

# LES BASES DU C#

## Partie 1

# Sommaire

1. LES TYPES DE VARIABLES

2. LES TABLEAUX

3. LES CONDITIONS

4. LES BOUCLES

5. REF ET OUT



## 1. Les types de variables

# Les types de variables

Type	Description
byte	Entier de 0 à 255
short	Entier de -32768 à 32767
int	Entier de -2147483648 à 2147483647
long	Entier de -9223372036854775808 à 9223372036854775807
float	Nombre simple précision de -3,402823e38 à 3,402823e38
double	Nombre double précision de -1,79769313486232e308 à 1,79769313486232e308
decimal	Nombre décimal convenant particulièrement aux calculs financiers (en raison de ses nombres significatifs après la virgule)
char	Représente un caractère
string	Une chaîne de caractère
bool	Une valeur booléenne (vrai ou faux)

# Conversion des types par cast

```
using System;

namespace TypeConversionApplication {
    class ExplicitConversion {
        static void Main(string[] args) {
            double d = 5673.74;
            int i;

            // cast double to int.
            i = (int)d;
            Console.WriteLine(i);
            Console.ReadKey();
        }
    }
}
```

# conversion des types par méthode

```
using System;

namespace TypeConversionApplication {
    class StringConversion {
        static void Main(string[] args) {
            int i = 75;
            float f = 53.005f;
            double d = 2345.7652;
            bool b = true;

            Console.WriteLine(i.ToString());
            Console.WriteLine(f.ToString());
            Console.WriteLine(d.ToString());
            Console.WriteLine(b.ToString());
            Console.ReadKey();
        }
    }
}
```

# Conversibilité

De	Vers
sbyte	short, int, long, float, double ou decimal
byte	short, ushort, int, uint, long, ulong, float, double ou decimal
short	int, long, float, double ou decimal
ushort	int, uint, long, ulong, float, double ou decimal
int	long, float, double ou decimal
uint	long, ulong, float, double ou decimal
long	float, double ou decimal
ulong	float, double ou decimal
char	ushort, int, uint, long, ulong, float, double ou decimal
float	double

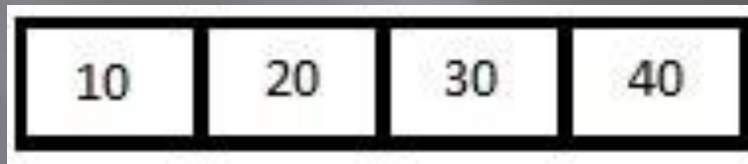


## 2. Les tableaux



# Les tableaux à une dimension

```
int [] myArray = new int [] {10, 20, 30, 40}
```



Indice	0	1	2	3	4	5
<i>valeur</i>	2	3	5	7	11	13

# Lire et écrire dans un tableau

## **ECRIRE**

```
// écriture dans le 11ème élément  
montableau[10] = 10;
```

## **LIRE**

```
// lecture du 13ème élément  
int valeur = montableau[12];
```

# Les tableaux à deux dimensions

```
int[,] array2D = new int[,] { { 1, 2 }, { 3, 4 }, { 5, 6 }, { 7, 8 } }
```

1	2
3	4
5	6
7	8

# Parcourir un tableau

→ foreach

```
foreach (type nom_de_variable in nom_de_l'ensemble)  
{  
  instructions  
}
```



### 3. Les conditions

# Les conditions

Les opérateurs de comparaison :

Opérateur	Description
==	Egalité
!=	Différence
>	Supérieur à
<	Inférieur à
>=	Supérieur ou égal
<=	Inférieur ou égal
&&	ET logique
	OU logique
!	Négation

# Les conditions

```
if (condition)
{
// code à exécuter si la condition est satisfaite
}
else
{
// code à exécuter si la condition n'est pas satisfaite
}
```

# Les conditions

```
switch (variable)
{
    case 1:
        {
            // code à exécuter dans la cas numéro 1
        }
        break ;

    case 2 :
        {
            // code à exécuter dans le cas numéro 2
        }
        break ;

}
```



