



# Full-Stack Web Developer Training Program

Front-end Cohort - Angular Development

code  LABS  
POWERED BY  codefi





# Class 02

## JavaScript Foundations

### Part 1

# Data Types

# Data Types

- Data types are a classification that specifies which type of value a variable has and what type of mathematical, relational, or logical operations can be applied to it (without causing an error).

```
// * string * \  
"Anything wrapped in quotes is a string";  
"They can be double quotes or single quotes";  
"2";
```

```
// * string * \  
"Anything wrapped in quotes is a string";  
"They can be double quotes or single quotes";  
"2";
```

```
// * number * \  
1; // 1.0  
10.0;  
13.9;
```

```
// * string * \  
"Anything wrapped in quotes is a string";  
"They can be double quotes or single quotes";  
"2";
```

```
// * number * \  
1; // 1.0  
10.0;  
13.9;
```

```
// * boolean * \  
true;  
false;
```

```
// * string * \\
"Anything wrapped in quotes is a string";
"They can be double quotes or single quotes";
"2";
```

```
// * number * \\
1; // 1.0
10.0;
13.9;
```

```
// * boolean * \\
true;
false;
```

```
// * special * \\
undefined;
null;
NaN;
```



```
user = {  
  name: "John",  
  password: 12345,  
  age: 35  
}
```

```
user = {  
    name: "John",  
    password: 12345,  
    age: 35  
}
```

```
user = ["Suzy", "Rachel", "Mark"]
```

# Data Types in JS

```
graph TD; A[Data Types in JS] --> B[Primitive]; A --> C[Non-Primitive]; B --> D[String]; B --> E[Number]; B --> F[Boolean]; B --> G[Null]; B --> H[Undefined]; C --> I[Object]; C --> J[Array]; C --> K[Function];
```

**Primitive**

**String**

**Number**

**Boolean**

**Null**

**Undefined**

**Non-Primitive**

**Object**

**Array**

**Function**

# Variables

# Variables

- A variable is a name of a storage location
- Types of variables
  - let
  - const
  - var

# Variables

```
let user = {  
  name: "John",  
  age: 67  
}
```

# Operators

# Operators

- Operators are capable of manipulating or evaluating an expression.



# Operators

- Operators are objects capable of manipulating or evaluating an expression.
- +
- -
- /
- \*
- >
- <
- ===
- !==

# Operators

```
// * Addition *  
console.groupCollapsed("Addition");  
console.log(2 + 2); // 4  
console.log("good" + " day"); // "good day"  
console.log(2 + "day"); // "2day"  
console.log(2 + "2"); // "22"  
console.groupEnd();
```

# Conditional Statements

- Conditional Statements are a programming expression or command that instructs a computer to run code if a specified statement is truthy or not.

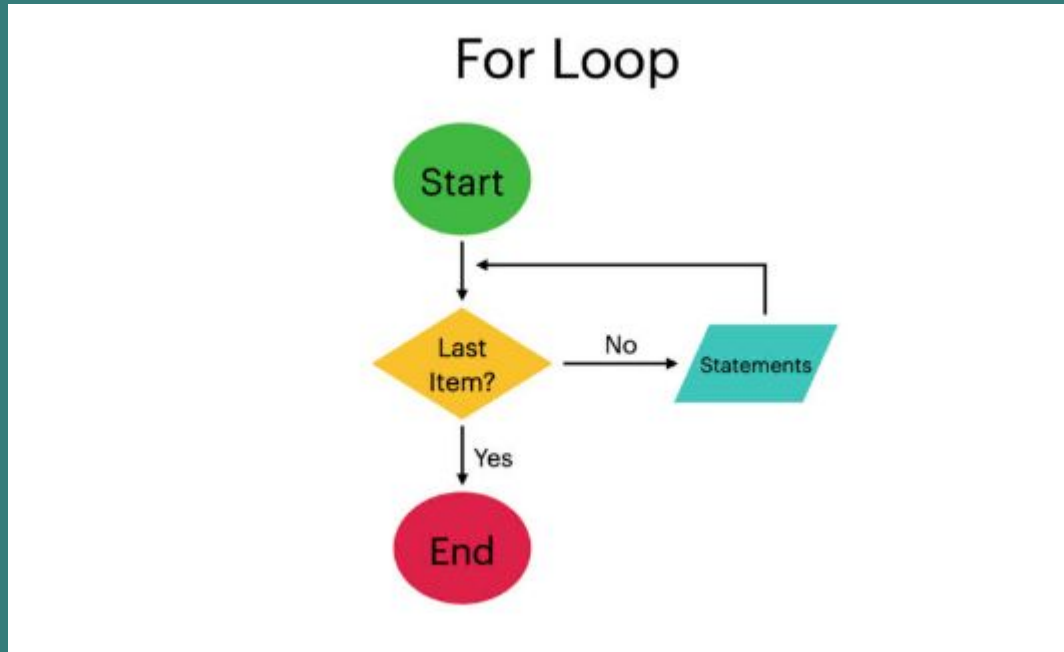
# Conditional Statements

- Conditional Statements are a programming expression or command that instructs a computer to run code if a specified statement is truthy or not.

```
let num = 5;  
if (num < 10){  
  console.log("num is less than 10")  
}
```

# Loops

- Loops are used in JavaScript to perform repeated tasks based on a condition.



# Overview

- Data Types - a way to express data in JavaScript
- Variable - a name of a storage place
- Operators - symbols that allow us to compute an expression (+, -, /, ect)
- Conditional Statements - logic execution blocks (if this, then do this)
- Loops - executing a block of code until a condition is met

**Demo**