

$$\begin{array}{r} 1 \\ 0 \\ 5 \\ \hline 1 \\ 7 \end{array}$$

- ☐ Build a click event demo.
- ☐ Build a stack trace demo.
- ☐ Update recording links.
- ☐ How big a JS program can be run in a browser?
- ☐ Number Methods:
  - Number.toString()
  - Number.toExponential()
  - Number.toFixed()
  - Number.toPrecision()
  - Number.valueOf()
  - parseInt(), parseFloat()
- ☐ String Interpolation:
  - Uses backquotes, curly braces and dollar sign:
  - let number = 5; // some value called number that is in-scope
  - let s = `\${number} of times`;
- ☐ Backslash as escape character: '\n'
  - Backslash as backslash: '\\'
- ☐ indexOf(), lastIndexOf()
  - indexOf() can take two arguments for peeling off multiple instances of matched strings.
  - let s = 'hello world';
  - let n = s.indexOf("world");
    - returns the position of the first occurrence of a specified value in a string.
    - ==Searches forward==
  - let n = s.lastIndexOf("world");
    - returns the position of the last occurrence of a specified value in a string.
    - ==Searches backward==

A good HW question:  
Implement lastIndexOf() using indexOf()  
and compare its performance with stock function.

■ `string.search()` returns position of match

```
let s = 'This is the day';
```

```
let n = s.search('th'); // n is 8
```

also takes /directives/regular expressions:

```
let s = 'This is the day';
```

```
let n = s.search(/Th/i); // n is 0 with case insensitivity 'i'
```

■ `string.slice(start, end)` returns substring as NEW string

Use negative number to select from the end of the string.

Flip two negative subscripts and add one to use positional subscripts.

■ `string.substring(start, length)` returns substring of given length

■ See <https://stackoverflow.com/questions/2243824/what-is-the-difference-between-string-slice-and-string-substring>

■ `string.replace(oldFrag, newFrag)` returns string with replacement

■ `/frag/` delineates regular expression, no quotes

■ `/frag/ig` or `/frag/gi` means 'global, case-insensitive', commutes

■ `string.replace(oldFrag, newFrag)` returns string with replacement

■ Utility string methods

```
string.toUpperCase()
```

```
string.toLowerCase()
```

```
string.concat()
```

```
string.trim()
```

```
string.charAt()
```

```
string.charCodeAt()
```

```
string.split()
```

■ Unicode and ASCII are identical for the first 128 values

■ References:

JavaScript Regular Expression

[https://www.w3schools.com/jsref/jsref\\_obj\\_regexp.asp](https://www.w3schools.com/jsref/jsref_obj_regexp.asp)

JavaScript String Methods

[https://www.w3schools.com/jsref/jsref\\_obj\\_string.asp](https://www.w3schools.com/jsref/jsref_obj_string.asp)

