

# Introduction to JavaScript

JavaScript is a versatile programming language used to create interactive web pages. It adds dynamism to websites, making them more user-friendly and engaging.

 by saowanee jongjaijit





# History and Evolution of JavaScript

1

## Early Years (1995)

Originally called Mocha, later renamed to LiveScript and finally JavaScript, it was developed by Brendan Eich at Netscape Communications.

2

## Standardization (1997)

ECMAScript was introduced as a standard for JavaScript, ensuring compatibility across different web browsers.

3

## Modern JavaScript (2009)

The release of ECMAScript 5 marked a significant update with features like strict mode and JSON support.



```

(
o:fviteoleesn>
t:teoflesstechte/(destefferey)>
t:cctotoy/ipalceifn)>
t:tctalesrthelies:feses(farglece(glcth)>
rce:freeth)>
t:cctaleortcher/ple.sein)>
t:cctalepatcheocloes/inll car.eint>
t:sctaleprtcalec6ccasts,lft.slen)>
t.focalecctect./bessting finoce fortility;ait)>
tark:jccaleyctbof/festtalact.ctrfant>
tity{ccaleerchosrtect.fife.ein}>
  jrtalsutongtec oflect in cleant>
  jrcateyeckecterteffing lotr,eft)>
  irtalearchoclecfness intcloge.estlocs/int>
  jacalearchocerrffeltallt,tigt eftay)nale Sud>
  jecaleyeckectecffetetian fleate,retesfint>
  jrtalayetoche/Cleate aso flces/int>
  jotalerchatoc/des/lfth.tag lattte,pnt>
  jacalayatoloyes/lass/iflatter calll: wis fnt)>
  jatalSATCDeetfdeveling (tan't -Ac officer, see Bad>
  jacalayetPesrfantcip)gecliet, tratir.itt: )
) galfin.com.pn)>
) saloyesgalistaciudGrancer > ();
jacalestecler3 "7eurtngecite" expaster,ix a(l);
frta:learccnt.'lach)>
t:ingltecleateh,aint>
  fctaesperish.pre.elit())>
t:jotalerstoolefcbestting 'with.wath:ohgalced)>
t:jocalercteclong.left>

```

# Syntax and Language Fundamentals

1

## Case Sensitivity

JavaScript is case-sensitive, so 'myVariable' is different from 'MyVariable'.

2

## Semicolons

While not strictly required, semicolons are commonly used to separate statements.

3

## Comments

Single-line comments use `//` and multi-line comments are enclosed in `/* */`.

4

## Keywords

Reserved words like 'var', 'let', 'const', and 'function' have specific meanings.

# Variables, Data Types, and Operators

## Variables

Variables store data using 'var', 'let', or 'const' keywords, each with different scoping rules.

## Data Types

Common data types include numbers, strings, booleans, arrays, objects, and null.

## Operators

Operators perform operations on data, such as arithmetic, comparison, logical, and assignment.



# Control Flow and Conditional Statements

1

## Conditional Logic

'if', 'else if', and 'else' statements control program flow based on conditions.

