



# Thapa Technical

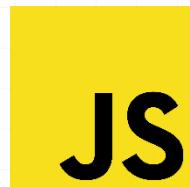
# JavaScript

# Complete Course

[Kodyfier.com](https://Kodyfier.com) - Online Classes

**THAPA TECHNICAL**

# COURSE TOPICS



## BASICS

- How Website Works?
- What is JavaScript?
- History of JavaScript
- Values & Variables
- Data Types in JavaScript
- Concat & Type Coercion
- Operators & Expression
- If Statements & Loops
- Functions in JavaScript
- Arrays in JavaScript
- Strings in JavaScript
- Math Object
- Date & Time in JavaScript

## ADVANCED

- EcmaScript 2015 - 2024
- Window Objects –  
BOM vs DOM
- Events Objects in JavaScript
- localStorage in JavaScript
- Timing Based Events
- Objects in JavaScript
- OOPs in JavaScript
- Event Propagation
- Advanced Functions
- JSON & FETCH API & other APIs
- Promises, Async-Await
- Error Handling in JavaScript

## PRO LEVEL

- How JavaScript Works ?
- 100+ Interview Questions
- 50+ Tips & Tricks
- Notes + List of  
Deprecated properties
- 150+ Animated Slides

PROJECT -  
**ECOM WEBSITE WITH  
HTML, CSS &  
JAVASCRIPT** ❤️

Subscribe: ThapaTechnical

# How to get Most from Our JavaScript Course?

**Code Along:** Avoid passively watching the videos. You'll learn **zero JavaScript skills** by just observing.

**Code along with me!** Get your hands dirty and practice coding yourself.

**Use the Timeline:** Utilize YouTube's **timeline** feature to skip sections or revisit topics as needed.

**Problem-Solving Practice:** Attempt coding **challenges independently before watching** the solutions.

**Play and Learn:** Don't be afraid to mess around with code to understand it better.

**Find Help:** If you're stuck, look up explanations or ask for help online (**Comment Section / Discord**).

**Think Back and Practice:** Look back on what you've learned and practice it again (**After 5days**).

**Set Targets:** Decide what you want to achieve and take small steps to get there. Don't rush.



**Kodyfier.com** - Online Classes

**THAPA TECHNICAL**

# My recommendation - JavaScript Course

**Make a Plan:** Consider how much time you can dedicate each day. For instance, watching 1 hour daily will finish the course in 12 days, while 3 hours daily will complete it in 4 days. Adjust your schedule to fit your learning pace.

**What I want:** You could aim to watch 2 hours of videos and practice for 1 hour each day.

In just 6 days, you'll have a basic understanding of JavaScript.

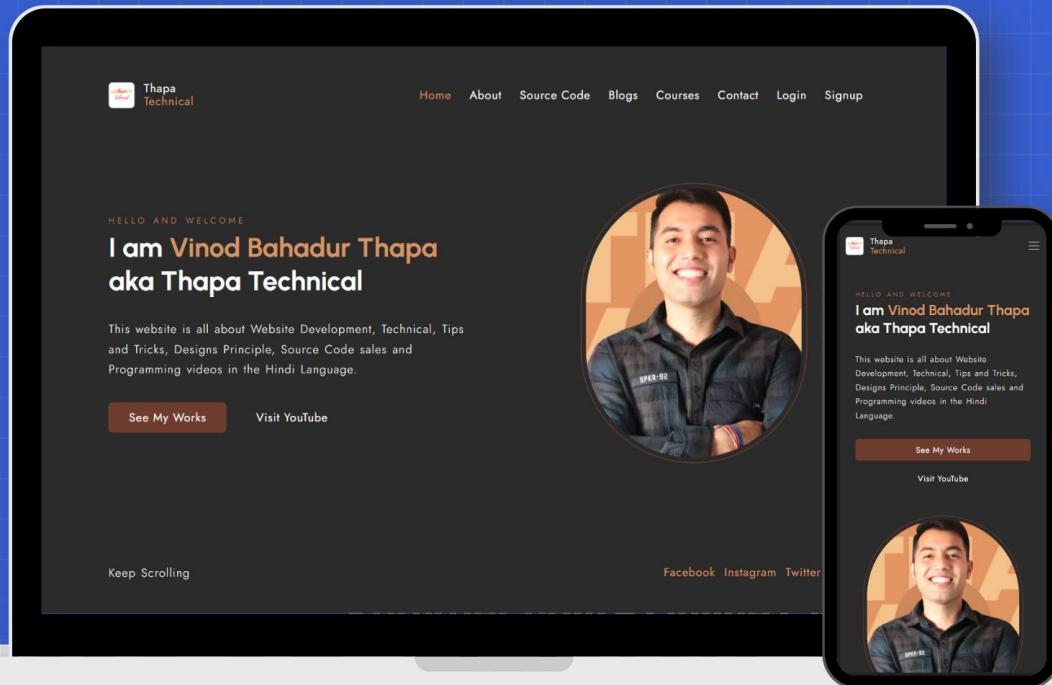
**Bonus:** I added JS quizzes in my website, you can go there and see how much you learned.

[Kodyfier.com](http://Kodyfier.com) - Online Classes

THAPA TECHNICAL

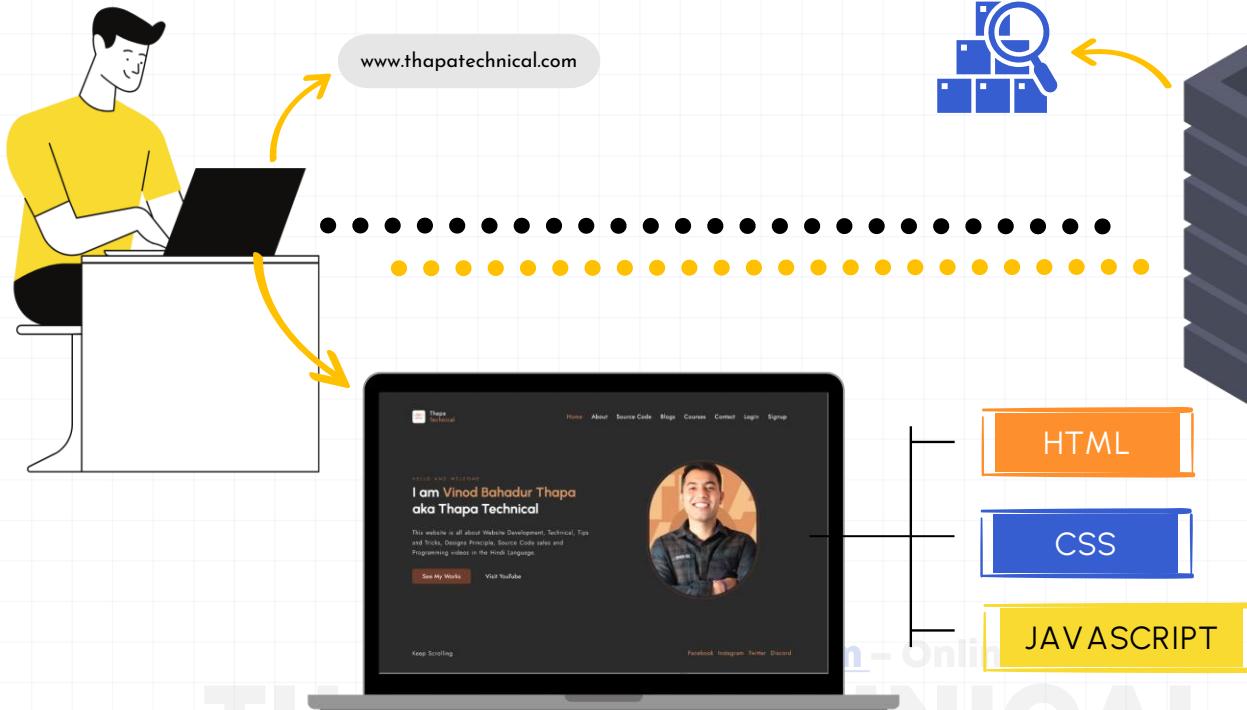
# 01

# HOW WEBSITE WORKS?



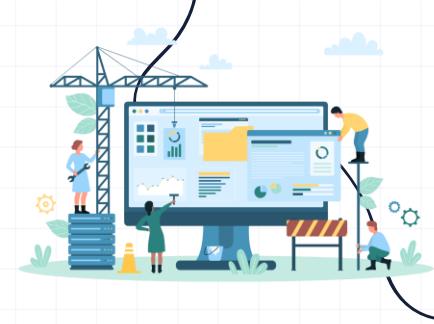
## THAPA TECHNICAL

# CLIENT VS SERVER



THAPA TECHNICAL

# BUILDING BLOCK OF WEBSITE



## HTML

Provides the structure and content of a webpage.

## CSS

Styles and designs the appearance of the webpage

## JS

Adds interactivity and dynamic behavior to the webpage.



# REAL LIVE EXAMPLE



Kodyfier.com – Online Classes  
**DATA TECHNICAL**

# What is **JavaScript**?

**JavaScript** improves the user experience of the **web page** by converting it from a static page into an **interactive** one.

OR

**JavaScript** is used to update and change both HTML and CSS. It adds **behavior** to web pages.

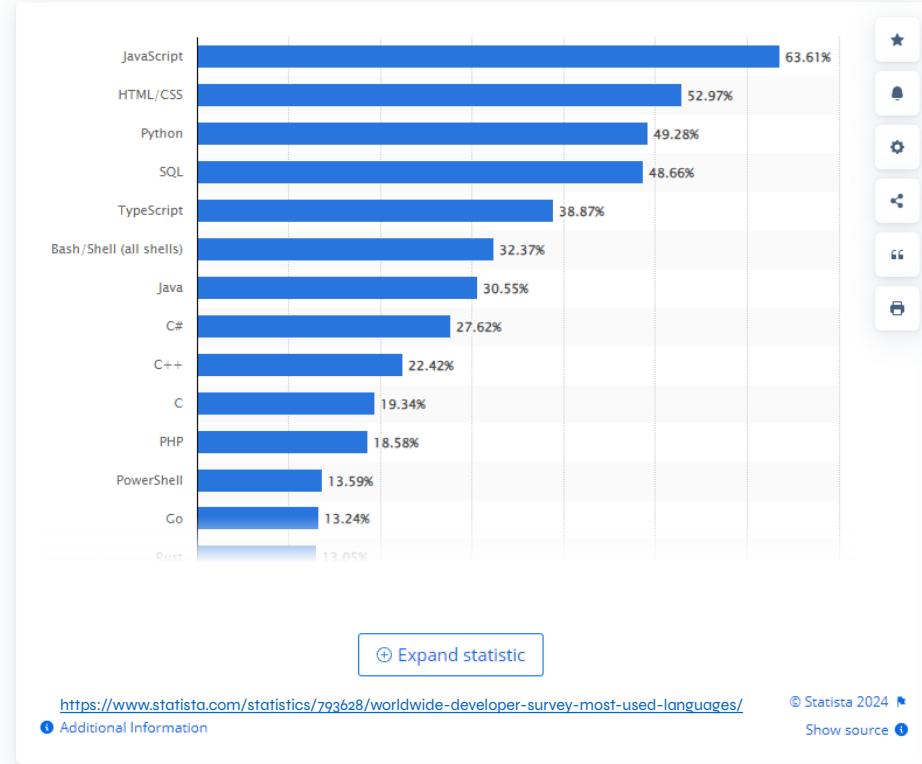
[Kodyfier.com](http://Kodyfier.com) - Online Classes

THAPA TECHNICAL

[Subscribe: ThapaTechnical](#)

# Why JavaScript?

Most used programming languages among developers worldwide as of 2023



Subscribe: ThapaTechnical

# History of JavaScript

In 1995 - Created by **Brendan Eich** at **Netscape** in just **10** days.

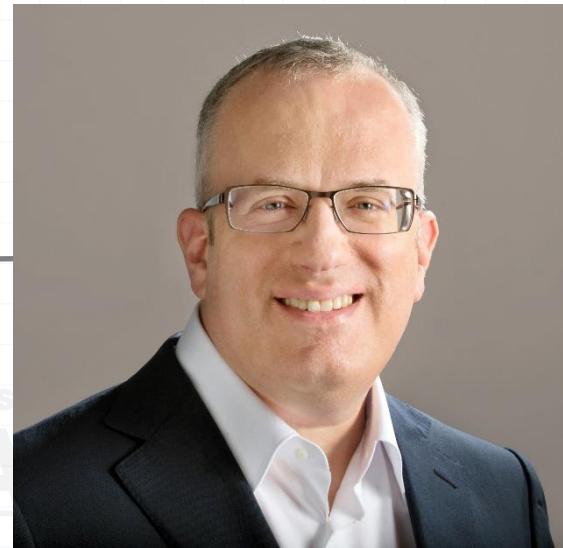


Switching from **LiveScript** to **JavaScript** was a smart move to make it sound cooler and piggyback on **Java's** fame, while also cozying up to Sun Microsystems.

