

CS 22A

JavaScript for Programmers

© 2021, Dr. Baba Kofi Weusijana
(Bah-bah Co-fee Way-ou-see-jah-nah)

All Rights Reserved

Some slides were originally created by Craig Duckett

Today (Day03)

- Announcements
- Variables
- Guidelines
- Debugging
- JavaScript reference/guide

Announcements

- **Reminder:** Late work is not acceptable after this 2nd week of the quarter. If you are behind, catch-up by Monday next week
 - Don't skip anything. Do your work in the order set in the modules on the Home page
- If you are having any trouble getting one of the textbooks or getting something installed, send me a direct message via Pronto

Computer Science Club

- Meets Tuesdays, 3:30PM to 5:30PM, in room 5607
- Join at:
<https://cs.foothillstemclubs.org/>

Online Tutoring

- From the STEM Success Center for CS Students:
<https://foothill.edu/stemcenter/tutoring-schedules/schedule-cs-online.html>
- Can also use the NetTutor link in our Canvas coursesite's navigation strip

Office Hours

- Dr. Weusijana's **ONLINE** Office Hours are listed on the Syllabus
- If the pace seems too fast please contact me
 - Via Canvas assignment comment 1st, then by Canvas Inbox
 - Anonymous contact form:
<https://goo.gl/forms/qFXKiTQk6RD71Hm42>

Naming Variables

- Using Allowed Characters
 - The following variable names are valid:

numberOfOranges

\$numberOfOranges

_numberOfOranges

number_of_oranges

numberOfOranges2

- The following variable names are **not** valid:

#numberOfOranges

1numberOfOranges

number of Oranges

numberOfOranges 2

Avoid Using JavaScript Keywords when Naming Variables

abstract	delete	goto	null	throws
as	do	if	package	transient
boolean	double	implements	private	true
break	else	import	protected	try
byte	enum	in	public	typeof
case	export	instanceof	return	use
catch	extends	int	short	var
char	false	interface	static	void
class	final	is	super	volatile
const	finally	long	switch	while
continue	float	namespace	synchronized	with
debugger	for	native	this	
default	function	new	throw	

Variable Data Types

- Types: number, string, boolean, and null.
 - Objects/Arrays too, but more on that later
- JavaScript does not force you to declare the type of variable when you define it
- JavaScript allows virtually **any** value to be assigned to **any** variable
- YOU need to be **very** careful:
YOU need to **TEST** your code

Primitives vs Objects

- Primitive values are not objects, have no methods, and can't be changed (immutable)
 - string, number, boolean, null, undefined, symbol
- Primitive wrapper objects in JavaScript
 - Except for null and undefined, all primitive values have object equivalents that wrap around the primitive values:
 - [String](#) for the string primitive.
 - [Number](#) for the number primitive.
 - [Boolean](#) for the Boolean primitive.
 - [Symbol](#) for the Symbol primitive.
- The wrapper's [valueOf\(\)](#) method returns the primitive value.

Variable Data Types: Numbers

var variableName = number;

- Examples:

var payCheck = 1800;

var phoneBill = 35.50;

var savings = 1.50;

var spareTime = -24.5;

Variable Data Types: Numbers

```
var bigNumber = 4.52e5; // this means move  
the decimal point 5 places to the right, so  
this becomes 452,000
```

- **NOTE:** JavaScript may return an answer to a calculation using **exponential notation** (like many calculators).

Guidelines

- Use **camelCase** for variables & functions/methods
- Open **curly braces** on the same line
- **Always** use blocks: even if it contains only 1 line
- Define your functions **before** you call them
- **Always** use semicolons to end a statement
- **Always** use **var** when declaring a variable

Debugging

- Make sure you are **always** testing in **FirefoxDeveloperEdition** while so you can see error messages on the **console**
 - Don't use other browsers unless you are directed to do so because your work will be graded using Firefox**DeveloperEdition**
 - Can't figure out what's going on with your code? Use `console.log()`.
 - Also the console is a great place to check on what objects exist and what functions work!

Debugging

- Still can't figure out what's going on? Use the FirefoxDeveloperEdition **Debugger** tab.
- You can set breakpoints and **watch the code run**
 - Watch variables/objects change
- You can run JavaScript live from the Console
- I will demo debugging later in the quarter once we start coding as part of in-class-exercises

JavaScript Guide & Reference

- The official Mozilla JavaScript Guide and Reference:
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>
 - Made by the people who made JS!
 - Up-to-date
 - Even mentions how other Web browsers interpret JS
- **AVOID W3CSchools for JavaScript!**
 - But great for HTML5 & CSS3

To Do

- Catch-up & then do the next assignments for the Week 2 Module
 - We need to stay on schedule so we can do the team projects and it delays grading