2

## Loops

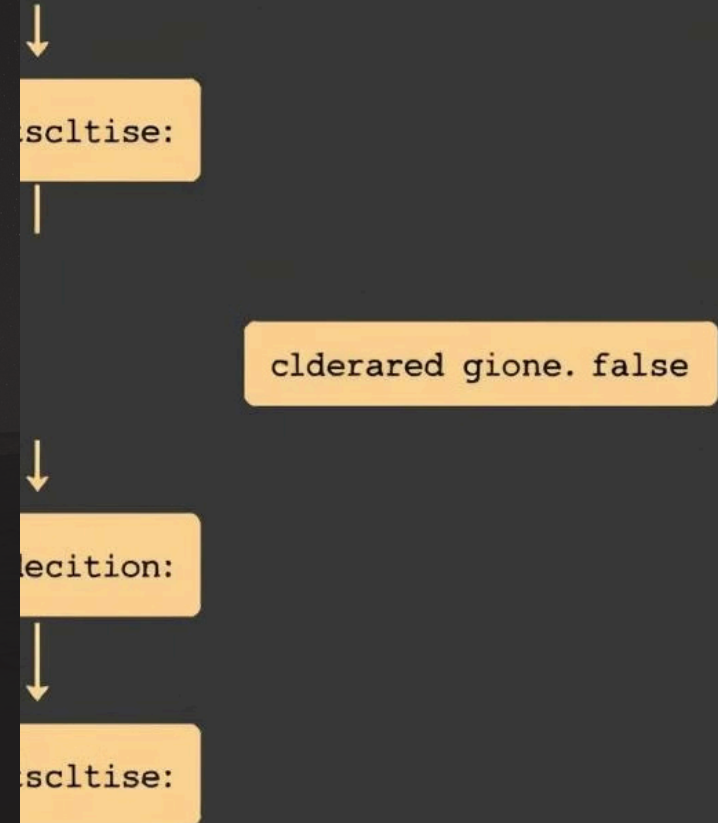
'for', 'while', and 'do...while' loops repeat code blocks until a condition is met.

3

## Switch Statements

A switch statement provides an efficient way to compare a value against multiple cases.

-Sieeee Sarettlo



# Function Call Stack

```
fontecm. >?)
```

```
fontæf: '1)
```

```
fontsent: ine( >?)
```

```
font '1)
```

```
fontsen: '1)
```

```
fost
```

## Functions and Scope

### Functions

Functions encapsulate reusable code blocks and perform specific tasks.

### Scope

Variables declared within a function have local scope, while those declared outside have global scope.

### Parameters and Arguments

Functions can accept input values through parameters and return results.



# Arrays and Objects

Arrays

Ordered collections of elements, accessible by index.

Objects

Key-value pairs representing data structures with properties and methods.



# DOM Manipulation and Events



## Document Object Model (DOM)

The DOM represents a webpage as a tree-like structure, allowing JavaScript to interact with HTML elements.



## Events

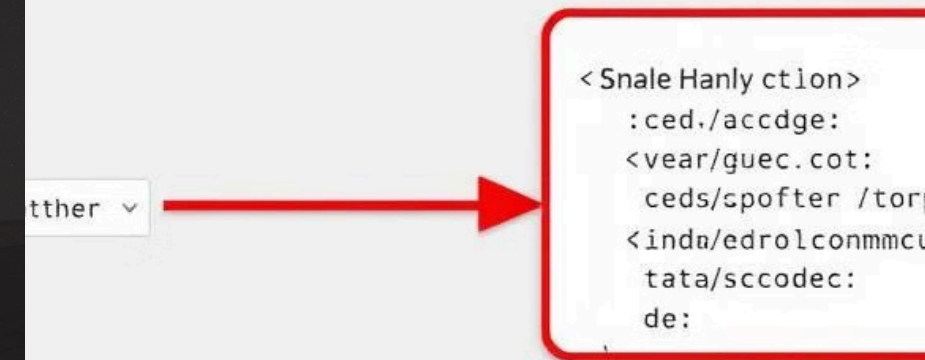
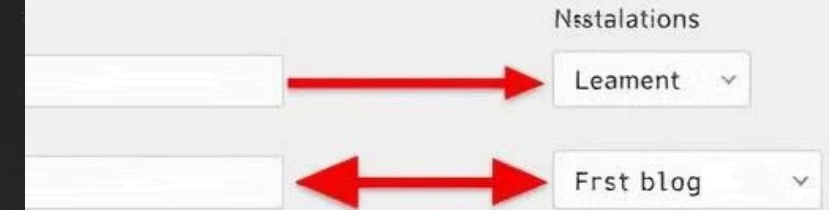
Events are actions that occur on a webpage, such as clicks, mouse movements, and key presses.



## Event Listeners

Event listeners allow JavaScript code to react to specific events and execute functions when they occur.

ns





# Asynchronous Programming with Callbacks, Promises, and Async/Await

1

## Callbacks

Functions passed as arguments to other functions, executed after an asynchronous operation completes.