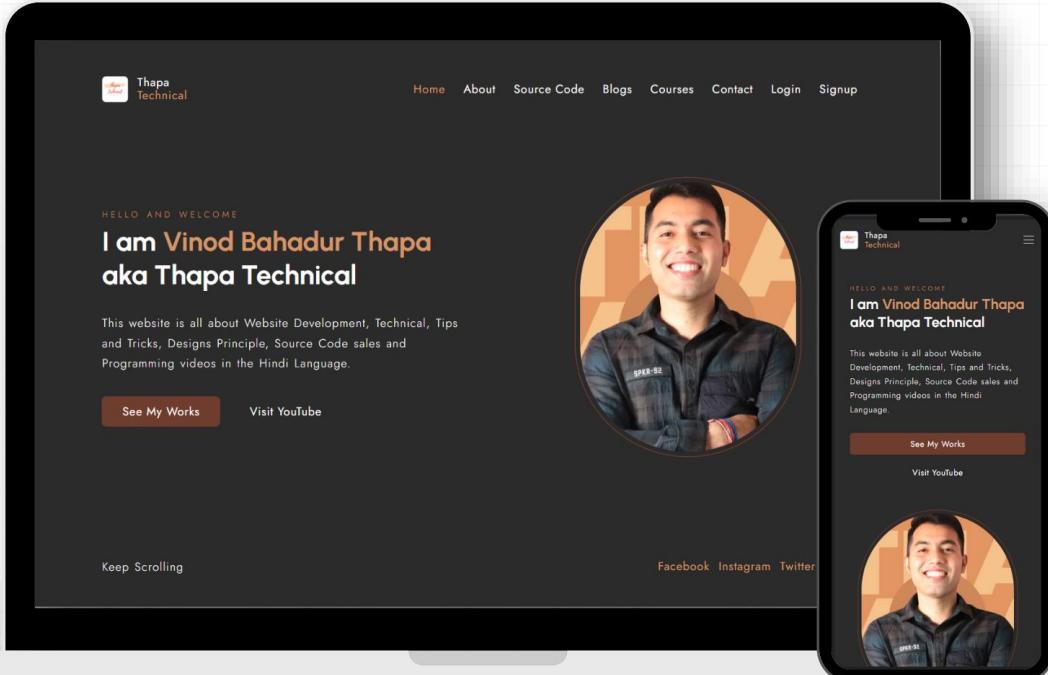


**JAVASCRIPT**  
JavaScript was created

**ECMA INTERNATIONAL**  
It was then submitted to **ECMA International** for standardization, which resulted in ECMAScript



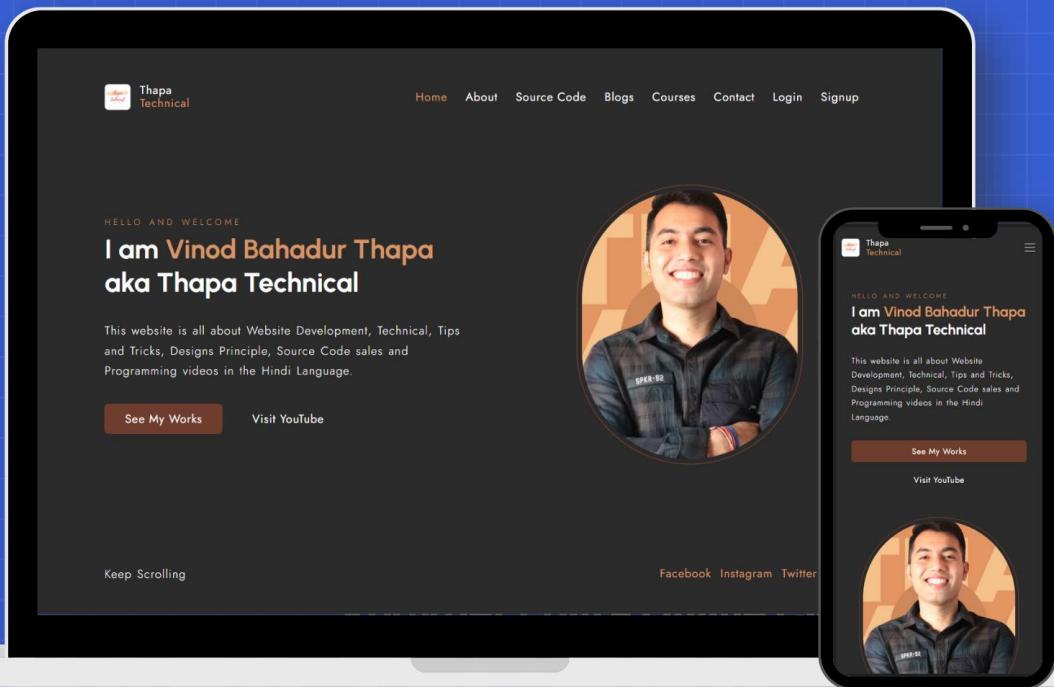
# Let's write Our First JavaScript Code in Console



Subscribe. [ThapaTechnical](#)

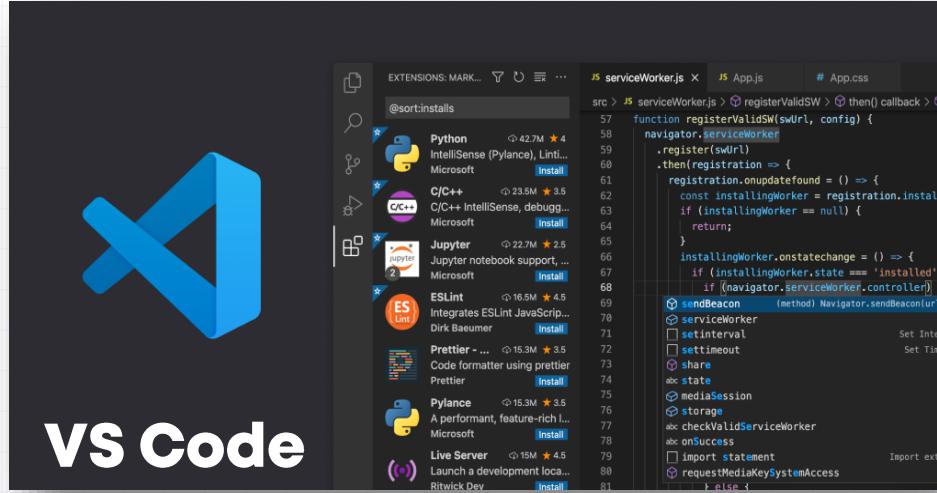
# WAYS TO WRITE JAVASCRIPT

03



THAPA TECHNICAL

# We need a Code Editor

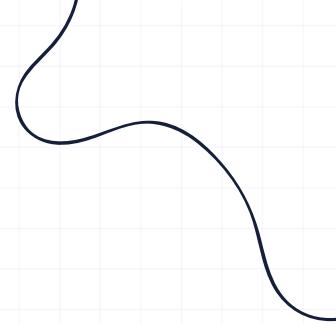


Kodyfier.com - Online Classes

HAPA TECHNICAL

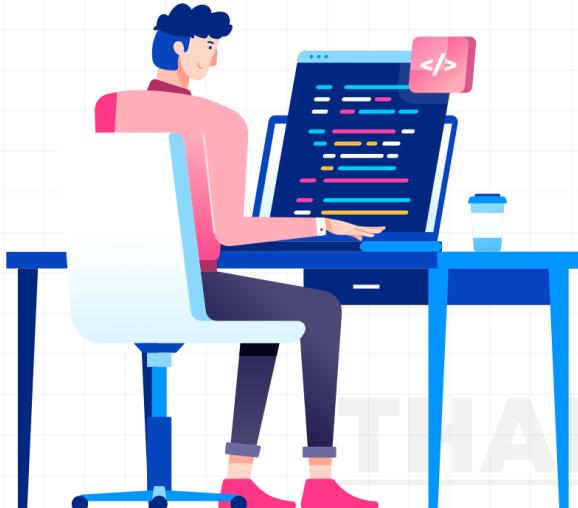
# Inline JavaScript

```
<button onclick="alert('Hello')">Click me</button>
```



# Internal JavaScript

```
<script> console.log('Hello, world!'); </script>
```



# External JavaScript

```
<script src="script.js"></script>
```

[Kodyfier.com](https://kodyfier.com) - Online Classes

THAPA TECHNICAL

JAVASCRIPT

\* Values &  
Variables

[Kodyfier.com](http://Kodyfier.com) – Online Classes

**THAPA TECHNICAL**

## Variable Name

A variable is a container(box) that holds a value.

```
var myName = "vinod"
```

Variable (key)

Value

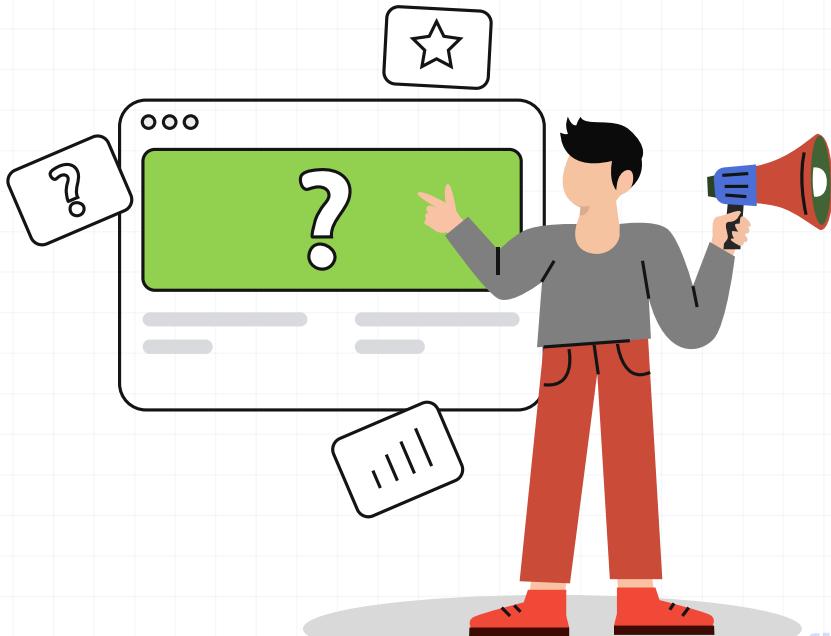
[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# Naming Variables: Rules and Best Practices

- \* Variable names must start with a letter, an **underscore** (\_) or a **dollar sign** (\$).
- \* Variable names cannot contain spaces.
- \* By convention, JavaScript variable names are written in **camelCase**.
- \* Variables cannot be the same as **reserved keywords** such as if or const.
- \* Variable names are **case sensitive**.
- \* Variable names can be as long as you need



# Challenge Time?

Are you ready for the challenge.



THAPA TECHNICAL

[Subscribe: ThapaTechnical](#)

Pause the video & try yourself

# Questions



```
var my(firstName = "John";
```

```
var _myLastName$ = "Doe";
```

```
var 123myAge = 25;
```

```
var $cityName = "New York";
```

```
var my@Email = “Thapa@me.com”;
```

# Answers

Kodyfier

IHAPA TEC

Pause the video & try yourself

# Questions



```
var my(firstName = "John";
```

```
var _myLastName$ = "Doe";
```

```
var 123myAge = 25;
```

```
var $cityName = "New York";
```

```
var my@Email = "Thapa@me.com";
```

# Answers

\* This is a valid variable name.

\* This is a valid variable name.

\* This is not a valid variable name.

\* This is a valid variable name.

\* This is not a valid variable name.

# JAVASCRIPT

# \* Data Types

[Kodyfier.com](http://Kodyfier.com) – Online Classes

**THAPA TECHNICAL**

## Data Types

Data Types

Primitive Data Types

String 1

Number 2

Boolean 3

Undefined 4

Null 5

Bigint 6

Symbol 7

Object Data Types

An Object 1

An Array 2

A Date 3

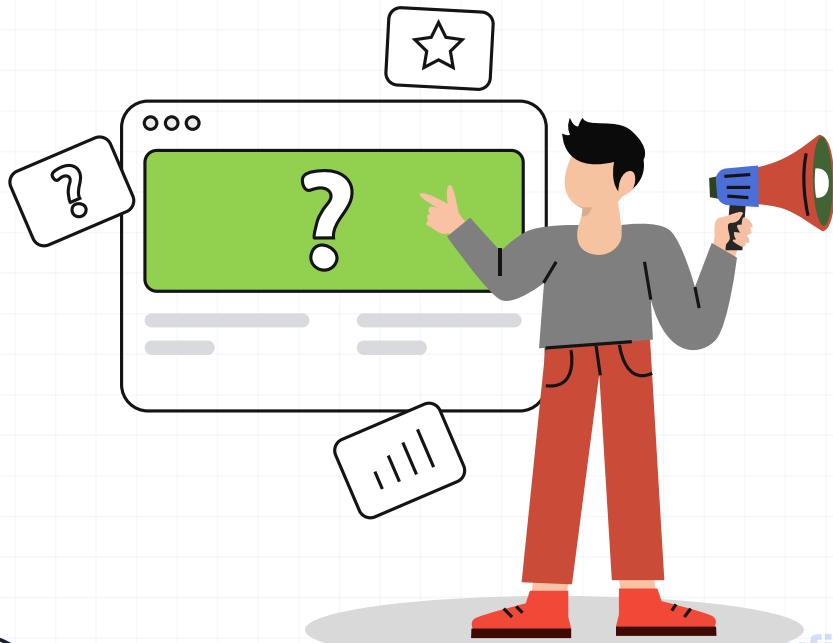
Kodytier.com - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# Interview Questions – Data Types

- 1: What is the difference between **null** and **undefined** in JavaScript ?
- 2: What is the purpose of **typeof** operator in JavaScript ?
- 3: What is the result of `typeof null` in JavaScript ?
- 4: What are **primitive data types** in JavaScript ?
- 5: Explain the concept of **truthy and falsy** values in JavaScript. Provide examples ?



# Challenge Time?

Are you ready for the challenge.



THAPA TECHNICAL

Subscribe: ThapaTechnical

# Wait!!!



Explore more  
for a solid  
understanding

I want you to understand it thoroughly.



