**JavaScript Beginning to Mastery \_Syllabus.**

* **JavaScript vs EcmaScript**
* **Install vscode and run first program**

Contents

[Basics of Programming in JavaScript 1](#_Toc111110595)

[Arrays in JavaScript 2](#_Toc111110596)

[Objects in JavaScript 2](#_Toc111110597)

[Functions in JavaScript 3](#_Toc111110598)

[Very Important Array Methods 3](#_Toc111110599)

[More useful things 4](#_Toc111110600)

[Object Oriented JavaScript / Prototypal Inheritance 4](#_Toc111110601)

[How JavaScript Works 5](#_Toc111110602)

[DOM Tutorial 5](#_Toc111110603)

[Asynchronous JavaScript 6](#_Toc111110604)

# **Basics of Programming in JavaScript**

* **Hello world program**
* **Declare variable using var**
* **More about variable**
* **Let**
* **Const**
* **String Indexing**
* **Useful string methods**
* **Template Strings**
* **Null, undefined, BigInt, typeof**
* **Booleans and Comparison Operator**
* **Truthy and Falsy Values**
* **If else statement**
* **Ternary Operator**
* **&& || operator**
* **Nested if else**
* **If elseif else**
* **Switch statement**
* **While loop**
* **While loop examples**
* **For loop**
* **For loop examples**
* **Break and continue keyword**
* **Do while loop**

# **Arrays in JavaScript**

* **Intro to arrays**
* **Push pop shift unshift**
* **Primitive vs reference data types**
* **Clone array & spread operator**
* **For loop**
* **use const for creating arrays**
* **While loop in array**
* **For of loop**
* **For in loop**
* **Array destructuring**

# **Objects in JavaScript**

* **Intro to objects**
* **Dot vs Bracket Notation**
* **Iterate objects**
* **Computed properties**
* **Spread operator in objects**
* **Object Destructuring**
* **Objects inside Array**
* **Nested Destructuring**

# **Functions in JavaScript**

* **Function declaration**
* **Function Expression**
* **Arrow Functions**
* **Function declarations are hoisted (covered in great detail , later in this course)**
* **Function inside function**
* **Lexical Scope**
* **Block Scope Vs Function Scope**
* **Default Parameters**
* **Rest Parameters**
* **Parameter Destructuring**
* **Very brief intro to callback functions(covered in great detail , later in the course)**
* **Functions returning Functions**

# **Very Important Array Methods**

* **Foreach method**
* **Map method**
* **Filter**
* **Reduce**
* **Sort**
* **Find**
* **Every**
* **Some**
* **Fill method**
* **Splice method**

# **More useful things**

* **Iterables**
* **Sets**
* **Maps**
* **Object.assign**
* **Optional chaining**

# **Object Oriented JavaScript / Prototypal Inheritance**

* **Methods**
* **This keyword, Window object**
* **Call , apply and bind method**
* **Some warnings**
* **This inside arrow functions**
* **Short syntax for methods**
* **Factory functions & discuss some memory related problems**
* **First solution to that problem**
* **Why that solution isn’t that great**
* **What is \_\_proto\_\_ , [[prototype]]**
* **What is prototype**
* **Use prototype**
* **New keyword**
* **Constructor function with new keyword**
* **More discussion about proto and prototype**
* **Class keyword**
* **Example using class keyword**
* **Super keyword**
* **Method overriding**
* **Getters and setters**
* **Static methods and properties**

# **How JavaScript Works**

* **Global Execution context**
* **This and window in global execution context**
* **Hoisting**
* **Are let and const are hoisted ? What is a reference Error ?**
* **Function execution context**
* **Scope chain and lexical environment**
* **Intro to closures**
* **Closure example 1**
* **Closure Example 2**
* **Closure Example 3**

# **DOM Tutorial**

* **HTML and CSS Crash course ( Around 30-40 minutes)**
* **Async vs defer**
* **Select elements using id**
* **querySelector**
* **textContent & innerText**
* **Change the styles of elements using js**
* **Get and set attributes**
* **Select multiple elements and loop through them**
* **innerHTML**
* **Deeply understand DOM tree, root node , element nodes, text nodes**
* **classList**
* **Add new elements to page**
* **Create elements**
* **Insert adjacent elements**
* **Clone nodes**
* **More methods to add elements on page**
* **How to get the dimensions of the element**
* **Intro to events**
* **This keyword inside eventListener callback**
* **Add events on multiple elements**
* **Event object**
* **How event listener works**
* **Practice with events**
* **Create demo project**
* **More events**
* **Event bubbling**
* **Event Capturing**
* **Event delegation**
* **Create Project using event delegation**

