



جامعة حماه  
المعهد التقني للحاسوب  
السنة الأولى

محاضرة 1

برمجة ٢

عملي

قسم البرمجيات

إعداد:

م. أريج فياض

م. رفا البنات

## ١. أنماط المعطيات في c# وتسمى built in:

int	عدد طبيعي
char	محرف
float	عدد صحيح ذو فاصلة عشرية عائمة

### الجدول 3.1

هناك تنويعات لما ذكر سابقاً، مثل:

unsigned int	عدد طبيعي بدون إشارة
short int	عدد طبيعي صغير
long int	عدد طبيعي كبير
double	عدد صحيح ذو فاصلة عشرية مضاعف الدقة

## ٢. تعليمات الكتابة والقراءة:

- تعليمة الكتابة والاظهار على الشاشة السوداء التي تعرف ب console.
- نستخدم العبارة البرمجية التالية: `Console.WriteLine();` حيث دالة `()` `writeline` من الصف `console` تقوم بأخذ السلسلة النصية الممررة لها بين قوسين وطباعتها على الشاشة .
- كما يمكن تمرير أي متغير وستقوم بطباعة قيمته .
- في حال أردنا كتابة العبارة التالية على الشاشة نقوم بكتابة مايلي:

```

ConsoleApplication1.Program
namespace ConsoleApplication1
{
    class Program
    {
        public static void Main(string[] args)
        {
            Console.WriteLine("hello from c# course");
            Console.ReadKey();
        }
    }
}

```

hello from c# course

- في حال أردنا طباعة جملة نصية وبعدها قيمة متحول ما نستخدم ما يعرف بال concatenation أي الربط وذلك باستخدام إشارة + بالطريقة التالية:

```

class Program
{
    public static void Main(string[] args)
    {
        string username = "ali";
        Console.WriteLine("hello from c# course " + username);
        Console.ReadKey();
    }
}

```

file:///C:/Users/hp/Documents/Visual Studio 2012/Proje  
hello from c# course ali

**ملاحظة :** دالة ReadKey() لكي لا تُغلق الشاشة السوداء بسرعة ،فهي تنتظر الضغط على أي مفتاح لتغلق الشاشة.

- في حال أراد المستخدم أن يدخل قيمة ما نستخدم تعليمة القراءة Console.ReadLine كما يلي:

```

namespace ConsoleApplication1
{
    class Program
    {
        public static void Main(string[] args)
        {
            string username;

            Console.WriteLine("please enter your name.... ");

            username = Console.ReadLine();

            Console.WriteLine("hello " + username + "...");

            Console.ReadKey();
        }
    }
}

```

please enter your name....  
sami  
hello sami...

Console.ReadLine(); تقرأ من المستخدم دائما سلسلة نصية أي من نمط String وفي حال أدخل المستخدم رقم نقوم بعملية تحويل من string الى رقم بطريقتين :

- باستخدام دوال التحويل converting method:

```

namespace ConsoleApplication1
{
    class Program
    {
        public static void Main(string[] args)
        {
            int number;

            Console.WriteLine("please enter your number.... ");

            number = Convert.ToInt32(Console.ReadLine());

            Console.WriteLine("hello your number is " + number);

            Console.ReadKey();
        }
    }
}

```

please enter your number....  
256896  
hello your number is 256896

- او باستخدام دالة parse الخاصة بالنمط المراد التحويل إليه:  
على سبيل المثال اذا أردنا التحويل من string المعادة من شاشة ال console الى float نكتب:  
`float number = float.Parse(Console.ReadLine());`