Rodyfier.com - Online Cr

## THAPA TECHNICAL

Subscribe: ThapaTechnical

## More on Data Types

- a. 10 + "20"
- b. 9 - "5"
- c. "Java" + "Script"
- d. " " + " "
- e. " " + 0
- f. "vinod" - "thapa"
- g. true + true
- h. true + false
- i. false + true

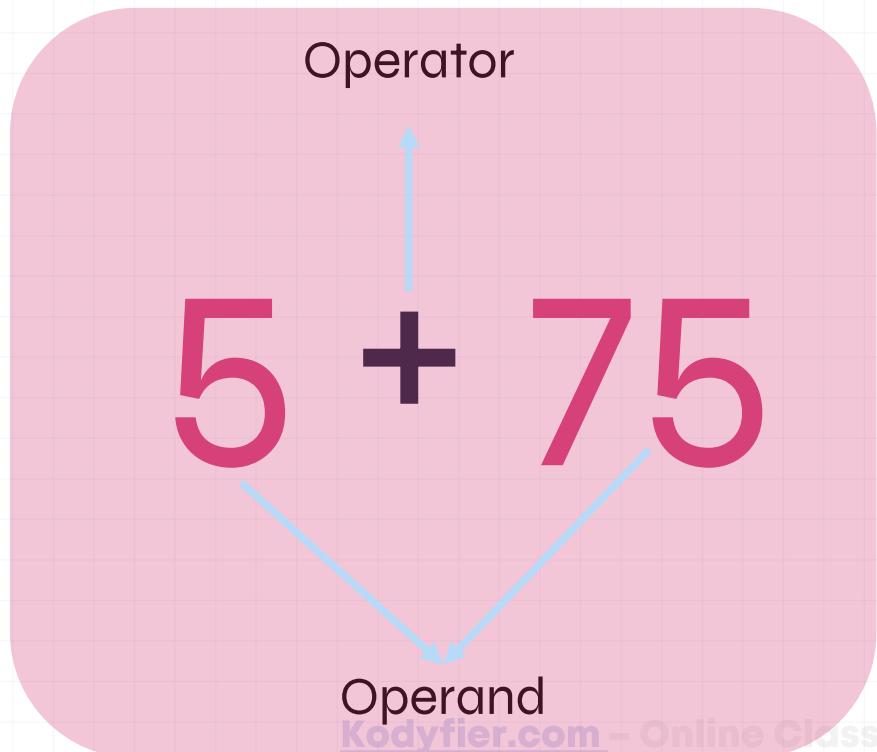
JAVASCRIPT

# \* Expressions & operators

[Kodyfier.com](http://Kodyfier.com) – Online Classes

**THAPA TECHNICAL**

# Expression



[Kodyfier.com](https://Kodyfier.com) - Online Classes

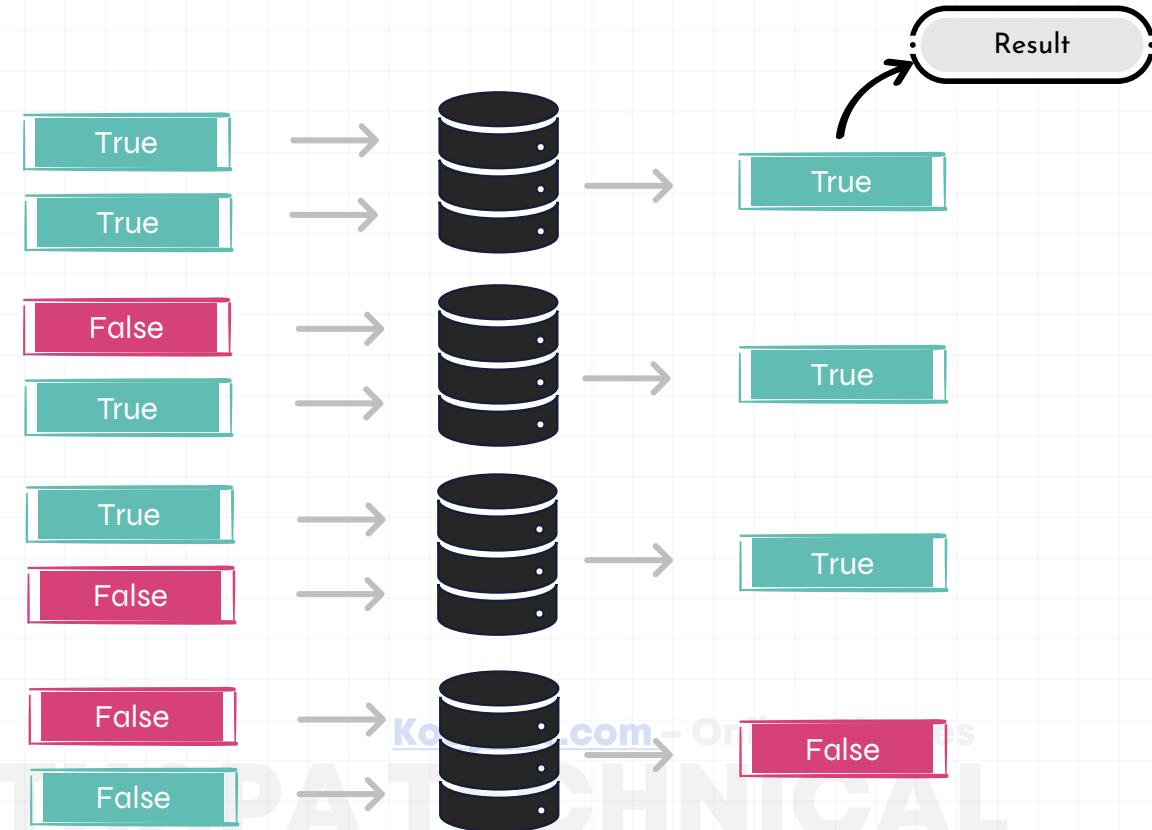
THAPA TECHNICAL

Subscribe: ThapaTechnical

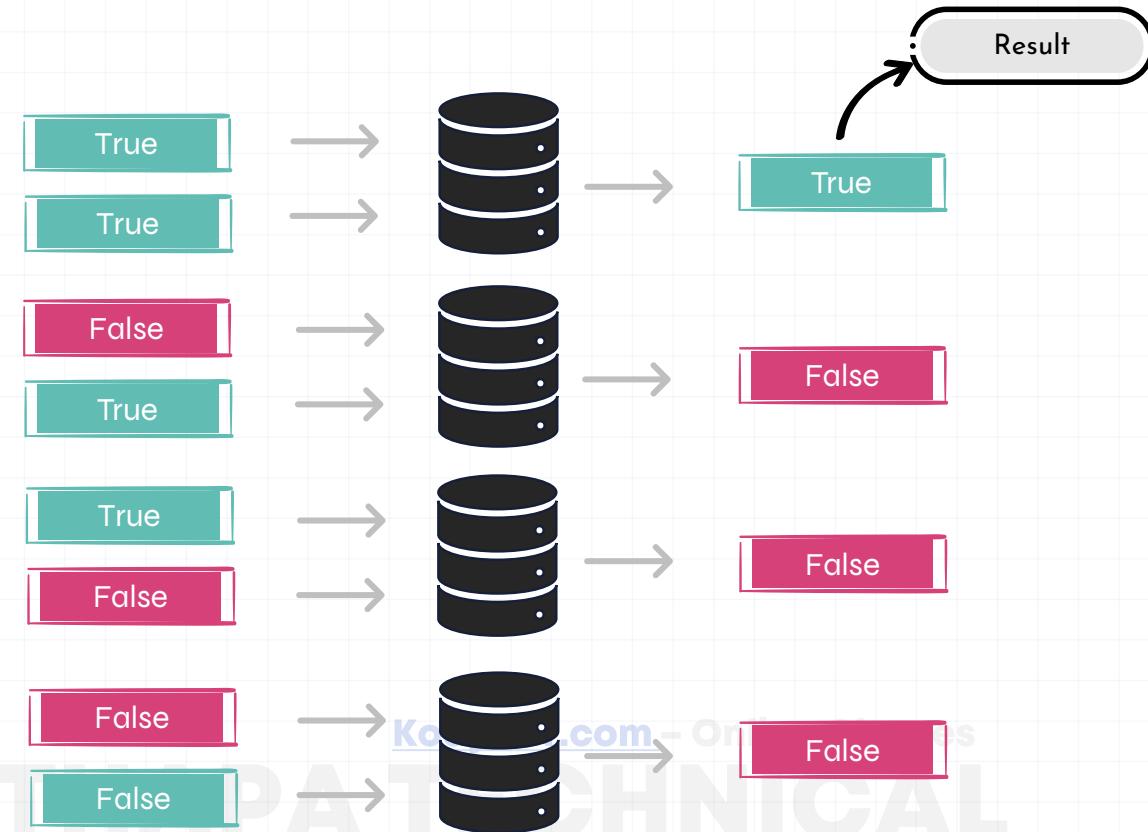
# Types of Operators



# How Logical OR Operator Works?



# How Logical AND Operator Works?



## Ternary Operator

Syntax:

condition ? expressionIfTrue : expressionIfFalse;

We get the output, If  
condition is true

We get the output, If  
condition is false



[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# Wait!!!



# Interview Questions

I want you to understand it thoroughly.



[kodyfier.com](https://kodyfier.com) - Online Cr

## THAPA TECHNICAL

Subscribe: ThapaTechnical

console.log("5" - 3);

2 (Type Coercion)

console.log(2 < 12 < 5);

True (Exp. evaluates from  
Left to Right)

console.log("20" + 10 + 10);

'201010' (same as 2nd)



# JAVASCRIPT Control Statement & THAPA TECHNICAL loops

[Kodyfier.com](http://Kodyfier.com) – Online Classes

# Control Statements & Loops

1 If.. Else Statement

2 Switch Statement

3 While Loop

4 Do While Loop

5 For Loop

6 For In / For Of Loop  
(Later in Arrays)

[Kodyfier.com](http://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# Syntax – If Else

```
if (condition) {  
    // Code to be executed if the condition is true  
} else {  
    // Code to be executed if the condition is false  
}
```

# Example – If Statement

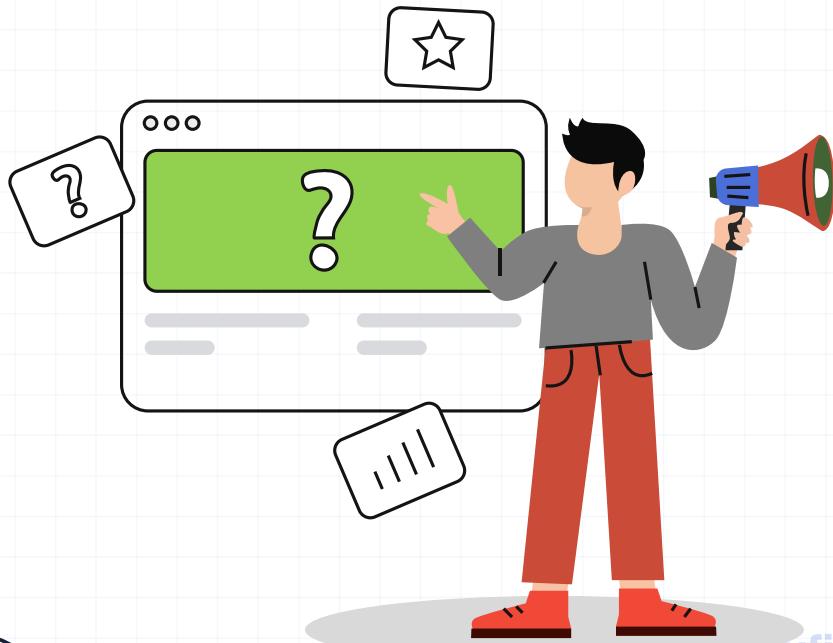
```
var temp = 40;
```

```
if (temp > 30) {  
    console.log("Let's go to Beach 🏊♂️")  
} else {  
    console.log("Watch TV at Home 😊")  
}
```



# If Else Statement

- 1: Write a program to check if a number is even or odd.
- 2: Write a program to check if a number is prime.
- 3: Write a program to check if a number is positive, negative, or zero.



# Challenge Time?

Are you ready for the challenge.



THAPA TECHNICAL

[Subscribe: ThapaTechnical](http://ThapaTechnical.com)

Challenge No:

# Switch Statement

Q: Write a JavaScript **switch statement** that takes a variable `areaOfShapes` representing different shapes, and based on its value, calculates and logs the area of the corresponding shape. Consider three shapes: '`Rectangle`,' '`Circle`,' and '`Square`.' For '`Rectangle`,' use variables `a` and `b` as the sides; for '`Circle`,' use a variable `r` as the radius; and for '`Square`,' use variable `a` as the side length. If the provided shape is not recognized, log a message saying, '`Sorry the shape is not available.`' Test your switch statement with `areaOfShapes` set to '`Square`' and sides `a` and `b` set to `5` and `10`, respectively. Ensure that the correct area (`25` in this case) is logged to the console.

# Syntax - While Loop

```
while (condition) {  
    // Code to be executed as long as the  
    condition is true  
}
```

[Kodyfier.com](http://Kodyfier.com) - Online Classes

THAPA TECHNICAL

[Subscribe: ThapaTechnical](#)

# Syntax – Do-While Loop

```
do {  
    // Code to be executed at least once  
} while (condition);
```

[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

[Subscribe: ThapaTechnical](https://ThapaTechnical.com)

# Syntax - For Loop

```
var num = 10;           num < 10           num++  
↑                 ↑                 ↑  
for(initializer; condition; iteration)  
{  
    // Code to be executed  
}
```

# Syntax – While Loop

```
let i=1;  
while (i<=10) {  
    console.log(i);  
    i++;  
}
```

Initialization → *i=1*

Condition → *(i<=10)*

Iteration → *i++*

[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# Syntax – Do-While Loop

```
let i=1;
```

Initialization

```
do{
```

```
    console.log(i);
```

Iteration

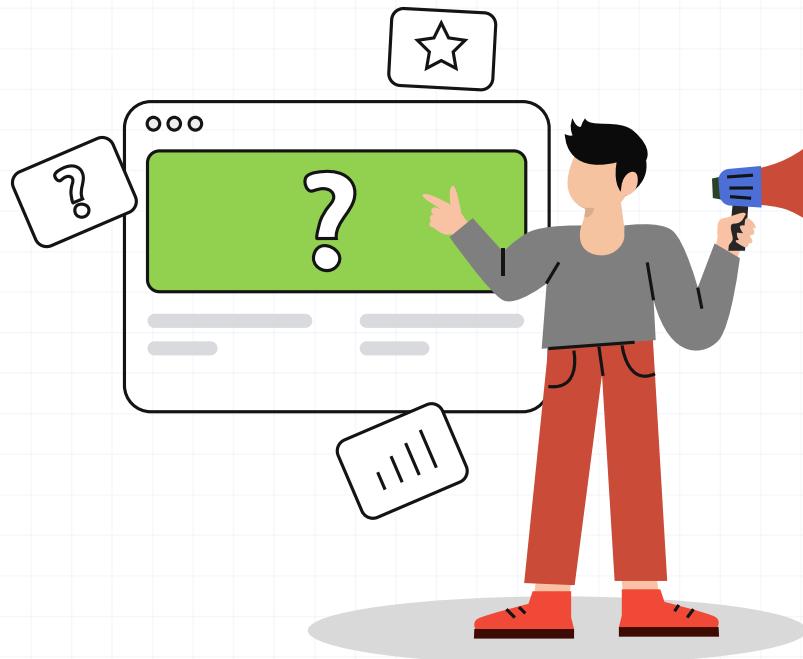
```
i++;
```

Condition

```
} while (i<=10)
```

# Syntax – For Loop

```
initializer    condition    iteration  
↑             ↑             ↑  
for (let i=1; i<=10; i++){  
    console.log(i);  
}
```



# Challenge Time?

Are you ready for the challenge.



