# Les Variables et Constantes

## Déclaration :

### Variables :

|  |  |
| --- | --- |
| **PseudoCode** | **Java** |
| VAR nomVariable : Type <- Valeur  nomVariable : défini le nom de la variable  Type : défini le type de la variable | Type nomVariable ( = Valeur)  ( = Valeur) : non obligatoire |

### Constante :

|  |  |
| --- | --- |
| **PseudoCode** | **Java** |
| VAR nomVariable : Type <- Valeur  nomVariable : défini le nom de la variable  Type : défini le type de la variable | Type nomVariable ( = Valeur)  ( = Valeur) : non obligatoire |

|  |  |
| --- | --- |
| **PseudoCode** | **Java** |
| CONST nomVariable : Type <- Valeur  nomVariable : défini le nom de la constante  Type : défini le type de la constante | final Type nomVariable ( = Valeur)  ( = Valeur) : non obligatoire  final : défini que c'est une constante |

## Type de variables :

|  |  |
| --- | --- |
| **PseudoCode** | **Java** |
| N | int Défini une valeur entière (non décimale)  Insérer un décimal plante le programme |
| N | float  Un nombre décimal court Le "." représente le séparateur |
| N | double  Un nombre décimal long  Le "." représente le séparateur |
| C | char  Un seul et unique caractère  A définir entre ' ' |
| T | String  Une chaine de caractère  A définir entre " " |
| B | boolean  Une variable de type booléen  Deux valeurs possible: true ou false |

## Affectation :

|  |  |
| --- | --- |
| **PseudoCode** | **Java** |
| nomVariable <- Valeur  La valeur doit être du même type que la variable | nomVariable = Valeur  La valeur doit être du même type que la variable |

## Déclaration :

|  |  |
| --- | --- |
| **PseudoCode** | **Java** |
| Variables locales :  CONST auteur : T <- Toto  VAR letter : C <- b  VAR age : N <- 25  VAR poids : N <- 4,65  VAR estVrai : B <- VRAI  fin variables | final String auteur = "Toto";  char letter = 'b'; int age = 25;  double poids = 4.65;  float poids2 = 4.65;  boolean estVrai = true; |

## Convertir une valeur :

|  |  |
| --- | --- |
| **PseudoCode** | **Java** |
| Pas possible de convertir une valeur en PseudoCode, on conserve toujours le même type. | Automatiquement :  int myInt = 9 ;  double myDouble = myInt;  Manuellement :  double myDouble = 9.54;  int myInt = (int) myDouble ;  System.Out.println(myInt); // outputs = 9 |