JSVARIABLES

WHAT

Variables are used **to store data values**. A variable need to be **declared** by a keyword (var, let, const). Next need to **be initialized**, assigning values by **equal (=) sign.** Variables are subjected to **Hoisting** behavior. A variable declared without a value will have the value undefined. Use **camelCase**.

	VAR	LET	CONST
SCOPE	Global / Local / Function Scope => available all inside the function	**Block Scope** => available only inside the specific block scope (ex. a for loop block, inside the function)	**Block Scope** => available only inside the specific block scope (ex. a for loop block, inside the function)
ASSIGNEMENT	**Redeclaretion**: YES **Reinitialization**: YES Can be reassigned, by re-declaring the variable => risk of overwriting	**Redeclaretion**: NO **Reinitialization**: YES Can be reassigned, but can't be re-declared	**Redeclaretion**: NO **Reinitialization**: NO Can't be reassigned
HOISTED*	Yes	No, variables LET and CONST are **hoisted to the top of the block, but NOT initialized**. Using them before being declared, will return **ReferenceError.**	No, variables LET and CONST are **hoisted to the top of the block, but NOT initialized.** Using them before being declared, will return **ReferenceError.**
REFERENCE BEFORE DECLARATION	Yes	No	No

*Hoisting is **THE DEFAULT BEHAVOIR** of **moving declarations to the top** of the current scope.

annalisacecchini.com | annalisacecchini.dev@gmail.com | (+39) 349.251.23.35