

## JavaScript Keywords

Keyword	Description	Example
var	Declares a variable (function-scoped)	var x = 10;
let	Declares a block-scoped variable	let y = 20;
const	Declares a block-scoped constant	const z = 30;
function	Declares a function	function greet() { return 'Hi'; }
return	Returns a value from a function	return x + y;
if	Specifies a block of code to execute if condition is true	if (x > 5) { console.log(x); }
else	Specifies a block of code if condition is false	if (x > 5) {} else { console.log('No'); }
switch	Selects one of many blocks to be executed	switch(x) { case 1: break; }
case	Defines a case in switch	case 1: console.log('One'); break;
default	Default case in switch	default: console.log('None');
for	Creates a loop with initialization, condition and final expression	for (let i = 0; i < 5; i++) { }
while	Creates a loop while a condition is true	while (x < 10) { x++; }
do...while	Executes block at least once, then repeats if condition is true	do { x++; } while (x < 10);
break	Exits a loop or switch	if (x > 5) break;
continue	Skips current iteration in loop	if (x < 5) continue;
try	Marks a block of code to test for errors	try { riskyFn(); }
catch	Handles the error	catch (e) { console.log(e); }
finally	Always executes after try/catch	finally { cleanup(); }
throw	Throws a custom error	throw new Error('Something went wrong');
class	Declares a class	class Animal { constructor() {} }
constructor	Special method for creating and initializing an object	constructor(name) { this.name = name; }
extends	Used for class inheritance	class Dog extends Animal { }
super	Calls the parent constructor	super(name);
import	Imports functions/objects from another file	import x from './file.js';

export	Exports functions/objects to be used in other files	export const x = 5;
new	Creates an instance of an object	let d = new Date();
typeof	Returns type of a variable	typeof 5 // 'number'
instanceof	Checks if object is instance of class	x instanceof Array // true
this	Refers to current object	this.name = 'JS';
delete	Deletes an object property	delete obj.prop;
in	Checks if property exists in object	'x' in obj // true