



End papers...

- Mistakes, obviously, show us what needs improving. Without mistakes, how would we know what we had to work on?

~Peter McWilliams, *Life 101*

- Mistakes are the portals of discovery.

~James Joyce (1882 - 1941)



Announcements

- New labs with quizzes
 - * Take the quiz as many times as you want to improve your score
 - * Bring questions to office hours and drop-in labs
 - Drop-in labs on Wednesdays *before* lecture are very sparsely attended—2 to 3 people!
10:30am to 12:30pm in MGH 430



Project 2A

- JavaScript Storyteller



Iteration Principles

Again, and again, and again



Looping back through some fundamental concepts

BROWSERS, HTML, & JAVASCRIPT



JavaScript

- We are now in a foreign land where you don't speak the language
- You recognize some sign posts—html—but JavaScript is brand new



JavaScript and HTML

Similarities

- Upload to Web space (Students server)
- HTML
- View in any Web browser
- Can use Dreamweaver or free downloads:
 - * Notepad++, Notepad2, jEdit
 - * Links on FIT 100 Computing page

Differences

- Validating doesn't help you fix JavaScript
- Test in Firefox
 - * Error console



Why JavaScript?

- Javascript is a programming language that many Web browsers can understand, or interpret.
- JavaScripts can be written with a simple text editor like NotePad++ or TextWrangler, or Dreamweaver, and tested in a Web browser.
- A Javascript program is a list of commands or statements that the browser runs to add features to an HTML document.



The goals of programming:

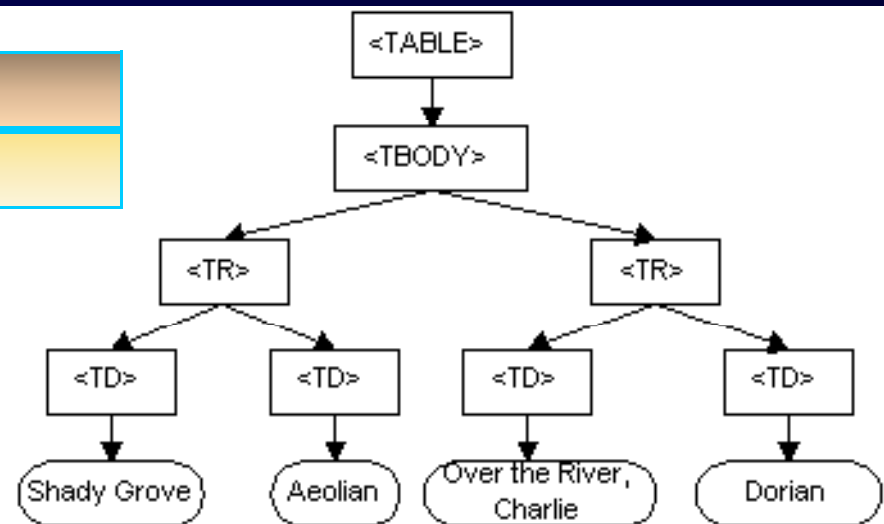
- Increase user interaction in three ways:
 - * Accept user input.
 - A user can invoke actions and/or enter data. Programmers call these actions and/or data *user input*, or simply *input*.
 - * Calculate using user input.
 - The computer can calculate a number, create some text, etc.
 - * Display results.
 - The results shown to the user are called output data, or simply output. Sometimes the output will change the graphical user interface (GUI) of the program, which changes what the user sees and can do with the program.



Running JavaScripts

- How the Web browser reads an HTML document that contains Javascript:
 1. The web browser starts to read through the HTML document from the first character in the top left to the last character in the bottom right.
 2. As the web browser reads through the document, the browser builds a document tree from the HTML elements it encounters.