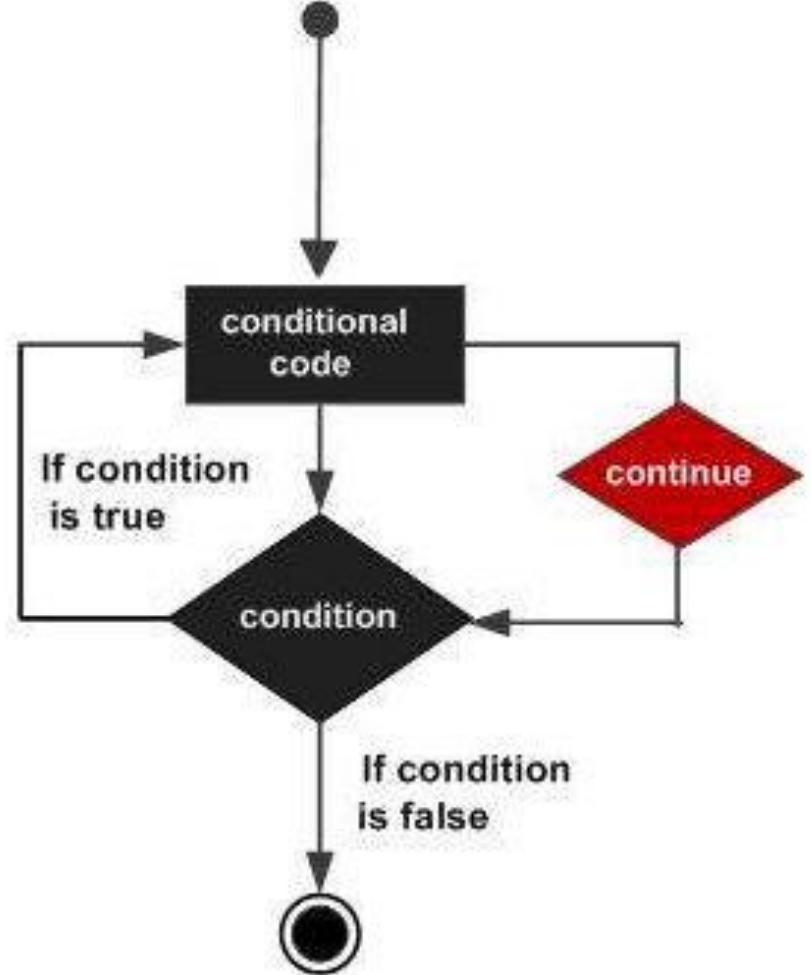
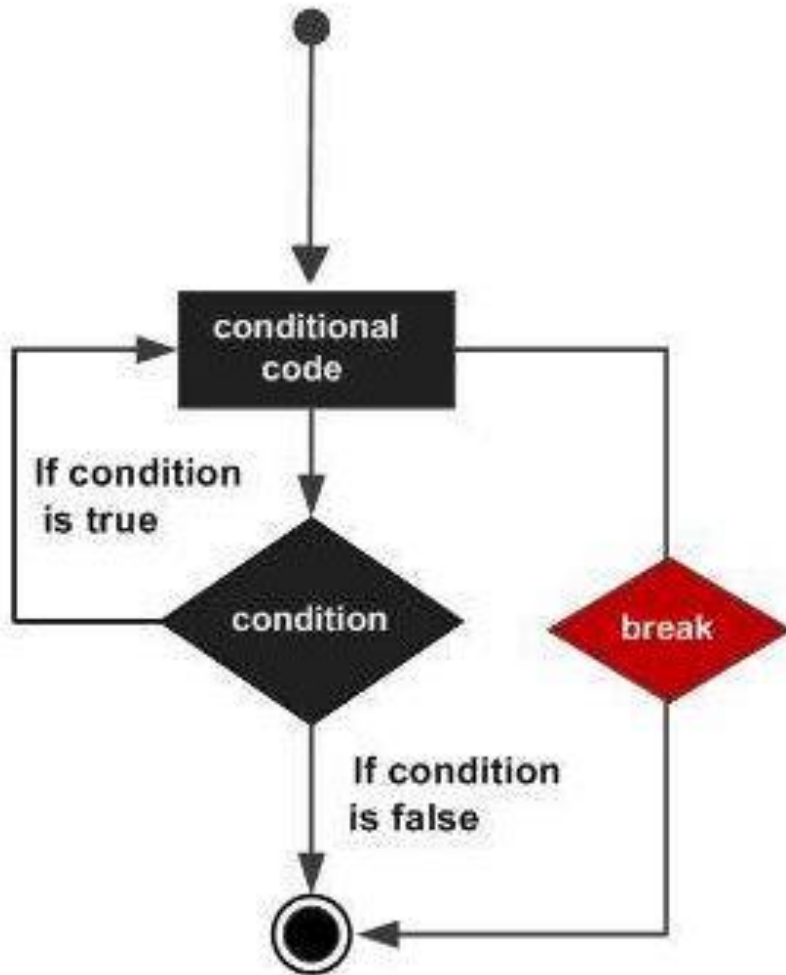


## 4. Les boucles

# Les boucles

- While
- Do ... While ...
- For ...
- Foreach ... In ...
- Break
- Continue

# break et continue





## 5. Différences entre REF et OUT

# Mots clés ref et out



# THE BASICS OF THE C#

## Part 2

# 1. Accessibility level

- ▣ The 4 main keywords :
- ▣ **public** : not limited
- ▣ **internal** : limited in the same assembly
- ▣ **private** : limited in the same class
- ▣
- ▣ **protected** : limited in the same class and the derived class

## 2. Enumerations

- ▣ **public** by default
- ▣ Declaration :
- ▣ enum days
- ▣ {
- ▣ Sat,Sun,Mon,Tue,Wed,Thu,Fri
- ▣ };
- ▣ days.Mon=2



## 3.Procedures

```
void MyProcedure(){  
    Console.WriteLine(« ça  
    marche !!! ») ;  
}
```

## 4.Functions

```
int calcul(){  
    Instruction1  
    ...  
    Instruction n  
}
```

# FUNCTIONS

- ▣ public int **CalculPrixTTC** (prixHt,tauxTva){
- ▣ double prixHt,tauxTva ,tva;
- ▣ tva=prixHt\*(tauxTva/100);
- ▣ return prixHt+tva;
- ▣ }

# 5. Classes

- ▣ For example
- ▣ `public class MyClass`
- ▣ `{`
- ▣ `string myName ;`
- ▣ `string myFirstName ;`
- ▣ `int myAge;`
- ▣ `}`

# Sources internet et livres

[openclassrooms.com](http://openclassrooms.com), [docs.microsoft.com](http://docs.microsoft.com), [c-sharpcorner](http://c-sharpcorner.com),  
[prof.bpesquet.fr](http://prof.bpesquet.fr), [dotnetdojo.com](http://dotnetdojo.com), [pise.info/csharp/](http://pise.info/csharp/),  
[tutorialspoint.com/csharp/](http://tutorialspoint.com/csharp/)

Les fondamentaux du langage c sharp

MERCI