### ٣. أنواع العمليات :

١. **عمليات رياضية:** وهي التي نستخدم فيها الرموز +, -, \*, /, %, وهي التي نستخدم فيها الرموز +, -, \*, /, %, وهي التي نستخدم فيها الرموز +, -, \*, /, %.
  ٢. **عمليات الاسناد:** وهي التي تخزن قيمة في متحول `x=45;`
  ٣. **عمليات المقارنة:** ونستخدم فيها <, >, <=, >=, ==, !=.
  ٤. **عمليات شرطية:** ويقصد بها If statement
- مثال عنها إذا أردنا وضع شرط معين على قيمة مدخلة من المستخدم ان تكون أصغر من ١٠  
فإذا تحقق الشرط يطبع عبارة ما وفي حال عدم التحقق ينتقل التنفيذ الى عبارة else وينفذ ماتحويه  
من تعليمات .

```

Console.WriteLine("please enter your number.... ");
number = Convert.ToInt32(Console.ReadLine());
if (number < 10)
{
    Console.WriteLine("hello your number is " + number);
}
else
{
    Console.WriteLine("please enter your number less than 10 ....");
}
Console.ReadKey();

```

```

file:///C:/Users/hp/Documents/Visual Studio 2012/Pro...
please enter your number....
5
hello your number is 5

```

- في حال ادخل المستخدم رقم اكبر من ١٠ نلاحظ مايلي:

```

please enter your number....
15
please enter your number less than 10 ....

```

### ٤. الحلقات :

حيث نستطيع من خلالها تنفيذ عبارة أو عدة عبارات برمجية لعدد من المرات. تدعم سي شارب مثل باقي لغات البرمجة نوعين من الحلقات التكرارية من حيث عدد التكرار، فهناك الحلقات ذات العدد المحدد من المرات (for حلقة) والتي نعلم فيها عدد مرات التكرار بشكل مسبق، والحلقات ذات العدد

غير المحدد من المرات (حلقة do-while وحلقة while) التي يكون فيها عدد مرات التكرار غير مُحددًا طالما الشرط لديها محقق.

- لنكتب برنامج يجمع الاعداد من ١ الى ١٠ بالطريقة الأولى while:

```
public static void Main(string[] args)
{
    int n=10;
    int i=1;
    int sum=0;

    while (i <= n)
    {
        sum += i;

        i++;
    }

    Console.WriteLine("sum = {0}", sum);
    Console.ReadKey();
}
```

- وبطريقة ثانية باستخدام for :

```
public static void Main(string[] args)
{
    int n=10;
    int i;
    int sum=0;

    for (i = 1; i < n; i++)
    {
        sum += i;

        i++;
    }

    Console.WriteLine("sum = {0}", sum);
    Console.ReadKey();
}
```

- والطريقة الثالثة do while :

```
public static void Main(string[] args)
{
    int n=10;
    int i=1;
    int sum=0;

    do
    {
        sum += i;

        i++;
    } while (i <= n);

    Console.WriteLine("sum = {0}", sum);
    Console.ReadKey();
}
```

## ٥. المصفوفات:

هي مجموعة من المتغيرات التي تنتمي إلى نفس النوع .

نقول عن مصفوفة بأنها رقمية عندما تحتوي عناصرها على أرقام ، ونقول عن مصفوفة بأنها نصية عندما تحتوي عناصرها على سلسلة نصية .

للمصفوفات عدة أنواع وهي:

### (١) المصفوفات أحادية البعد:

يكون التصريح عنها بالشكل التالي : `Type [] array = new Type [n];`

حيث n عدد عناصر المصفوفة .

مثال: مصفوفة رقمية مكونة من ثلاث عناصر:

```
int[] array = new int[3] { 2, 3, 4 };
Console.WriteLine("the first number:" + array [0]);
Console.WriteLine("the second number:" + array[1]);
Console.WriteLine("the third number:" + array[2]);
Console.ReadKey();
```

```
the first number:2
the second number:3
the third number:4
```

كما يمكننا عرض عناصر المصفوفة باستخدام حلقة for :

```
int[] array = new int[3] { 2, 3, 4 };
for (int i = 0; i < 3; i++)
    Console.WriteLine(array[i]);
Console.ReadKey();
```

## (٢) المصفوفات ثنائية البعد:

يكون التصريح عنها بالشكل التالي : `Type [,] array = new Type [n,m];`

مثال: ليكن لدينا مصفوفة رقمية مكونة من سطرين وثلاثة أعمدة ، فيكون التعبير الرياضي المعبر عنها:

(0,0)	(0,1)	(0,2)
(1,0)	(1,1)	(1,2)

ويكون عدد العناصر هو  $m*n$

نقوم بالتصريح عن المصفوفة الثنائية ، ثم ندخل العناصر بطريقتين كالتالي.

