

In the first lesson of JavaScript primitives where we talked about the 5 basic datatypes, which are string, number, Boolean, undefined & null. Here in this lesson we will learn in detail about Null & Undefined.

They are both just values of which a variable can be set to and they both mean nothingness in a sense, but there is a big difference between them. The code below illustrates that difference.

## Null and Undefined

```
//The two other primitives are null and undefined

//Variables that are declared but not
//initialized are undefined
//The following variables are undefined:
var name;
var age;

//null is "explicitly nothing"
var currentPlayer = "charlie";
currentPlayer = null;    //game over
```

Let's start with undefined, if we declare a variable like

```
var name;
```

but we do not set it to any value or strings, then its declared but not initialized to a value, then it is considered undefined.

Console coding:

```
var age;
```

```
age;
```

➔ *Undefined*

Null is another way of expressing nothingness, but its different because null means explicitly empty or explicitly nothingness, versus undefined which just means that something does not have a value yet.

So, on the picture, we are setting a current player variable as “charlie” in a game but then when the player dies, the current player variable is set to NULL, to make it explicitly clear that there is no current player.

Null and Undefined are similar, they both mean nothingness, but there is a big difference between them where NULL means explicitly nothing, versus Undefined something is empty or does not have a value yet.