



Lesson 5 if

مرحبا زملائي المبرمجين!
 إذا كنت تحب البرمجة ، قل "أحب البرمجة!"
 هذا صحيح! يتحدث هذا الدرس عن الأمر **if** وكيف يمكننا استخدامه في الكود!
 إنها بسيطة مثل اسمها. فكر في الأمر مثل المثال السابق ، إذا كنت لا تحب البرمجة ، فلن
 نقول "أحب البرمجة!" لن نقرأ هذا حتى .
 لنقم بإنشاء كود مثل ذلك :

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

```
int main( ) {
    printf("Do you love programming? Enter a number bigger than
4 if yes! ");
    int x;
    scanf("%d", &x);
    if(x > 4)
        printf("YAY! YOU LOVE PROGRAMMING! \n");
    else
        printf("oh no!\n");
}
```

(قم بتجربة الكود بنفسك واضغط هنا)

في السطر الأول ، نوجه المستخدم ببساطة إلى المدخلات التي يحتاجها هذا البرنامج.
 ثم نقوم بحجز المتغير.
 بعد ذلك ، نأخذ مدخلات من المستخدم ونخزنها.
 في جزء الـ " **if** " ، يمكنك رؤيته على أنه نوع من البوابة التي لا تفتح إلا عند استيفاء
 المعايير الموجودة داخل الأقواس (). في هذه الحالة ، إذا كان الرقم الذي تم إدخاله أكبر
 من 4 ، فإن البوابة "تفتح" وأي كود بعد سطر الـ **if** يتم تنفيذه.



لكن في حالة لم يتم تحقق الشرط داخل ال **if** حين ذلك يبحث البرنامج عن الخيارات الأخرى و التي تكون موجودة بعد كلمة **else** و التي بمعنى **وإلا** ، فنجد في المثال السابق إذا لم يتم استيفاء المعايير داخل ال **if** ، ينتقل إلى السطر التالي في الكود ، والذي يكون في هذه الحالة **"else"** فيجد داخله أمر طباعة آخر يقوم بتنفيذه و يطبع **"!oh no"**

هل لاحظت أننا استخدمنا علامة **>** للمقارنة بين قيمتين ؟

نعم فنحن لدينا بعض العلامات التي تسمى بال **operators** كل علامة منهم لها وظيفة معينة فمثلا

- علامتي أكبر من و أصغر من (**<**) , (**>**) يتم إستخدامهم للمقارنة بين الأرقام
 - عندما نقول (**x > y**) معناها إذا كانت **x** أكبر من **y**
 - عندما نقول (**x < y**) معناها إذا كانت **x** أصغر من **y**
- علامة يساوى (**==**) يتم استخدامها للتحقق من ما إذا كانت القيمتان على طرفي العلامة متساويان أم لا .
 - عندما نقول (**x == y**) معناها إذا كانت **x** تساوى **y**
- علامة لا يساوى (**!=**) هي عكس علامة (**==**) يتم استخدامها للتحقق من ما إذا كانت القيمتان على طرفي العلامة غير متساويان أم لا .
 - عندما نقول (**x != y**) معناها إذا كانت **x** لا تساوى **y**

لاحظ عند التحقق من ما إذا كانت القيمتان على طرفي العلامة متساويان أم لا إستخدمنا (**==**) و ليس علامة (**=**) واحدة لأن علامة **=** الواحدة هكذا معناها ضع القيمة التي على يمين العلامة داخل المتغير الذي على يسار العلامة (مثلما عرفنا في درس ال **variables**)

لذلك فإن **x = y** معناها ضع نفس قيمة المتغير **y** داخل المتغير **x** فلو كانت ال **y** تساوى 5 إذا ال **x** أصبحت تساوى 5
أما **x == y** كما ذكرنا معناها إذا كانت قيمة **x** تساوى قيمة **y**

أمر else :

لنفترض أنك تريد شراء البقالة. أولويتك هي شراء الأفوكادو ، لكن إذا نفذ المخزون ، ستشتري الخوخ بدلاً منه. لذلك ، إذا كان هناك أفوكادو ، قم بشرائها ، وإلا ، اشترِ الخوخ.



لذا فإن كلمة "else" في الكود السابق تعني أنه إذا لم يكن الرقم الذي تم إدخاله أكبر من 4 ، أي أنه لم يتم استيفاء المعايير ، فسيتم تنفيذ الكود الموجود داخل الـ else و سيطبع " Oh no! "

ملحوظة : ليس بالضرورة وجود else بعد if، لكن تكتبها إذا كنت في حاجة لذلك في البرنامج الخاص بك.

