

PROGRAMMATION C#

Syntaxe

- Chaque instruction doit se terminer par un point-virgule ;
- Un bloc de code est contenu à l'intérieur d'accolades { }
- Cela vous sert à dire quel code doit être à l'intérieur d'une fonction, d'une boucle, d'une condition
- **Attention** juste avant un bloc de code on ne met pas de point-virgule, exemple:

```
if (nom == « »)    <— ! Pas de point-virgule ici
{
    Console.WriteLine(« Le nom est vide»);
}
```

- **Remarque** : on peut mettre les accolades à la ligne ou sur la même ligne, cela n'a pas d'importance. Pas besoin non plus de respecter les tabulations. Exemple :

```
if (nom == « ») {
    Console.WriteLine(« Le nom est vide»);
}
```

— ou encore : —

```
if (nom == « ») { Console.WriteLine(« Le nom est vide»); }
```

- Pour les conditions, vous êtes obligé d'utiliser des parenthèses

Types de variables

string	// chaîne de caractères : « toto »
int	// nombre entier : 0, 1, 2, 3, 4 (peut être négatif, mais pas de virgules)
float	// nombre à virgules : 1.5f
bool	// booléen : vrai ou faux (true / false)

WriteLine / ReadLine

```
Console.Write("Quel est votre nom? ");    // Afficher sans passer à la ligne
string nom = Console.ReadLine();          // L'utilisateur rentre des informations
Console.WriteLine("Vous vous appelez " + nom);    // Concaténation de chaîne
Console.WriteLine($"Vous vous appelez {nom},\nvous avez {age} ans"); // Chaînes
formatées (\n : sert à retourner à la ligne)
```

Commentaires

```
// Commentaire sur une ligne
/* Commentaire sur
    plusieurs lignes */
```

Conversions

```
int age = 30;
Console.WriteLine("Votre age est: " + age);    // Conversion implicite de int vers string

string age_str = "30";
int age_int = int.Parse(age_str);             // Conversion de string vers int.
                                              // Utiliser un bloc try/catch en cas d'erreur
```

Boucle While

Boucle tant que la condition est vraie

```
string nom = "";
while (nom == "") {
    Console.Write("Quel est votre nom? ");
    nom = Console.ReadLine();
}
```

Boucle For

Boucle un nombre de fois

```
for (int i = 0; i < 4; i++) {    # de 0 (inclus) à 4 (exclu) : 0, 1, 2, 3
    Console.WriteLine(i);
}
```

Fonctions

```
// Définition (taille est un paramètre optionnel)
void AfficherInfosPersonne(int age, string nom, float taille = 0) {
    Console.WriteLine("Vous vous appelez " + nom + ", vous avez " + age + " ans »);
}
```

```
// Appel
AfficherInfosPersonne("Jean", 25)
```

// **Return** : Retourner une valeur ou sortir de la fonction directement

Conditions

```
==    // égal
!=    // différent de
<=    // inférieur ou égal
<     // inférieur
>=    // supérieur ou égal
>     // supérieur
&&    // ET
||     // OU
```

```
if (age == 18)
{
    Console.WriteLine("Vous êtes tout juste majeur");
}
else if (age == 17)
{
    Console.WriteLine("Vous serez bientôt majeur");
}
else if ((age >= 12) && (age < 18))
{
    Console.WriteLine("Vous êtes adolescent");
}
else if ((age == 1) || (age == 2))
{
    Console.WriteLine("Vous êtes un bébé");
}
```

Exception

```
try
{
    age_num = int.Parse(age_str);
}
catch
{
    Console.WriteLine("Erreur : Vous devez rentrer un age valide.");
}
```

Configurer la console en UTF8

Pour bien afficher les caractères spéciaux

```
Console.OutputEncoding = System.Text.Encoding.UTF8;
```