FIT 100: Fluency with Information Technology

LAB 9 / 10: JavaScript Objects, Methods, Conditionals and Expressions

Required Reading for Lab 9/10

- W3Schools: http://www.w3schools.com/js/js_conditionals.asp

Review Reading for Lab 9/10

- Chapter 17 and 18 of the FIT course pack

1. Establish a Date object name. 1
2. Use the getDate() method to assign the current day to a new variable 3
3. Test your variable by printing it to your webpage. 4
4. Create a conditional in plain English... 5
5. Write a JavaScript conditional that will change your background color. 6
6. Add an *else* Statement to your Conditional. 7
7. Replace document.getSeconds() with a variable. 8
8. Use an Expression to add a label before your current second............................ 9
9. Use an expression to print out the complete date on your webpage................. 10

1. Establish a Date object name.

The Date Object:

There are many pre-defined objects in JavaScript and the Date object is one of them. The Date object has many methods associated with it, including getDay(), getMonth(), and getSeconds(). Because methods are something that an object *does* you can probably guess what these methods would do. (In a moment we will try it out.)

Note: See the appendix to the JavaScript book for a list of other methods for Date or go to W3Schools: http://www.w3schools.com/js/js_datetime.asp and scroll down to the information on the Date object and methods.

Instances:

In JavaScript, before we can use a built-in object, we must create a variable that will hold this particular instance of the object. Just as there are many different instances of a book in real life, in JavaScript when we talk about an object we must name the particular instance of the object. This allows us to uniquely identify that particular instance of the object by using a variable name.

Format:

```
variablename = new Object()
```

A. Adding onto lab8.html, establish the beginning and end of your script. You will put this script in the body of your HTML above the <hr> tag and below the <br> tag. Reference Lab 8, Exercise 1 if you need to.

Note: Remember to protect your script from showing up with older, non JavaScript browsers by using comments!

B. Choose a name that will hold your date variable. Once you have done so, assign that name a new date instance using the format used above.

   Example:

```
YourVariableNameForTheDate = new Date()
```

2. Use the getDate() method to assign the current day to a new variable

Assignment:

To get today's date you must use the getDate() method on the instance of the object (Date) that you have created. Once you know how to get the date, you must assign that information to a variable that will hold the value for you. As you know from your variable assignment, you can assign a variable a specific value (such as "dog" or 5) or you may assign it values based on the results of any statement. For example:

modified = document.lastModified.

document.lastModified is assigned to the value of the property, lastModified and will be stored in the variable modified.

Note: Remember that when you use a method of an object you use the following syntax: **object.method()** Remember that lastModified is a **property** and therefore does not have the trailing ().

C. Create (declare) another new variable to hold the day of the month.

D. Assign this new variable returned by the getDate method:

    YourVariableNameForTodaysDate = YourVariableNameForTheDate.getDate()

Remember: Like everything in JavaScript variable names and methods are case sensitive. Note that getDate needs to be spelled with just that capitalization while your variable name should have the exact capitalization that you used when you declared it.

3. Test your variable by printing it to your webpage.

E. Print the current date to your webpage using the document.write() method.

F. Save and test your page. Does it print the day of the month to your webpage?

Note: The location of your script tag will determine where the document.write() method will print the day of the month. If it is not in the place that you would like it to be, try adding another <script> tag somewhere else on your page and print the current date in that location.

G. Use the document.write() method to add formatting and text before the current date. Two examples are:

```
document.write(“Today’s date is: ”)
```

and

```
document.write(“<p>”)
```

Note: **To write HTML onto your webpage from within the <script> tag you must print it to your document using the document.write() method. Remember to use quotes around any formatting or text.**

4. Create a conditional in plain English.

For this exercise we are going to write out, in plain English, what we want JavaScript to help us do in our next exercise. We know from Chapter 17 and lecture that conditionals are ways that we allow a program to make decisions. To get a taste of conditionals, we are going to write some JavaScript that will cause our webpage to have different background colors depending on the time. While this may not be a terribly useful conditional, it is easy to test.

We know from our earlier work with the Date object that we are able to use several methods to get specific information relating to the current date and time. One of these methods will allow us to get the current second. Because we know that computers are best able to work with True/False decisions, we need to phrase our condition as being dependent on a true/false outcome. For example:

> If the cost of a piece of cake is less then $3.00, buy a piece of cake.

"Is the cake less then $3.00?" True or False? If True, then the next part of the statement should be executed and the cake is purchased. If false, nothing happens.

H. Just after the code statements you have written for Exercise 3, start a multi-line comment in JavaScript.

Note: Remember that the start of a multi-line comment is: /* and the end is: */

I. In this comment, write in plain English, a conditional that will check what second it is and then change the background to a specific color. Remember that you can use whatever comparison operators (review lesson 5 for more information) you want here. Also, remember to use specifics such as the exact color the background will change to.

