

SEPTEMBER 15 – 21, 2019



المبادرة السعودية للمطورين

تعلم .. فكر .. حاول .. أبداع

المبادرة السعودية للمطورين

مسار الـ JavaScript

مشرف المسار:

محمد الأسمرى

JS

ملاحظات قبل بدء الدروس:

- على المتدربين نشر كل يوم الجزئية التي تم كتابتها من النص البرمجي في الـ **Github** تحت **Topic** بعنوان **saudidev.org** كما تم توضيحه في دروس الـ **Github** سابقاً

- على المتدربين نشر كل يوم مقدار التقدم وصورة لما تم تعلمه وتطبيقه على **Twitter** تحت الهاشتاقات:
#المبادرة_السعودية_للمطورين
_100#يوم_برمجة
#100DaysOfCode

تمنياتنا لك بالتوفيق
المبادرة السعودية للمطورين

اليوم السابع والعشرون

(jump - return)

هي طريقة ارجاع قيمة الى function وتقوم بإنهاء تنفيذ function
طريقة صياغته

```
return expression;
```

وهي تستخدم داخل function ومن الخطأ استخدامها خارجها

دعونا نكتب مثال بسيط للتربيع

```
var x;  
function square(x){  
    return x*x;  
}  
document.write(square(3));
```

هنا عرفنا function باسم square وينفذ عملية ارجاع قيمة التربيع

اما طريقة استدعاءها فهي square(3);

وسنشرح لاحقاً function بتفصيل اكبر

عندما قلنا أنها تنهي تنفيذ ال function , نعني انها ستتوقف عن اكماله جرب أن تضع طباعة جملة قبل return وبعده ولاحظ المثال أدناه

في حال كانت من دون expression فسترجع undefined او null , فالمثال أدناه استمر في الطباعة حتى تمت مقاطعته من return كما أن ال function نفسها سترجع قيمة undefined. جرب تنفيذ الكود أدناه مرة ب counter() ومرة ب document.write(counter()) "أزل التعليق وضعه في الآخر"

```
function counter() {  
    تكرار الى ما لا نهاية // for (var count = 1; ; count++) {  
        حتى 5 // document.write(count + "A<br>");  
        if (count === 5) {  
            return;  
        }  
        حتى 4 // document.write(count + "B<br>");  
    }  
    لن تظهر أبداً // document.write(count + "C<br>");
```

```
counter();
```

```
//document.write(counter());
```

اليوم الثامن والعشرون

(jump - throw)

هي طريقة لاضافة استثناءات برمجية بحيث تتحكم في القيم التي تريد التعامل معها وتفرق بينها وصيغتها كالتالي

throw expression;

خذ على سبيل المثال الكود أدناه , في حال اني أردت التعامل مع الارقام الموجبة فقط وضعنا throw لكي يتعامل مع الارقام السالبة كخطأ في حال استخدامها مع ملاحظة الخطأ سيظهر في ال console

```
var x;  
function positive(x) {  
  if (x >= 0) document.write("the number is positive");  
  else throw "x must not be negative";  
}  
positive(-1);
```

اليوم التاسع والعشرون

try/catch/finally

طريقة لمعالجة الاستثناءات

try يكون فيه الكود الذي نريد أن نعالجه من الاخطاء والاستثناءات

catch عندما يحصل الاستثناء يتم التعامل معه فيها كما يجب ان يتم تعريف الخطأ بين القوسين (e) على سبيل المثال

finally لتنظيف الكود حتى يضمن تنفيذه بغض النظر عن ماحدث في try وسيتم تنفيذها دائماً

catch & finally اختيارية في الاختيار ويجب استخدام واحد على الاقل مع try

صياغتها كالتالي:

```
try {
}
catch (e) {
}
finally {
}
```

هنا على نفس مثالنا في الدرس السابق أردنا معرفة الارقام السالبة وطباعتها في رسالة تنبيه عوضاً عن ارسالها الى console

```
var x;
function positive(x) {
  if (x >= 0) return x;
  else throw "x must not be negative";
}
try {
  var n = Number(prompt("Please enter a positive integer", ""));
  positive(n);
  alert(n + " is positive");
} catch (y) {
  alert(y);
  //document.write(y);
}
```

```
finally{
```

```
  alert("Finally, I got it");
```

```
}
```

اليوم الثلاثون

(Object)

في حياتنا اليومية نواجه الكثير من الاوبجكت وعلى سبيل المثال السيارات, السيارات لها خصائص ووظائف , الخصائص كاللون والموديل والوزن وغيرها , ووظائف كالتشغيل والاطفاء والتحرك وغيرها.

فلو أردنا تعريفها تكون الكتالي

```
var car = {type:"GMC", model:"2019", color:"white"};
```

مثال آخر

```
var person = {
  firstName: "Mohammed",
  lastName: "Alali",
  age: 30,
  eyeColor: "brown"
};
```

هكذا عرفنا object بخصائصه , لو أردنا الوصول له لاحد الخصائص بهذه الطريقة

```
objectName.propertyName
```

أو

```
objectName["propertyName"]
```

جرب

```
var person = {
  firstName: "Mohammed",
  lastName: "Alali",
  age: 30,
  eyeColor: "brown"
};
```

```
document.write(person.firstName);
```

وكذلك

```
var person = {
  firstName: "Mohammed",
```

```
lastName: "Alali",
```

```
age: 30,
```

```
eyeColor: "brown"
```

```
};
```

```
document.write(person["firstName"]);
```

اليوم الواحد والثلاثون

(Object with method)

كما قلنا سابقاً أن object له خصائص ووظائف وتعرفنا على كيفية تعريف الخصائص , في هذا الدرس سنتعلم كيفية تعريف وظائف

وسنستخدم مثالنا السابق ونضيف عليه وظيفة الاسم كامل

```
var person = {  
  firstName: "Mohammed",  
  lastName: "Alali",  
  age: 30,  
  fullName : function() {  
    return this.firstName + " " + this.lastName;  
  }  
};
```

اما طريقة استدعاءها كالتالي

```
var person = {  
  firstName: "Mohammed",  
  lastName: "Alali",  
  age: 30,  
  fullName : function() {  
    return this.firstName + " " + this.lastName;  
  }  
};
```

```
document.write(person.fullName());
```

اليوم الثاني والثلاثون واليوم الثالث والثلاثون

نهاية الاسبوع الخامس

استخدم try/catch/finally في القسمة على الصفر
بحيث نتجاوز ظهور Infinity بأي رسالة خطأ تراها مناسبة