

SEPTEMBER 8 – 14, 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 العشرون

الحلقات (Loops)

While

هي طريقة لتكرار جزء من الكود البرمجي ولها أربعة أشكال:
while, do/while, for, and for/in (سنحدث عن for/in لاحقاً)

مبدأ while هو أن في حين أن expression يتحقق فسيتم تنفيذ الـ statement

```
while (expression)
```

```
statement
```

على سبيل المثال أدناه سيطلع الأرقام من 0 إلى 9 , حيث أنه عندما يصل للرقم 10 لن يتحقق expression ولن يتم statement

```
var count =0;
```

```
while (count < 10) {
```

```
document.write(count);
```

```
count++;
```

```
}
```

اليوم الواحد والعشرون

(do/while)

هي أقل استخدام من while والفكرة هي أن تنفيذ statement يتم قبل expression وتفيد أننا ننفذ ثم ندخل في التكرار وصيغتها كالتالي

```
do  
statement  
while (expression);
```

هذا المثال الاول

```
var count =0;  
do {  
document.write(count);  
count++;  
}while (count < 10);
```

وهذا المثال الثاني

```
var count =0;  
do {  
document.write(count);  
//count++;  
}while (count++ < 10);
```

لاحظ أن في المثال الأول تم طباعة الأرقام من 0 إلى 9 أما المثال الثاني من 0 إلى 10

اليوم الثاني والعشرون

(For)

طريقة التكرار في for مختلفة عن while حيث أن تكون صيغتها كالتالي

```
for(initialize ; test ; increment)  
statement
```

هذا المثال الأول

```
for(var count =0 ; count < 10 ; count++){  
document.write(count);  
}
```

كما أننا نستطيع أن نعرف أكثر من initialize و عداد داخل for

```
var i,j,multi;  
for(i = 1, j = 6 ; i < 4 ; i++, j--){  
multi = i * j;  
document.write(multi+"<br>");  
}
```

ايضاً نستطيع تعريف initialize قبل for وتكون بالطريقة التالية

```
var count =0;  
for( ; count < 10 ; count++){  
document.write(count);  
}
```


اليوم الثالث والعشرون

(Jumps)

هذه كلمات محجوزة (break/continue/return/throw/try/catch/finally) تمتاز بخصائص القفز وتجاوز الاوامر ولكل واحدة منها استخدام خاص بها

وقبل هذا لا بد أن نعرف Labeled Statements

أي statement نستطيع أن نضع عليها label من خلال تعريفها التالي

تستخدم التعليمة label لتوفير معرف (id) للتعليمة البرمجية المراد الرجوع اليها من موقع آخر في البرنامج. على سبيل المثال، يمكنك استخدام label لتحديد حلقة، ثم استخدام التعليمة break أو continue للإشارة إلى ما إذا كان البرنامج ينبغي أن يوقف الحلقة أو الاستمرار في تنفيذها.

identifier: statement

على سبيل المثال

```
var i=0;
HowToLabel : while ( i < 3 ) {
    document.write("i= "+i+"<br>");
    i++;
}
```

break

تستخدم التعليمة break لانتهاء الحلقة، أو switch، أو بالتزامن مع التعليمة label.

عند استخدام break من دون label ستقوم بانتهاء الحلقة (while ،do-while ،for أو switch). وانتقال التحكم الى التعليمة البرمجية التالية.

عند استخدام break مع label سيتم انتهاء تنفيذ التعليمة البرمجية المحددة من طرف label

يمكن التعبير عن التعليمة break من خلال طريقتين:

```
break;
break label;
```

جرب المثال التالي والذي سيتوقف رغم أن i إلى 1000

```
for(var i = 0; i < 1000; i++) {
    document.write("Hello<br>");
    if (i==2) break;
}
```

وهنا سنرى بالطريقتين

```
var x = 0;
var z = 0;
```

```
labelCancelLoops: while (true) {  
    document.write("Outer loops(x): " + x + " ");  
    x += 1;  
    z = 1;  
    while (true) {  
        document.write("Inner loops(z): " + z+ "<br>");  
        z += 1;  
        if (z === 3 && x === 3) {  
            break labelCancelLoops;  
        } else if (z === 3) {  
            break;  
        }  
    }  
}
```

التعليمة break تقوم بانهاء الحلقة نهائيا عندما يتم تحقق الشرط المحدد.

اليوم الرابع والعشرون

(Jump- continue)

continue

يمكن استخدام التعليمة continue لاستئناف الحلقة do-while (for، مع labe)، بعد توقفها لغرض معين.

عند استخدام continue من دون label، ستقوم بانتهاء التكرار الحالي للحلقة do-while (while، او for) وتنتقل الى التكرار التالي. على النقيض من التعليمة break، التعليمة continue لا تنهي تنفيذ الحلقة تماماً. عند استخدامها مع الحلقة while فحلقة البرنامج ستعود إلى الشرط. وعند استخدامها مع الحلقة for فحلقة البرنامج ستعود إلى تعليمة الزيادة (increment-expression) عند استخدام continue مع label، سيتم تطبيقها على الحلقة المحددة بواسطة label.

يمكن التعبير عن التعليمة continue من خلال طريقتين:

```
continue;
```

```
continue label;
```

لاحظ هذا المثال (لدينا الحلقة while مع التعليمة continue التي ستنفذ حينما تصبح i تساوي 3. ولدينا n الذي سيحتوي على القيم التالية: 1, 3, 7 و 12)

```
var i = 0;
var n = 0;
while (i < 5) {
  i++;
  if (i == 3) {
    continue;
  }
  n += i;
  document.write("n="+n+" i="+i+"<br>");
}
```

التعليمة label checkiandj تحتوي على التعليمة label checkj. عند مصادفة continue سيقوم البرنامج بانتهاء التكرار الحالي ل checkj ويبدأ التكرار من جديد. وفي كل مرة يتم مصادفة continue، تستمر checkj بالتكرار حتى يعود الشرط ب false. عندما سيتم العودة ب false، ما تبقى من التعليمة checkiandj سيكمل، و checkiandj ستستمر بالتكرار حتى يعود الشرط ب false. عندما سيتم العودة ب false، سينتقل البرنامج الى التعليمة البرمجية التي تلي checkiandj.

```
var i = 0;
var j = 10;
```

checkiandj:

```
while (i < 4) {
```

```
    document.write(i);
```

```
    i += 1;
```

```
    checkj:
```

```
        while (j > 4) {
```

```
            document.write(j);
```

```
            j -= 1;
```

```
            if ((j % 2) == 0) {
```

```
                continue checkj;
```

```
            }
```

```
            document.write(j + " is odd.");
```

```
        }
```

```
        document.write("i = " + i);
```

```
        document.write("j = " + j);
```

```
    }
```

تقوم التعليمة continue بتوقيف الحلقة للتحقق من الشرط المحدد، وعندما يتم تنفيذ الشرط، يتم استئناف الحلقة.

اليوم الخامس والعشرون واليوم السادس والعشرون

نهاية الاسبوع الرابع

اكتب كود for loop داخلها while 2 (الاول label and use break) (والثاني label and use counting)