Sunday, June 6, 2021

JavaScript & jQuery

Terminology

- **Primitive**: A simple element that can’t change.
  - JavaScript has 6 types of primitives.
- **Immutable**: Something that can’t change. All primitives are immutable.
  - **Types of Primitive**
  - **NUMBER-** Numerical value
  - **STRING-** A set of characters enclosed in quotations. “Hello”
  - **BOOLEAN-** True or False
  - **NULL-** Nothing
  - **UNDEFINED-** Hasn’t been defined in our code yet.

OPERATORS - Special character that indicates an action to be performed.

- =, +, *, %, /
- **Modulo** - Returns the remainder
- **NaN** - Not a number
- **Infinity** is also a number
- **Semicolons ;**
- **Keyboard shortcuts - MAC: cmd + option+ J**

JavaScript Variables

- **Variables:** Thought of as containers used to store information . Allow for a way to label data with a descriptive name.
  - Variables should begin with a letter

- **Lower Camel Case**: Use *lowerCamelCase when naming JavaScript variables. Start with lowercase letter, and if the variable name is more than one word, remove all spaces and capitalize the first letter of each subsequent word .*

- ***let**: Modern way of declaring a variable that changes in JavaScript*
- ***const:*** The modern way of declaring a variable that should never change in JavaScript. This is short for constant.

Strings

- A JavaScript data type that represents the exact text of whatever is enclosed in the quotes. “Hello”
  - Strings can include letters, punctuation, and numbers
  - Strings are surrounded with quotation marks
  - Variables can be set equal to strings
  - To escape characters use the \ characters
- **Method**: A method is an action run on a piece of data; you can think of it as a message you send to a piece of data, and the result is the response.
- **Return value**: The return value is the method’s response.
- **Argument**: Some methods take one or more arguments that provide the method with additional information to help it perform it’s action.

- String Methods

- **Chaining Methods:** Calling a method directly on the return value of another method.
- **Concatenation:** Combining two Strings together into one String.

Methods

A few useful string methods:

- *charAt ()*; -Returns the character at a particular location in a String.
- *toUpperCase ()*; - Converts a String to uppercase.
- *toLowerCase ()*; - Converts a String to lowercase.
- *concat ()*; - Combines two strings.

```
"supercalifragilisticexpialidocious".toUpperCase();
```

```
const word = "foo";

word.concat(“bar");
```

Assignment and Comparison Operators

- **Boolean**: *true* and *false*
- **Assignment operator:** Changes the value of the variable on the left of the operator.
- **Comparison operator:** Does not change any values, but returns a boolean (*true* or *false)* depending on whether the statement evaluates as true or false.

Assignment Operators

- = assign variable on left of operator value on right of operator
- += increase value of variable on left of operator by value on right of operator
- −= decrease value of variable on left of operator by value on right of operator
- ∗= multiply value of variable on left of operator by value on right of operator
- /= divide value of variable on left of operator by value on right of operator

Comparison Operators

- === is equal to
- !== is *not* equal to
- > greater than
- >= greater than or equal to
- < less than
- <= less than or equal to

Data Types (Primitives)

Primitives (see above)

Functions

- *parseInt* ( ) :Converts a string into a number
- *toString* ( ) : Converts a number into a string