



lesson 41 Enum

إذا اردنا عمل برنامج لترتيب المواعيد ، يوم السبت لديك ميعاد صباحاً أو مساءً ، وباقي أيام الأسبوع يوجد مواعيد بها ، فى حالة أيام الأسبوع فهى ليست أعداد `int` ، وإنما كلمات فهنا إذا أردنا عمل `for` لتمر على الايام فسيكون الأمر صعباً لأنها ليست أعداد ، لذلك لعمل برنامج بسهولة نريد أن يرى الكمبيوتر يوم السبت أنه رقم 1 والأحد أنه رقم 2 والاثنين رقم 3

بالتالى نريده أن يرى الكلمات على أنها أعداد
هناك شئ يسمى `enum` و هو `data type` يستختم بشكل رئيسى لتعيين أسماء لقيم عددية من نوع `int` ، و ذلك لسهولة قراءة الكود و التعديل عليه .

سوف نقوم بعمله كالتالى :

```
int main( ) {
    enum week { sat, sun, mon, tue, wed, thu, fri };
    هنا أصبحت sat=0 لأنه يبدأ الترتيب الطبيعي من الصفر//
    printf("%d\n", sat);
    printf("%d\n", fri);
}
```

output:

0

6

(قم بتجربة الكود بنفسك واضغط هنا)

مثال آخر لل `enum` :

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

```
int main()
{
    enum year{Jan, Feb, Mar, Apr, May, Jun, Jul,
```



Aug, Sep, Oct, Nov, Dec};

```
int i;
for (i=Jan; i<=Dec; i++)
    printf("%d ", i);

return 0;
}
```

output:

0 1 2 3 4 5 6 7 8 9 10 11

هنا يبدأ العد من صفر ولكن يمكننا ان نغير العد

```
int main( ) {
    enum week { sat = 5, sun, mon=11, tue, wed, thu, fri };
    printf("%d\n", sat);
    printf("%d\n", fri);
}
```

output:

5
15

(قم بتجربة الكود بنفسك واضغط هنا)

```
int main( ) {
    enum { sat , sun, mon=11, tue, wed, thu, fri };
    printf("%d\n", sat);
}
```



```
printf("%d\n", tue);  
}
```

output:

0
12

(قم بتجربة الكود بنفسك واضغط هنا)

عند إعطاء قيمة معينة لاسم داخل ال `enum` فإن باقى الأسماء التي تأتي بعده تزداد ب 1 عن القيمة التي تم إعطاؤها للاسم.

باستخدام **enum** يمكننى عمل أرقام متسلسلة لأى اسماء !