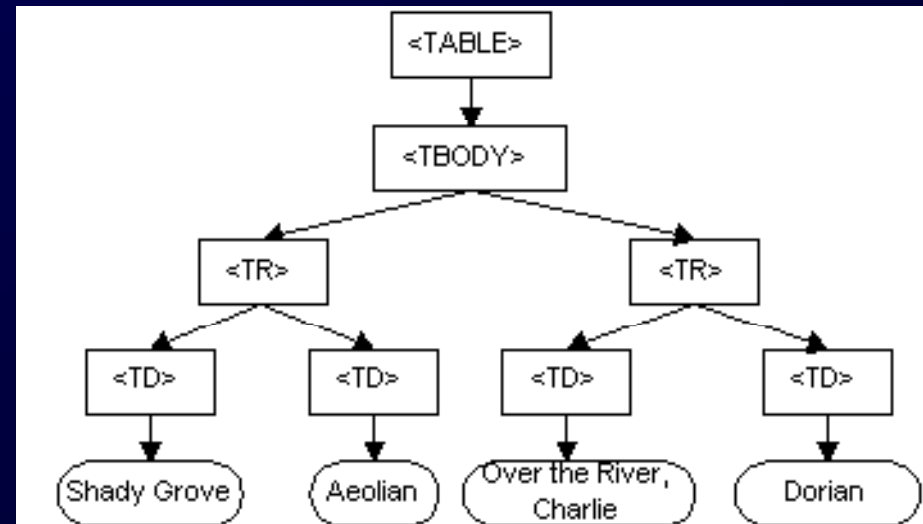
Shady Grove	Aeolian
Over the River, Charlie	Dorian





Running Javascript

- How the Web browser reads an HTML document that contains Javascript:
 3. If the browser encounters a script element, it executes the code that is specified by the src attribute. If there is no src attribute defined, the browser executes the code that is in the contents of the script element.
 4. If the script generates HTML output, this output is then added to the HTML document tree.





Play it again, Sam.

ITERATION



Definitions

- Iteration, or looping, is the process of repetition:
 - * looping through a series of statements to repeat them



Major Types of Iterations

- For loop
 - * Count up
 - * Count down
- While loop
 - * Count up
 - * Count down
- Do...While



Repetition is good

FOR LOOPS



The **for** Loop Basic Syntax

```
for (<initialization>; <continuation>; <next iteration>)  
{  
    <statement list>  
}
```

- Text that is not in *metabrackets* <> must be given literally
- The whole sequence of statements in the statement list is performed for each iteration
 - * Computer completes the whole statement sequence of the <statement list> before beginning the next iteration



The Iteration Variable

- *Control specification*: the three operations in the parentheses of the **for** loop
 - * Control the number of times the loop iterates
 - * by using an *iteration variable* (must be declared)



JavaScript Rules for **for** Loops (cont'd)

- The World-Famous Iteration

- * JavaScript uses the same `for` loop statement as other programming languages, so thousands of loops with this structure are written every day:

```
for ( j = 0; j < n; j++ ) {...}
```

- * Most frequently written **for** loop of all time
- * Easy to see iteration count:
 - Always n times



The Iteration Variable (cont'd)

- Example:

```
for ( j = 1 ; j <= 3 ; j = j + 1 ) {  
    <statement list>  
}
```

- Here's what happens:
 - * The first operation is the *<initialization>*
 - Sets the iteration variable's value for the first iteration of the loop. Done only once.
 - * The next operation is *<continuation>*
 - Test. If the test has a false outcome, the *<statement list>* is skipped.
 - If the test has a true outcome, the *<statement list>* is performed. When the statements are complete, the
 - * *<next iteration>* operation is performed
 - Repeats with the continuation test, performs same sequence of steps.



Running through a `for` loop

Table 21.1 The sequence of operations on `j` from the `for` loop with control specification (`j=1; j<=3; j=j+1`)

Operation	Operation Result	Role
<code>j = 1</code>	<code>j</code> 's value is 1	Initialize iteration variable
<code>j <= 3</code>	true, <code>j</code> is <i>less than</i> 3	First <i><continuation></i> test, continue
<code>j = j + 1</code>	<code>j</code> 's value is 2	First <i><next iteration></i> operation
<code>j <= 3</code>	true, <code>j</code> is <i>less than</i> 3	Second <i><continuation></i> test, continue
<code>j = j + 1</code>	<code>j</code> 's value is 3	Second <i><next iteration></i> operation
<code>j <= 3</code>	true, <code>j</code> is <i>equal to</i> 3	Third <i><continuation></i> test, continue
<code>j = j + 1</code>	<code>j</code> 's value is 4	Third <i><next iteration></i> operation
<code>j <= 3</code>	false, <code>j</code> is <i>greater than</i> 3	Fourth <i><continuation></i> test, terminate



How a **for** Loop Works

- Consider a computation on declared variables **j** and **text**

```
text = "She said ";  
for ( j = 1; j <= 3; j = j + 1 )  
{  
    text = text + "Never! ";  
}  
alert(text);
```



