

## # Complete JavaScript Syllabus - Basic to Advanced

### ### Introduction

- What is programming language?
- What is front-end?
- What is back-end?
- Introduction
  - HTML
  - CSS
  - JavaScript
- Role of HTML, CSS, and JavaScript in WEB development?

### ### History of JavaScript

- JavaScript history
- ECMAScript
- Versions of JavaScript & History
- Backward & Forward compatible

### ### Project setup

- Separation of concern
- Setting up starter files

## ## JavaScript Core

### ### JavaScript Features

- High Level Language
- Garbage Collected
- Interpreted Language (JIT)
- Multi Paradigm
- Prototype based functions
- First Class Function
- Dynamically Type
- Single Threaded
- Non-Blocking Event Loop
- Platform Independent

### ### Interaction

- console.log
- document.write()
- window.alert()
- innerHTML

### ### Value, Variable and Data types

- Value, Variable and Data Types
- Difference between Primitive and Non-Primitive Data Types ?
- Primitive Data types (In built data types)
  - Number
  - String
  - Undefined
  - Boolean
  - Symbol

- BigInt
- Non-primitive Data types
  - Object
  - function

### ### Truthy and Falsy Values

- What are the truthy and falsy values in JavaScript ?
- Falsy values : undefined, 0, null, '', false, NaN
- Using Falsy and Truthy values in conditional statements

### ### Type Conversion

- Implicit type conversion/Automatic Type conversion/Coercion
- Explicit type conversion/Manual Type conversion
  - Number
    - String to Number
    - Boolean to Number
    - Undefined to Number
  - String
    - Number to String
    - Boolean to String
    - Undefined to String
  - Boolean
    - String to Boolean
    - Number to Boolean
    - Undefined to Boolean

### ### JavaScript Popup Boxes

- Alert Box
- Confirm Box
- Prompt Box

### ### Identifiers

- What is an identifier ?
- Rules for creating identifier
- Reserved keywords

### ### Comments

- What is comment ?
  - Single line comment
  - Multi line comment
- Coding guidelines for comments

### ### Statement

- What is a statement in programming ?
- How to write a single line of statement ?
- How to write a multi-line statement or code-block ?
- Use of Semicolon and whitespace in statement.

### ### let, const and var

- let
- const

- var
- Difference between let, const and var?
- When to use let, const and var ?

### ### JavaScript operators

- Arithmetic operator
  - \+ Addition
  - \- Subtraction
  - \\* Multiplication
  - \*\* Exponentiation (ES2016)
  - / Division
  - % Modulus (Division Remainder)
  - ++ Increment
  - -- Decrement
- Assignment operator
  - =
  - +=
  - -=
  - \*=
  - /=
  - %=
  - \*\*=
- Comparison operator
  - == equal to
  - === equal value and equal type
  - != not equal
  - !== not equal value or not equal type
  - \> greater than
  - < less than
  - \>= greater than or equal to
  - <= less than or equal to
  - ? ternary operator
- Logical operators
  - && logical and
  - || logical or
  - ! logical not
- Operator precedence
  - [Operator Precedence Table] ([https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Operator\\_Precedence#table](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Operator_Precedence#table))
  - Truth table of &&, || and !
  - Loose equality operator ==
  - Strict equality operator ===
  - Typeof operator
  - Ternary operator

### ### Conditional statements

- if
- else if
- else
- Grouping multiple conditions using logical operator
- Nested If else
- Switch case

### ### use strict

- What is the significance of use strict ?
- How JavaScript behave without use strict ?

### ### Looping and Switch

- For Loop
- While Loop
- Do while loop
- Loop inside loop
- Backwards Loop
- For of loop
- For in loop
- Break and continue

### ### JavaScript String

- What is a String ?
- Uses of single quote `` and double quotes "" in string
- Escape characters
- How to create a String ?
  - String Literal
  - String Object
- String Literal vs String Object ?
- String length
- String to Array
- String Template Literal
- String functions
  - charAt
  - slice
  - substring
  - substr
  - replace
  - repeat
  - toUpperCase
  - toLowerCase
  - concat
  - trim
  - trimEnd
  - trimStart
  - padStart
  - padEnd
  - split
  - indexOf
  - valueOf
  - lastIndexOf
  - startsWith
  - endsWith
  - search
  - match
  - includes

### ### Scope

- What is Scoping
- Lexical Scoping

- Different types of scopes in JavaScript ?
  - Global Scope
  - Functional scope
  - Block scope