[Kodyfier.com](http://Kodyfier.com) - Online Class

THAPA TECHNICAL

Subscribe: ThapaTechnical

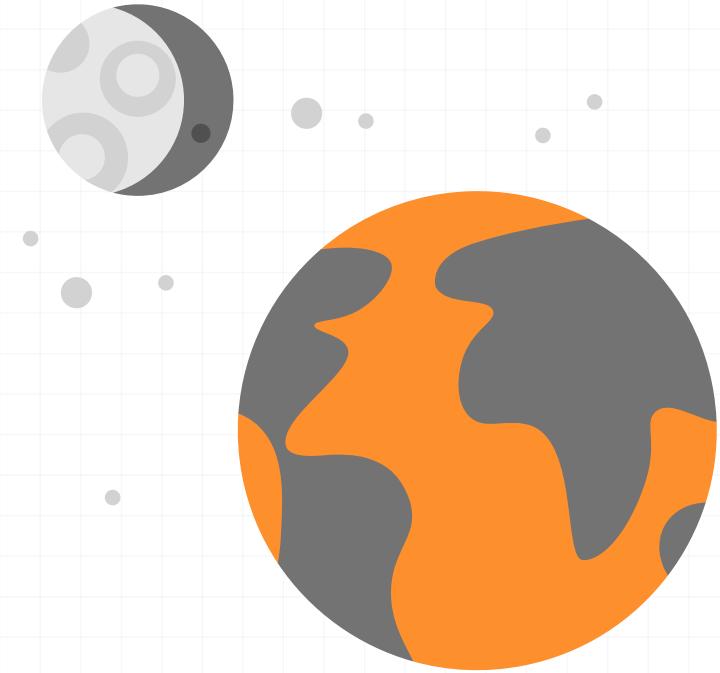
# For Loop

Program To check if a year is  
a **leap year**, 

If a year is **divisible by 4** and  
not **divisible by 100**, or

If a year is **divisible by 400**,  
then it is a leap year.

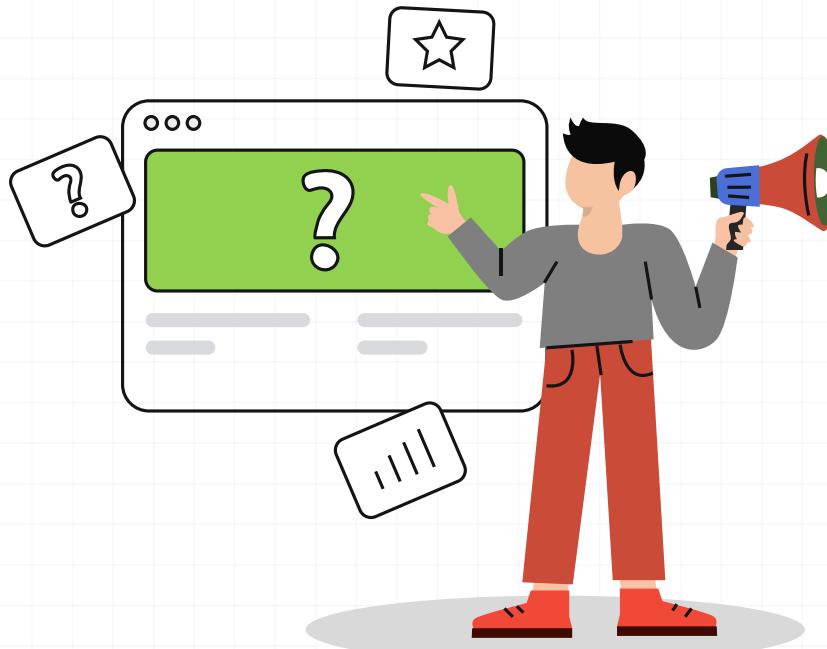
Otherwise, it is not a leap year.



[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical



# Challenge Time?

Are you ready for the challenge.



[Kodyfier.com](https://Kodyfier.com) - Online Class

THAPA TECHNICAL

Subscribe: ThapaTechnical

# For Loop

	J=1	J=2	J=3	J=4	J=5
I=1	*				
I=2	*	*			
I=3	*	*	*		
I=4	*	*	*	*	
I=5	*	*	*	*	*

logic    I == J    =>    print \*

THAPA TECHNICAL

Subscribe: ThapaTechnical

# For Loop

	J=1	J=2	J=3	J=4	J=5
I=1	*				
I=2					
I=3					
I=4					
I=5					

logic    I == J    =>    print \*

THAPA TECHNICAL

Subscribe: ThapaTechnical

# For Loop

	J=1	J=2	J=3	J=4	J=5
I=1	*				
I=2	*	*			
I=3					
I=4					
I=5					

logic    I == J    =>    print \*

THAPA TECHNICAL

Subscribe: ThapaTechnical

# For Loop

	J=1	J=2	J=3	J=4	J=5
I=1	*				
I=2	*	*			
I=3	*	*	*		
I=4					
I=5					

logic    I == J    =>    print \*

THAPA TECHNICAL

Subscribe: ThapaTechnical

# For Loop

	J=1	J=2	J=3	J=4	J=5
I=1	*				
I=2	*	*			
I=3	*	*	*		
I=4	*	*	*	*	
I=5					

logic    I == J    =>    print \*

THAPA TECHNICAL

Subscribe: ThapaTechnical

# For Loop

	J=1	J=2	J=3	J=4	J=5
I=1	*				
I=2	*	*			
I=3	*	*	*		
I=4	*	*	*	*	
I=5	*	*	*	*	*

logic    I == J    =>    print \*

THAPA TECHNICAL

Subscribe: ThapaTechnical

# JAVASCRIPT \* Functions

[Kodyfier.com](http://Kodyfier.com) – Online Classes

**THAPA TECHNICAL**

# JavaScript Function

In JavaScript, a **function** is a block of reusable code that performs a specific task or set of tasks. Functions are used to **organize code into modular** and manageable pieces, promote **code reuse**, and make programs **more readable**.

```
function functionName(parameters) {  
    // code to be executed  
    return result; // optional return statement  
}
```

[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# JavaScript Function

In JavaScript, a **function** is a block of reusable code that performs a specific task or set of tasks. Functions are used to **organize code into modular** and manageable pieces, promote **code reuse**, and make programs **more readable**.

```
function functionName(parameters) {  
    // code to be executed  
    return result; // optional return statement  
}
```

[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# What we will cover

1 Function Declaration

4 Function Argument

7 Return Keyword

2 Function Invocation

5 Function expressions

8 IIFE (Immediately Invoked Function expression)

3 Function Parameter

6 Anonymous Function

9 More we will see in advanced

# Syntax – Function Declaration

```
function  
Keyword      function Name  
function greet() {  
  
    console.log("Welcome to Thapa Technical JS  
        Course ");  
  
}  
          }  
          function Body
```

[Kodyfier.com](http://Kodyfier.com) - Online Classes

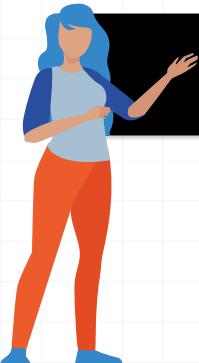
THAPA TECHNICAL

Subscribe: ThapaTechnical

# Syntax – Function Invocation

```
function greet() {  
    console.log(" Welcome to Thapa Technical JS Course ");  
}  
  
greet()
```

We need to call the function name



Welcome to Thapa Technical JS Course

[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

# Syntax – Function Parameter

```
function greet(parameter1) {  
    console.log(" Best JS Course ");  
}
```

We need to add values here (parameter)

We need to call the function name

```
greet()
```

# Syntax – Function Parameter

```
function greet(parameter1, parameter2) {  
    console.log(" Best JS Course ");  
}
```

We need to call the function name  
*greet()*

# Syntax – Function Parameter

```
function greet(parameter1, parameter2, ...)  
{  
    console.log(" Best JS Course ");  
}
```

We need to call the function name  
*greet()*

# Syntax – Function Argument

```
function greet(parameter1, parameter2, ...)  
{  
    console.log(" Best JS Course ");  
}
```

 We need to call the function name

**greet(argument1)**

# Syntax – Function Argument

```
function greet(parameter1, parameter2, ...)  
{  
    console.log(" Best JS Course ");  
}
```

 We need to call the function name

**greet(argument1, argument2)**

# Syntax – Function Argument

```
function greet(parameter1, parameter2, ...)  
{  
    console.log(" Best JS Course ");  
}
```

 We need to call the function name

```
greet(argument1, argument2, ...)
```

## Interview Questions – Function

- 1: **Reverse a String:** Write a function to reverse a given string **without** using built-in reverse methods.
  
- 2: **Palindrome Check:** Create a function to determine if a given string is a **palindrome** (reads the same backward as forward).
  
- 3: **Calculator Function:** Write a JavaScript function calculator that takes two numbers and an operator as parameters and returns the result of the operation. The function should support addition, subtraction, multiplication, and division.

World Best JavaScript

JAVASCRIPT

\* ECMAScript

[Kodyfier.com](https://Kodyfier.com) – Online Classes

**THAPA TECHNICAL**

# EcmaScript Timeline



## JAVASCRIPT

JavaScript was created

1997

## ECMA INTERNATIONAL

It was then submitted to [ECMA International](#) for standardization, which resulted in EcmaScript

2015

## ES6 / EcmaScript 2015

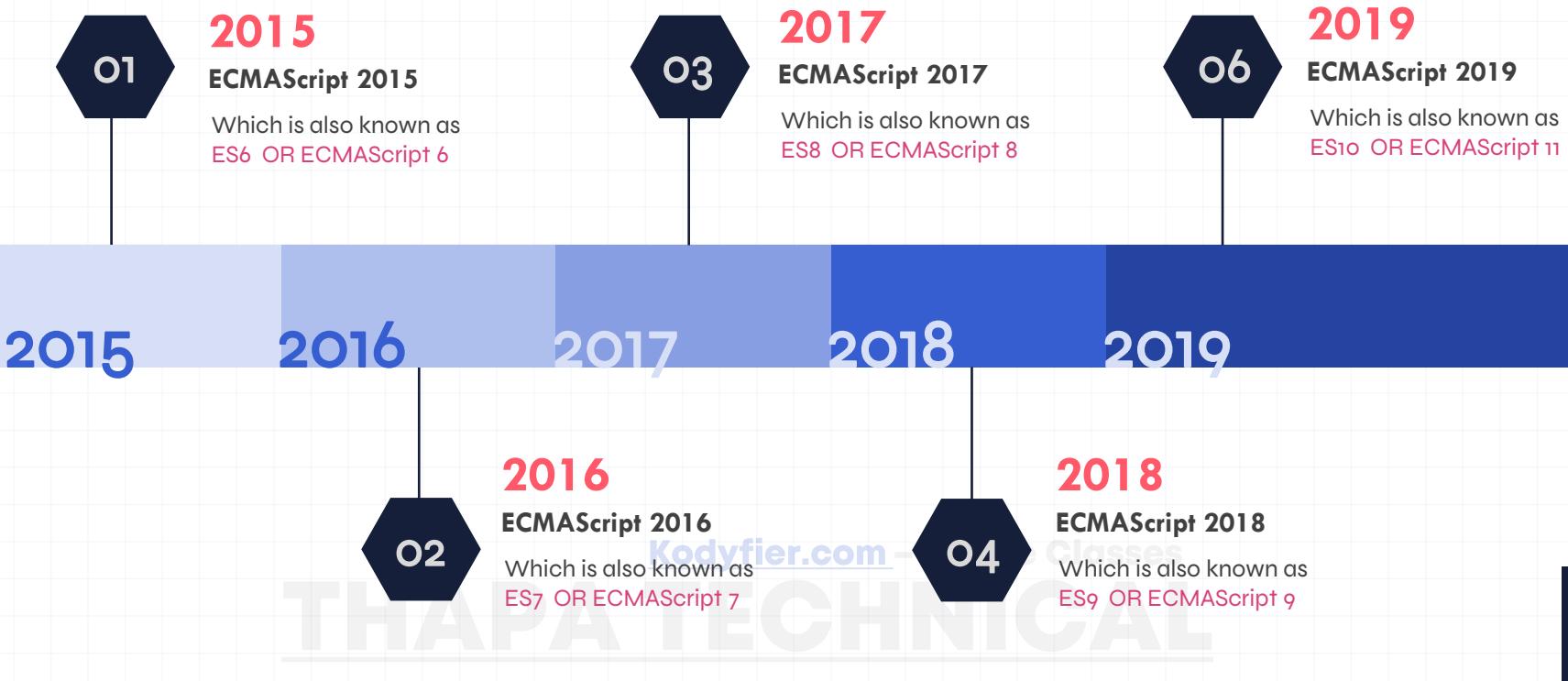
Decided to switch to [annual](#) releases of EcmaScript

[Kodyfier.com](#) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# Timeline



# Timeline



2020

2021

2022

2023

2024

01

# ECMAScript 2015 / ES6



LET AND CONST



TEMPLATE STRINGS



DEFAULT ARGUMENTS



ARROW FUNCTION



DESTRUCTURING



OBJECT PROPERTIES



REST OPERATORS



SPREAD OPERATORS

[Kodyfier.com](http://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

## Interview Questions – Function

- 1: **Reverse a String:** Write a function to reverse a given string **without** using built-in reverse methods.
  
- 2: **Palindrome Check:** Create a function to determine if a given string is a **palindrome** (reads the same backward as forward).
  
- 3: **Calculator Function:** Write a JavaScript function calculator that takes two numbers and an operator as parameters and returns the result of the operation. The function should support addition, subtraction, multiplication, and division.

World Best JavaScript

# JAVASCRIPT \* ARRAYS

[Kodyfier.com](https://Kodyfier.com) – Online Classes

**THAPA TECHNICAL**

# JavaScript Array



Ram



Hari



Sita



Bishal



Gita

Imagine you want to store collection of people names.

[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

const person1 = Ram 

const person2 = Hari 

const person3 = Sita 

const person4 = Bishal 

const person5 = Gita 

You will think of doing something like this.

Kodyfier.com - Online Classes

THAPA TECHNICAL

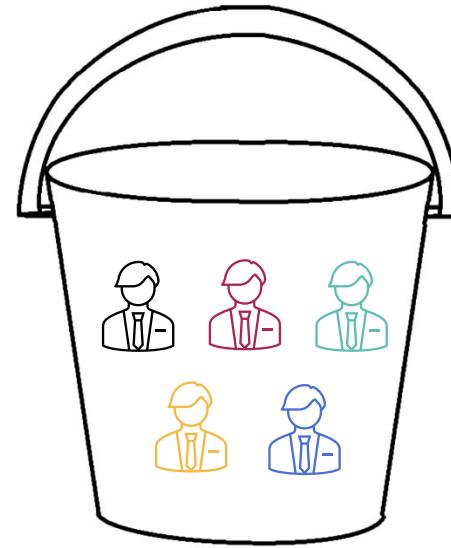
```
const person1 = Ram
```

```
const person2 = Hari
```

```
const person3 = Sita
```

```
const person4 = Bishal
```

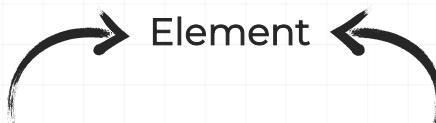
```
const person5 = Gita
```



What if we could store all of these into a bucket?

