



I want clear notes for Javascript Variables (Let , Const , Var) with basic codes simply point wise i want every line should be related to the code and understandable way

JavaScript Variables: var, let, const—Clear Notes with Examples

- **var** is function-scoped and can be re-declared and updated.

```
var a = 5;           // Declare a variable "a" with value 5
var a = 10;          // Redeclare "a" (allowed with var)
a = 15;              // Update "a" (allowed with var)
console.log(a);      // Output: 15
```

- **let** is block-scoped and can be updated, but not re-declared in the same scope.

```
let b = 20;          // Declare "b" with value 20
// let b = 25;        // ERROR: Cannot redeclare "b" in the same block
b = 30;              // Update "b" to 30
console.log(b);      // Output: 30
```

- **const** is block-scoped, must be initialized, and cannot be updated or re-declared.

```
const c = 40;        // Declare "c" as a constant with value 40
// const c = 50;      // ERROR: Cannot redeclare "c"
// c = 60;             // ERROR: Cannot update "c"
console.log(c);      // Output: 40
```

- **var** gets hoisted (moved to the top), initialized with `undefined`. You can use it before declaration (but value is `undefined`).

```
console.log(d);      // Output: undefined
var d = 70;          // Declared after usage
```

- **let** and **const** are hoisted but **not initialized**. Using before declaration causes error (Temporal Dead Zone).

```
// console.log(e); // ERROR: Cannot access 'e' before initialization
let e = 80;         // OK after declaration
```

- **let** and **const** respect block scope (inside `{}`), but **var** does not.

```

{
  var f = 90;    // "f" available outside block (no block scope)
  let g = 100;   // "g" only inside block
  const h = 110; // "h" only inside block
}
console.log(f);  // Output: 90
// console.log(g); // ERROR: "g" is not defined outside block
// console.log(h); // ERROR: "h" is not defined outside block

```

- Use **let** for variables that you expect to change, and **const** for constants. Avoid **var** in modern JavaScript.

Summary Table

Keyword	Scope	Re-declaration	Updatable	Hoisted	Use Case
var	Function	Yes	Yes	Yes	Older JS, avoid in ES6
let	Block	No	Yes	No	Variables (changeable)
const	Block	No	No	No	Constants (no change)