



## Lesson 9 Do While

نستخدم **do while** إذا كنا نريد تنفيذ الكود عدة مرات ولا نعرف كم عدد الدورات ، لكن يتوقف الدوران عند اكتمال الشرط

الفرق بين **do while** وبين **while** أن **while** تفحص الشرط أولاً ثم تنفذ الكود لكن **do while** يتم تنفيذها مرة واحدة على الأقل لأنها تنفذ الكود مباشرة أول مرة ثم تقوم بفحص الشرط

```
do{
```

```
// هنا يتم تنفيذ الاكواد قبل فحص الشرط
```

```
} while (الشرط);
```

مثال باستخدام ال **while** :

```
#include <stdio.h>
```

```
int main( ) {
```

```
int x = 5 ;
```

```
while ( x > 10){
```

```
printf("%d \n", x );
```

```
}
```



```
}
```

سنجد أن البرنامج لن يطبع شئ حيث أن الشرط لا يتحقق فلن يدخل البرنامج في ال  
while loop

أما عند تطبيق نفس المثال باستخدام **do while** :

```
#include <stdio.h>
```

```
int main( ) {
```

```
int x = 5 ;
```

```
do {
```

```
printf("%d \n", x );
```

```
} while ( x > 10);
```

```
}
```

سنجد أن البرنامج سيقوم بطباعة 5 مرة واحدة ثم لن يطبع شئ حيث أن الشرط لا يتحقق  
فلن يدخل البرنامج في ال loop مرة أخرى

ملحوظة هامة إذا أمعنت النظر في الكود ستجد أن عند استخدام **do while** تم وضع  
علامة ( ; ) بعد جملة ال **while** وهذه هي الطريقة الصحيحة لكتابة ال **do while**

مثال آخر باستخدام ال **do while** :  
برنامج للسوبر ماركت يحسب تكلفة المشتريات



```
#include <stdio.h>
```

```
int main( ) {
```

```
    int item, sum = 0;
```

```
    // هنا قمنا بحجز مكانين في الذاكرة
```

```
    do {
```

```
        printf("Enter Item price : ");
```

```
        scanf("%d", &item);
```

```
        // هنا قام بإدخال سعر السلعة
```

```
        sum += item;
```

```
        // معناها ان يتم اضافة السلعة الى المجموع
```

```
    } while (item != 0);
```

```
    printf("Total = %d \n ", sum);
```

```
}
```

( قم بتجربة الكود بنفسك واضغط هنا )

قمنا بتنفيذ الكود بداخل الاقواس { } ثم فحصنا الشرط في **while**