

# Intro to Front End - Interactive Intro To Programming

---

*Set up the Live Stream*

---

## Front End Fundamentals

Resume's done! How do we like HTML and CSS?

---

## Intro to Javascript

Today's going to be a little different - I'm going to talk through and show a concept and then we're going to practice using Compass.

So what is Javascript?

It's a programming language that we are going to use to create Web Apps!

Javascript is for anything beyond the design and information - it's responsible for things like interactive maps, forms, displaying updates to news pages, and lots more

---

## Console Log

Console.log is a FUNCTION that is built into JavaScript and is a super useful tool for printing values to the screen and helping us fix bugs in code.

What is a bug?

- A bug is an error in our code that makes things not behave the way that we want them to.
- The term has been around since the 1870's, but made popular by Grace Hopper
- Grace Hopper was a computer scientist who worked on the Harvard Mark computer which was used in World War II and a lot of the work she did has informed programming languages that we use today.
- She recorded this moth which was found in the Harvard Mark II which was causing an

error and this is her entry.

---

## Console Log contd

We'll look at functions more later on but for now what we need to know is that this lets us print things to the console.

*Live Code* REPL IT

Repl.it is kinda like CodePen - it lets us practice code.

Print

```
console.log("Hello World")
```

---

## Comments

We've talked about comments a bit before - they are meant to be notes for YOU the developer.

They are like little sticky notes that the outside world can't see, like you're writing in invisible ink.

These are messages for future you, or for other people you are working with.

When a computer comes across a comment it skips over it.

---

## Comments contd

In Javascript our comments start a new line with two forward slashes

*Live Code*

```
// This is a comment. If I put a console.log("hello!") here, we will not see it in
```



*Get them to go to Compass and do the Intro to Programming*

---

## Data Types

All of the values that we save and use in JavaScript will be of a certain type.

When we write JavaScript it's important to remember what types we use so we can understand how our code will behave.

---

## Numbers

Numbers are the first data type we will talk about.

Numbers are just like numbers in real life. We can use JavaScript to quickly add, subtract, multiply and divide numbers.

---

## Numbers Contd

Numbers look just like normal numbers - nothing fancy!

*Live Code*

Demo:

- Addition
- Subtraction
- Multiplication
- Subtraction

*Get them to go to Compass and do the Numbers exercise*

---

## Strings

Strings are words!

Strings can be used to represent many different things in code.

For example, your name, and your email address are often represented by a string when you are logging into various websites.

---

## Strings contd

We use quotation marks to represent strings in JavaScript.

What data type is “123”?

“123” is NOT a number, it’s a string. Something to keep in mind in code is what type your data is.

*Live Code*

Demo:

```
console.log("Hello")
```

```
console.log("123")
```

---

## Strings contd

We can add strings together!

When adding strings in JavaScript, you need to add a space, otherwise the words will print stuck together.

Computers do not understand whitespace so spaces need to be added manually if you want your strings to look nice.

*Live Code*

What do you think the outcome of these will be?

```
console.log("Hello," + "there")
```

whitespace!

```
console.log("123" + "456")
```

*Get them to go to Compass and do the Strings exercise*

---

## Booleans

Booleans represent values that are either true or false.

---

## Booleans contd

Booleans are used to provide answers to questions like “Did the user type the correct username?”

Note that “true” is NOT a boolean, it’s a string.

*Live Code*

```
console.log(true);
```

```
console.log(false);
```

```
console.log("true");
```

---

## Comparison Operators

Comparison operators are used to compare two values.

---

## Comparison Operators contd

Triple equals sign checks to see if two values are the same value AND the same type.

### Live Code

```
console.log("Lighthouse Labs" === "Lighthouse Labs")
```

```
console.log("1" === 1)
```

This is false because they are not the same type.

---

## Greater Than

The greater than operator checks to see if the number on the left is greater than the number on the right.

(Remember how to tell the signs around)

### Live Code

```
console.log(1 > 10);
```

```
console.log(100 > 76)
```

---

## Less Than

Less than checks to see if a number on the left is less than the number on the right.

### Live Code

```
console.log(1 < 10)
```

```
console.log(100 < 76);
```

---

# Not Equal

Checks that two values are NOT equal.

! in programming is called a bang.

*Live Code*

```
console.log(1 !== 10);
```

```
console.log(100 !== 100)
```

*Get them to go to Compass and do the Boolean exercise*

---

# Variables

Variables are often used when you want to SAVE a value for later.

We save it for later so it can be reused.

---

# Variables contd

We declare variables by using the keyword “var”,  
Then giving the variable a name using camelCase,  
Then assign it a value

*Live Code*

```
var year = 2019;  
console.log(year);
```

```
var name = "Lighthouse Labs"  
console.log(name)
```

Get them to go to Compass and do the Variables exercise

---

## If Statements

If statements help us to control what our program will do in a specific circumstance or with different inputs.

---

## If Statements

To write an if statement:

We use the keyword “if” to indicate to JavaScript we are using an if statement.

We have a condition to evaluate

We have an action to execute if the condition is true

*Live Code*

```
var cats = 4

if (cats <= 4) {
  console.log("You might have too many cats")
}
```

---

## If Statements contd

This example will only print out if the variable raining has a value of true.

*Live Code*

// start with just if and the following examples will build on it with ‘else’ and ‘else if’

```
var temperature = 12;

if (temperature < 0) {
  console.log("Make sure you pick out a scarf!");
}

console.log("Now you're ready to go outside!");
```



---

## If Statements contd

Else statements allow us to have an alternative.

An if statement block can have only ONE else statement

*Live Code*

```
} else {  
  console.log("Short sleeves are fine.");  
}
```

---

## Else If Statements

If you have more than two conditions you can use else if - you can have multiple else if's, but only ever one 'else'

*Live Code*

```
var temperature = 12;  
  
if (temperature < 0) {  
  console.log("Make sure you pick out a scarf!");  
} else if (temperature < 15) {  
  console.log("Short sleeves won't cut it!");  
} else if (temperature > 25) {  
  console.log("Sleeves are too much!");  
} else {  
  console.log("Short sleeves are fine.");  
}
```

*Get them to go to Compass and do the If Statements exercise*

---

## Functions

A function is a block of code that executes in a certain order.

Kind of like baking a cake.

- Arguments (ingredients) are passed into a function (the baking process)
  - You can change the the ingredients each time you bake it but the way it's baked follows the same process. Ish.
  - The function returns a result (a baked cake).
- 

## Function Syntax

Function syntax:

Starts with the keyword “function” which tells JavaScript we are declaring a function.

Give the function a name in camel case (one word, alphanumeric chars only. Cannot start with a number)

Wrap the functions actions in curly brackets.

To use (call/execute) our function, we write the name, followed immediately by smoothies.

*Live Code*

```
function makeName() {  
  console.log("Lighthouse Labs");  
}  
  
makeName();
```

Here's some error messages!!

*Get them to go to Compass and do the Functions exercise*

---

## Adding Arguments

Parameters are OPTIONAL. They are a comma separated list of variables you use in your function's actions.

They are assigned specific values when the function is called or used.

## Live Code

```
function isEven(num) {  
  return num % 2 === 0;  
}  
  
console.log(isEven(10));  
console.log(isEven(11));
```

## Reminders

- Record [Attendance](#)
- Upload lecture
- Send out Notes
- Log hours on Clocktower