Kodyfier.com - Online Classes

THAPA TECHNICAL



```
const persons = [ "Ram", "Hari", "Sita", "Bishal", "Gita" ];
```

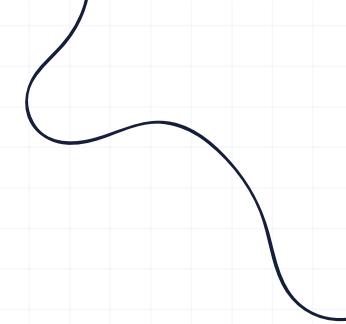


That's what array is for.

JavaScript array is an object that represents a collection of similar type of elements.

Each value(name) will be called as an Element.

THAPA TECHNICAL



Index Number

0

1

2

3

4

```
const persons = [ "Ram", "Hari", "Sita", "Bishal", "Gita" ];
```



In arrays, each element is represented by an index which starts with zero.

[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Index Number

0

1

2

3

4

```
const persons = [ "Ram", "Hari", "Sita", "Bishal", "Gita" ];
```



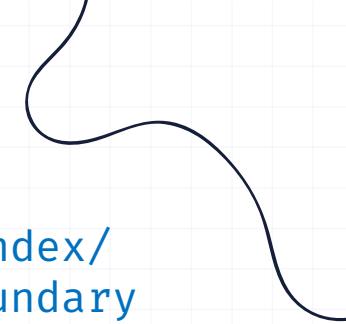
```
persons[0]; // Ram
```

```
persons[1]; // Hari
```

And we can access each element by using indexes

[Kodifyer.com](https://www.kodifyer.com) - Online Classes

THAPA TECHNICAL



Lower Index/  
Lower Boundary

```
const persons = [ "Ram", "Hari", "Sita", "Bishal", "Gita" ];
```



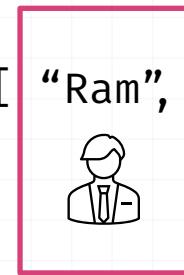
Upper Index/  
Upper Boundary

First element or head: Refers to the element at index 0.

Last element or tail: Refers to the element at the last index,  
which can be obtained using `array.length - 1`.

Lower Index/  
Lower Boundary

```
const persons = [ "Ram", "Hari", "Sita", "Bishal", "Gita" ];
```



Upper Index/  
Upper Boundary

```
persons[-1] // ERROR
```

```
persons.at(-1) // Gita
```

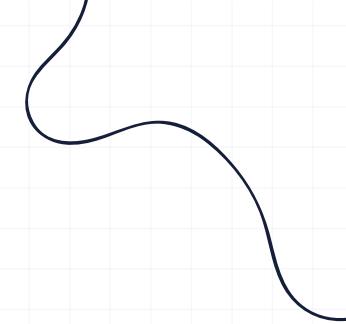
```
persons.at(-2) // Bishal
```

ECMAScript 2022 also introduces new `.at()` method in arrays  
which helps to index from last elements too easily.

THAPATECHNICAL

## What we will cover

- \* Creating Arrays / Accessing Elements / Modifying Elements
- \* Array Traversal / Iterating Over Arrays
- \* How to Insert, Add, Replace and Delete Elements in Array(CRUD)
- \* Searching in an Array
- \* Filter in an Array
- \* How to Sort and Compare an Array
- \* Very Very Important Array Methods



Index Number	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
--------------	----------	----------	----------	----------	----------

```
const persons = [ "Ram", "Hari", "Sita", "Bishal", "Gita" ];
```



In arrays, each element is represented by an index which starts with zero.

[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

# Push()

**Push Method:** The Push Method that adds one or more elements to the end of an array.

Syntax: `push(Element)`

```
persons.push('Gita')
```

Index Number	=	0	1	2	3
--------------	---	---	---	---	---

```
const persons = ["Ram", "Hari", "Sita", "Bishal"];
```



[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# Push()

**Push Method:** The Push Method that adds one or more elements to the end of an array.

Syntax: `push(Element)`

Index Number

0

1

2

3

4

```
const persons = ["Ram", "Hari", "Sita", "Bishal", "Gita"];
```



Gita added at the end of the array

`persons.push('Gita')`

[Kodyfier.com](https://kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# Pop()

Pop Method: Method that removes the last element from an array.

Syntax: `pop(Element)`

Index Number

0

1

2

3

4

```
const persons = ["Ram", "Hari", "Sita", "Bishal", "Gita"];
```



Gita will be removed  
`persons.pop('Gita')`

[Kodyfier.com](http://Kodyfier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# Pop()

**Pop Method:** Method that removes the last element from an array.

Syntax: `pop(Element)`

Index Number = 0 1 2 3

```
const persons = [ "Ram", "Hari", "Sita", "Bishal" ];
```



[Kodyfier.com](https://Kodyfier.com) - Online Classes

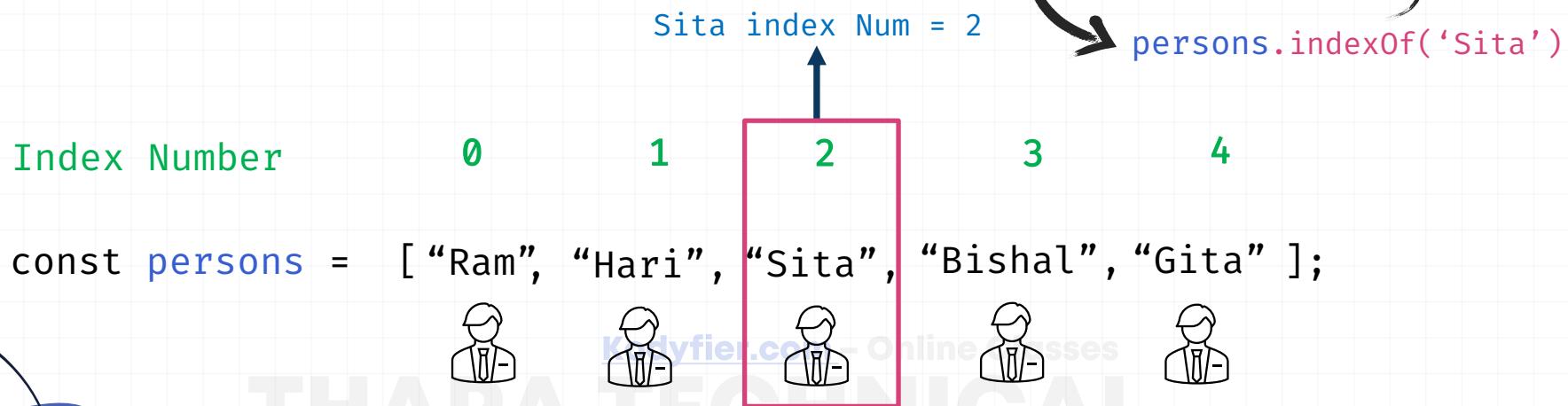
THAPA TECHNICAL

Subscribe: ThapaTechnical

# indexOf()

**indexOf Method:** The `indexOf` method returns the first index at which a given element can be found in the array, or `-1` if it is not present.

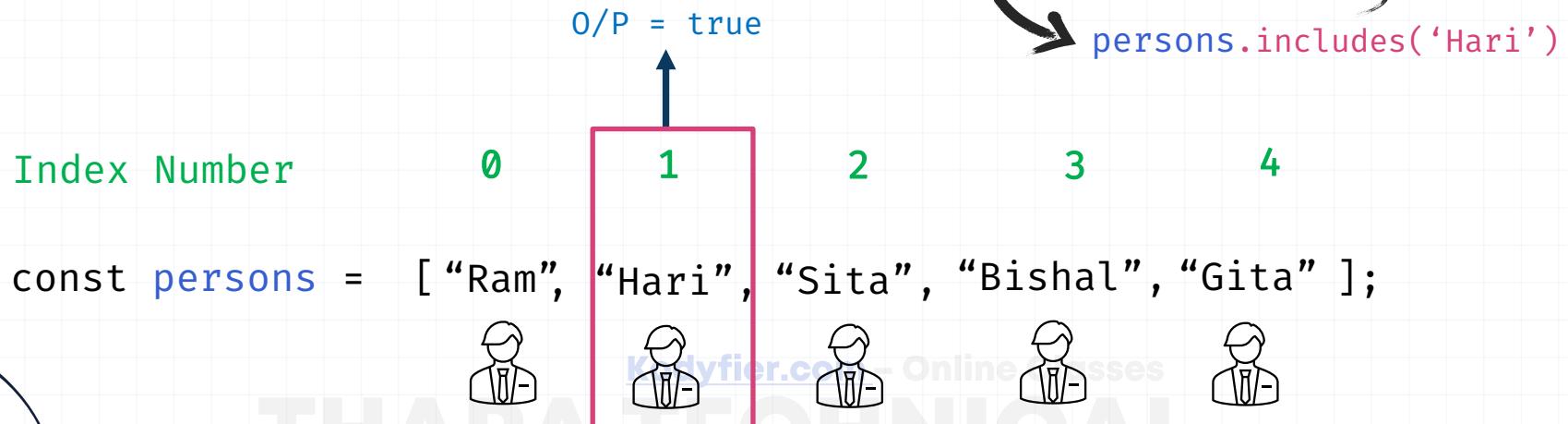
Syntax: `indexOf(searchElement, fromIndex)`



# Includes()

**Includes Method:** The includes method checks whether an array includes a certain element, returning true or false.

Syntax: `Includes(searchElement, fromIndex)`

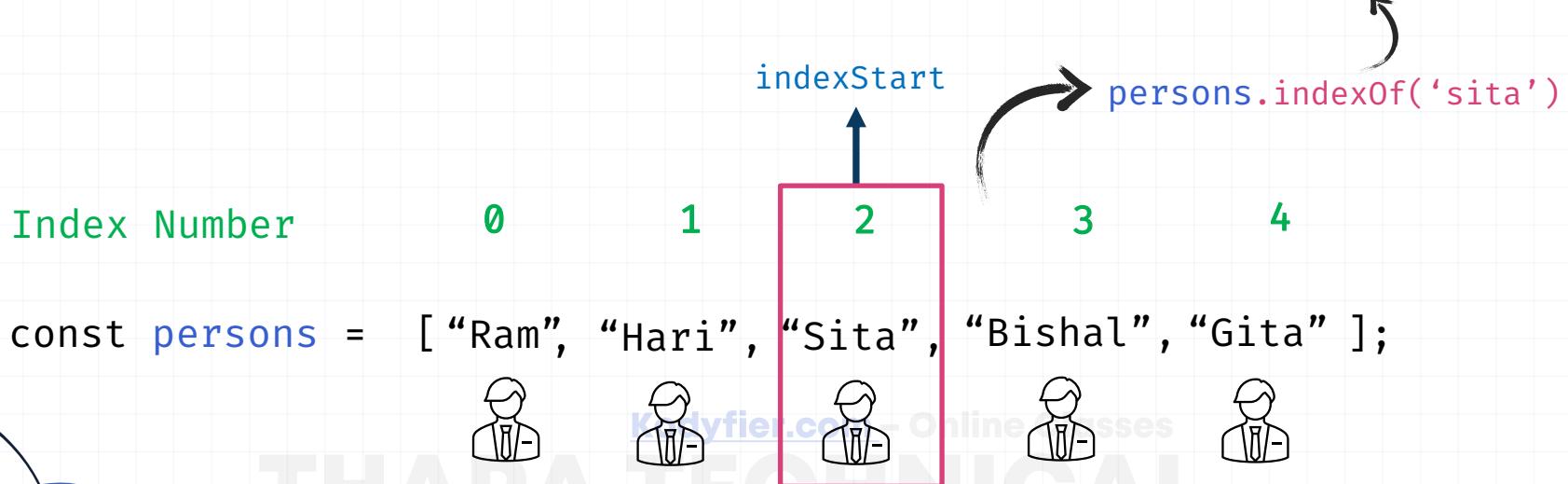


# indexOf()

**indexOf Method:** The `indexOf` method returns the first index at which a given element can be found in the array, or `-1` if it is not present.

Syntax: `indexOf(searchElement, fromIndex)`

Look only the 1<sup>st</sup> letter

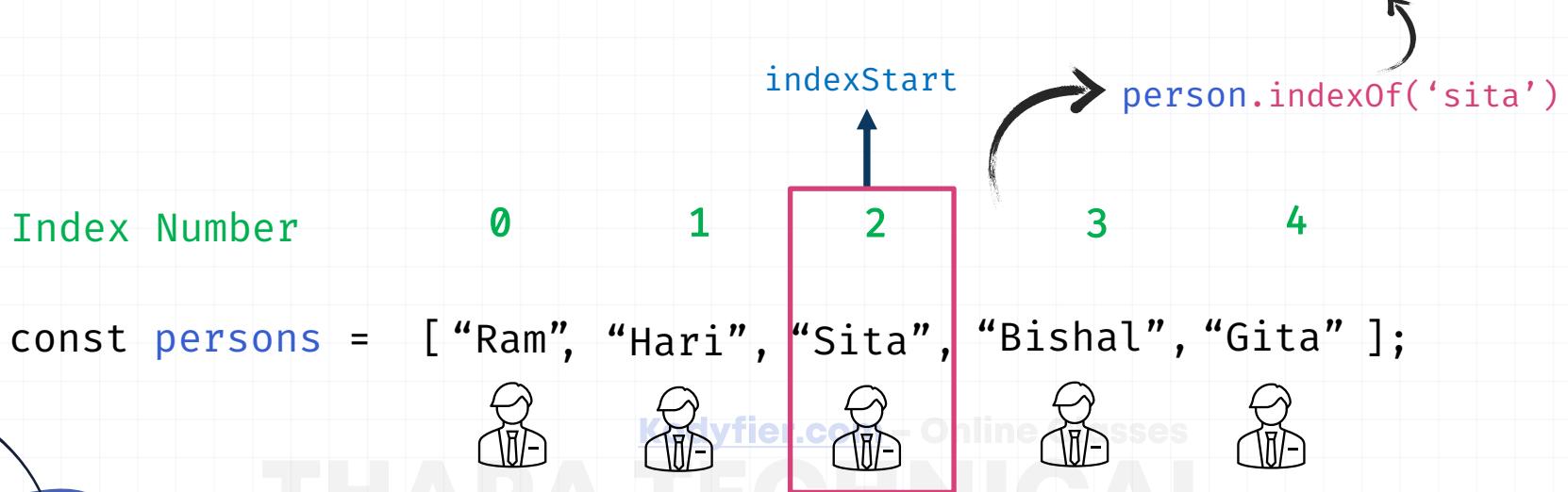


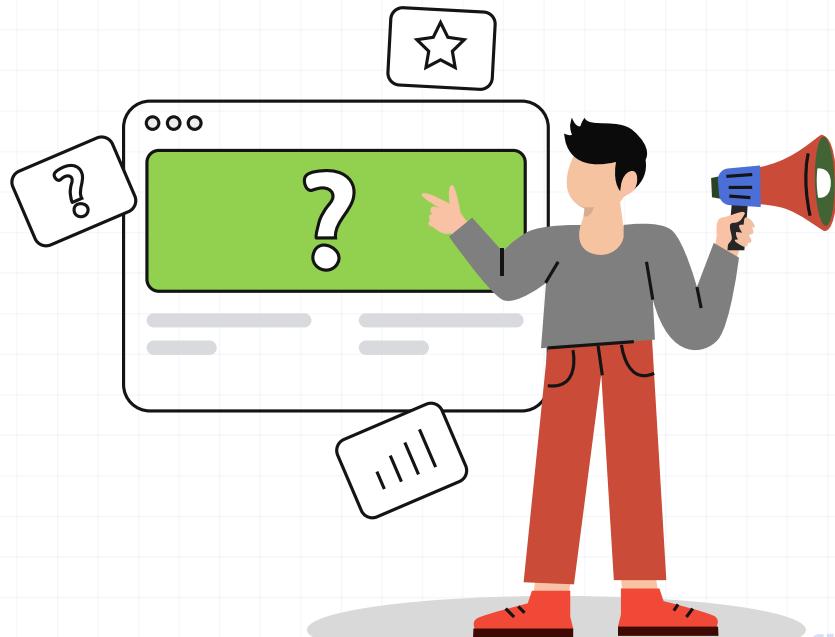
# indexOf()

**indexOf Method:** The `indexOf` method returns the first index at which a given element can be found in the array, or `-1` if it is not present.