# **Asynchronous JavaScript**

* **Is JavaScript a synchronous or asynchronous programming language ?**
* **SetTimeout()**
* **SetTimeout() with 0 millisecond**
* **Callback Queue**
* **SetInterval and create little project with setInterval**
* **Understand callbacks in general**
* **Callbacks in asynchronous programming**
* **Callback Hell and Pyramid of doom**
* **Intro to promises**
* **Microtask Queue**
* **Function that returns promise**
* **Promise and settimeout**
* **Promise.resolve and more about then method**
* **Convert nested Callbacks to flat code using promises**
* **Intro to Ajax, HTTP Request**
* **XHR requests**
* **Error handling in XHR requests**
* **XHR request Chaining**
* **Promisifying XHR requests and chaining using then method**
* **Fetch API**
* **Error Handling in Fetch API**
* **Consume Promises with async and Await**
* **Split code into multiple files using ES6 modules.**
* **Congratulations**
* **Now you know javascript in Great Details**
* **What next ?**

**Video Tutorial Timestamp :**

**Part 1 ( 11 : Hours )**

[**00:00**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=0s) **Introduction**

[**01:09**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=69s) **Syllabus**

[**11:12**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=672s) **History of JavaScript**

[**21:34**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=1294s) **Download and Install vscode**

[**23:34**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=1414s) **How to execute javascript code**

[**29:34**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=1774s) **Hello World program**

[**33:29**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=2009s) **Intro to variables**

[**42:05**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=2525s) **Rules for naming variables**

[**49:20**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=2960s) **Let keyword**

[**54:02**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=3242s) **Const keyword**

[**56:41**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=3401s) **String Indexing**

[**01:00:40**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=3640s) **Useful string methods**

[**01:09:11**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=4151s) **typeof , string to number, number to string conversion** [**01:16:06**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=4566s) **String Concatenation**

[**01:19:01**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=4741s) **Template Strings** [**01:22:34**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=4954s) **undefined null bigint** [**01:31:00**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=5460s) **Booleans and comparison Operators** [**01:37:01**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=5821s) **if else , truthy and falsy values** [**01:46:36**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=6396s) **Ternary operator** [**01:50:12**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=6612s) **And Or Operator** [**01:55:51**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=6951s) **Nested If else** [**02:02:39**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=7359s) **if elseif** [**02:08:59**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=7739s) **Switch statement** [**02:14:44**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=8084s) **While loop** [**02:21:36**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=8496s) **While loop example** [**02:28:46**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=8926s) **Intro to for loop** [**02:32:27**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=9147s) **for loop examples** [**02:34:29**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=9269s) **Break and Continue Keyword** [**02:38:17**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=9497s) **Do while loop** [**02:41:56**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=9716s) **Intro to arrays** [**02:52:05**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=10325s) **push pop shift unshift** [**03:00:03**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=10803s) **Primitive vs reference data types** [**03:16:29**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=11789s) **Clone array, concat, spread operator** [**03:26:12**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=12372s) **For loop in array** [**03:32:47**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=12767s) **Use const for creating arrays** [**03:37:46**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=13066s) **While loop in array** [**03:39:32**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=13172s) **For of loop** [**03:43:25**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=13405s) **For in loop** [**03:46:46**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=13606s) **Array destructuring** [**03:56:46**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=14206s) **Intro to objects** [**04:06:00**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=14760s) **Dot vs bracket notation** [**04:11:09**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=15069s) **How to iterate objects** [**04:18:21**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=15501s) **Computed Properties** [**04:23:14**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=15794s) **Spread operator in array and objects** [**04:33:59**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=16439s) **Object destructuring** [**04:42:13**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=16933s) **Object inside array** [**04:46:03**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=17163s) **Nested destructuring** [**04:51:08**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=17468s) **Function declaration** [**05:11:43**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=18703s) **Function Expression** [**05:16:22**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=18982s) **Arrow Functions** [**05:24:46**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=19486s) **Hoisting intro ( later covered in detail)** [**05:29:22**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=19762s) **Functions inside functions** [**05:33:21**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=20001s) **Lexical Scope** [**05:43:15**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=20595s) **Block scope vs function scope** [**05:53:41**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=21221s) **Default parameters** [**05:56:19**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=21379s) **Rest parameters** [**06:02:35**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=21755s) **Param destructuring** [**06:05:54**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=21954s) **callback functions** [**06:13:09**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=22389s) **Function returning function** [**06:18:48**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=22728s) **forEach method** [**06:35:48**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=23748s) **map method** [**06:43:57**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=24237s) **filter method** [**06:47:55**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=24475s) **reduce method** [**07:04:34**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=25474s) **sort method** [**07:26:31**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=26791s) **find method** [**07:33:12**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=27192s) **every method** [**07:39:25**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=27565s) **some method** [**07:43:18**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=27798s) **fill method** [**07:46:08**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=27968s) **splice method** [**07:52:20**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=28340s) **Iterables, array like objects** [**07:57:15**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=28635s) **sets** [**08:11:13**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=29473s) **map data structure** [**08:29:10**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=30550s) **clone using Object.assign** [**08:32:02**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=30722s) **optional chaining** [**08:38:42**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=31122s) **Create your own methods** [**08:52:00**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=31920s) **This keyword** [**08:55:44**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=32144s) **call, apply, bind methods** [**09:05:49**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=32749s) **small warning** [**09:08:55**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=32935s) **arrow functions and this** [**09:12:36**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=33156s) **short syntax** [**09:13:24**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=33204s) **create functions to create multiple objects** [**09:23:03**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=33783s) **store methods in different object** [**09:28:37**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=34117s) **solution using object.create** [**09:45:55**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=35155s) **What is prototype** [**09:56:21**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=35781s) **Use prototype** [**10:04:18**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=36258s) **New keyword** [**10:17:17**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=37037s) **hasOwnProperty** [**10:19:35**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=37175s) **more about prototype** [**10:26:43**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=37603s) **class keyword** [**10:35:12**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=38112s) **class practice and extends keyword** [**10:42:48**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=38568s) **super keyword** [**10:47:18**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=38838s) **same method in base class** [**10:49:35**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=38975s) **getters and setters** [**11:02:47**](https://www.youtube.com/watch?v=chx9Rs41W6g&t=39767s) **static methods and properties**