How a **for** Loop Works

- Consider a computation on declared variables **j** and **text**

```
text = "She said ";  
for ( j = 1; j <= 3; j = j + 1 )  
{  
    text = text + "Never! ";  
}  
alert(text);
```

Starting point



How a **for** Loop Works

- Consider a computation on declared variables **j** and **text**

```
text = "She said ";  
for ( j = 1; j <= 3; j = j + 1 )  
{  
    text = text + "Never! ";  
}  
alert(text);
```

Stop condition



How a **for** Loop Works

- Consider a computation on declared variables **j** and **text**

```
text = "She said ";  
for ( j = 1; j <= 3; j = j + 1 )  
{  
    text = text + "Never! ";  
}  
alert(text);
```

Step size or
increment



How a **for** Loop Works

- Demo:

```
text = "The two-year-old said ";  
for ( j = 1; j <= 3; j = j + 1 )  
{  
    text = text + "No! ";  
}  
alert(text);
```



JavaScript Rules for **for** Loops

- The Iteration Variable: `j = 1;`
 - * Must be declared, and follow rules for variable identifiers
 - * i, j, and k are the most common choices
- The Starting Point
 - * Iteration can begin anywhere, including negative numbers



JavaScript Rules for **for** Loops

- Continuation/Termination Test $j \leq 3$
 - * *Test* is any expression resulting in a Boolean value (true/false)
 - * Continuation must involve iteration variable to avoid infinite loop
- Step Size $j = j + 1$
 - * Amount of change from one iteration to the next
 - * Often called the *increment* or *decrement*
 - Increment: $j + 1$
 - Decrement: $j - 1$



Experiments with Flipping Coins

- To practice **for** loops, we experiment with flipping electronic coins
- We can use the function **randNum(2)**, which returns either 0 (tails) or 1 (heads)
- Set up an iteration in which our `randNum()` function is performed 100 times, and statistics gathered

Experiments with Flipping Coins (cont'd)



```
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
<head>
  <meta http-equiv="content-type"
    content="text/html; charset=utf-8" />
  <meta http-equiv="Content-Style-Type" content="text/css" />
  <title>For loop example with coin toss</title>
  <script type="text/javascript">
    var heads=0, tails=0;           //Counters
    var i;                          //Iteration variable
    for (i=0; i<100; i++ )
    {
      if (randNum(2) == 1)
        heads++;
      else
        tails++;
    }
    alert("Heads: " + heads + " and Tails: " + tails);
    function randNum(range)
    {
      return Math.floor(range*Math.random());
    }
  </script>
</head>
```



Demonstration

- Coin toss...



Experiments with Flipping Coins (cont'd)

- `i` ranges from 0 to 99, so the loop iterates 100 times
- Conditional statement checks and records the outcome of random number generation
- When random number is 1, count of heads is increased by 1 (`heads++;`)
- When random number is 0, count of tails is increased by 1 (`tails++;`)



Experiments with Flipping Coins (cont'd)

- A Nested Loop
 - * To run several trials, consider the entire loop we just looked at as one Trial
 - * Create another **for** loop containing this Trial unit, adding a couple of needed statements
 - * We have a loop within a loop (*nested loop*) which causes the Trial loop (0-99) to run five times

Experiments with Flipping Coins

```
4 <html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
5 <head>
6   <meta http-equiv="content-type"
7     content="text/html;charset=utf-8" />
8   <meta http-equiv="Content-Style-Type" content="text/css" />
9   <title>For loop example with coin toss</title>
10  <script type="text/javascript">
11    var heads = 0, tails = 0;
12    var i, j, text='';           //Iteration vars
13    for (j = 0; j < 5; j++)     //Outer loop start
14    {
15      for (i=0; i<100; i++)      //Trial line 1
16      {
17        if (randNum(2) == 1)    //Trial line 2
18          heads++;              //Trial line 3
19        else                    //Trial line 4
20          tails++;              //Trial line 5
21      }                          //Trial line 6
22
23      text = text + 'Trial ' + j + ': '; //Add line to message that will print at end
24      for (i = 0; i < (Math.abs(heads-50)); i++)
25      {
26        text = text + '*';      //Add to message
27      }
28      text = text + '\n';      //Add line break (html <br /> cannot be used for alerts)
29      heads = 0; tails = 0;    //Additional
30    }                          //Outer loop end
31    alert(text);
32    function randNum(range)
33    {
34      return Math.floor(range*Math.random());
35    }
36  </script>
37 </head>
```

