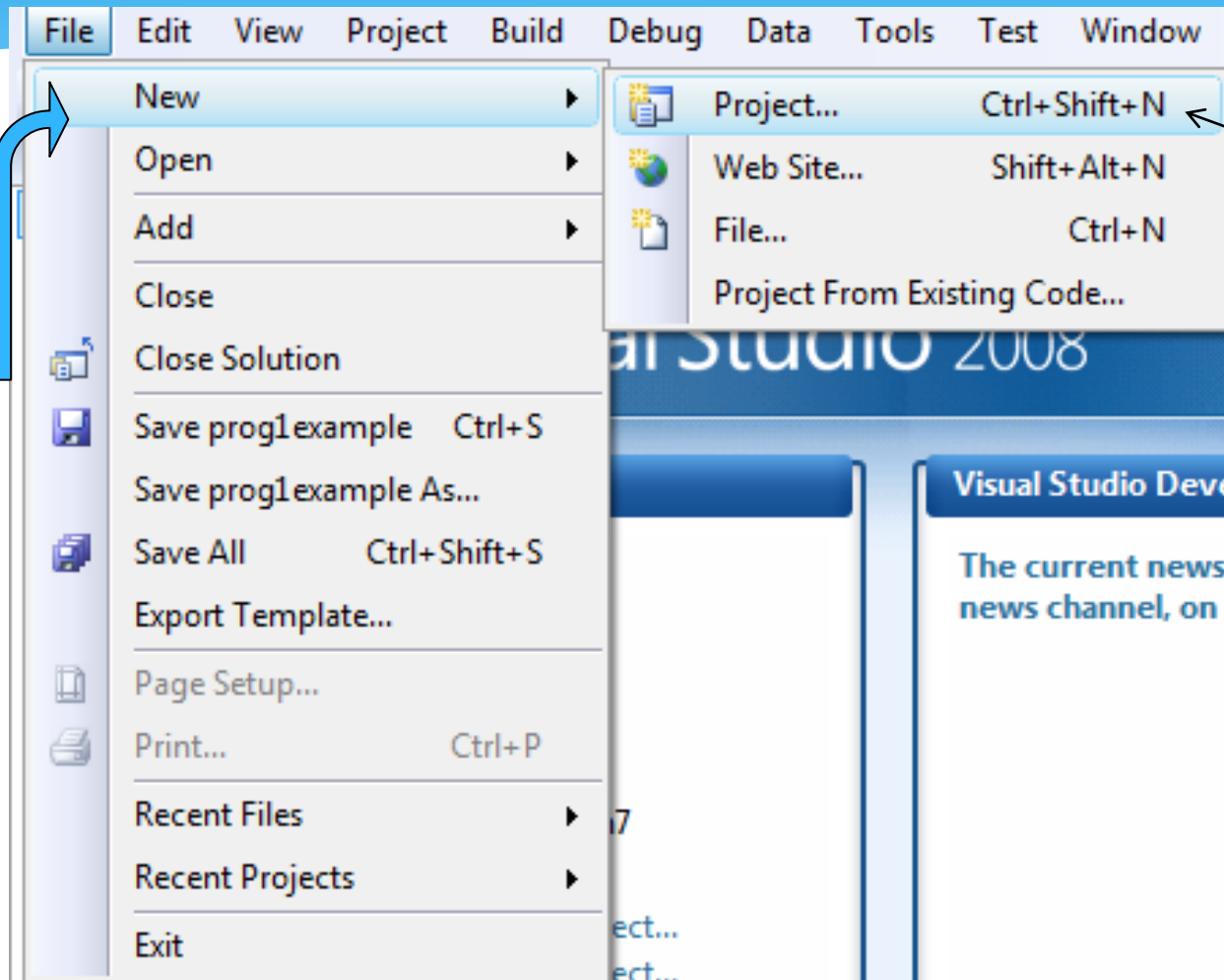
