

Closures

- هي عبارة عن دالة بتحتفظ بالحالة المحيطة بها والتي بتعبر عن المتغيرات الموجودة في نطاق الدالة الخارجية التي موجود فيها الدالة الداخلية .
مثال:

```
function outer() {  
  var a = 5;  
  
  function inner() {  
    var b = 10;  
    alert(a + b);  
  }  
  
  return inner;  
}  
  
var inner = outer();  
console.log(inner());
```

في المثال دا في دالتين وكل واحد له النطاق بتاعها وفي الدالة الخارجية في متغير **a** قيمته 5 وفي الدالة الداخلية في متغير **b** وقيمته 10 والدالة بتعرض جمع المتغيرين بس المفروض ان متغير **a** موجود في نطاق الدالة الخارجية بس و الدالة الداخلية تتعرف علي متغير **b** بس وهنا ببيان دور ال **closure** وهو ان الدالة الداخلية عند تعريفها بتعمل غلق علي المتغيرات التي هتحتاجها و موجوده خارجها فا في الحالة دي الدالة الداخلية هتتدبر علي المتغير **a** و هتكون النتيجة في ال **alert** هي **15 = 10+5**

Call vs bind vs apply

Call():

- هي دالة معرفه مسبقا في الجافا سكريبت ويمكن لاي دالة اخري استخدامها فهي تقوم بعمل استدعاء (Invocation) للدوال .
 - ولها دور اخر هو تغيير سياق الدوال
- مثال:

```
3
4  var User_Mark = {
5      id: 5,
6      name: 'Mark',
7      printUserName: function(){
8          console.log('The User Name Is ' + this.name);
9      }
10 };
11 User_Mark.printUserName();
12
13
```

- في المثال دا this بتشير لل object الي اسمه User_Mark

The User Name Is Mark

فا دا هيكون ال output

- و لكن في المثال دا :

```
var User_Mark = {
    id: 5,
    name: 'Mark',
    printUserName: function(){
        console.log('The User Name Is ' + this.name);
    }
};

// User_Mark.printUserName();

var User_Ahmed = {
    id: 9,
    name: 'Ahmed'
};

User_Mark.printUserName.call(User_Ahmed);
```

The User Name Is Ahmed

- الناتج سيكون هنا
 - ودا بسبب ان دالة Call غيرت سياق كلمه this وبقت بتشير ل Object الي اسمه User_Ahmed .
 - الدالة call بتاخد عده مدخلات واهمهم هو اول واحد الي بيكون ال Object الي بتشير ليه كلمه this و باقي المدخلات بيتم تمريرها للدالة و بيكون ال syntax بتاعها :
function.call(contextualObj, param1, param 2 ,...)
-

Bind():

- هي تشبه الدالة call() في تغيير سياق الدوال لكن الاختلاف بيكون في التوقيت بتاع التنفيذ فا الدالة call بيكون التنفيذ علطول لكن الدالة bind بتقوم بإرجاع دالة جديده .
- مثال:

```
function test(arg){  
  console.log(this.number, arg);  
}  
  
let bindedFn = test.bind({number: 99}, "argument");  
  
bindedFn();
```

Output:

99 'argument'

Syntax of function bind():

boundedFunc = function.bind(contextualObj,
param1, param2, ...)

Apply():

- مثل باقي الدوال فا الدالة apply تشبه الباقي في تغيير سياق الدوال و لكن تختلف في ال syntax حيث تأخذ ايتين مدخلات فقط الاول بيكون لل object كما في الدالة call و بيكون الثاني عبارته عن مصفوفة (array) ويتم تمرير عناصرها للدالة .

function.apply(contextualObj, paramsArray)

مثال:

```
function test(...arguments){  
  console.log(this.num, arguments);  
}  
  
test.apply({num: 100}, [1,2,3]);
```

Output:

100 ▶ (3) [1, 2, 3]