Syntax: `indexOf(searchElement, fromIndex)`

Look only the 1<sup>st</sup> letter





# Interview Question

forEach vs Map in JavaScript?



THAPA TECHNICAL

Subscribe: ThapaTechnical

# Syntax – forEach

```
array.forEach(function  
callback(currentValue, index, array) {  
    // Your logic here  
}, thisValue);
```

# Here's a breakdown of each part:

`array`: The array on which the `foreach` method is called.

`callback`: A function that will be called once for each element in the array.

`currentValue`: The `current element` being processed in the array.

`index` (optional): The index of the `current element` being processed.

`array` (optional): The array `foreach` was called upon.

`thisValue` (optional): A value to use as `this` when executing the callback function.

# Syntax – forEach

```
array.forEach((currentValue, index, array)  
=> { // Your logic here }, thisValue);
```

[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

[Subscribe: ThapaTechnical](#)

# Syntax – Map()

```
array.map(function callback(currentValue,  
index, array) {  
    // Your logic here  
}, thisValue);
```

# Syntax – Map()

```
array.map((currentValue, index, array) =>  
{  
    // Your logic here  
}, thisValue);
```

# Here's a breakdown of each part:

`array`: The array on which the `map` method is called.

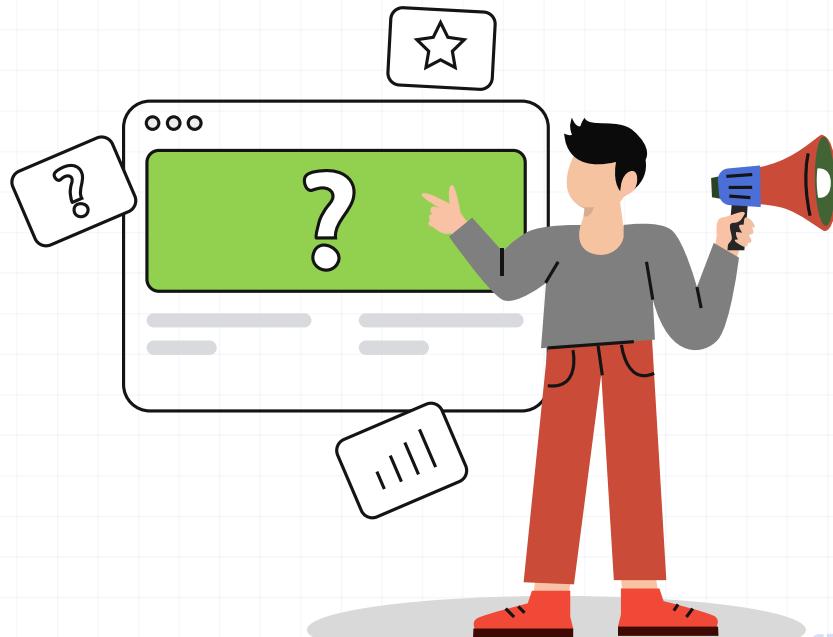
`callback`: A function that will be called once for each element in the array.

`currentValue`: The `current element` being processed in the array.

`index` (optional): The index of the `current element` being processed.

`array` (optional): The array `map` was called upon.

`thisValue` (optional): A value to use as `this` when executing the callback function.



# Interview Question

Let's test our knowledge.



THAPA TECHNICAL

Subscribe: ThapaTechnical

# Interview Questions – Array CRUD

- 1: Add Dec at the `end` of an array?
- 2: What is the `return` value of splice method?
- 3: `Update` march to March (update)?
- 4: `Delete` June from an array?

```
const months = ['Jan', 'march', 'April', 'June', 'July'];
```

[Kodyfier.com](https://Kodyfier.com) - Online Classes

THAPA TECHNICAL

[Subscribe: ThapaTechnical](https://ThapaTechnical.com)

## Interview Questions – Array Filter

Q: Given an array of products where each product has a **name** and a **price**, write a function that uses the **filter method** to return an array containing only the products with a **price less than or equal to 500**.

```
const products = [  
    { name: "Laptop", price: 1200 },  
    { name: "Phone", price: 800 },  
    { name: "Tablet", price: 300 },  
    { name: "Smartwatch", price: 150 },  
];
```

## Interview Questions – Array Filter

1: Using the `map` method, write a function that takes an array of strings and returns a new array where each `string is capitalized`.

2: Using the `map` method, write a function that takes an array of numbers and returns a new array where each `number is squared`, but only if it's an even number.

3: Using the `map` method, write a function that takes an array of names and returns a new array where each name is prefixed with "Mr".

[Kodifier.com](http://Kodifier.com) - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

## Interview Questions – Array Reduce

Write a JavaScript function that **calculates the total price** of items in a shopping cart. The function should take an array of item prices as input and **return the total price**.

JAVASCRIPT  
\* **STRINGS**

[Kodyfier.com](http://Kodyfier.com) – Online Classes

**THAPA TECHNICAL**

# What we will cover

\* String & its properties

\* String Search Methods

\* Extracting String Characters

\* Other Useful Methods

\* Escape Character

\* Extracting String Parts

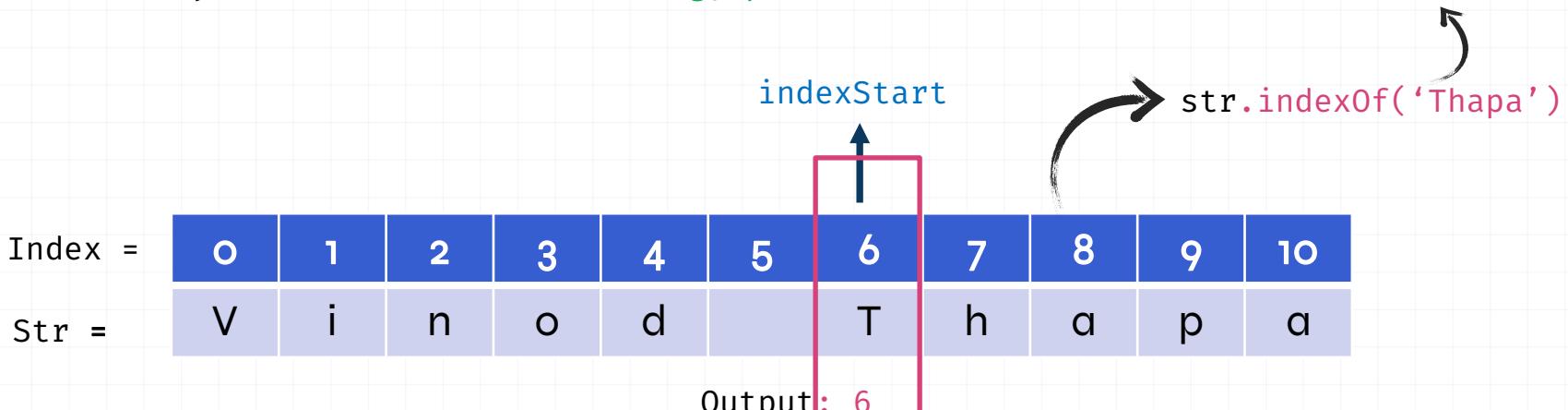
\* Replacing String Content

# indexOf()

The `indexOf()` method returns the index (position) of the first occurrence of a string in a string, or it returns -1 if the string is not found:

Syntax: `indexOf(searchString, position)`

Look only the 1<sup>st</sup> letter

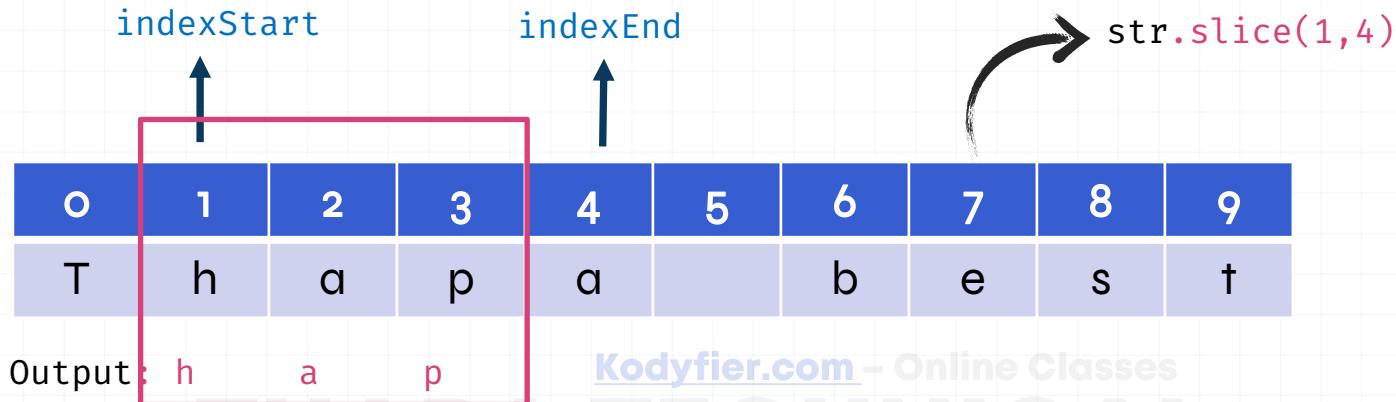


# slice()

`slice()` extracts a part of a string and returns the extracted part in a new string.

1: JavaScript counts **positions from zero**.

2: `slice()` extracts up to but **not including indexEnd**.

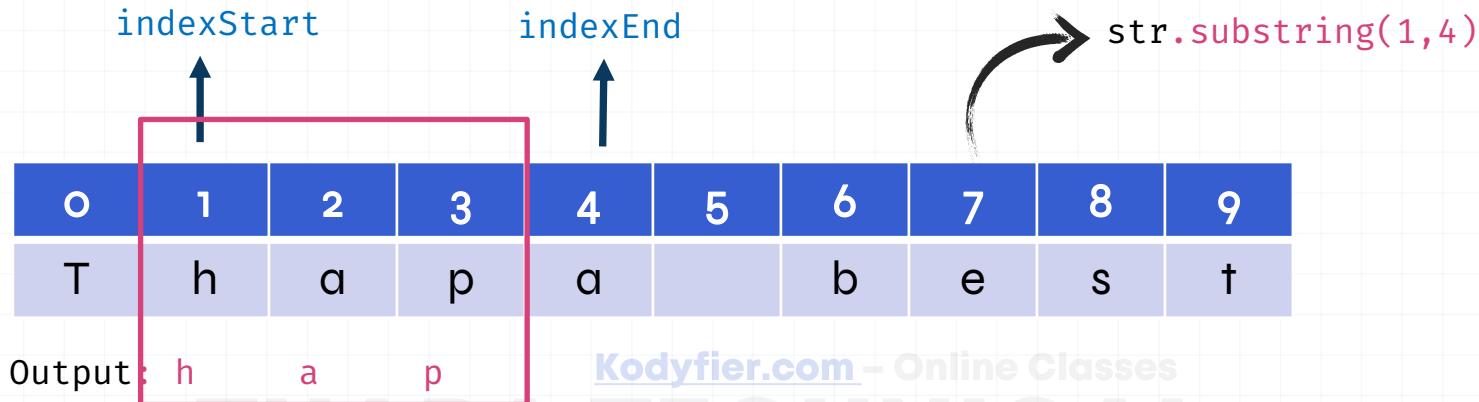


# substring()

`substring()` extracts a part of a string and returns the extracted part in a new string.

1: JavaScript counts **positions from zero**.

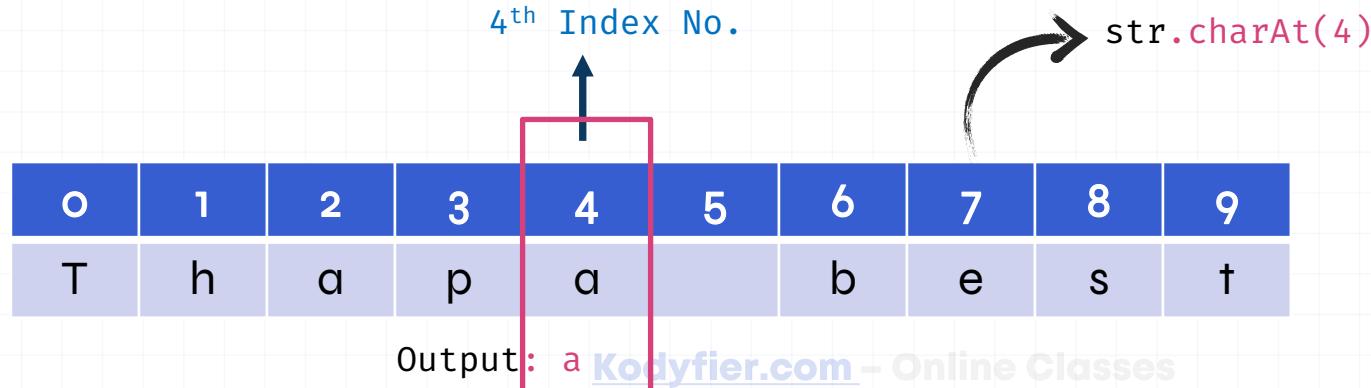
2: `substring()` extracts up to but **not including indexEnd**.



