

1. VARIABLES ET TYPES

DÉFINITION D'UNE VARIABLE

Une variable en programmation est un espace mémoire qui porte un nom, dans lequel est stockée une valeur.

Les variables peuvent changer de valeur au cours du temps.

Chaque variable a un type qui définit la nature de la valeur stockée dans l'espace mémoire associée.



PORTÉ D'UNE VARIABLE

La porté d'une variable est la portion de code source où elle est accessible, par exemple, la portée d'une variable (non globale) s'étend de sa définition à la fin du bloc où elle est définie.

VARIABLE LOCALE

Une variable est locale lorsqu'elle appartient à un bloc d'instruction : à l'intérieur d'une méthode, d'une boucle, d'un test. Elle s'utilise à l'intérieur du bloc à laquelle elle appartient après qu'elle ait été déclarée et initialisée.

VARIABLE GLOBALE

Une variable est globale lorsqu'elle n'appartient à aucun bloc d'instructions. En langage Processing, une variable globale se déclare avant le `setup()`, s'initialise dans le `setup()` et s'utilise à n'importe quelle endroit du programme après qu'elle ait été déclarée et initialisée.

TYPE

En programmation, le type d'une variable définit la nature des valeurs que peut prendre cette variable, ainsi que les opérateurs qui peuvent lui être appliqués.

TYPE DE BASE (PRIMITIF)

Les types de base correspondent à des données qui peuvent être traitées directement par le processeur, sans conversion ou formatage préalable. Les principaux types du langage Processing dérivent du Java :

- Type **booléen (boolean)**: valeurs vrai ou faux — ou respectivement 1 ou 0.
- Type **entier (int)** signé ou non signé : valeurs codées sur 32 bits en Java/Processing.
- Les **caractères (char)** codés sur 16 bits en Java/Processing.
- Type **réel en virgule flottante (float)** codé sur 32 bits en Java/Processing.

Le tableau suivant résume les différents types :

Type	description	taille	ensemble de valeurs	val. init.
<i>Nombres entiers</i>				
byte	Octet	8 bits	de -128 à 127	0
short	Entier court	16 bits	de -32 768 à 32 767	
int	Entier	32 bits	de -2 147 483 648 à 2 147 483 647	
long	Entier long	64 bits	de -2^{63} à $2^{63} - 1$	
<i>Nombres flottants</i>				
float	Flottant simple précision	32 bits	de $-3,4028235 \times 10^{38}$ à $-1,4 \times 10^{-45}$, 0 et de $1,4 \times 10^{-45}$ à $3,4028235 \times 10^{38}$	0.0
double	Flottant double précision	64 bits	de $-1,7976931348623157 \times 10^{308}$ à $-4,9 \times 10^{-324}$, 0 et de $4,9 \times 10^{-324}$ à $1,7976931348623157 \times 10^{308}$	
<i>Autres types</i>				
char	Caractère	16 bits	Tous les caractères Unicode	'\u0000'
boolean	Valeur booléenne	1 bit	false, true	false

TYPE COMPLEXE

Les types complexes comprennent l'ensemble des types non-primitifs. Il s'agit des types paramétrés comme les tableaux, ou les types hiérarchiques comme les classes dans la programmation orientée objet.

Exemple de types complexes sous Processing :

- **String** : chaîne de caractères
- **PImage** : gestion des images
- **Sound** : gestion du son
- **video** : gestion de la vidéo
- etc.

SYNTAXE

```
type nomDeLaVariable = valeur ;
```

Exemple :

```
int i = 10 ;  
float x = 1.54 ;  
char c = 'A' ;  
String s = "Bonjour !" ;
```

2. ARCHITECTURE GÉNÉRALE D'UN PROGRAMME SOUS PROCESSING

STRUCTURE GÉNÉRALE D'UN PROGRAMME :

```
// appel aux librairies  
  
// déclaration des variables globales  
  
void setup(){ instructions }  
  
void draw(){ instructions }  
  
// méthodes personnelles
```

VOID SETUP()

Il s'agit d'une méthode qui permet d'initialiser les variables globales et les paramètres d'affichage.

VOID DRAW()

Il s'agit d'une méthode réalisant une boucle qui s'exécute par défaut 30 fois par secondes permettant de réaliser un bloc d'instructions qui agissent généralement sur le contenu de la fenêtre d'affichage.