**JavaScript**

* **Variable**
* Var : works similarly to let it declares a variable but it does so in an older.
* let : declare variable in block scope that allow we can change value later.
* **Const :** declare variable in block scope that not allow to reassigned new.
* **Data types**
* **String :** for letter
* **Number :** for number (int , float)
* **Boolean :** for true or false
* **Array :** for store multiple values
* **Object :** for more complex data structure
* **Null :** for unknow value
* **Undefined :** for unassigned values
* **Functions**
* **Declaration :** is one of the ways to define a function. It's a statement that creates a named function which can be called later in the code. Function declarations are hoisted to the top of their scope. This means you can call the function before it's defined in the code.
* **Expression :** main difference between a function expression and a function declaration is the *function name*, which can be omitted in function expressions to create *anonymous* functions. A function expression can be used as an [IIFE](https://developer.mozilla.org/en-US/docs/Glossary/IIFE) (Immediately Invoked Function Expression) which runs as soon as it is defined.
* **arrow functions :** An arrow function in JavaScript is a shorter and more concise way to write functions, introduced in ES6 (ECMAScript 2015). It uses the => syntax, which looks like an arrow, hence the name.
* **Conditionals**
* **If :** Check if a condition is true.
* **Else :** What to do if none are true.
* **Switch :** is used to perform different actions based on different conditions.
* **Loops**
* **For loop :** Used when the number of iterations is known.
* **While loop :** Used when the number of iterations is not known in advance.
* **For…**.**of** : Used to iterate over **iterable objects** like arrays, strings, etc.
* **Arrays and objects manipulation**
* **Map** : Transforms each element of an array and returns a **new array**.
* **Filter** : Filters elements based on a condition and returns a **new array**.
* **Reduce :** Reduces an array to a **single value** (number, object, string, etc.)
* **Scope and hoisting**
* Scope : is the **context** in which variables, functions, and objects are accessible in your code.
* Hoisting : **Hoisting** is JavaScript's default behavior of **moving declarations to the top** of their scope (not the assignments).