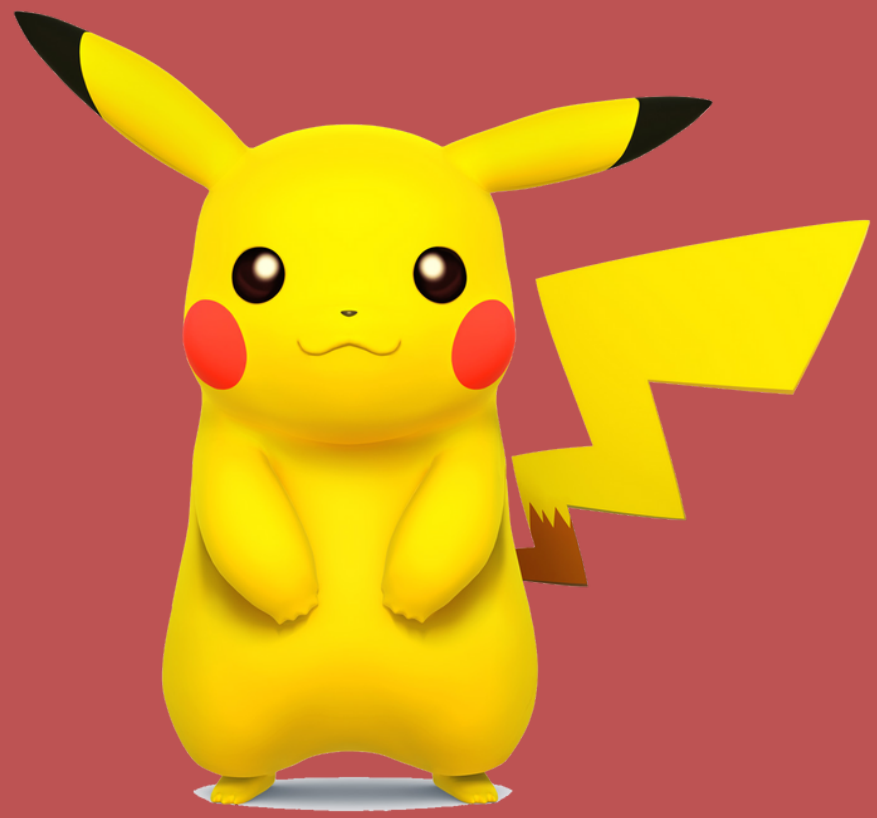
VAR, LET AND CONST IN JAVASCRIPT

JS



Simran Bedi

@psimran

JS

SCOPE

VAR - THE SCOPE OF A VAR VARIABLE IS FUNCTIONAL SCOPE.

LET - THE SCOPE OF A LET VARIABLE IS BLOCK SCOPE.

CONST - THE SCOPE OF A CONST VARIABLE IS BLOCK SCOPE.

JS

EXAMPLE

```
{
  let x = 10;
  const y = 20;
  var z = 30;

  console.log(x); // Output: 10
  console.log(y); // Output: 20
  console.log(z); // Output: 30
}

console.log(x); // Error: x is not defined
console.log(y); // Error: y is not defined
console.log(z); // Output: 30
```

JS

UPDATION & REDECLARATION

VAR - IT CAN BE UPDATED AND RE-DECLARED INTO THE SCOPE.

LET - IT CAN BE UPDATED BUT CANNOT BE RE-DECLARED INTO THE SCOPE.

CONST - IT CANNOT BE UPDATED OR RE-DECLARED INTO THE SCOPE.

JS

EXAMPLE OF UPDATION

```
let a = 10;
const b = 20;
var c = 30;

a = 5;
b = 15; // Error: Assignment to constant variable
c = 25;

console.log(a); // Output: 5
console.log(b); // Uncaught TypeError: Assignment to constant variable.
console.log(c); // Output: 25
```

JS

EXAMPLE OF REDECLARATION

```
var x = 10;
let y = 20;
const z = 30;

var x = 40; // Redeclaration with 'var' is allowed
let y = 50; // Error: Identifier 'y' has already been declared
const z = 60; // Error: Identifier 'z' has already been declared

console.log(x); // Output: 40
console.log(y);
console.log(z);
```

JS

DECLARATION WITHOUT INITIALIZATION

VAR - IT CAN BE DECLARED WITHOUT INITIALIZATION.

LET - IT CAN BE DECLARED WITHOUT INITIALIZATION.

CONST - IT CANNOT BE DECLARED WITHOUT INITIALIZATION.

JS

EXAMPLE

```
var x;
let y;
const z;

console.log(x); // Output: undefined
console.log(y); // Output: undefined
console.log(z); // Output: SyntaxError: Missing initializer in const declaration
```

JS

HOISTING

VAR - HOISTING DONE, WITH UNDEFINED AS ‘DEFAULT’ VALUE

LET - HOISTING IS DONE, BUT NOT INITIALIZED

CONST - HOISTING IS DONE, BUT NOT INITIALIZED

JS

EXAMPLE

```
console.log(x); // Output: undefined
console.log(y); // Output: ReferenceError: Cannot access 'y' before initialization
console.log(z); // Output: ReferenceError: Cannot access 'z' before initialization

var x = 10;
let y = 20;
const z = 70;
```

JS

SIMRAN BEDI

@psimran

LIKE SHARE AND FOLLOW