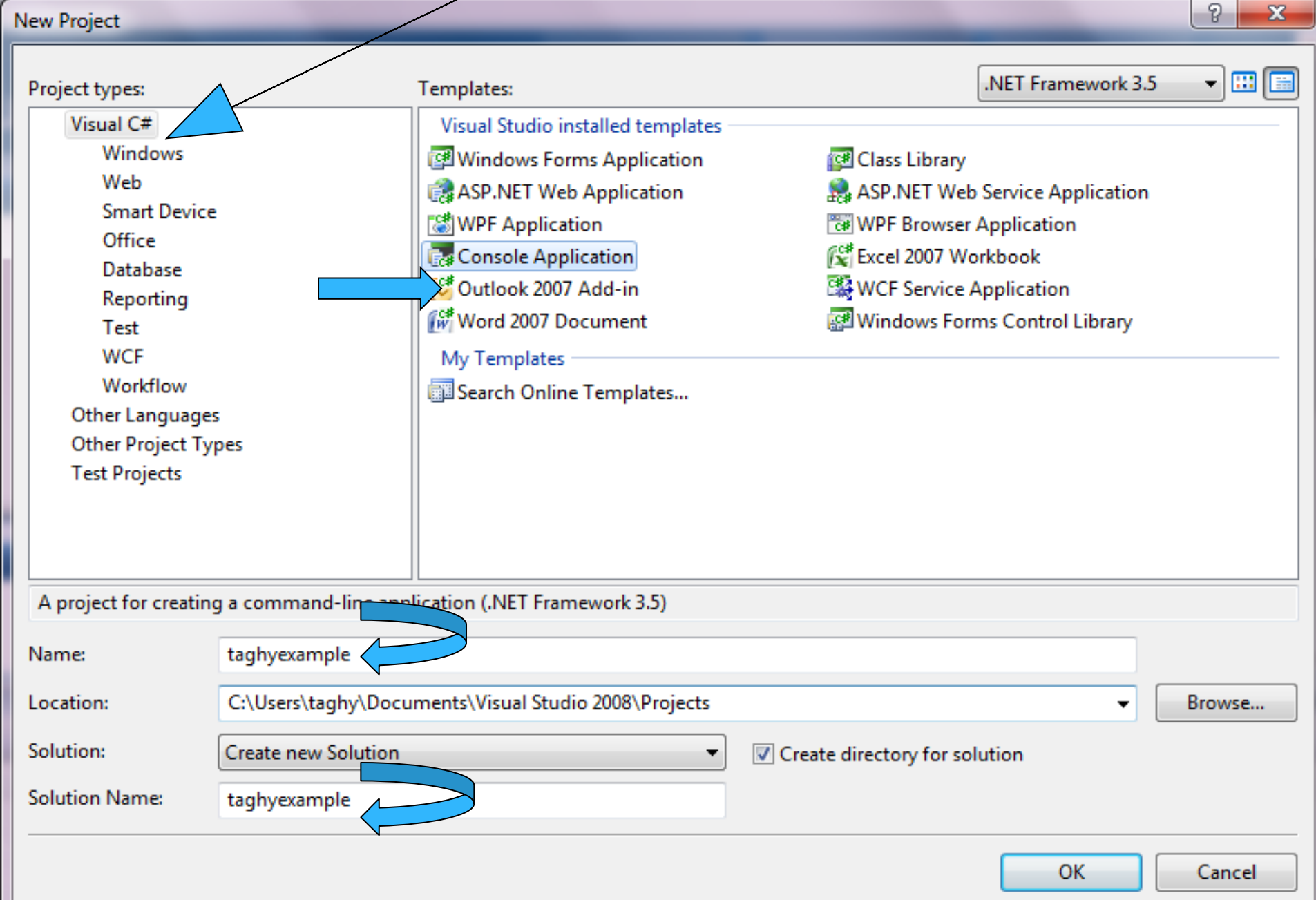