J. Read over the statement you just wrote. Is the outcome predictable? If you know the second, will you always know what color the background is going to be?

5. Write a JavaScript conditional that will change your background color.

Now that you have thought out your conditional, we are going to set about writing this conditional. As you know from reading Chapter 17 of the FIT book, the basic conditional form is:

```
if (T/F Expression)
        {
                code statements;
        }
```

A quick note about curly brackets and semicolons in JavaScript:

If your T/F expression is true, your conditional will run all code statements within the following curly brackets. Semicolons (the ; character) allow you to separate each code statement so that you may do several things in response to a true result of your conditional. Because JavaScript, like HTML, does not take spacing into account when it reads your code, semicolons are important when you want to do two or more separate things in order.

a. Construct the first line of your conditional (This can be placed directly after Exercise 4)

For example:

```
if (dateobjectvariable.getSeconds() = = 12)
        {

        }
```

> ***Note:*** **It would be silly to use the *is equal to* operator here because of the unlikely event that the page would be loaded when the second equaled 12. We suggest using a greater or less than operator.**

b. Add the code statement between the curly brackets that will change the background color. Remember that you should first identify the object you want to effect (the document) and then the property.

c. Save your page and test it. Does the color change as you want it to?

6. Add an *else* Statement to your Conditional.

In your last exercise you figured out how to change the background of your form if the same conditional was true. If your conditional was false, then your screen stayed white. Sometimes when you write a conditional you want to specify an alternate thing to happen if your T/F statement is not true. Using our earlier example of buying cake, take a look at the following conditional:

> If the cost of a piece of cake is less then $3.00, buy a piece of cake. If it isn’t, then buy a brownie.

The syntax to achieve this result involves using an else statement. The syntax is as follows:

```
if (T/F expression)
        {
         code statements;
        }
else
        {
         code statements;
        }
```

d. Add an else statement to your conditional that will change the background color to something else. It should look something like this:

Remember that your T/F expression is going to be different then the one listed below!

```
if (dateobjectvariable.getSeconds() = = 12)
        {
         document.bgColor=“blue”;
        }
else
        {
         document.bgColor=“red”;
        }
```

e. Save your work and reload your document. Did it change color? If you continue to reload your document does it eventually change color back and forth?

f. Add a statement that will print the current second to your webpage. This will help you test your code. You may add this statement after the end of your if/then/else statement:

document.write(dateobjectvariable.getSeconds())

Note: **You may want to add additional formatting to put this statement on a new line. Add the <p> tag by using the following: document.write(“<p>”) before the one writing the current second.**

7. Replace dateobjectvariable.getSeconds() with a variable.

In Exercise 2 we created a variable that held the value of the current day. We then used that variable to write the current day to our webpage. In Exercises 5 and 6 we referenced the current second using the getSeconds() method. Now we are going to replace this reference with a variable.

g. Create a variable that will hold the current second.
    i. var YourVariableName

h. Assign the current second to the variable.

i. Looking at your conditional, replace the first instance of dateobjectvariable.getSeconds() with your new variable name.

j. Save you work and refresh your webpage. Did it work the same?

k. Replace the other instance of dateobjectvariable.getSeconds() with your variable. Again, your program should work just the same.

8. Use an Expression to add a label before your current second.

Expressions:

"An *expression* is any valid set of literals, variables, operators, and expressions that evaluates to a single value; the value can be a number, a string, or a logical value."
– Netscape Core JavaScript Guide

What does that mean exactly? Here are some examples to help you out:

1 + 1 Expression with two literals and an operator (+). Evaluated it equals 2.

(for the next examples, y2k holds the value 2000.)

y2k + 1 Expression with one variable (y2k) and a literal (1). Evaluated it equals 2001

An example of an expression that results in a string might be:

"The year was " + y2k Evaluated it equals "The year was 2000"

Note: Adding a number to a string always results in a string.

I. Modify your document.write() method that prints the current second with an expression to add "The current second is " before you print the current second.

   Document.write("The current second is " + YourVariableNameForTheSecond)

Remember that the addition operator adds the variable directly after the last character of the string. It is often useful to put a space as the last character in a string.

9. Use an expression to print out the complete date on your webpage.

This is a Challenge! You are not required to do this exercise!

Print the current date on your webpage in the following format:

Month-day-Year
Example: 3-15-2004

Important:
One thing you need to know is that the method getMonth() counts from 0, not 1. Therefore, January is 0, February is 1, March is 2, etc. They way to correct for this is to use an expression to add 1 to the month.