

JavaScript Hoisting

- هو أسلوب تستخدمه محركات ال js لحفظ اسماء الاكواد مثل الوظائف والمتغيرات في الذاكرة قبل تنفيذ باقي الاوامر ويحفظهم في اعلي الصفحة وبعد ذلك سيتم استخدامهم ..

- وهنا يتم تعريف ال function و Variable ثم استخدامهم :

- Declaring a Function .

- function myFunction () { }
 - myFunction()

- Declaring a variable .

- let a = "a" ;
 - console.log(a)

-ويتم استخدامهم ببساطة بعد تعريفهم ولكن ال hoisting يتم استخدامهم قبل تعريفهم مثال :-

Function

```
-console.log(newFunction);  
-function newFunction () {  
    return "hello"  
}
```

-ولكن دا يطبق علي ال function من النوع العادي وليس يطبق علي
expression function and arrow function

Variables

-console.log(a);
-var a = 5 ; ⇒ undefined

-ولكن دا يطبق علي المتغير من نوع ال Var فقط وليس يطبق علي Let and Const
-لانه هيشوف المتغير دا بدون value ⇒ var a ;
-وبيكون الاولوية لتعريف ل Functions ثم ...Variables

```
carName = "bmw";  
let carName ⇒ This will result in a ReferenceError:
```

```
carName = "bmw";  
const carName ⇒ This will result in a ReferenceError:
```

Deep copy

القدرة علي اخذ نسخة من الداتا المخزنة داخل array او object عن طريق الmethod
copywithin , ولكن هذه الطريقة غير مستحبة لان اي تعديل في النسخة يؤثر علي الداتا
الاصلية لانه يعتبر نسخة عميقة

shallow copy

القدرة علي اخذ نسخة من الداتا المخزنة داخل array او object عن طريق Spread [...] وهذه الطريقة مستحبة لان اي تعديل في النسخة لا يؤثر علي الداتا الاصلية لانه يعتبر
نسخة سطحية

او هناك طريقة اخري وهي طريقة JSON.stringify JSON.Parse

