

Intro to JavaScript

- Pre-Class Survey – Problem Solving Review:
 - When solving a problem, it's best to
 - just start coding, the solution will unfold
 - put it off and let someone else figure it out
 - question endlessly the reason it's a problem
 - take the extra time to really understand the problem
 - After understanding the problem, the next step is to
 - extract the relevant information
 - identify what the user is expecting to see
 - figure out how to produce the expected output
 - all of the above
 - Solving problems requires
 - critical thinking
 - implementing memorized solutions
 - applying predefined solutions
 - reinventing the wheel
 - JavaScript is the only programming language that runs in the browser?
 - True
 - False
 - JavaScript was written in 10 days in May of 1995?
 - True
 - False
- Introduction to Variables, Types and Constants.
 - JavaScript is a loosely or dynamically typed programming language. Variables in JavaScript are not directly associated with any particular value type, and any variable can be assigned (and re-assigned) values of all types
 - Variables:
 - Variable have names so that we can uniquely retrieve their contents:
 - Variables must have unique names
 - Variable Names cannot be reserved words within JavaScript
 - Cannot start with a number
 - Must start with a letter, the \$ sign, or the _ underscore
 - Can contain both upper and lowercase letters
 - Names are case sensitive – so Name is different than name
 - Variables are containers for storing data values (numbers, strings, Booleans, etc.)
 - In programming, just like in algebra, we use variable to hold values or unknowns
 - ie: let x represent the number of apples; let numPeople equal 7, etc.
 - Also just like in algebra, we use variables in equations...ie: $x = \text{numApples} / \text{numPeople}$
 - As mentioned, in JavaScript, variables are loosely typed, meaning we don't have to declare what type of information is stored within it – this is quite different than most languages, but provides the flexibility needed for a web language
 - ie: let answer equal anything
 - `answer = 47; console.log(answer);`
 - `answer = " is the meaning of life!"; console.log(answer);` this is valid code

- Variable Types:
 - JavaScript still has Variable Types
 - A Variable's Type is determined by its content
 - Primitives (report as their type):
 - Number (64 bit floating-point)
 - String
 - Boolean (True or False)
 - Null
 - Undefined
 - BigInt
 - Symbol
 - Objects (report as objects, except for functions)
 - Examples of Objects are:
 - Data Structures
 - Functions (which report as Functions)
 - Arrays
- Constants:
 - Constants are Read-Only
 - Constants are variables that cannot be changed
 - ie: `const PI = 3.14159265359;`
 - By Industry Standards, Constants use all-caps (all uppercase)
- Declaring Variables:
 - "var" – function scoped, uses hoisting, problematic due to duplicate and scope
 - "let" – block scoped, similar to most other languages
 - "const" – block scoped and read-only, memory efficient
- Understanding Basic Operators.

Precedence	Operator type	Associativity	Individual operators
20	Grouping	n/a	(...)
18	new (without argument list)	right-to-left	new ...
17	Postfix Increment	n/a	... ++
	Postfix Decrement		... --
16	Prefix Increment	right-to-left	++ ...
	Prefix Decrement		-- ...
15	Exponentiation	right-to-left	... ** ...
14	Multiplication	left-to-right	... * ...
	Division		... / ...
	Remainder		... % ...
13	Addition	left-to-right	... + ...
	Subtraction		... - ...
11	Less Than	left-to-right	... < ...
	Less Than Or Equal		... <= ...

	Greater Than		... > ...
	Greater Than Or Equal		... >= ...
	in		... in ...
10	Equality	left-to-right	... == ...
	Inequality		... != ...
	Strict Equality		... === ...
	Strict Inequality		... !== ...
6	Logical AND	left-to-right	... && ...
5	Logical OR	left-to-right
4	Conditional (ternary)	right-to-left	... ? ... : ...
3	Assignment	right-to-left	... = ...
			... += ...
			... -= ...
			... **= ...
			... *= ...
			... /= ...
			... %= ...
			... <<= ...
			... >>= ...
			... >>>= ...
			... &= ...
			... ^= ...
			... = ...
1	Comma / Sequence	left-to-right	... , ...