ويمكننا اظهار جميع العناصر باستخدام foreach

```
int[,] array = new int[2, 3];
array[0, 0] = 10;
array[0, 1] = 20;
array[0, 2] = 30;
array[1, 0] = 40;
array[1, 1] = 50;
array[1, 2] = 60;
foreach (int n in array)
    Console.WriteLine(n);
Console.ReadKey();
```

```
int[,] array = new int[2, 3] { { 10, 20, 30 },
{ 40, 50, 60 } };
Console.WriteLine(array[0,1]);
Console.ReadKey();
```

```
Console.WriteLine(array[0,1]);
Console.ReadKey();
```

## أمثلة:

- اكتب برنامج لادخال أسماء ثلاث أشخاص إلى مصفوفة أحادية ثم قم باستبدال الاسم الثاني باسم جديد.

```
string[] names = new string[] { "ahmad", " ali", " khaled" };
names[1] = "mohamad";
Console.WriteLine("the new name is: " + names[1]);
Console.ReadKey();
```

- اكتب برنامج لإدخال n سطر و m عمود من المستخدم لمصفوفة ثنائية ، ثم حساب أكبر عدد في المصفوفة وأصغر عدد في المصفوفة وحساب المتوسط لقيم العناصر.

```

Console.WriteLine("enter n");
int n = int.Parse(Console.ReadLine());
Console.WriteLine("enter m");
int m = int.Parse(Console.ReadLine());

int[,] X = new int[n, m];

for (int i = 0; i < n; i++)
{
    for (int j = 0; j < m; j++)
    {
        X[i, j] = int.Parse(Console.ReadLine());
    }
}

for (int i = 0; i < n; i++)
{
    for (int j = 0; j < m; j++)
    {
        Console.Write(X[i, j] + " ");
        Console.WriteLine();
    }
}

int max = X[0, 0];
int min = X[0, 0];
int sum = 0;
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < m; j++)
    {
        if (X[i, j] > max)
        {
            max = X[i, j];
        }
    }
}

Console.WriteLine("The Max is " + max);
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < m; j++)
    {
        if (X[i, j] < min)
        {
            min = X[i, j];
        }
    }
}

Console.WriteLine("The min is " + min);

for (int i = 0; i < n; i++)
{
    for (int j = 0; j < m; j++)
    {
        sum = sum + X[i, j];
    }
}

Console.WriteLine("The average is " + sum/n*m);

// for
Console.ReadKey();

```



- اكتب برنامج لحساب مجموع القطر الثانوي لمصفوفة ثنائية:

```
int sum2=0;
int j1 = m - 1;

for (int i = 0; i < n; i++)
{

    sum2 = sum2 + X[i, j1];
    j1--;

}

Console.WriteLine("The secondry daigonal is " + sum2);
```

- اكتب برنامج للبحث عن عنصر ضمن مصفوفة وفي حال العثور عليه يظهر لنا true والا يظهر لنا false

```
Console.WriteLine("enter the number for search");
int number = int.Parse(Console.ReadLine());
bool result = true;
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < m; j++)

        if (X[i, j] == number)
        {
            result = true;
        }

        else
        { result = false; }

}

Console.WriteLine("the number is " + result);
```

وظيفة 

اكتب برنامج لطباعة العناصر الموجودة فوق القطر الرئيسي في مصفوفة ثنائية.

اكتب برنامج لطباعة العناصر الموجودة تحت القطر الرئيسي في مصفوفة ثنائية.