JavaScript String Methods … by Fred Brack

```
// Cut-and-Paste all the following if you wish and experiment.
// Remember, string positions start at 0, not 1!

// Use ONE of the following function definitions for SAY (for say(whatever))

// Use this one anywhere
function say(msg) { alert(msg) }

// Use this one anywhere you bring up the debugging console (Ctrl-Shift-I or -K)
function say(msg) { console.log(msg) }

// Use this one with https://www.webtoolkitonline.com/javascript-tester.html
function say(msg) {
  var screen = document.getElementById( "screen" );
  var div = document.createElement( "div");
  div.innerHTML = msg;
  screen.appendChild( div );
}

// These are “primitive values” not objects, but methods work on them!
firstName = "Fred";
lastName  = "Brack";
fullName  = firstName + " " + lastName;
pattern  = /r/;  // a “regular expression” denoted by the slashes
patterng = /r/g; // add the ‘global’ modifier
pattern2 = "r";  // a simple string

result = fullName.includes(pattern2);
say("INCLUDES: is string "+pattern2+" in "+fullName+"? ==> "+result);
say("... Only works with plain strings")
//INCLUDES: is string r in Fred Brack? ==> true
//... Only works with plain strings

result = pattern.test(fullName);
say("TEST: is pattern "+pattern+" in "+fullName+"? ==> "+result);
say("... Can take regular expressions as argument, making it more powerful")
//TEST: is pattern /r/ in Fred Brack? ==> true
//... Can take regular expressions as argument ==> more powerful

result = fullName.indexOf(pattern2);
say("INDEXOF: find pos of string "+pattern2+" in "+fullName+" ==> "+result);
say("... Only works with plain strings")
//INDEXOF: find pos of string r in Fred Brack ==> 1
//... Only works with plain strings

result = fullName.lastIndexOf(pattern2);
say("LASTINDEXOF: find pos of string "+pattern2+" in "+fullName+" ==> "+result);
//LASTINDEXOF: find last pos of string r in Fred Brack ==> 6

result = fullName.search(pattern);
say("SEARCH: find pattern "+pattern+" in "+fullName+" ==> "+result);
say("... Can take regular expressions as argument, making it more powerful")
//SEARCH: find pattern /r/ in Fred Brack ==> 1
//... Can take regular expressions as argument ==> more powerful

result = fullName.substr(0,4);
say("SUBSTR: extract the first 4 characters of "+fullName+" ==> "+result);
//SUBSTR: extract the first 4 characters of Fred Brack ==> Fred
```

```
result = fullName.substring(1,4); // up to but not including 4
say("SUBSTRING: extract characters 1 through 3 from "+fullName+" ==> "+result);
//SUBSTRING: extract characters 1 through 3 from Fred Brack ==> red

result = fullName.toUpperCase();
say("TOUPPERASE: convert "+fullName+" to uppercase ==> "+result);
//TOUPPERASE: convert Fred Brack to uppercase ==> FRED BRACK

result = fullName.toLowerCase();
say("TOLOWERCASE: convert "+fullName+" to lowercase ==> "+result);
//TOLOWERCASE: convert Fred Brack to lowercase ==> fred brack

result = fullName.startsWith("F");
say("STARTSWITH: does "+fullName+" start with string 'F'? ==> "+result);
//STARTSWITH: does Fred Brack start with string 'F'? ==> true

result = fullName.endsWith("ck");
say("ENDSWITH: does "+fullName+" end with string 'ck'? ==> "+result);
//ENDSWITH: does Fred Brack end with string 'ck'? ==> true

result = fullName.charAt(1);
say("CHARAT: what character is at pos 1 in "+fullName+" ==> "+result);
//CHARAT: what character is at pos 1 in Fred Brack ==> r

result = fullName.charCodeAt(1);
say("CHARCODEAT: what is the decimal character code at position 1 in "+fullName+"
==> "+result);
//CHARCODEAT: what is the decimal character code at position 1 in Fred Brack
==> 114

result = fullName.match(patterng);
say("MATCH: find pattern "+patterng+" in "+fullName+" ==> "+result);
//MATCH: find pattern /r/g in Fred Brack ==> r,r

result = fullName.replace(patterng,”x”);
say("REPLACE: replace pattern "+patterng+" in "+fullName+" with "+replacement+" ==>
"+result);
//REPLACE: replace pattern match /r/g in Fred Brack with x ==> Fxed Bxack

result = fullName.slice(1,4); // from position 1 to 3 (not including position 4)
say("SLICE: extract (slice) positions 1 through 3 from "+fullName+" ==> "+result);
//SLICE: extract (slice) positions 1 through 3 from Fred Brack ==> red

result = fullName.split(" ");
say("SPLIT: split "+fullName+" into parts separated by blanks, returning array ==>
"+result);
//SPLIT: split Fred Brack into parts separated by blanks, returning array
==> Fred,Brack

x = "   "+fullName+"      ";
result = x.trim();
say("TRIM: trim whitespace from beginning and end of '"+x+"' ==> "+result);
//TRIM: trim whitespace from beginning and end of ' Fred Brack ' ==> Fred
Brack

result = "*".repeat(10);
say("REPEAT: Here are a bunch of asterisks ==> "+result);
//REPEAT: Here are a bunch of asterisks **********

//END - Fred Brack, fbrack@bracksco.com, last updated Sep 13, 2020
```