أمر else if :

لنفترض أنك تريد شراء البقالة و ذهبت إلى السوبر ماركت، فقلت أنه لو معك 50 جنية أو أكثر ستشتري اللبن و البيض ، أما لو معك 20 جنية أو أكثر ستشتري اللبن فقط و لو كان معك 10 جنيهات أو أقل ستشتري البيض فقط .
في هذا المثال يوجد عدة احتمالات للقرار الذي سوف تتخذه و لكل احتمال شرط معين، كيف نقوم بذلك في البرنامج :

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

```
int main( ) {
    printf("How much money do you have ? ");
    int x;
    scanf("%d", &x);
    if(x >= 50)
        printf(" Buy milk and eggs \n");
    else if (x >= 20)
        printf(" Buy milk only \n");
    else if (x <= 10)
        printf(" Buy eggs only \n");
}
```



نفهم من المثال السابق أن عندما يكون هناك أكثر من شرط و على أساسهم سيقوم البرنامج بالتصرف بطريقة معينة، نستخدم `else if` مع `if` الأساسية و سيتم تنفيذ الأمر الذى تحقق الشرط الخاص به.

(**لأقوياء الملاحظة**) ربما لاحظت أننا قمنا بإستخدام علامتان مختلفتان معا (`>=`) و (`<=`) فما معنى ذلك ؟
 علامة أكبر من أو يساوى (`>=`) معناها هل القيمة التى على اليسار أكبر من أو تساوى القيمة على اليمين ؟
 علامة أصغر من أو يساوى (`<=`) معناها هل القيمة التى على اليسار أصغر من أو تساوى القيمة على اليمين ؟

يمكن أن تسأل نفسك ماذا لو أدخل شخص ما رقم 70، فى هذه الحالة الشرط الأول و الثانى متحققين، حيث أن ال 70 أكبر من ال 50 و فى نفس الوقت أكبر من 20، إذا يجب تنفيذ الأوامر التى بداخلهما و يقوم البرنامج بطباعة الجملتين معا

Buy milk and eggs

Buy milk only

لكن فى حقيقة الأمر سيطبع البرنامج الجملة الأولى فقط

Buy milk and eggs

و ذلك لأن ما إن تحقق شرط ما فى الشروط الموجودة لن ينظر البرنامج إلى باقى الشروط طالما تم استخدام مصطلح `else`، بمعنى أنه كان من الممكن أن يتم طباعة الجملتين هكذا إذا كان البرنامج على هذا الشكل :

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

```
int main( ) {  
    printf("How much money do you have ? ");
```



```
int x;
scanf("%d", &x);
if(x >= 50)
    printf(" Buy milk and eggs \n");
if (x >= 20)
    printf(" Buy milk only \n");
if (x <= 10)
    printf(" Buy eggs only \n");
}
```

فى هذه الحالة كل if مستقلة بذاتها، فكلما يتحقق شرط ما سوف ينفذ البرنامج ما بداخل ال if و فى حالة تم إدخال رقم مثل 70 سيتم طباعة الجملة الأولى و الثانية :

Buy milk and eggs

Buy milk only

لذلك أمر else و else if مفيدان للغاية فى تسهيل عمل البرنامج و تحقيق شروطه بسهولة و دقة.

آخر أمر عليك معرفته حالياً أنه ليس دائماً سيكون هناك أمر واحد فقط يتم تنفيذه داخل ال if أو ال else و ال else if ، إذا كان هناك أكثر من أمر يتم تنفيذه نستخدم الأقواس المموجة { } لنكتب بداخلها الأوامر مثل :

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

```
int main( ) {
    printf("How much money do you have ? ");
    int x;
    scanf("%d", &x);
    if(x >= 50) {
```



```
printf(" Buy milk and eggs \n");
printf(" Get extra sweets \n");
printf(" Go back home \n");
}
else if (x >= 20){
    printf(" Buy milk only \n");
    printf(" put gas in the car \n");
}
else if (x <= 10)
    printf(" Buy eggs only \n");
}
```

يمكنك استخدام الأقواس { } دائما حتى لو كان هناك أمر واحد للتنفيذ و لكنك غير ملزم بذلك، و لكنك ملزم بذلك إذا كان هناك أكثر من أمر للتنفيذ

الأخطاء الشائعة في الكود:

- استخدام = بدلا من ==
- كتابة علامة المقارنة بطريقة خاطئة مثل (<=) بدل من (>=)
- نسيان استخدام الأقواس { } في حالة وجود أكثر من أمر

الآن أنت تعرف إذا كان لديك شرط ، ماذا ستفعل !