

Array

1. One Dimension Array

```
<ArrayType> [ ] Arrayname = new <arrayType>  
[<size>];
```

Ex:

```
int [ ] x =new int [3];  
int [ ] x={9,7,6};  
int [ ] x =new int [5] {3,4,5,6,7};
```

```
int [ ] x =new int [10] {3,4,5,6,7};
```

×

EX:

```
const arraysize=5  
int [ ] myaarray=new int [arraysize] {6,8,7,8,8};
```

```
int [ ] x =new int [3];  
x[0]=21;  
x[1]=63;  
x[2]=7;
```

```
float [ ] f= new [2.5f, 5.5f, 3.5f]
```

```
char [ ] ch ={'a', 'b', 'c'}
```

لايجاد حجم المصفوفة :

```
Console.WriteLine(<array name>.Length) ;
```

Ex:

```
static void Main(string[] args)
{
    string[] friendname = { "huda", "saja", "noor" };
    int i;
    Console.WriteLine ("Here are {0} of my friend:" ,
friendname.Length );
    for (i =0; i <friendname.Length ;i++)
    {
        Console.WriteLine (friendname [i]);
    }
}
```

Ex:

```
static void Main(string[] args)
{
    int[] x = new int[5];
    for (int i = 0; i < 5; i++)
    {
        Console.Write("Enter value at location {0} in array :", i + 1);
        x[i] = int.Parse(Console.ReadLine());
    }
    Console.WriteLine();
    for (int j=0; j<5 ; j++)
        Console.WriteLine ("Value of location {0} is :{1}", j+1,x[j]);
}
```

2. Two dimensional Array

- Rectangular array
- Jagged array

Rectangular array

_Type [,] name = new Type [size,size]

Ex:

```
int [ , ] x=new int [2,3] { {1,4,2},{52,12,9}};
```

```
int [ , ] x=new int [2,3];  
x[0,0]=14;
```

```
double [ , ] distances =new double [4,4]
```

اعطاء قيم مباشرة

```
-double [ , ] distances ={{0,300,500,880},  
                           {200,800,70,400},  
                           {800,400,120,490},  
                           {300,620,890,590}};
```

```
-For (i=0; i<3; i++)  
    For (j=0; j<3; j++)  
        Console.WriteLine(x[i][j]);
```

foreach loops:

```
foreach (<basetype> <name> in <array>)  
{  
    // can use <name> for each element  
}
```

EX:

```
static void Main(string[] args)  
{  
    string[] friendnames ={ "huda", "saja", "noor" };  
    int i;
```

```

        Console.WriteLine("Here are {0} of my friend:",
friendnames.Length);
        foreach (string friend in friendnames )
        {
            Console.WriteLine(friend);
        }
    }
}

```

- ان الفرق الرئيسى بين استخدام هذه الطريقة عن استخدام for هو ان حلقة foreach تؤمن لنا وصولاً محمياً (للقراءة فقط) لمحتويات المصفوفة، أي لا يمكننا تعديل محتويات المصفوفة ضمن الحلقة فلا يمكننا ان نكتب:

```

        foreach (string friend in friendnames )
        {
            Friend ="suha";
        }

```

EX:

```

static void Main(string[ ] args)
{
    int[] x = new int[5] { 6, 7, 8, 2, 4 };
    int sum = 0;
    foreach (int y in x)
        sum += y;
    Console.WriteLine(sum);
}

```