### ### Functions Basics

- Type of functions ?
  - Function declaration
  - Function expression
  - Arrow function
  - Anonymous function
- Difference between function declaration and function expression ?
- Difference between function expression and arrow function ?
- Function invoking/calling
- Function calling from other function
- Function as values
- Function returning function
- Parameters
- Arguments
- Arguments Object in functions
- Default parameters
- Passing arguments: value vs reference

### ### Functions Advanced

- First Class function/Citizen
- High Order function
- Callback function
- setTimeout
- setInterval
- The call and apply methods
- The bind method
- Immediately invoked function expression
- Closures
- Hoisting
- Temporal Dead Zone (TDZ)
- DRY Principle
- Currying
- eval()
- recursion

### ## Debugging

- Debugging
- Overview of Google chrome developer tools
- Debugging points, adding a breakpoint
- Fixing errors
  - console.log
  - console.warn
  - console.error
  - console.table
- How to fix a bug, different steps:
  - Identifying bug
  - finding bug
  - fixing bug
  - Not repeat bugs
- Different type of errors

- Syntax Error
- Reference Error
- Type Error
- Internal Error
- Range Error
- URI Error

## ## Numbers and Dates

### ### Number

- 64-bit Floating Point Numbers
- Converting numbers
- Number System
  - Binary
  - Octal
  - Decimal
  - HexaDecimal
- Checking numbers
- Math and Rounding
- The Remainder operator
- Numeric Separators
- BigInt
  - Working with BigInt
  - Exceptions in BigInt
- Number class functions
  - toFixed
  - toString
  - valueOf
  - Number constructor
  - parseInt
  - parseFloat
  - isNaN
- Number Properties
  - MAX\_VALUE
  - MIN\_VALUE
  - POSITIVE\_INFINITY
  - NEGATIVE\_INFINITY
  - NaN
  - Infinity
- Internationalization Numbers

### ### Date

- What is Date ?
- Creating Dates using different ways of creating Date object
- Understanding milliseconds and other units of time
- Date setter methods
- Date getter methods
- Internationalization Dates

## ## JavaScript Data Structures

### ### Array

- What is an Array ?
- Need of an Array ?
- How to create an Array ?

- Array Literal
- Array Object
- Index in Array
- Array length property
- Array Declaration
- Looping Array
- Array functions
  - sort
  - push
  - pop
  - unshift
  - shift
  - toString
  - join
  - concat
  - splice
  - slice
  - sort
  - reverse
  - forEach
  - at
  - map
  - filter
  - reduce
  - find
  - findIndex
  - some
  - every
  - flat
  - flatMap

### ### Object

- What is an object ?
- Creating object
  - Object literal syntax
  - new keyword
- Annotation
  - Dot
  - Bracket
- Object properties
  - Key
  - Value
- Array in Object
- Function in Object
- Uses of this in Object
- Object methods
  - Keys
  - Values
  - Entries
- For Of Loop In Object
  - Object.keys()
  - Object.values()
  - Object.entries()

### ### Set

- What is a Set ?

- Creating set
- Elements order in Set
- Set Functions
  - size()
  - has()
  - delete()
  - other functions
- Index in set
- Printing set values using for of loop
- Creating set to array
- forEach method
- WeakSet

### ### Map

- What is a Map ?
- Creating new map
- Adding value in map
- Chaining in map
- Functions
  - get()
  - has()
  - size()
  - clear()
- Array as key
- Iteration of Map
- Object to map
- Map to array
- forEach function on map
- WeakMap

### ## Modern Features

#### ### Destructuring Arrays

- What is destructuring
- Reverse values using destructuring
- Returning two values from function
- Destructuring of nested array
- Setting default values in Destructuring
- Destructuring Objects
- Extract value in Destructuring
- Different property name in Destructuring
- Nested Object Destructuring
- Destructuring in function

#### ### The Spread Operator

- Copy Array
- Join 2 Arrays
- String to array using spread operator
- Passing array in function and spread to parameters using spread
- Shallow copy using spread operator

#### ### The Rest Parameter

- Assign values
- Rest element last element



- Assign values in object
- Variable arguments in function

### ### Short Circuiting

- Use of ||
- Replace with ternary operator
- Working of short circuiting with non nullish values
- Use of &&
- Calling function using &&
- The Nullish Coalescing Operator ??
- Logical Assignment Operator
  - ||=
  - &&=
  - ??=

### ### Enhanced Object literals

- Exactly same name
- Function in object
- Compute property name

### ### Optional Chaining

- Multiple condition in if condition
- Work for nullish
- Checking if method exist
- Checking array is empty