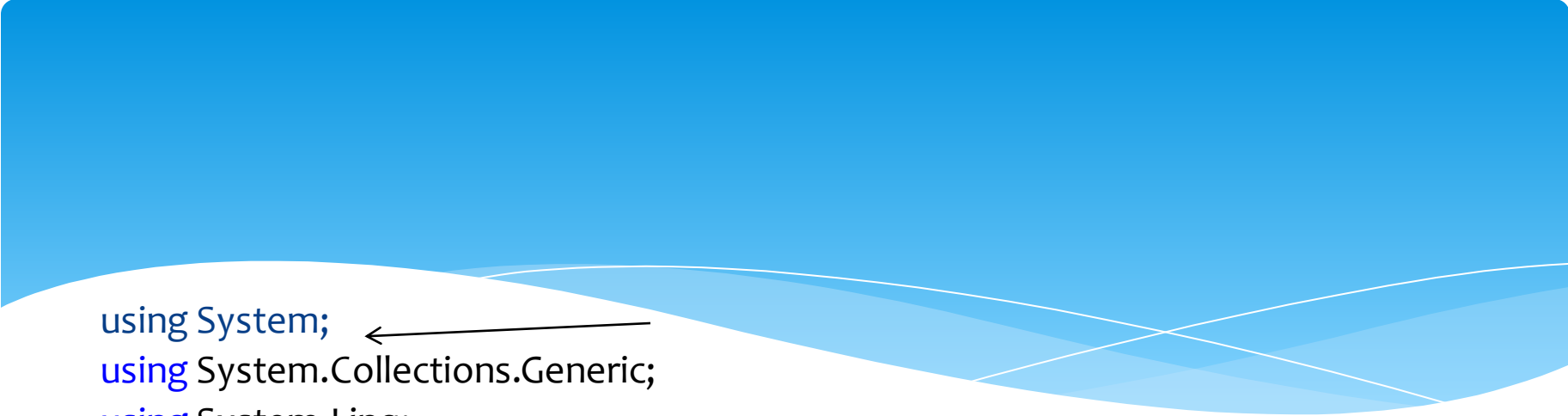
محاضرات برمجة ١

Eng.Taghreed Harfoush

الدخول الى بيئة C#







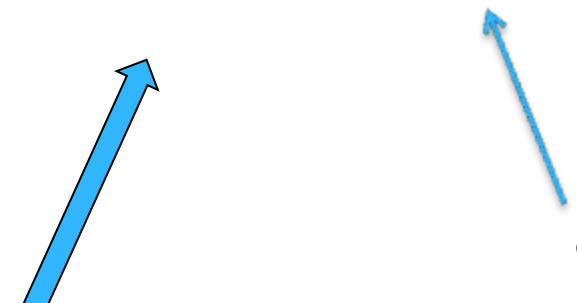
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
```

