



**Coding**  
Bootcamps



# **Intro to Node.js, Express.js and MongoDB**

By Kaustubh Ghadge from [Coding Bootcamps](#)

# Session Outline

1. Welcome
2. About this Course
3. Quick Programming Refresher
4. Setup and Installation

**1.**

**Welcome**

# Hello!

**I am Kaustubh Ghadge  
("KG")**

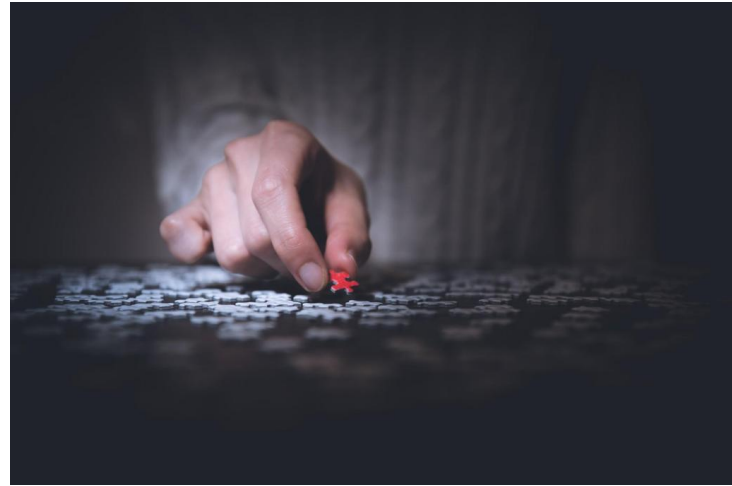
A senior instructor at  
[coding-bootcamps.com](https://coding-bootcamps.com)