## ## JavaScript DOM and BOM

### ### DOM (Document Object Model)

- Introduction
- DOM functions
  - getElementById
  - getElementsByTagName
  - getElementsByClassName
  - querySelector
  - querySelectorAll
  - write()
- Properties
  - innerHTML
  - attribute
  - style.property
  - textContent
- Forms
  - Forms validation
  - Properties
    - Disabled
    - Max
    - Min
    - Pattern
    - Required
- Type of Events
  - Onclick
  - Onchange
  - Other events

- Mouse events
  - Onmousedown
  - Onmouseup
  - Other events
- Event Listener
  - addEventListener
  - Event bubbling
  - Event capturing
  - Event Delegation
- Navigation
  - parentNode
  - childNodes
  - firstChild
  - lastChild
  - nextSibling
  - previousSibling
- DOM Nodes
  - createElement
  - createTextNode
  - appendChild

### ### JavaScript BOM

- Window
- History
- Navigator
- Screen
- Location
- Timing
- Cookies
- LocalStorage

### ## RegExp

- What is Regex
- search()
- exec()
- test()
- match()

### ## Json & XML

- What is JSON?
- Rules for creating valid JSON?
- What is XML?
- Rules for creating valid XML?

### ### JavaScript Inside

- JavaScript Engine
- Call Stack
- Execution Context
- Memory/Heap
- Compiler
- Interpreter
- Compiler Vs Interpreter
- Event Loop
- Execution Context consists of 3 things:

- Variable Environment
  - let, const and var declarations
  - functions
  - Arguments Objects
- Scope Chain
- this keyword
- Execution Context divides in two parts
  - Type of execution context
    - Global
    - Functional
- Creation Phase
- Code Phase
- Scope Chain:
  - Scoping: How our programs variables are organized and accessed
- 3 types:
  - Global Scope
  - Local/Function Scope
  - Block Scope
- this key word
  - this in global scope
  - this in function
  - this in object
  - this in arrow function
  - this in inside function inside object
- Primitive vs Object
  - Understanding of how primitive and non-primitives are stored in memory
  - Copying object
  - Copy first level properties
    - Shallow copy
    - Deep copy

## OOPS

### JavaScript OOPs

- OOPs Concepts
  - Object
  - Class
  - Abstraction
  - Encapsulation
  - Inheritance
  - Polymorphism
- OOPs in JavaScript
  - Constructor functions and new operator
    - Prototypes
    - Native Prototype
    - Prototypal inheritance and prototype chain
    - Prototypal inheritance on Built-in objects
  - ES6 classes
    - Setters and Getters
    - Static methods
    - Object.create
    - Inheritance between classes
  - Object.create function
  - Inheritance
    - Through constructor functions
    - Using ES6 classes

- Using object.create
- Encapsulation
  - Protected Properties and Methods
  - Private Class Fields and Methods
- Chaining methods
- Mixins

## ## Web

- How the web works?
  - Server
  - Client
  - Request
  - Response
  - API
  - REST Basics
  - SOAP
  - CORS

## ### AJAX

- What is AJAX?
- XMLHttpRequest (XHR)
- Calling an api using AJAX
- Handling response

## ### Asynchronous JavaScript

- What is Asynchronous ?
- Promise
  - Promise and Fetch API
    - Download progress
    - Abort
    - Consuming Promises
    - Chaining Promises
  - Asynchronous
    - The Event Loop
    - Create a Simple Promise
    - Consuming Promise using Async/Await
    - Returning values from Async functions
  - Running promises in Parallel
  - Promise Combinators
    - race
    - allSettled
    - any

## ### Error Handling

- Error Handling
  - try
  - catch
  - finally
- Error object
- Throwing errors
- Custom Errors
- Error Propagation
- Async Await error handling
- Handling Rejected Promises
- Handling AJAX errors
- Global Error handling

### ### Modules

- Modules
  - Exporting and importing in Modules
  - Top-Level await
  - The Module Pattern

### ### JavaScript Tools and Techniques

- Bundling With Parcel and NPM Scripts
- Babel
- Transpiling
- Polyfilling
- Transpiling vs Polyfilling

### ### Security:

- Cross-Site Scripting (XSS)
- Cross-Site Request Forgery (CSRF)
- Content Security Policy (CSP)
- HTTPS and Secure Cookies

### ## Advanced topics

- Generators
- Proxy objects
- Reflection
- Web workers
- Service workers
- Web Sockets
- Memoization
- Immutable Data Structures
- Functional Reactive Programming (FRP)
- WebRTC (Real-Time Communication)
- WebAssembly