# charAt()

The `charAt()` method returns the character at a specified index (position) in a string

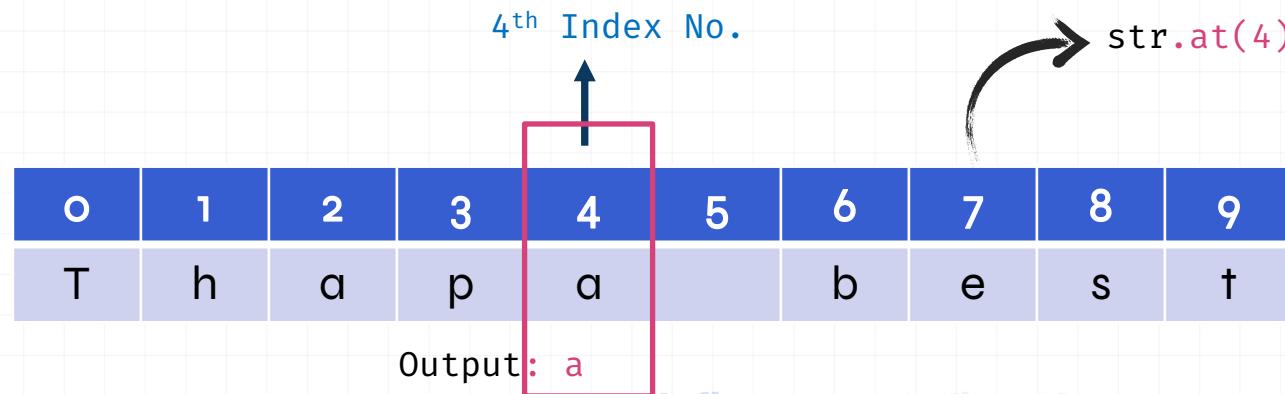
1: JavaScript counts **positions from zero**.



# at()

The `at()` method returns the character at a specified index (position) in a string

- 1: It allows the use of negative indexes while `charAt()` do not.



# at()

The `at()` method returns the character at a specified index (position) in a string

- 1: It allows the use of negative indexes while `charAt()` do not.

o	1	2	3	4	5	6	7	8	9
T	h	a	p	a		b	e	s	t
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

4<sup>th</sup> Index No. → `str.at(-4)`

KodOutput: b - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical

# Interview Questions – Strings

- 1: Write a JavaScript function that prints the letters 'a' through 'z' in the console. You should use a loop to iterate through the letters and print each one on a new line.
  
- 2: Write a function to count the number of vowels in a string?
  
- 3: Write a function to check if all the vowels presents in a string or not?

# Interview Questions – Strings

Write a JavaScript function `isPangram` that takes a string as input and returns true if the string is a pangram (`contains all letters of the alphabet`) and false otherwise. The function should be case-insensitive and ignore spaces.

## Constraints:

- 1: The input string will consist of alphabetic characters and spaces.
- 2: The function should handle both uppercase and lowercase letters.
- 3: Consider the English alphabet with 26 letters.

# JAVASCRIPT

# \* MATH OBJECT

[Kodyfier.com](http://Kodyfier.com) – Online Classes

**THAPA TECHNICAL**

# Difference Between Round, Floor & Ceil

## Math.round()

Rounds to the nearest integer.

Ex:

```
console.log(Math.round(4.5));  
// Output: 5  
console.log(Math.round(4.1));  
// Output: 4
```

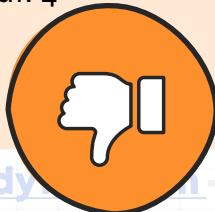


## Math.floor()

Always rounds down to the nearest integer.

Ex:

```
console.log(Math.floor(4.9));  
// Output: 4  
console.log(Math.floor(4.1));  
// Output: 4
```



## Math.ceil()

Always rounds up to the nearest integer.

Ex:

```
console.log(Math.ceil(4.2));  
// Output: 5  
console.log(Math.ceil(4.9));  
// Output: 5
```



# Interview Questions – Strings & Functions

1: Write a JavaScript function that prints the letters 'a' through 'z' in the console. You should use a loop to iterate through the letters and print each one on a new line.

2: Write a JavaScript function `isPangram` that takes a string as input and returns true if the string is a pangram (contains all letters of the alphabet) and false otherwise. The function should be case-insensitive and ignore spaces.

**Constraints:**

- 1: The input string will consist of alphabetic characters and spaces.
- 2: The function should handle both uppercase and lowercase letters.
- 3: Consider the English alphabet with 26 letters.

JAVASCRIPT

\* **Window in JS  
DOM & BOM**

[Kodyfier.com](http://Kodyfier.com) – Online Classes

**THAPA TECHNICAL**

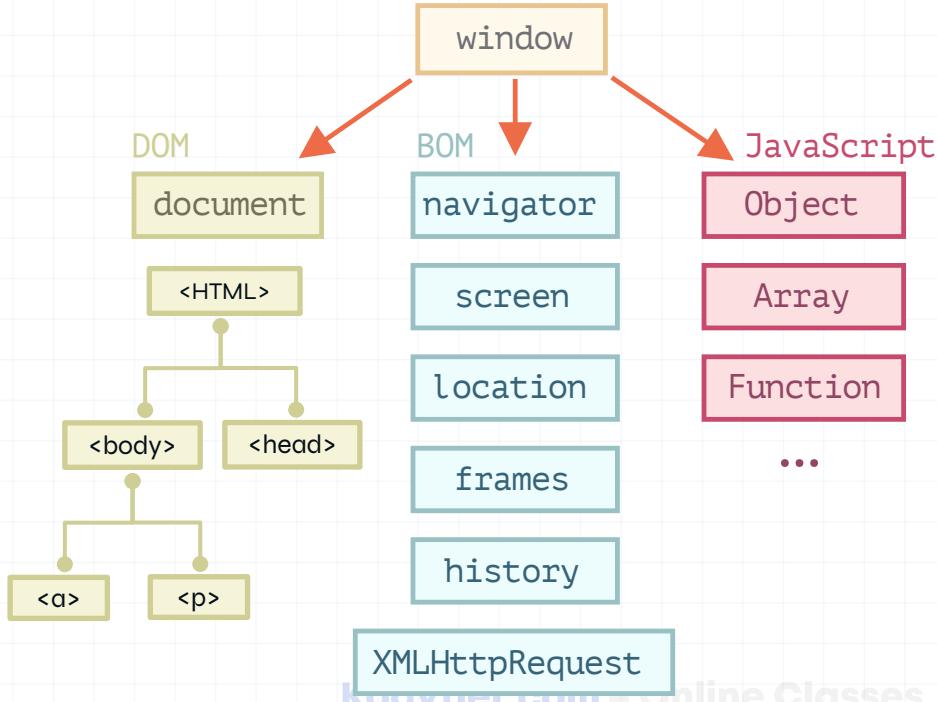
# Window

- 1: Window is the main container, or we can say the **global Object** and any operations related to entire browser window can be a part of window object.
- 2: All the members like objects, methods or properties. If they are the part of window object, then **we do not refer the window object.**
- 3: Window has methods, properties and object. ex `setTimeout()` or `setInterval()` are the methods, where as Document is the object of the Window and It also has a screen object with properties describing the physical display.

# Document

- 1: Whereas the **DOM** is the child of window object
- 2: Where in the DOM **we need to refer the document**, if we want to use the document object, methods or properties
- 3: Document is just the **object of the global object** that is Window, which deals with the document, the HTML elements themselves.

# Window Global Object



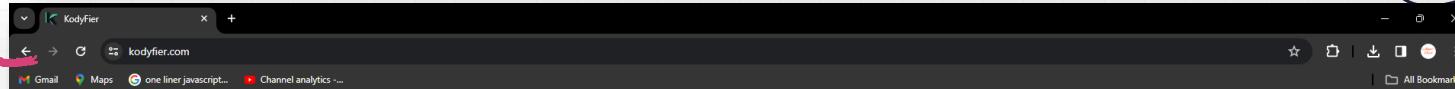
<https://javascript.info/dom-nodes#:text=An%20example%20of%20the%20DOM&text=Every%20tree%20node%20is%20an%3E%20are%20its%20children%C20etc.>

# Window Global Object

The screenshot shows a web browser window with the URL [kodyfier.com](http://kodyfier.com) in the address bar. The page content is as follows:

- KodyFier** logo in the top left.
- Home About Courses Contact** navigation menu in the top right.
- IT TRAINING INSTITUTE IN PUNE** main title with green feather icons on either side.
- A **Get Started** button with a green rounded rectangle and white text.
- Four service icons with labels: **Live Classes** (play button), **Modern Projects** (document icon), and **Industry Level** (location pin).
- A large circular graphic featuring a smiling man holding a book, set against a background of concentric circles.
- A descriptive text block about the institute's mission and offerings.

The BOM



The DOM

The screenshot shows the homepage of 'KodyFier IT TRAINING INSTITUTE IN PUNE'. The page features a green header with the logo 'KodyFier' and navigation links for 'Home', 'About', 'Courses ▾', and 'Contact'. The main title 'IT TRAINING INSTITUTE IN PUNE' is centered with decorative green leaf-like icons on either side. Below the title, a paragraph describes the institute as an ideal place for upgrading skills in the tech industry. To the right, there's a circular graphic with concentric circles containing a photo of a smiling man holding a book. On the left, there's a 'Get Started' button and a list of course offerings: 'Live Classes' (with a video camera icon), 'Modern Projects' (with a document icon), and 'Industry Level' (with a person icon). A green arrow points from the text 'The DOM' down towards the website content.

## Window Object:

The window object represents the global window in a browser.

Both the [Browser Object Model \(BOM\)](#) and the [Document Object Model \(DOM\)](#) are part of the window object.

## Window Object:

The window object represents the global window in a browser. Both the **Browser Object Model (BOM)** and the **Document Object Model (DOM)** are part of the window object.

## Browser Object Model (BOM):

The BOM represents the browser as an object and provides methods and properties for interacting with the browser itself (not directly related to the content of a web page).

Examples of BOM features include `window.navigator` for browser information, `window.location` for URL manipulation, and `window.alert` for displaying alerts.

## Window Object:

The window object represents the global window in a browser.

Both the **Browser Object Model (BOM)** and the **Document Object Model (DOM)** are part of the window object.

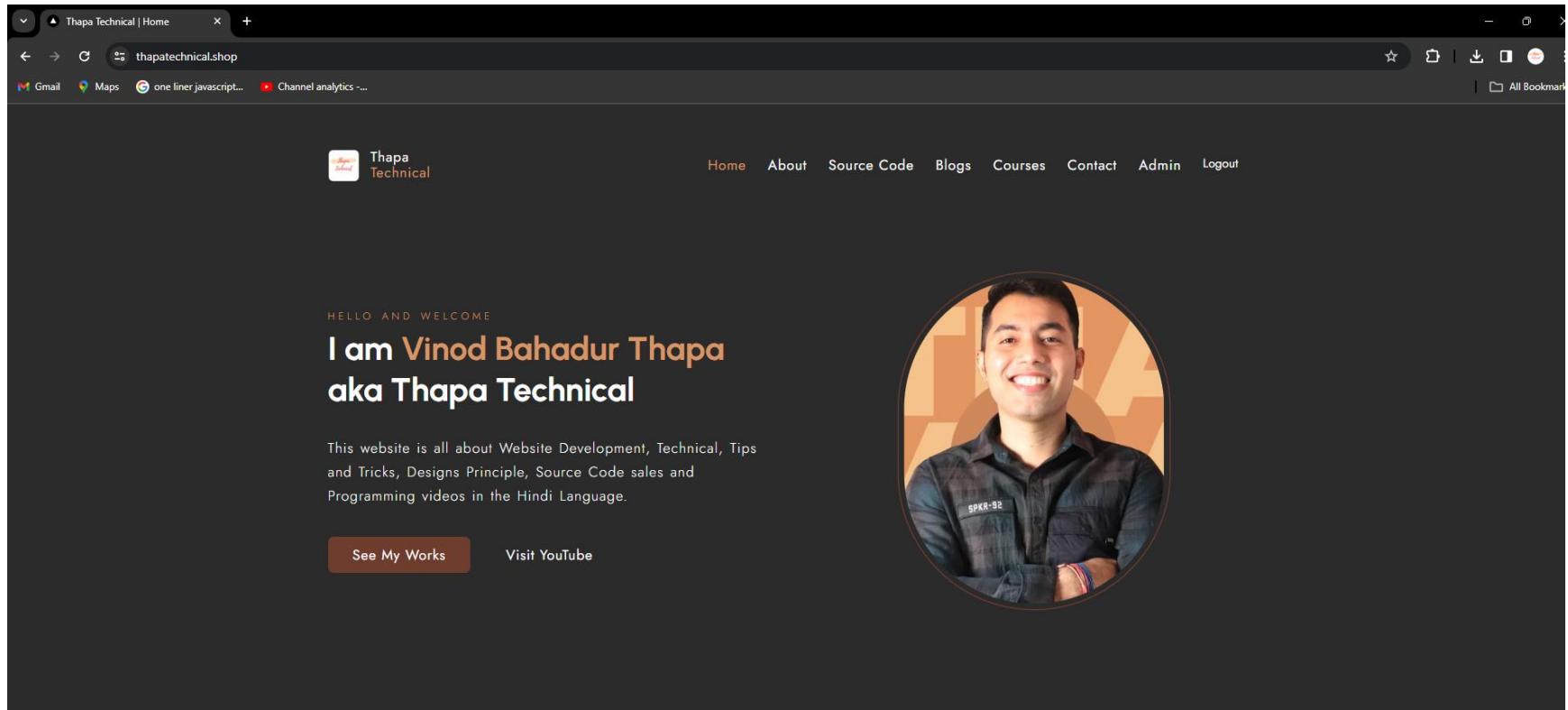
## Document Object Model (DOM):

The DOM represents the structured document as a tree of objects, where each object corresponds to a part of the document (such as elements, attributes, and text).

