

# **COMPUTATIONAL THINKING (VARIABLES & CONDITIONALS)**

Felix Cohen

## QUIZ RECAP

- What tag is used to include a .js file?
- What is jQuery?
- How do you add jQuery to a webpage?
- How do you add comments to JavaScript code?
- What is the `$()` function?

## AGENDA

- Intro to Computational Thinking
- Introducing the Console
- Variables and Data Types
- Conditional Statements

# **COMPUTATIONAL THINKING**

# OBJECT ORIENTATION

Think about the car again.

## PROBLEM SOLVING

JavaScript (really all programming languages) programs executes:

- linearly
- decisions
- storage
- Loops
  - For loops
  - While loops (and do...while loops)

# **THE JAVASCRIPT CONSOLE**

# HOW WE WILL WORK

Using the console to learn the basics

Cookie monster to show snippets of code in the bigger picture.

The magic of jQuery brings us back to the browser. It can modify HTML and CSS.

More on jQuery later.



# WHERE IS THE JAVASCRIPT CONSOLE?

Why would you use this?

# OUTPUT TO THE CONSOLE

```
console.log("Message to log");
```

# INPUT FROM USERS

```
alert('Hello world');  
prompt('What is your first name?');
```

## TRY IT!

In Developer Tools console type:

- `console.log("Hello world");`
- your output goes between the parentheses

ARITHMETIC IN JAVASCRIPT

Operator	Description	Example
+	Addition	1+1
-	Subtraction	3-2
*	Multiplication	5 * 3
/	Division	10 /2
++	Increment	5 ++
--	Decrement	5 --
%	Modulus	1%2

# **DATA TYPES & VARIABLES**

# WHAT IS A VARIABLE?

A 'bucket' for data

Can hold booleans, numbers or strings of text

Programming language needs to be told a word is a variable.

## VARIABLES

Declaration:

- ▶ `var age;`

Assignment:

- ▶ `age = 21;`

Declaration and initialization

- ▶ `var age = 21;`



## RE-ASSIGNMENT

```
var name = "Jo";
```

```
name = "Mich";
```

## BOOLEAN

Binary, two possible values:

- ▶ true
- ▶ false

Has driver license:

- ▶ If driver has license: true
- ▶ If driver does not have license: false

## VARIABLE CONVENTIONS

Variables start with a lower case letter

If they contain multiple words, subsequent words start with an upper case letter

► e.g: `var numberOfStudents = 10;`

## OBJECTS

```
car={make:"Ford",model:"Fiesta",cc:  
100,colour:"blue"};
```

# DEBUG VARIABLES – UNDEFINED

```
var name = "Jo";
```

```
name.surname;
```

- ▶ surname property is not on name, therefore it's undefined

# DEBUG VARIABLES – NULL

- ▶ `var colour = null;`
- ▶ `var size; //This is null and undefined`

# DEBUG VARIABLES – NULL VS UNDEFINED

```
var amount;  
amount is null  
amount is also undefined
```

## STRING

Stores textual information

- ▶ Double quotes
  - ▶ "How is the weather today?"
- ▶ Single Quotes
  - ▶ 'Warm'



## QUOTES IN STRINGS

Double vs single quoted strings:

- ▶ 'They "purchased" it'
- ▶ "It's a beautiful day"

Escaping

"They \"purchased\" it"

'It\'s a beautiful day'

## NUMBERS

Represent numerical data

- ▶ int: 42
- ▶ float: 3.14159265

Signed

- ▶ int: +6
- ▶ float: -8.2

Can perform arithmetic on number data types

# A FEW BASIC OPERATIONS

Length of a string:

- ▶ `var name = "Jo";`

- ▶ `name.length`

  - ▶ `= > 2`

- ▶ Can be done directly on the string:

  - `"Jo".length`

  - ▶ `= > 2`

# LAB TIME

data\_types

# DATA TYPE CONVERSION

When/ why would you convert a data types?

## CONVERSION: STRING TO NUMBER

```
var intString = "4";
```

```
var intNumber = parseInt(intString);
```

```
var floatString = "3.14159";
```

```
var floatNumber = parseFloat(floatString);
```

- These work:  
    parseInt("4");  
    parseFloat("3.14159");  
    parseInt("3.5"); //gives 3

# CONVERT: NUMBER TO STRING

`var number = 4;`

▶ `number.toString(); => "4"`

OR

`number + ""; => "4"`

# **CONDITIONAL LOGIC**



# COMPARISONS

- Why would you need to compare.
- Passwords!

# COMPARISONS – EQUALITY

Are two things equal or identical?

`10 == 10`

`true`

`10 == 5`

`false`

`"hi" == "hi"`

`true`

**x = 3**

## Logical Operators

Operator	Description	Comparing	Returns
<b>==</b>	equal to	<b>x == 8</b>	<b>FALSE</b>
<b>===</b>	exactly equal to (value and type)	<b>x === "3"</b>	<b>FALSE</b>
		<b>x === 3</b>	<b>TRUE</b>
<b>!=</b>	is not equal	<b>x != 8</b>	<b>TRUE</b>
<b>!==</b>	is not equal (neither value nor type)	<b>x !== "3"</b>	<b>TRUE</b>
		<b>x !== 3</b>	<b>FALSE</b>
<b>&gt;</b>	greater than	<b>x &gt; 8</b>	<b>FALSE</b>
<b>&lt;</b>	less than	<b>x &lt; 8</b>	<b>TRUE</b>
<b>&gt;=</b>	greater than or equal to	<b>x &gt;= 8</b>	<b>FALSE</b>
<b>&lt;=</b>	less than or equal to	<b>x &lt;= 8</b>	<b>TRUE</b>

# CONDITIONALS

- What is a Conditional?
- Why would we use it? (Remember your car example)

## AN EXAMPLE

### Logic

You can come in if:

- you are a GA students

### JavaScript Syntax

```
var student = "GA"

if (student == "GA") {
  console.log("Come on in!");
}
```

# THE SYNTAX - IF

```
if(true) {  
  console.log("The condition is true");  
}
```

```
if(false) {  
  console.log("You won't see this");  
}
```

# CONDITIONALS – IF ELSE

```
if(condition is true) {
```

```
    console.log("The condition is true");
```

```
} else {
```

```
    console.log("The condition was false")
```

```
}
```

## IF/ELSE-IF/ELSE

```
var topic = "JS";
```

```
if (topic == "JS") {  
    console.log("You're learning JavaScript");  
} else if (topic == "JavaScript") {  
    console.log("You're still learning JavaScript");  
} else {  
    console.log("You're learning something else");  
}
```



## COMPARING MULTIPLE CONDITIONS

### THE TRUTH TABLE - &&

```
if (name == "GA" && password == "YellowPencil") {  
  console.log("can access the internet")  
}
```

AND - &&	TRUE	FALSE
TRUE	true	false
FALSE	false	false

## COMPARING MULTIPLE CONDITIONS THE TRUTH TABLE - ||

```
if (day == "Tuesday" || day == "Thursday") {  
  console.log ("You have class today")  
}
```

OR -	TRUE	FALSE
TRUE	true	true
FALSE	true	false

# LAB TIME

coa\_conditionals

ex\_ifElse

Conditional Cookie Monster