```
namespace taghyexample
{
    class Program
    {
        static void Main(string[] args)
        {
        }
    }
}
```

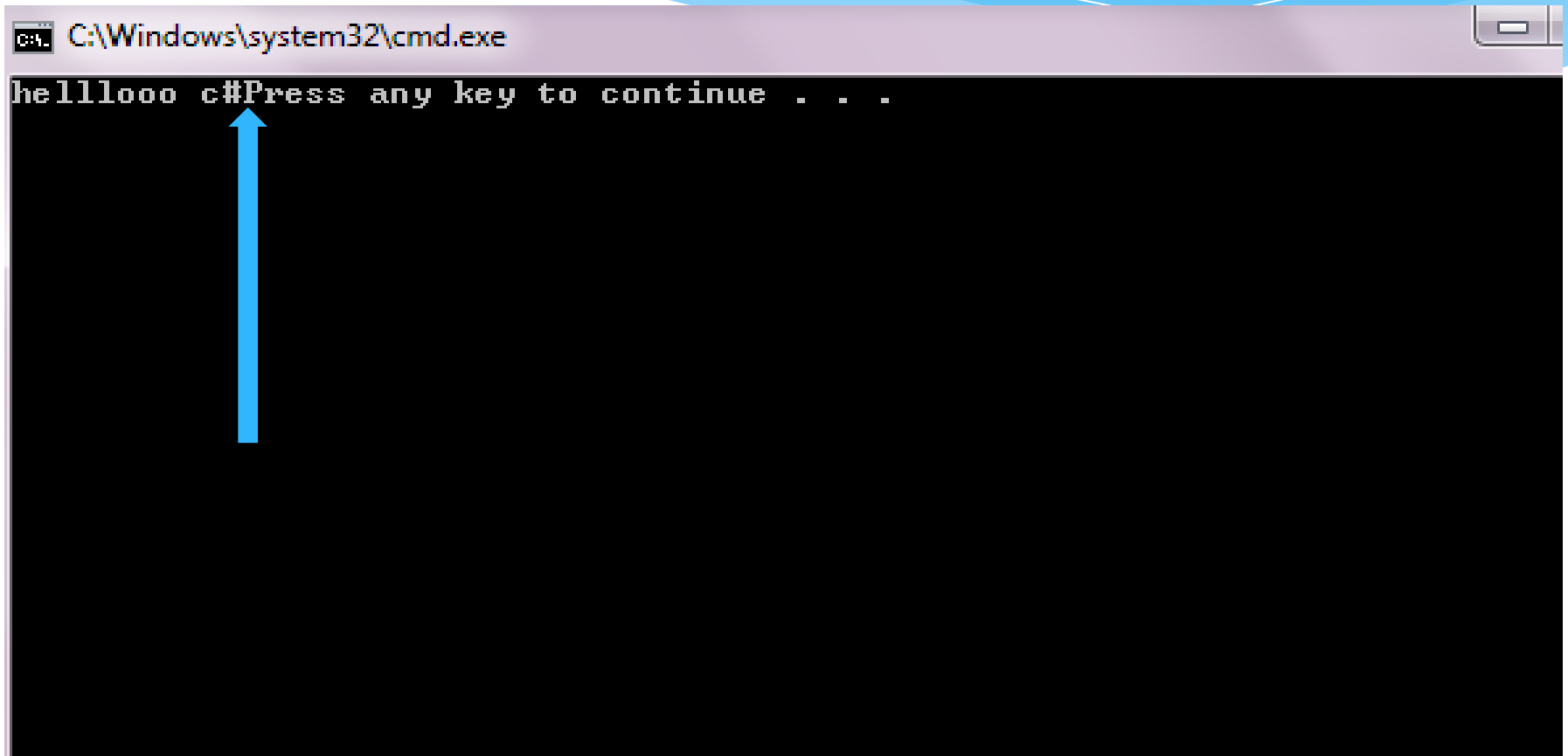
اول برنامج في C#

```
using System.Collections.Generic;  
using System.Linq;  
using System.Text;
```

```
namespace taghyexample  
{  
    class Program  
    {  
        static void Main(string[] args)  
        {Console.WriteLine("helllooo c#");  
        }  
    }  
}
```



يطبع النص ويبقى مؤشر الكتابة على نفس السطر

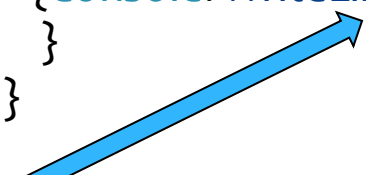