Experiments with Flipping Coins

```
4 <html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
5 <head>
6   <meta http-equiv="content-type"
7     content="text/html; charset=utf-8" />
8   <meta http-equiv="Content-Style-Type" content="text/css" />
9   <title>For loop example with coin toss</title>
10  <script type="text/javascript">
11    var heads = 0, tails = 0;
12    var i, j, text=''; //Iteration vars
13    for (j = 0; j < 5; j++) //Outer loop start
14    {
15      for (i=0; i<100; i++) //Trial line 1
16      {
17        if (randNum(2) == 1) //Trial line 2
18          heads++; //Trial line 3
19        else //Trial line 4
20          tails++; //Trial line 5
21      } //Trial line 6
22
23      text = text + 'Trial ' + j + ': '; //Add line to message that will print at end
24      for (i = 0; i < (Math.abs(heads-50)); i++)
25      {
26        text = text + '*'; //Add to message
27      }
28      text = text + '\n'; //Add line break (html <br /> cannot be used for alerts)
29      heads = 0; tails = 0; //Additional
30    } //Outer loop end
31    alert(text);
32    function randNum(range)
33    {
34      return Math.floor(range*Math.random());
35    }
36  </script>
37 </head>
```

• Demo...

Experiments with Flipping Coins

```
4 <html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
5 <head>
6   <meta http-equiv="content-type"
7     content="text/html; charset=utf-8" />
8   <meta http-equiv="Content-Style-Type" content="text/css" />
9   <title>For loop example with coin toss</title>
10  <script type="text/javascript">
11    var heads = 0, tails = 0;
12    var i, j, text=''; //Iteration vars
13    for (j = 0; j < 5; j++) //Outer loop start
14    {
15      for (i=0; i<100; i++) //Trial line 1
16      {
17        if (randNum(2) == 1) //Trial line 2
18          heads++; //Trial line 3
19        else //Trial line 4
20          tails++; //Trial line 5
21      } //Trial line 6
22
23      text = text + 'Trial ' + j + ': '; //Add line to message that will print at end
24      for (i = 0; i < (Math.abs(heads-50)); i++)
25      {
26        text = text + '*'; //Add to message
27      }
28      text = text + '\n'; //Add line break (html <br /> cannot be used for alerts)
29      heads = 0; tails = 0; //Additional
30    } //Outer loop end
31    alert(text);
32    function randNum(range)
33    {
34      return Math.floor(range*Math.random());
35    }
36  </script>
37 </head>
```

Experiments with Flipping Coins

```
4 <html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
5 <head>
6   <meta http-equiv="content-type"
7     content="text/html; charset=utf-8" />
8   <meta http-equiv="Content-Style-Type" content="text/css" />
9   <title>For loop example with coin toss</title>
10  <script type="text/javascript">
11    var heads = 0, tails = 0;
12    var i, j, text=''; //Iteration vars
13    for (j = 0; j < 5; j++) //Outer loop start
14    {
15      for (i=0; i<100; i++) //Trial line 1
16      {
17        if (randNum(2) == 1) //Trial line 2
18          heads++; //Trial line 3
19        else //Trial line 4
20          tails++; //Trial line 5
21      } //Trial line 6
22
23      text = text + 'Trial ' + j + ': '; //Add line to message that will print at end
24      for (i = 0; i < (Math.abs(heads-50)); i++)
25      {
26        text = text + '*'; //Add to message
27      }
28      text = text + '\n'; //Add line break (html <br /> cannot be used for alerts)
29      heads = 0; tails = 0; //Additional
30    } //Outer loop end
31    alert(text);
32    function randNum(range)
33    {
34      return Math.floor(range*Math.random());
35    }
36  </script>
37 </head>
```



Experiments with Flipping Coins (cont'd)

- A Diagram of Results
 - * To show how far off a perfect 50-50 score a trial is, display with diagram
 - * Compute the distance from 50-50 and show that number using asterisks

```
text = text + 'Trial ' + j + ': ';  
for (i = 0; i < (Math.abs(heads-50)); i++) {  
    text = text + '*';  
}  
text = text + '\n';  
alert(text);
```



Quick Write

- Closed book
- Your own work
- Put away
 - * Laptops
 - * Cell phones
 - * Notebooks
 - * Books
 - * Etc.
- Stop talking
- Eyes to yourself
- Raise your hand if you have a question