

Java

CM3-2 : Java, types

Mickaël Martin Nevot

V1.0.0



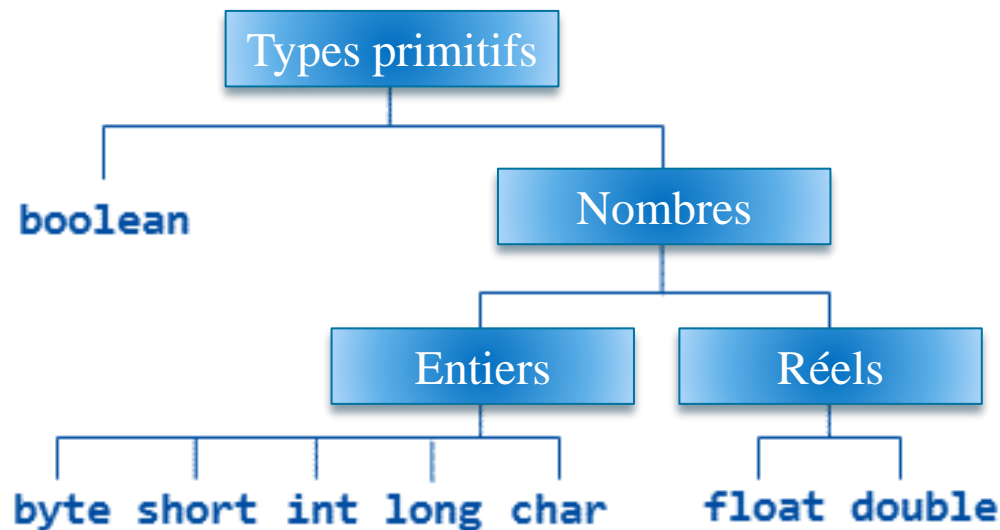
Cette œuvre de [Mickaël Martin Nevot](#) est mise à disposition selon les termes de la [licence Creative Commons Attribution – Pas d'Utilisation Commerciale – Partage à l'Identique 3.0 non transposé](#).

Java

- I. Prés.
- II. POO
- III. Objet
- IV. Java
- V. **Types**
- VI. Héritage
- VII. Outils
- VIII. Exceptions
- IX. Polymorphisme
- X. Thread
- XI. Avancé

Type primitif

- N'est pas un objet
- Occupe une place fixe en mémoire
- Dispose d'un *alter ego* objet et d'une méthode de conversion
- Est converti automatiquement en référence (*autoboxing*)
- Conversion de type explicite (*cast*) : (`type`)



Types primitifs

- Entiers :

• <code>byte</code>	-128 à 127	1 octet
• <code>short</code>	-32768 à 32768	2 octets
• <code>int</code>	-2147483648 à 2147483647	4 octets
• <code>long</code>	-9223372036854775808 à 9223372036854775807	8 octets

- Flottants :

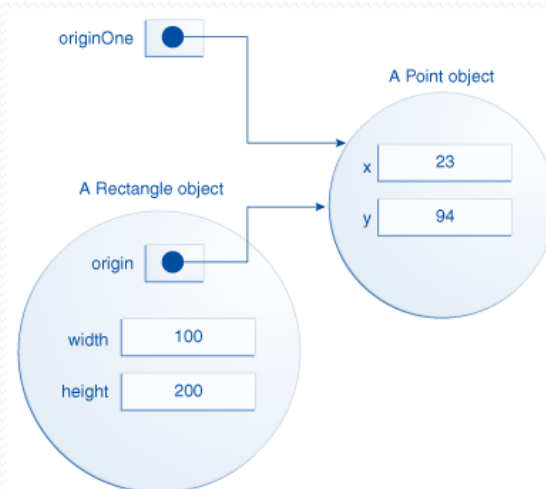
• <code>float</code>	variable	4 octets
• <code>double</code>	variable	8 octets

- Autres :

• <code>boolean</code>	<code>true</code> , <code>false</code>	selon la JVM
• <code>char</code> (Unicode, c.-à-d. a)	0 à 65535	2 octets

Référence

- Référence vers un objet : il n'existe **pas de variable objet**
- Une référence déclarée pour un type d'objet ne peut référencer que des objets de ce type
- Une référence ne référence qu'un seul objet à la fois
- Un objet peut être référencé par plusieurs références
- Aucune référence : `null`



Variable et constante

- Variable (deux catégories : **primitive** ou **référence**) :
 - Identifiant
 - Type
- Constante :
 - Variable ne pouvant avoir qu'une seule affectation
 - **Non modifiable**
 - Mot clef `final`

```
final int n = 5;  
final int t;  
...  
t = 8;  
n = 10; // Erreur !
```



Tableau

- Considéré comme un objet
- Un seul type par tableau (primitif/objet)
- Indices commencent à zéro
- Mot clef `new` :
 - Alloue la mémoire en fonction de la taille (fixe)
 - Initialise à 0 (type primitif)
- Multidimensionnel (tableau de type tableau) :

Pas de dimensions à la déclaration

```
int[] myTab; // Déclaration.  
myTab = new int[3]; // Dimensionnement.  
myTab[0] = 1;  
myTab[2] = 5;
```



```
int myTab[] = {1, 0, 5};
```

Opérateurs

- Unaires :
 - Arithmétiques : +, -
 - Incrémentation/Décrémentation (pré, post) : ++, --
 - Transtypage : (`type`)
- Binaires :
 - Arithmétiques : +, -, *, /, %
 - Affectations (élargies) : =, +=, -=, *=, /=
 - Comparaisons : ==, >, >=, <, !=
 - Logiques : &&, ||, &, |
 - Concaténation : +

Crédits

Auteur

Mickaël Martin Nevot

mmartin.nevot@gmail.com



Carte de visite électronique

Relecteurs

Cours en ligne sur : www.mickaël-martin-nevot.com

