CSCI 341 : HANDS ON

UNIT: Strings

Activities

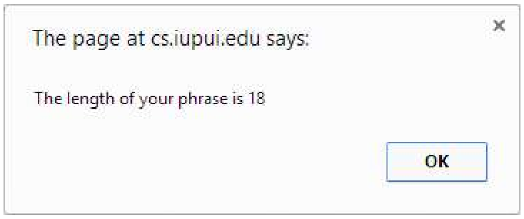
- A string is a type of object
- The string property
- Playing around with string methods
- Read and complete the assignment
- Check your work!

Activity One

Quite a few objects are built into the JavaScript language. As you can probably tell from the title, this hands on focuses on the String object. When you create a string variable, the JavaScript Magic Fairies actually convert a simple "var" to an object based version (complete with the keyword "new", like our Lab Expectations page has required of you). This is all pretty handy, for at least two reasons: 1) you don't actually have to use the keyword new, which is a constructor call. The Magic Fairies would be perfectly willing to handle this for you and 2) Once strings are turned into objects, you get any built in attributes and methods for free! (in other words, you don't have to write them yourself)

Perhaps the most frequently used String attribute is the Length property. As with all object properties, you can access the length property through dot name nomenclature; the object name followed by a dot followed by the name of the property. (strName.length) This all works because JavaScript string objects are shipped from the factory knowing how to report their length. If you need to know how long a string is, you can find out with the following code:

```
var strPhrase = "You da cannonball!";
var intPhraseLength = 0;
intPhraseLength = strPhrase.length;
alert("The length of your phrase is " + intPhraseLength);
```



The page at cs.iupui.edu says:

The length of your phrase is 18

Activity Two

There's even more fun to be had with String methods. While not providing as rich a string method tool kit as some object oriented language, JavaScript still presents the programmer with a useful set of built in utilities. As we look at some of these utilities, keep in mind that in JavaScript, ALL method calls on strings return a NEW string, leaving the original string untouched. Finally, let's remember from our objects lesson that all methods are preceded by a dot and followed by the symbols () as such: strName.method();

toUpperCase, toLowerCase -- converts to either upper or lower case letters

```
var text1 = "Ahoy World!";
var text2 = text1.toUpperCase(); //text2 is text1 converted to upper
var text2 = text1.toLowerCase(); //text2 is now text1 converted to lower
```

charAt(4) - would return the fifth character in a string (numbering begins at 0)

```
var str = "Ahoy world!";
var charResult = str.charAt(0);
alert(charResult);
// The result will be: "A"
```

indexOf("a") - would return the POSITION in the string of the first occurrence of lowercase a, numbering begins at 0. Returns a -1 if no instances are found

```
var str = "Whar be the treasure?";
var pos = str.indexOf("treasure");
alert(pos);
// The result will be: 12
```

substring(3) - returns remainder of string beginning from position 3

```
var str = "No me gusta pirates.";
var res = str.substring(3);
alert(res);
// The result will be: "me gusta pirates."
```

substring(3,19) - returns string subset from position 3 up to, but not including, postion 19

```
var str = "No me gusta pirates.";
var res = str.substring(3,19);
alert(res);
// The result will be: "me gusta pirates."
```

substr(0, 29) - returns a substring of length 29 characters starting at 0

```
var str = "A cannonball is headed for my leg.";
var res = str.substr(0,29);
alert(res);
// The result will be: "A cannonball is headed for my "
```

The trim method removes white space from both ends of a string

```
str = "Space ahoy!     ";
var n = str.trim();
// The result will be: "Space ahoy!"
```

The replace method replaces a specified value with another value in a string

```
str = "I have a hand!";
var n = str.replace("hand","hook");
// The result will be: "I have a hook!"
```

For more string examples, see pages 128 – 130 and page 65 of the textbook.

Assignment

Assignment (H.M.S. Stringsarefore): Admiral Captain Corcoran has been given a fleet of a dozen new pirate chasing ships which the queen her majesty wishes to name after her significant other (spouse / pet / other), but she cannot give every ship the same name. Create a program which accepts the name of her significant other and uses that name to derive different 12 ship names through JavaScript string operations. (use charAt, indexOf, substring, trim, replace, toUpperCase, toLowerCase) Store the names into an array and write them to the page as a list. Make sure the names put fear into the heart of every pirate!

Important Procedures for All Labs

Here are some general notes for perfection that you should follow for every assignment:

1. Please produce all web content to HTML5 standards.
2. Please **validate** all your files.
3. Be sure to update the header block comments for each file.
4. Be sure to check your browser's console / developer tools for error free code.
5. Test your code in Chrome and Edge at a minimum.
6. Use only your own original code for all labs.
7. Be sure to put your CSS and JavaScript in a separate files from your html.
8. Be sure to read through the lab rubric in Canvas.
9. Submit your lab in Canvas for grading.

Holler if you have any questions!

Mission Accomplished!

Fantastic work! You are now the very model of a modern Major-General. You're information variable, object oriented, and logical!