1 - Shallow and Deep copy :

Shallow copy : used to make a copy of an object or array , & it will affect two the copy one & the original one .

we using object.assign() for object copy EX :

const shallowCopy = Object.assign({}, original);

,

using (spread operator) ... for object or array EX :

const shallowCopyArray = [...originalArray];

.

Deep copy : used to create a new object or new array with the copy of another object or array value will not affect the other only affect the copy one. نفس القيمة ب اسم جديد.

we using JSON.parse() and JSON.stringify() for simple objects and arrays EX :

const deepCopy = JSON.parse(JSON.stringify(original-array));

.

we use shallow copy when you don’t need to modify the nested objects/arrays independently. It is typically faster and sufficient.

we use deep copy when you need completely independent copy of the original object or array.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2 - Hoisting :

var: Hoisted and initialized with undefined.

let and const: Hoisted but not initialized, leading to a temporal dead zone.