# **Hoisting**

1. In JS, before the code is executed, the variables get initialized to undefined.

2. Arrow functions enact as variables and get "undefined" during the memory creation phase while functions actually get run.

3. Hoisting: Mechanism in JS where the variable declarations are moved to the top of the scope before execution. Therefore it is possible to call a function before initializing it.

4. Whenever a JS program is run, a global execution block is created, which comprises of

2: Memory creation and Code execution.

Golden Rules:

1. Variable declarations are scanned and are made undefined

2. Function declarations are scanned and are made available

Undefined means variable has been declared but value is not assigned

Not defined means variable is not declared

findings

let and const variables and functions are also hoisted , hoisting just means that they are present in the memory before code execution , but the point here is that let and const are set to uninitialised during memory creation phase and var variables are initialised with undefined but var let const all are hoisted keep in mind

IMPORTANT

Variables defined with let and const are hoisted to the top of the block, but not initialized.

Meaning: The block of code is aware of the variable, but it cannot be used until it has been declared.

* Using a let variable before it is declared will result in a ReferenceError.

The variable is in a "temporal dead zone" from the start of the block until it is declared:

* Using a const variable before it is declared, is a syntax error, so the code will simply not run.