whats hoisted

Hoisting in JavaScript is a behavior where the declarations of variables (**var**, **let**, **const**), functions, and classes are **moved to the top of their scope**—either the global scope or the nearest function/block—*before* code execution actually starts.

However, **only the declarations are hoisted, not the assignments**.

How Hoisting Works

With `var`:

- The variable declaration is hoisted and initialized as `undefined`.
- You can reference the variable before its declaration, but you'll get `undefined` until it's actually assigned.

```
console.log(a);   // undefined (due to hoisting)
var a = 5;
console.log(a);   // 5
```

With `let` and `const`:

- Declarations are also hoisted, **but not initialized**.
- Trying to access them before their declaration throws a `ReferenceError`. This time gap, between the start of the block and the declaration, is called the **Temporal Dead Zone (TDZ)**.

```
console.log(b); // ReferenceError: Cannot access 'b' before initialization
let b = 10;
```

With Functions:

- **Function Declarations** are hoisted completely—their name and entire function body can be used before their definition.

```
greet();           // "Hello!"
function greet() {
  console.log("Hello!");
}
```

- **Function Expressions** (like `var greet = function() {...}`) only hoist the variable, not the function assignment, so accessing before assignment gives `undefined` or errors.

Summary Table

Declaration Type	Hoisted?	Usable Before Declaration?	Value Before Declaration
var	Yes	Yes	undefined
let	Yes	No (TDZ, throws ReferenceError)	-
const	Yes	No (TDZ, throws ReferenceError)	-
function (decl.)	Yes (fully)	Yes	function itself
function (expr.)	Yes (var only)	No (undefined or throws if called)	undefined

In short:

Hoisting means JavaScript "moves" declarations to the top of their scope before executing code. Use `let` and `const` to avoid confusion, as they don't behave as forgivingly (or as dangerously!) as `var` when accessed before declaration [1] [2] [3] [4].

⁂

1. https://www.w3schools.com/js/js_hoisting.asp
2. https://developer.mozilla.org/en-US/docs/Glossary/Hoisting
3. https://www.geeksforgeeks.org/javascript/javascript-hoisting/
4. https://www.programiz.com/javascript/hoisting