

FORMATION OCKAM RAZOR

LES NOMBRES, BOOLEENS ET CHÂÎNES

FLASH

PRINCIPE

En programmation, **les trois variables d'usages courant** sont les valeurs numériques, les valeurs booléennes et les chaînes de caractère.

DESCRIPTION

Avant toutes choses, il est important de définir ce qu'est une variable.

Les variables sont des valeurs que le programmeur peut rattacher à un nom de son choix (*Par exemple : "maVariable"*). À la suite de quoi, le programmeur utilisera ce même nom pour **retrouver la valeur de sa variable ou la modifier**.

En ActionScript 3, les valeurs **Numériques** sont nommées "**Number**", elles peuvent contenir des valeurs positives (*exemple : 15*) ou négatives (*exemple : -30*) de -infini à +infini.

Elles peuvent être entières ou avec décimales.

En ActionScript 3, les valeurs **Booléennes** sont nommées "**Boolean**", il s'agit d'une valeur vraie ou fausse comme un levier.

En ActionScript 3, les valeurs de **Chaîne de caractères** sont nommées "**String**", il s'agit de suite de caractères formant des mots ou des phrases et utilisables dans de nombreux cas de figures.

UTILISATION

Pour créer une variable en ActionScript 3, on utilise la formule suivante :

```
var maVariable:Type = valeur;
```

À la place de "**maVariable**", on met le nom de son choix.

À la place de "**Type**" on met le type de la variable (*Par exemple : Number*)

À la place de "**valeur**" on met la valeur initiale que l'on souhaite donner à la variable.

Exemple avec un nombre :

```
var monNombre:Number = 10;
```

À noter que pour un nombre, si on souhaite mettre des chiffres après la virgule, on n'utilise pas la virgule de ponctuation standard mais le point.

```
var monNombre:Number = 10.5;
```

À tout moment, vous pouvez modifier, dans votre code, la valeur d'une variable en utilisant : " = ".

Pour les booléennes, les deux valeurs possibles sont :

- **true** : vrai
- **false** : faux

Pour les chaîne, une valeur doit être entre guillemet :

```
var maChaine:String = "Youpie Doo !";
```

RACCOURCIS

Number

+= X → Ajoute **X** à la valeur

-= X → Retire **X** à la valeur

***= X** → Multiplie cette valeur par **X**

/= X → Divise cette valeur par **X**

Boolean

!= X → La valeur devient l'inverse de **X**

String

+= X → Concaténer **X** à la fin de la valeur

length → Donne le nombre de caractères de la chaîne

toUpperCase() → Met la chaîne en Majuscule

toLowerCase() → Met la chaîne en Minuscule

substr(X) → Retire les **X** derniers caractères de la valeur