

# JAVASCRIPT

## DAY 1 ASSIGNMENT

### 1. Difference between Let and Var.

	LET	VAR
scope	block scoped	function scoped
redeclaration	no	yes
hoisting	not hoisted	hoisted
global props creation	no	yes
global scope	not attached to window object	attached to window object
usage	for temporary variables that perish in a loop or function	for global/local variables

## 2. Why are null, array printed as object type in console.

**NULL** - Null is a primitive data type. But when we check 'typeof(null)' in console it returns "object". Since null is a primitive value allows us to conclude that this is a bug.

Loosely and strictly equals provide the correct return of false.

Null is considered an object for web safety reasons. The legacy code could be broken if this was to change that it was deemed better to just leave it alone, but it is not and will most likely become deprecated in the future.

**ARRAY** - An ARRAY lets us store multiple values in a single variable. But when we check 'typeof(array)' in console it returns "object".

Arrays are special type of objects in JS. Unlike regular objects, they have special features.

- Arrays have length property.
- Arrays have a special algorithm regarding defining new properties.
- Arrays have an additional object in their prototype chain i.e. Array.prototype.

Arrays have constructor function in JS, like any object in JS. Hence they are considered as objects,