GENERAL ASSEMBLY

FRONT-END JAVASCRIPT COMPUTATIONAL THINKING (VARIABLES & CONDITIONALS)

Emerson Taymor
co-founder, philosophie

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

EXERCISE: THINK LIKE A COMPUTER

GA wants to give you a robot! But first, we need to know that you can think like a computer.

Write the steps you would provide a robot so that it can check your email inbox.

Go ahead and give your robot a name.

PROBLEM SOLVING

JavaScript (really all programming languages) programs executes:

- linearly
  - One thing happens after another
- decisions
  - IF email notification received THEN check email.
- storage
  - Archive the email my friend sent.
- iterations
  - For the next 10 unread messages, delete

THE JAVASCRIPT CONSOLE

HOW WE WILL WORK

Using the console to learn the basics

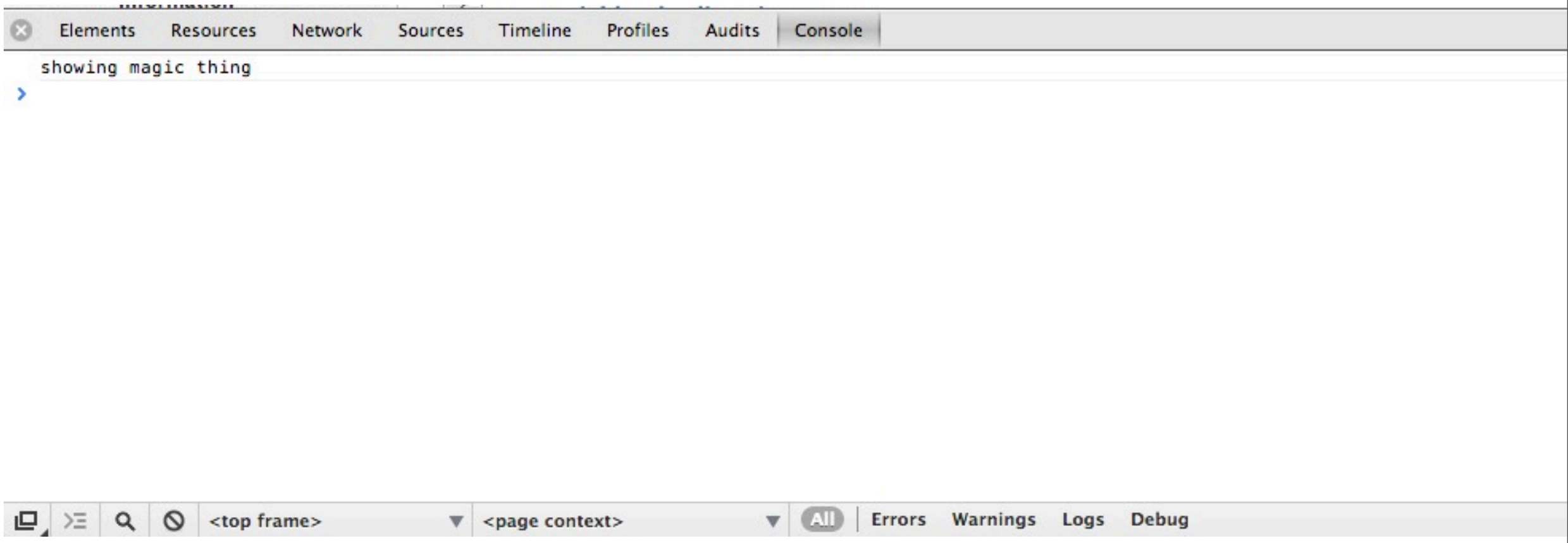
Cookie monster to show snippets of code in the bigger picture.

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

```
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

ARITHMETIC IN JAVASCRIPT

What is the order of precedence in operations?

- BOMDAS, BEMDAS - unless otherwise specified by parentheses, * / and % take precedence over + and -.

LAB TIME

line by line
variables by line

DATA TYPES & VARIABLES

WHAT IS A VARIABLE?

A variable is something that stores information that can be retrieved later.

Like Algebra

$x=5$

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;`

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

undefined means a variable has been declared, but not assigned a value

null is an assignment value. it represents no value

WHAT CAN VARIABLES STORE?

JavaScript variables are untyped

They can store data of any valid type

String

Number

Boolean

Array

Object

DATA TYPES

We will look at:

- string
- number
- boolean

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"
```

PROBLEM CONVERSION: NUMBER TO STRING

This results in an error:

- 4.toString();

This does not:

- (4).toString();

Why?

PROBLEM CONVERSION:
STRING TO NUMBER

```
var notANumber = parseInt("NAN");
var nullThing = null;
parseFloat(nullThing);
```

CONDITIONAL LOGIC

COMPARISONS

- Why would you need to compare.

COMPARISONS - EQUALITY

Are two things equal?

10 == 10

true

10 == 5
false

“hi” == “hi”

true

$x = 3$

Logical Operators			
Operator	Description	Comparing	Returns
==	equal to	x == 8	FALSE
===	exactly equal to(value and type)	x=== “3”	FALSE
		x===3	TRUE
!=	is not equal	x!=8	TRUE
!==	is not equal(neither value nor type)	x!== “3”	TRUE
		x!==3	FALSE
>	greater than	x>8	FALSE
<	less than	x<8	TRUE
>=	greater than or equal to	x>=8	FALSE
<=	less than or equal to	x<=8	TRUE

CONDITIONALS

- What is a Conditional?
- Why would we use it? (Remember your robot 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

GENERAL ASSEMBLY