```
C:\Windows\system32\cmd.exe
helllooo c#Press any key to continue . . .
```

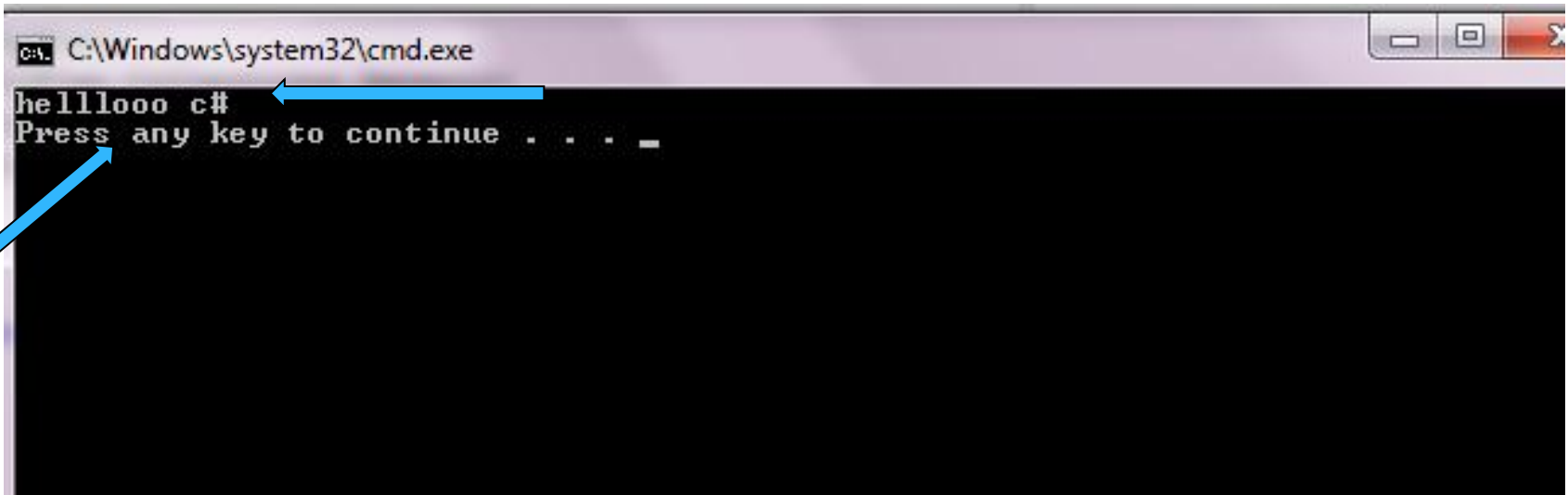
WriteLine

```
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace taghyexample
{
    class Program
    {
        static void Main(string[] args)
        { Console.WriteLine("helllooo c#"); }
    }
}
```



يطبع النص وينزل مؤشر الكتابة الى سطر جديد



A screenshot of a Windows command prompt window. The title bar shows the path `C:\Windows\system32\cmd.exe`. The command prompt displays the output of a C# program: `helllooo c#` followed by `Press any key to continue _`. Two blue arrows are present: one points to the end of the first line of output, and another points to the start of the second line of output.