**2.**

**About this Course**

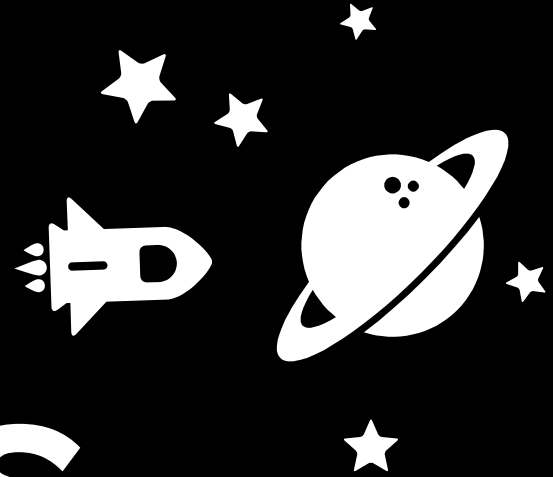
# Course Structure



# **How to follow this course?**



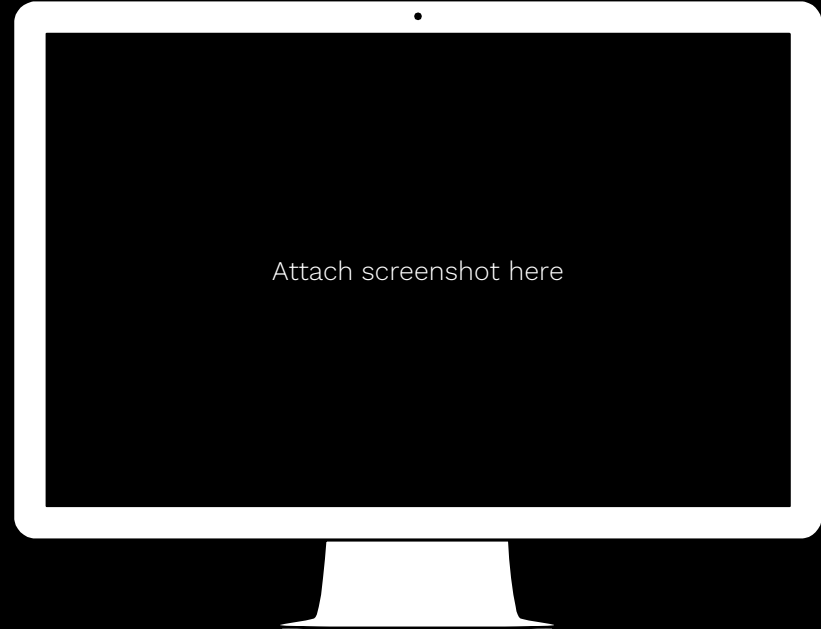
# PROJECTS





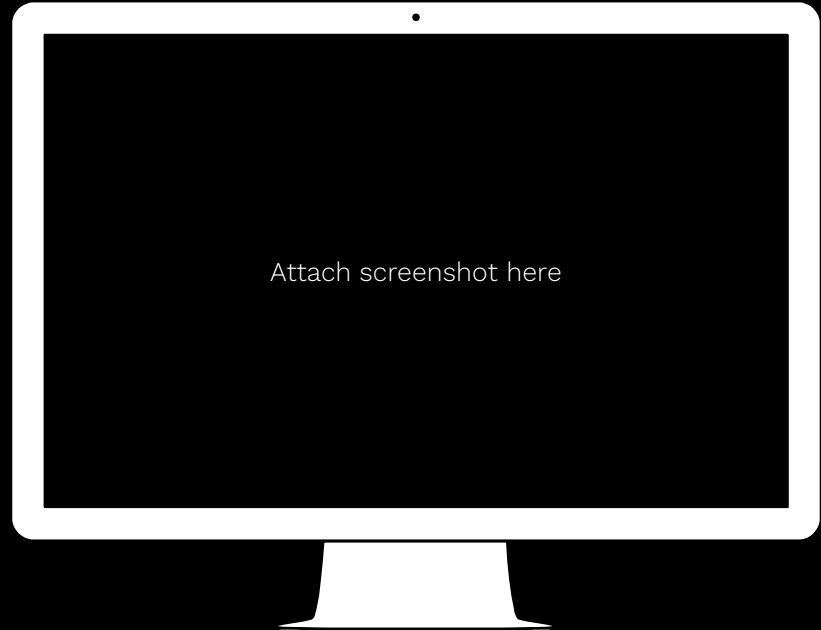
# RESTFUL API Using Express and MongoDB

[https://github.com/kaustubhghad  
ge/coding-  
bootcamp/tree/master/node-  
mongo-restful-api](https://github.com/kaustubhghadge/coding-bootcamp/tree/master/node-mongo-restful-api)



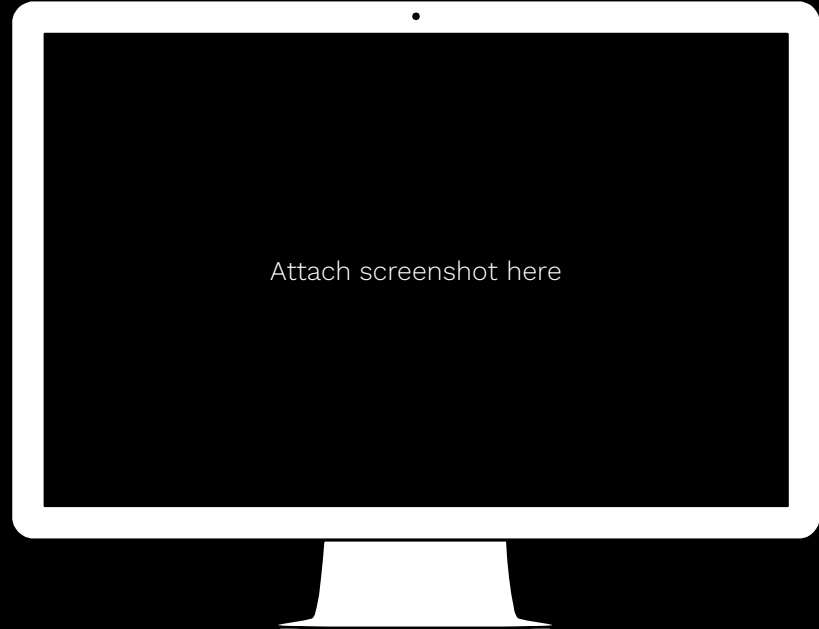
# User Authentication System [MongoDB]

[https://github.com/kaustubhghad  
ge/coding-  
bootcamp/tree/master/node-  
mongo-auth](https://github.com/kaustubhghadge/coding-bootcamp/tree/master/node-mongo-auth)



# User Authentication System [SQL]

[https://github.com/kaustubhghad  
ge/coding-  
bootcamp/tree/master/node-  
mongo-auth](https://github.com/kaustubhghadge/coding-bootcamp/tree/master/node-mongo-auth)



# Requirements

- Have a computer with macOS or Windows 10 or Ubuntu Linux with internet connection
- Fundamentals of programming
- Basic knowledge of Javascript

Visit our [course page](#) for more information.

# JavaScript Training

You should take the below course first if you are new to JS:

- [Intro to JavaScript](#)

# Linux Training

It is recommended to take the below course if you are new to Linux commands.