The DOM is primarily concerned with the content of the web page and allows JavaScript to interact with and manipulate the HTML elements.

So, while the DOM is focused on the content of the page, the BOM is focused on the browser environment. The window object serves as the global object that encompasses both the BOM and the DOM when working in a browser environment.

# The Window Object



A screenshot of a web browser displaying the homepage of the website [thapatechnical.shop](https://thapatechnical.shop). The page has a dark theme with orange and white text. At the top, there's a navigation bar with links for Home, About, Source Code, Blogs, Courses, Contact, Admin, and Logout. On the left, there's a "Hello and Welcome" message followed by the text "I am Vinod Bahadur Thapa aka Thapa Technical". Below this is a paragraph about the website's purpose. At the bottom left are two buttons: "See My Works" and "Visit YouTube". To the right is a large circular portrait of a smiling man, identified as the author. The background features a faint watermark of the word "THAPATECHNICAL".

Thapa Technical | Home

thapatechnical.shop

Gmail Maps one liner javascript... Channel analytics ... All Bookmarks

Home About Source Code Blogs Courses Contact Admin Logout

HELLO AND WELCOME

I am **Vinod Bahadur Thapa**  
aka **Thapa Technical**

This website is all about Website Development, Technical, Tips and Tricks, Designs Principle, Source Code sales and Programming videos in the Hindi Language.

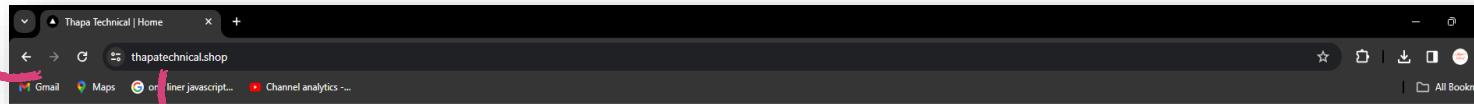
See My Works Visit YouTube

SPKR-92

THAPATECHNICAL

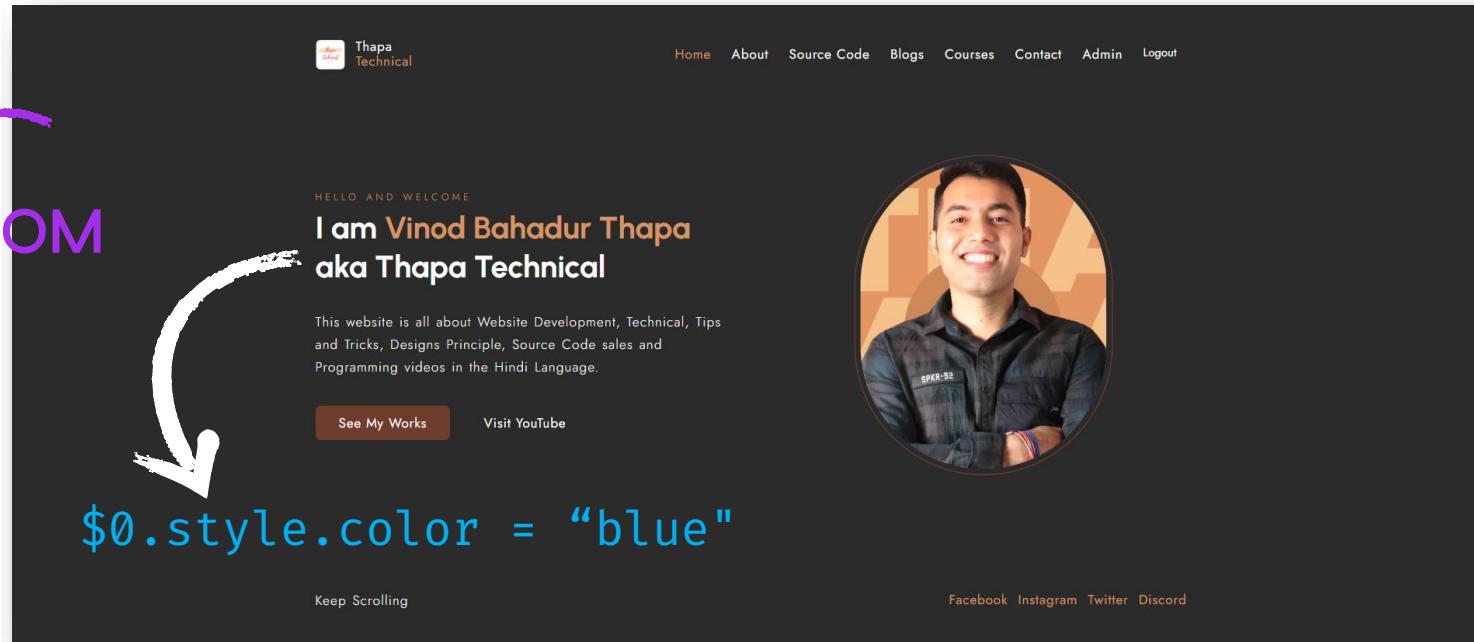
Subscribe: [ThapaTechnical](https://www.youtube.com/c/ThapaTechnical)

# The BOM



window.location.href

# The DOM



\$0.style.color = "blue"

Subscribe: [ThapaTechnical](#)

```
8   <body>
9     !—— /comment —→
10    <div>
11      Hello
12      <span>World</span>
13    </div>
14    <script></script>
15  </body>
```

```
> document.body.childNodes
< ▼ NodeList(7) [text, comment, text, div, text, script, text] ⓘ
```

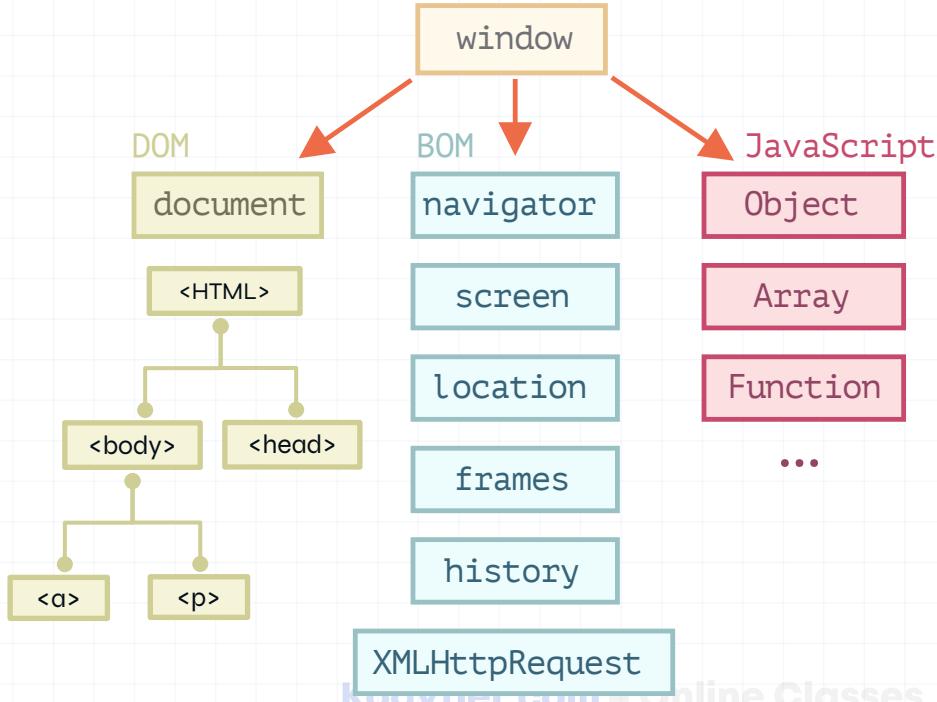
```
innerHTML: "\n      Hello\n      <span>World</span>\n      "
innerText: "Hello World"
```

```
textContent: null
```

[Kodyfier.com](https://Kodyfier.com) - Online Classes

Subscribe: [ThapaTechnical](https://ThapaTechnical.com)

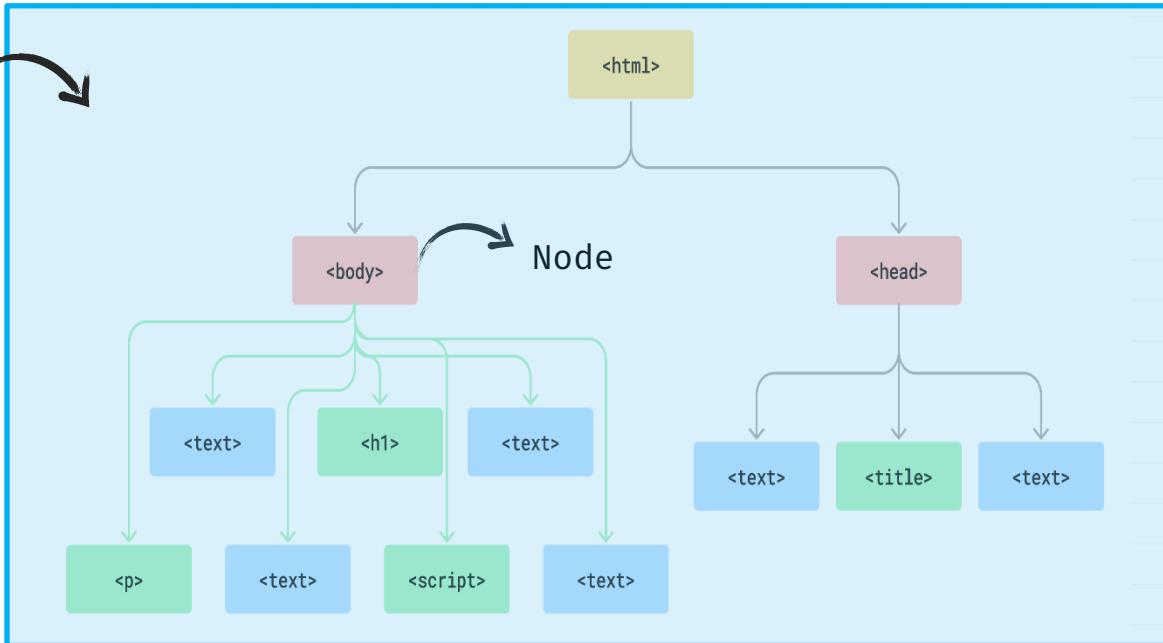
# Window Global Object



<https://javascript.info/dom-nodes#:text=An%20example%20of%20the%20DOM&text=Every%20tree%20node%20is%20an%3E%20are%20its%20children%C20etc.>

# BROWSER - DOM TREE

```
<!DOCTYPE HTML>
<html>
<head>
  <title>JavaScript</title>
</head>
<body>
  <h1>Best JS Course</h1>
  <p> DOM Tree Structure</p>
  <script> </script>
</body>
</html>
```



This entire DOM tree is then accessible to JavaScript as an object.

<https://javascript.info/dom-nodes#:text=An%20example%20of%20the%20DOM&text=Every%20tree%20node%20is%20an%3E%20are%20its%20children%2C%20etc.>

# Window Global Object



```
<!DOCTYPE HTML>
<html>
<head>
  <title>JavaScript</title>
</head>
<body>
  <h1>Best JS Course</h1>
  <p> DOM Tree Structure</p>
</body>
</html>
```

```
>   document.body.childNodes
<‐ ▼ NodeList(5) [text, h1, text, p, text] ⓘ
  ► 0: text
  ► 1: h1
  ► 2: text
  ► 3: p
  ► 4: text
    length: 5
  ► [[Prototype]]: NodeList
>   document.head.childNodes
<‐ ▼ NodeList(3) [text, title, text] ⓘ
  ► 0: text
  ► 1: title
  ► 2: text
    length: 3
  ► [[Prototype]]: NodeList
```



This entire DOM tree is then accessible to JavaScript as an object.

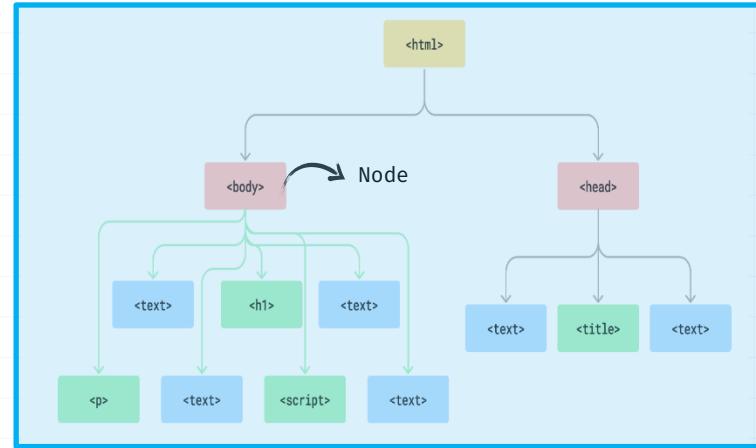
<https://javascript.info/dom-nodes#:~:text=An%20example%20of%20the%20DOM&text=Every%20tree%20node%20is%20an,%3E%20are%20its%20children%C20etc.>

THAPA TECHNICAL

Rudyner.com

The Document Object Model (DOM) is a **tree-like representation** of the HTML document. It provides a way **to interact with the HTML document using JavaScript**. The DOM provides **multiple properties** and **methods** to dynamically change the content of the HTML document using JavaScript

```
<!DOCTYPE HTML>
<html>
<head>
  <title>Javascript</title>
</head>
<body>
  <h1>Best JS Course</h1>
  <p> DOM Tree Sturcture</p>
</body>
</html>
```



## DOM Properties

document

getElementById(id)

getElementsByClassName(className)

getElementsByTagName(tagName)

querySelector(selector)

querySelectorAll(selector)

innerHTML

textContent

style

## DOM Methods

createElement(tagName)

appendChild(node)

removeChild(node)

addEventListener(event, function)

removeEventListener(event, function)

setAttribute(name, value)

getAttribute(name)

parentNode / parentElement

childNodes / children

firstChild / firstElementChild

lastChild / lastElementChild

nextSibling / nextElementSibling

previousSibling / previousElementSibling

closest(selector)

forEach (Array.from)

Thank you for your love and support! 😊 We hope you enjoy our world-class JavaScript course by Thapa Technical. If you find it helpful, could you please share the video with your friends too? Your support means the world to us! ✅

Here is the link: <https://youtu.be/13gLB6hDHR8>

Kodyfier.com - Online Classes

THAPA TECHNICAL

Subscribe: ThapaTechnical