Javascript

Variables are containers for storing data ( using var, const and let keywords ).

When adding a number and a string, Javascript will treat the number as a string.

A JavaScript function is defined with the function keyword, followed by a **name**, followed by parentheses **()**.

Function names can contain letters, digits, underscores, and dollar signs (same rules as variables). The parentheses may include parameter names separated by commas:  
**(parameter1, parameter2, ...)**

The code to be executed, by the function, is placed inside curly brackets: **{}** Function **parameters** are listed inside the parentheses () in the function definition. Function **arguments** are the **values** received by the function when it is invoked. Inside the function, the arguments (the parameters) behave as local variables. Functions often compute a **return value**. The return value is "returned" back to the "caller". Variables declared within a JavaScript function, become **LOCAL** to the function. Local variables can only be accessed from within the function.

A javaScript object is **an entity having state and behavior (properties and method)**. For example: car, pen, bike, chair, glass, keyboard, monitor etc. JavaScript is an object-based language. Everything is an object in JavaScript.

You define (and create) a JavaScript object with an object literal. You can access object properties using ‘object\_name.property’.

Objects can also have **methods**. Methods are **actions** that can be performed on objects. Methods are stored in properties as **function definitions**.

HTML events are **"things"** that happen to HTML elements. When JavaScript is used in HTML pages, JavaScript can **"react"** on these events.