

Page No.			
Date			

LET, CONST OR VAR

- * LET, CONST are introduce in ES6
- * That's why they are mother in JS.

1) * Let:- Let variable are re-accessible variable in JavaScript. If I give
 Let CM = 23; then we can re-access
 Let CM = 50; also

* Variables declared as let are in the block scope

eg:- Let Age = '30';
 Age = '31'; ← Re-accessible

2) * const:- Not re-accessible like Let the variable is fixed when we use in first time. If I use const variable next time the console show me error
 eg:- const birthyear = '1991';
 birthyear = '1993'; ← error generate

* variable declare as a const are in the block scope

3) * Var:- var is like Let but the var use before the Envation of ES6 that's why var is old variable type.

* var are in the function scope

have to

★ All in three we can use const for the good practice.

* we can use variable or declare variable without var, const, let
eg.

```
CM = '23';
```

```
console.log(CM);
```

output:- 23.

But for the good practice use variable (let, const, var)


```
const fixedValue = 1234;  
let new value = "chandan";  
var old value = "older";  
let not Declare;
```

```
console.log([fixed value, new value, old value,  
not declare]);
```

Output:-

Index	value
0	1 2 3 4
1	'chandan'
2	"older"
3	undefined

EXPLORER

OPEN EDITORS

WELCOME

indexjs

indexhtml

HITESH JAVASCRIPT

index.html

index.js

Welcome

indexjs

index.html

indexjs > ...

1 const fixedValue = 1234;

2 let newFormat = "chandan";

3 var olderValue = "1334";

4 let notDeclare;

5

6 //new format of console all value in one table

7 console.table([fixedValue, newFormat, olderValue, notDeclare]);

8

9 //fixedValue = 5678;

10 newFormat = "suraj";

11 olderValue = "0000";

12

13 console.table([fixedValue, newFormat, olderValue, notDeclare]);

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

(index)	Values
0	1234
1	'chandan'
2	'1334'
3	undefined

(index)	Values
0	1234

Code

Code

OUTLINE

TIMELINE

Ln 3, Col 25

Spaces: 7

UTF-8

CRLF

() JavaScript

Go Live