```
C:\Windows\system32\cmd.exe
helllooo c#
Press any key to continue . . . . _
```

انماط المعطيات (Data types)

C# type	.Net type	Size in bytes	Description
Byte	Byte	1	May contain integers from 0-255
int	Int32	4	Ranges from -2,147,483,648 to 2,147,483,647
short	Int16	2	Ranges from -32,768 to 32,767
long	Int64	8	Unsigned, ranges from 0 to 18,446,744,073,709,551,615
float	Single	4	Ranges from $\pm 1.5 \times 10^{-45}$ to $\pm 3.4 \times 10^{38}$ with 7 digits precision. Requires the suffix 'f' or 'F'
double	Double	8	Ranges from $\pm 5.0 \times 10^{-324}$ to $\pm 1.7 \times 10^{308}$ with 15-16 Digits Precision
char	Char	2	Contains any single Unicode character enclosed in single quotation mark such as 'c'

decimal	Decimal	12	Ranges from 1.0×10^{-28} to 7.9×10^{28} with 28-29 digits precision. Requires the suffix 'm' or 'M'
bool	Boolean	1	Contains either true or false
uint	UInt32	4	Unsigned, ranges from 0 to 4,294,967,295

المتحولات في C#

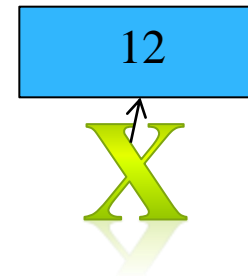
- * يملك المتحول محددتين في الذاكرة (عنوان، قيمة)
- * اسم المتحول هو عنوانه
- * القيمة المسندة اليه هي قيمة المتحول في الذاكرة
- * اما انها ثابتة او متغيرة
- * ثابتة أي معروفة من قبل المبرمج
- * متغيرة تحتاج للمستخدم لتحديد لها
- * التصريح عن متحول أي حجز مكان في الذاكرة باسم هذا المتحول تمهيدا لاعطاؤه قيمة ما.

الصيغة القواعدية للتصريح عن متحول

- a) <datatype> varname;
- b) <datatype> varname=value;

- * int x=12;
- * double y=4.9;
- * char z='b';

* مثال:



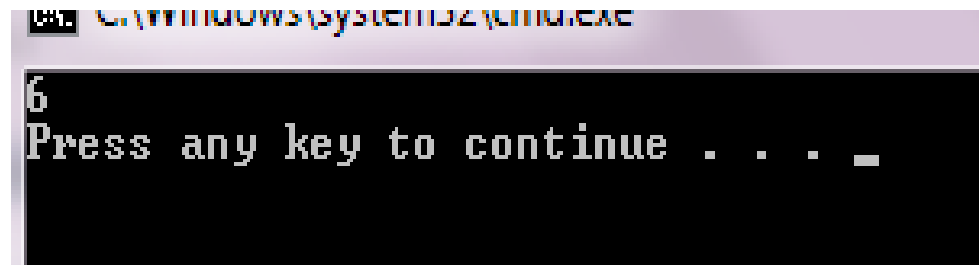
قواعد عامة للتصريح عن متحولات

<pre>int @x; false ! @ # \$ % ^ & * (+ > < ? ~</pre>	لا يجوز ان يبدأ اسم المتحول برمز
<pre>Int y h; f</pre>	لا يجوز ان يحتوي اسم المتحول على فراغات
<pre>Char 7x; f double 9s; f</pre>	لا يجوز ان يبدأ اسم المتحول برقم
<pre>int x; double x;</pre>	لا يجوز ان نصرح عن المتحول بنمطن مختلفين من البيانات في نفس البرنامج
<pre>double t='c'; f int b=5.7; f</pre>	لا يجوز ان نسند قيمة لمتحول مختلفة عن نمطه
<pre>int x; int y=8; Z=x+y; Console.WriteLine(z); //error</pre>	لا يجوز ان نستخدم متحول في عملية ما (حسابية ام منطقية) بدون معرفة قيمته قبل ذلك
<pre>int double ; f</pre>	لا يجوز ان يكون اسم المتحول هو احد الكلمات المحجوزة لصالح اللغة
<pre>int b; int B;</pre>	يجوز ان يكون لدينا متحولين بنفس الحرف الانكليزي لكن مع فرق الكبير والصغير لان هذه اللغة حساسة لحالة الاحرف

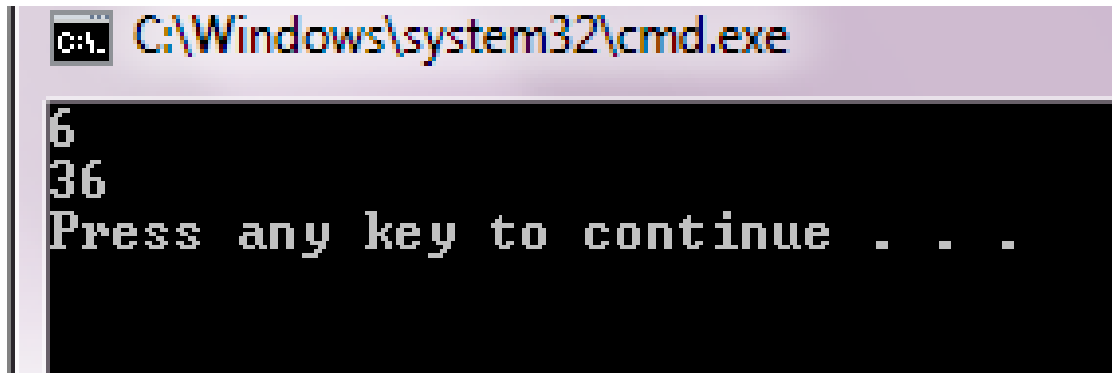
ملاحظة

عند اعطاء قيمتين متتاليتين لمتحول فان القيمة الجدد تحل مكان القيمة القديمة في الذاكرة.
مثال:

```
namespace taghyexample
{
    class Program
    {
        static void Main(string[] args)
        {
            int x = 6;
            Console.WriteLine(x);
        }
    }
}
```



```
namespace taghyexample
{
    class Program
    {
        static void Main(string[] args)
        {
            int x = 6;
            Console.WriteLine(x);
            x = 6*6;
            Console.WriteLine(x);
        }
    }
}
```



A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Windows\system32\cmd.exe'. The command prompt displays the output of the program: the number '6' on the first line, '36' on the second line, and the text 'Press any key to continue . . .' on the third line.


```
namespace taghyexample
{
    class Program
    {
        static void Main(string[] args)
        {
            int x = 6;
            Console.WriteLine("x="+x);
            x = 6*6;
            Console.WriteLine("x=" + x);
        }
    }
}
```

الخرج هو:

C:\Windows\system32\cmd.exe

x=6

x=36

Press any key to continue . . . _

الثوابت

- * هي متحولات تأخذ قيمة من قبل المبرمج ضمن شفرة البرنامج ولا يمكن تغييرها اثناء تنفيذ البرنامج ولا يتدخل المستخدم في تحديد قيمتها
- * يتم التصريح عن الثوابت بالصيغة القواعدية التالية مع استخدام الكلمة المحجوزة const:

* const datatype varname=value;

* مثال:

* `const double PP=3.14;`

العمليات الحسابية في C#

الرمز	العملية
+	جمع
-	طرح
*	ضرب
/	قسمة
%	باق القسمة الصحيحة
++	الزيادة بمقدار ١
--	النقصان بمقدار ١

العمليات المنطقية

الرمز	العملية
&&	AND
	OR
!	NOT

عمليات الاسناد

الرمز	العملية
$+=$	اسناد مع جمع
$-=$	اسناد مع طرح
$*=$	اسناد مع ضرب
$/=$	اسناد مع قسمة
$\%=$	اسناد مع باق القسمة الصحيحة
$=$	الاسناد البسيط
$==$	المساواة المنطقية

امثلة على العمليات السابقة

- * `Console.WriteLine(5+4);`
- * `Console.WriteLine(30*7);`
- * `Console.WriteLine(35/7);`
- * `Console.WriteLine(26%6);`
- * `Console.WriteLine(4++);`
- * `Console.WriteLine(9--);`

ملاحظات

- * لا يتم التصريح عن متحول في C# الا مرة واحدة ضمن البرنامج الواحد.
- * يوضع النص بين اشارات تنصيص مزدوجة "....."
- * يوضع المحرف بين اشارات تنصيص مفردة '.....'
- * في تعليمة الطباعة ما يكتب بين "....." يظهر كما هو مكتوب ضمن الشيفرة على شاشة التنفيذ.
- * لتنفيذ البرنامج يوجد اكثر من طريقة F5, CTRL+F5 من القوائم
Debug