- [Introduction to Linux OS](#)

# 3.

## **Quick Refresher**

Let's start with the first set of slides

# Programming Refresh

- Literals
- Expressions
- Types
- Variables
- Functions
- Comments
- this
- Loops
- Arrow Functions
- Scope
- Array
- Semicolons



# this

The JavaScript **this** keyword refers to the object it belongs to. It has different values depending on where it is used:

- In a method, **this** refers to the owner object.
- Alone, **this** refers to the global object.
- In a function, **this** refers to the global object.

# Scope

- Scope determines the accessibility (visibility) of variables.
- In JavaScript there are two types of scope:
  - Local scope
  - Global scope

JavaScript has function scope: Each function creates a new scope.

# Arrow Functions

Arrow functions allow us to write shorter function syntax. They were introduced in ES6.

# Arrow Functions

## Regular function

```
hello = function() {  
    return "Hello  
World!";  
}
```

## Arrow function

```
hello = () => {  
    return "Hello World!";  
}
```

# Loop

Loops can execute a block of code a number of times.

Types of loops in Javascript -

- **for** - loops through a block of code a number of times
- **while** - loops through a block of code while a specified condition is true
- **do/while** - also loops through a block of code while a specified condition is true

# Semicolons

Semicolons(;) denote end of an expression.

Semicolons aren't mandatory in JavaScript. But it is a good practice to end your expressions with semicolons.

# Comments

There are two ways we can add comments in JavaScript -

- Multi-line comment - **`/* */`**
- Single-line comment - **`//`**

# Literals

Literal is a value that is written in source code.

Example -

- Number
- String
- Boolean, etc.



# Expressions

Any unit of code that can be evaluated to a value is an expression. They're further classified according to their type such as boolean, string, arithmetic etc.

E.g. - `300 > 19;`     `// evaluates to boolean value true`

# Types

- To be able to operate on variables, it is important to know their type.
- Javascript variables can hold multiple data types and are dynamic.

E.g. `var length = 10 // type Number`

# Variables

JavaScript variables are containers for storing data values.

E.g. - `var length = 100`

Which means variable “length” stores the value 100.

# Functions

- A JavaScript function is a block of code designed to perform a particular task.
- A JavaScript function is executed when "something" invokes it (calls it).

E.g. - `function myFunction(p1, p2) {  
 return p1 * p2; // The function returns the  
 product of p1 and p2  
}`

# Arrays

An array is a special variable, which can hold more than one value at a time.

E.g. - `var array_name = [item1, item2, ...];`

# More JavaScript Training

For more in-depth JS training, the below course is highly recommended:

- [Intermediate JS with jQuery, JSON and Ajax](#)

**4.**

**Setup**

# Installation

Official packages for all the major platforms are available at <https://nodejs.org/en/download/>



# Node Version Manager

- **nvm** is a popular way to run Node.js. It allows you to easily switch the Node.js version, and install new versions to try and easily rollback if something breaks, for example.
- It is also very useful to test your code with old Node.js versions.

# Resources

## macOS

- **Node.js** - Install using [homebrew](#)
- **VSCode** - [official link](#)

## Windows 10

- **Node.js** - Install using [chocolatey](#)
- **VSCode** - [official link](#)

## Ubuntu

- **Node.js** - Install using [apt-get](#)
- **VSCode** - [official link](#)

# My Setup

- OS - macOS 10.15.4
- Node.js Version - v12.11.1
- Code Editor - Visual Studio Code

**RECAP**

# Next Session

- Basics of Functional Programming
- What is Node.js
- Why does Node.js exist?
- NPM
- Fundamentals of Node.js
- Error Handling

# Live Private Coaching Sessions

- [Web design and development tutoring sessions- Weekly and monthly plans](#)
- [JavaScript, jQuery, Node.JS, MongoDB, and Express.JS- Private tutoring sessions](#)

# More JavaScript Courses

- [Intermediate JS with jQuery, JSON and Ajax](#)
- [Intro to Angular.JS Framework](#)
- [Intro to React.JS Framework](#)
- [Vue.JS Framework](#)

# Thanks!

**Any questions?**

You can find me at

- [@kaustubh\\_ghadge](https://twitter.com/kaustubh_ghadge)
- [kaustubh.ghadge@gmail.com](mailto:kaustubh.ghadge@gmail.com)

**coding-bootcamps.com**



# Coding Bootcamps





NEXT:  
Quick Refresher