24/7", "multi-Tier", "30,000 foot", "B-to-B", "win-win"]; 
    var words2 = ["empowered", "value-added", "oriented", "focused", "aligned"]; 
    var words3 = ["process", "solution", "tipping-point", "strategy", "vision"]; 

    var rand1 = Math.floor(Math.random() * words1.length); 
    var rand2 = Math.floor(Math.random() * words2.length); 
    var rand3 = Math.floor(Math.random() * words3.length); 

    var phrase = words1[rand1] + " " + words2[rand2] + " " + words3[rand3]; 
    var phraseElement = document.getElementById("phrase"); 

    phraseElement.innerHTML = phrase;
}
window.onload = makePhrases;
</script>