

Constructors

Web Development Boot Camp
Lesson 10.1





What is programming?

Programming

The designing and building of an executable program that will accomplish a specific computing task. Essentially, programming is problem solving.

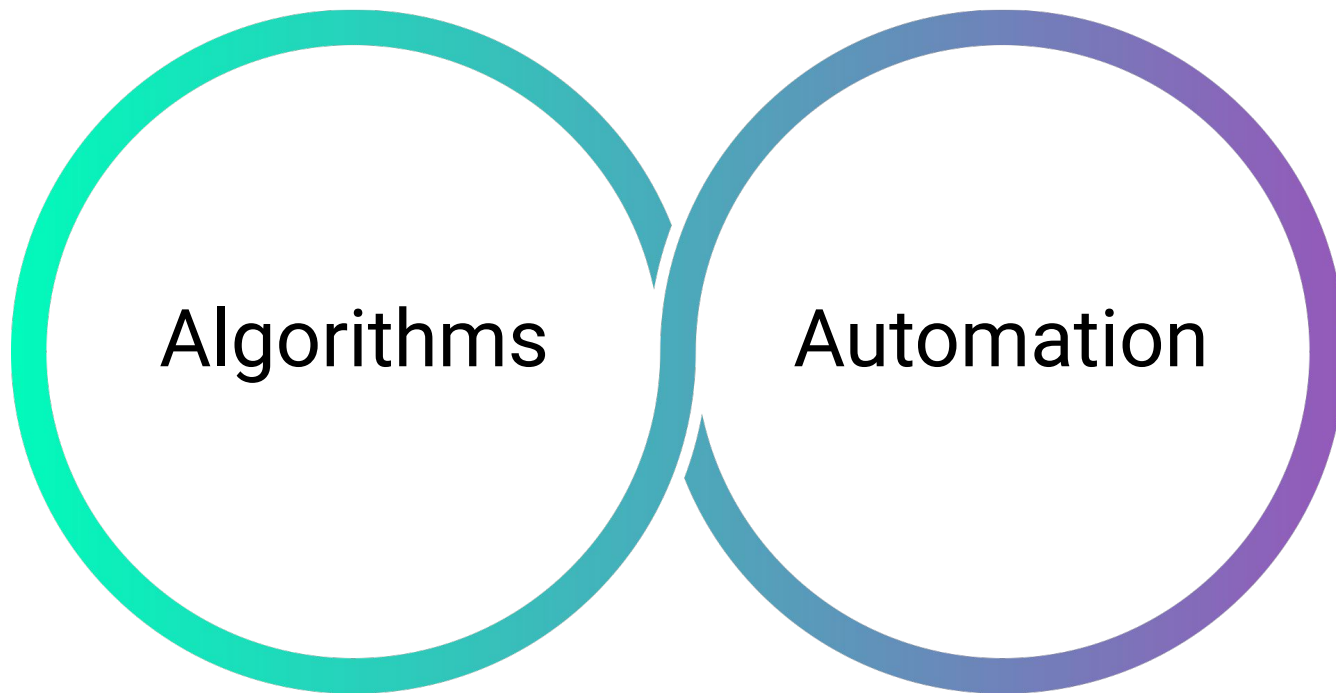




What problems do we solve?

Algorithms and Automation

Programming allows for us to solve almost any task or problem on a computer.
There are two primary categories:





What is DRY?

Don't Repeat Yourself!

Rewriting code wastes time, memory, and can confuse later readers and/or contributors to your code.

D on't
R epeat
Y ourself



What is an object?

Objects

Objects in JavaScript are unordered collections of related data built on a `key:value` structure where values can be any `data-type`, including functions.

```
const person = {  
  name: ['Bob', 'Smith'],  
  age: 32,  
  gender: 'male',  
  interests: ['music', 'skiing'],  
  bio: function() {  
    alert(this.name[0] + ' ' + this.name[1] + ' is ' + this.age + ' years old. He  
likes ' + this.interests[0] + ' and ' + this.interests[1] + '.');  
  },  
  greeting: function() {  
    alert('Hi! I\'m ' + this.name[0] + '.');  
  }  
};
```



**Why/what are objects important
in JavaScript?**

Everything is an Object!

Well, almost everything

data-types objects:

- Arrays
- Date
- Math
- ...and more!
- Even Functions are objects!

Primitive types are **not** objects:

- Boolean
- Null
- Undefined
- Number
- String
- Symbol



What is object-oriented programming?

Object-Oriented Programming (OOP)

OOP is a programming paradigm or pattern of programming centered around objects. Problems are thought of in a way in which a collection of objects work together to solve a problem. Objects can speak to one another, and this ability makes them suitable for managing and solving large and complex problems.

Encapsulation

Object data (and often, functions too) can be stored neatly (the official word is **encapsulated**)

Abstraction

creating a simple model of a more complex thing

Inheritance

we can create new classes based on other classes ie. Person is parent to student and teacher

OOP

```
graph TD; OOP((OOP)) --> Encapsulation[Encapsulation]; OOP --> Abstraction[Abstraction]; OOP --> Inheritance[Inheritance]; OOP --> Polymorphism[Polymorphism];
```

Polymorphism

The fancy word for the ability of multiple object types to implement the same functionality is **polymorphism**.

<Time to Code>