2

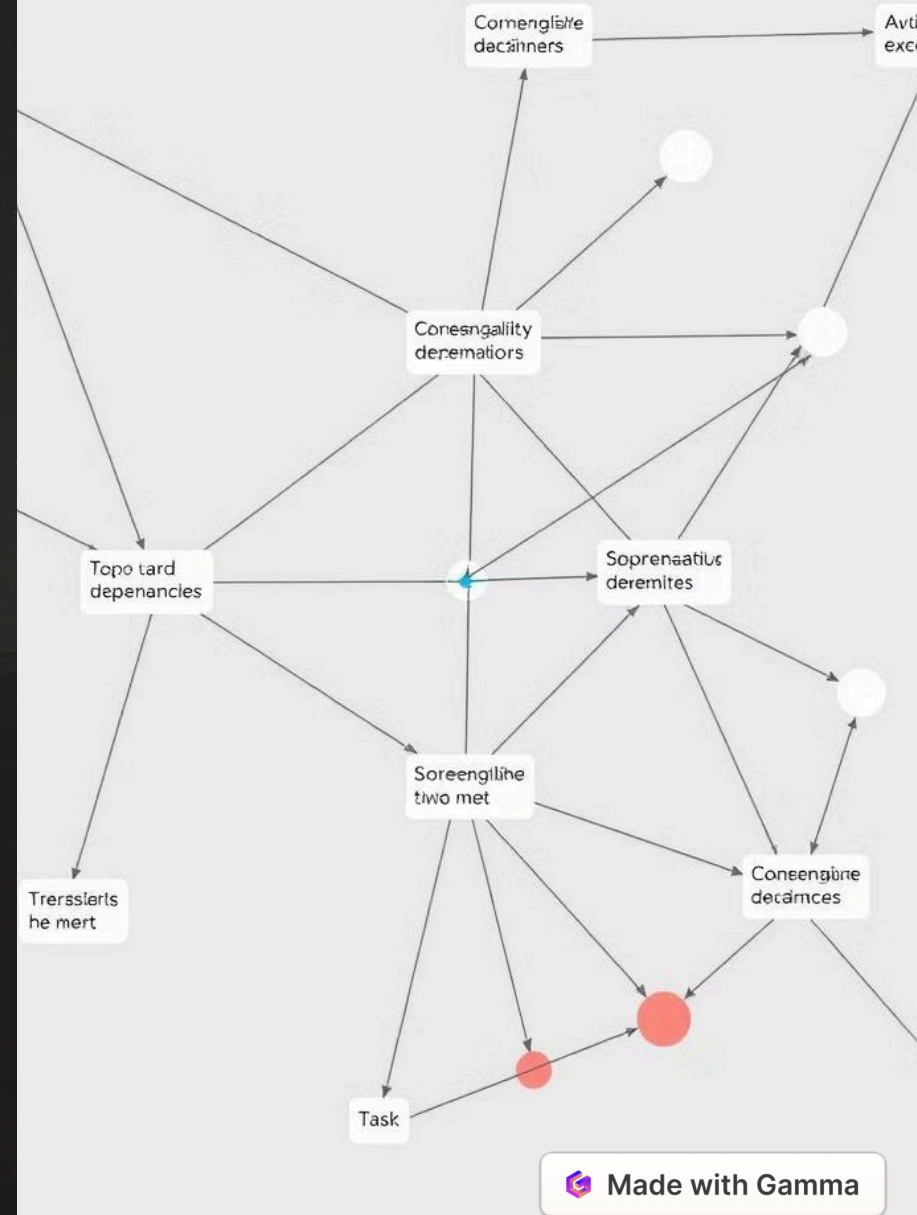
## Promises

Objects representing the eventual result of an asynchronous operation, handling success or failure.

3

## Async/Await

Syntactic sugar that simplifies asynchronous code, making it more readable and easier to manage.



# Modern JavaScript Frameworks and Libraries



React

## React

A popular JavaScript library for building user interfaces with a focus on component-based architecture.

Angular

## Angular

A comprehensive framework for building complex web applications with features like data binding and dependency injection.



## Vue

A progressive framework for building user interfaces, known for its ease of use and scalability.