**Part 2 ( 11 : Hours )**

[**00:00**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=0s) **Introduction** [**02:39**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=159s) **How javascript works overview** [**16:19**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=979s) **How javascript execute code** [**18:00**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=1080s) **Global execution context** [**32:11**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=1931s) **What happens to function declarations ?** [**45:33**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=2733s) **What is hoisting ?** [**49:07**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=2947s) **What happens to function expressions ?** [**56:18**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=3378s) **Are let and const are hoisted ?** [**01:08:36**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=4116s) **Quick summary of what we’ve learned so far** [**01:14:54**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=4494s) **Function execution context** [**01:31:46**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=5506s) **Scope chain** [**01:40:53**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=6053s) **Intro to closures** [**01:47:02**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=6422s) **Closures Example 1** [**02:07:57**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=7677s) **Closures Example 2** [**02:17:22**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=8242s) **Closures Example 3** [**02:31:55**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=9115s) **Closures Example 4** [**02:50:43**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=10243s) **Create Webpage using HTML & CSS** [**03:25:37**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=12337s) **Async vs Defer | Correct way of linking JS file** [**03:39:46**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=13186s) **What is DOM ?** [**03:47:38**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=13658s) **getElementById** [**03:52:23**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=13943s) **querySelector** [**03:57:48**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=14268s) **textContent and innerText** [**04:02:53**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=14573s) **Change the styles of elements** [**04:09:20**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=14960s) **get and set attributes** [**04:14:29**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=15269s) **querySelectorAll & getElementsbyClassName** [**04:20:01**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=15601s) **Iterate elements** [**04:32:14**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=16334s) **innerHTML** [**04:36:19**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=16579s) **DOM Tree** [**04:52:30**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=17550s) **Traversing DOM Tree** [**05:16:55**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=19015s) **classList ,add and remove, toggle classes** [**05:25:10**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=19510s) **Add HTML element using JavaScript** [**05:31:49**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=19909s) **createElement, append, prepend, remove, before, after** [**05:40:15**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=20415s) **insertAdjacentHTML** [**05:45:45**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=20745s) **Clone Nodes** [**05:48:58**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=20938s) **appendChild, insertBefore, replaceChild, removeChild** [**05:54:10**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=21250s) **static list vs live list** [**06:01:17**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=21677s) **How to get the dimensions of an element** [**06:04:40**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=21880s) **Intro to events** [**06:16:35**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=22595s) **This keyword inside eventListener callback** [**06:21:17**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=22877s) **Click event on multiple buttons** [**06:31:27**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=23487s) **Event object** [**06:45:02**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=24302s) **Events Behind the scenes** [**07:04:32**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=25472s) **practice with click events** [**07:06:51**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=25611s) **Little demo project** [**07:18:09**](https://www.youtube.com/watch?v=Llsq1y-HWs4&t=26289s)

**Part 3 ( 3 : Hours )**

[**00:00**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=0s) **Intro**

[**01:32**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=92s) **setTimeout**

[**25:53**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=1553s) **setInterval**

[**42:52**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=2572s) **callbacks in general**

[**55:26**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=3326s) **callbacks, callback hell, the pyramid of doom**

[**01:24:01**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=5041s) **Intro to promise**

[**01:58:33**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=7113s) **Function returning a promise**

[**02:00:12**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=7212s) **Promise and setTimeout**

[**02:02:34**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=7354s) **Promise.resolve and promise chaining**

[**02:09:37**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=7777s) **Callback hell to flat code**

[**02:22:42**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=8562s) **AJAX Basic Theory**

[**02:27:50**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=8870s) **XHR (XML HTTP Request)**

[**02:53:03**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=10383s) **XHR practice and error handling**

[**03:02:49**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=10969s) **XHR using promises**

[**03:13:09**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=11589s) **Fetch API**

[**03:24:00**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=12240s) **Consume Promises Using Async Await**

[**03:33:47**](https://www.youtube.com/watch?v=pAl_uGXKXq8&t=12827s) **ES6 Modules**