



# PROGRAMMATION ORIENTÉE OBJET JAVA

LICENCE 2 INGÉNIERIE - INFORMATIQUE

2020– 2021

Marie **NDIAYE** DIOP



# LES BASES DU LANGAGE JAVA

# INSTRUCTIONS, BLOCS ET BLANCS

- Les instructions Java se terminent par un « ; »
- Une instruction peut tenir sur plusieurs lignes
- Les blocs sont délimités par :
  - « { » pour le début de bloc
  - « } » pour la fin du bloc
- Un bloc permet de définir un regroupement d'instructions. La définition d'une classe ou d'une méthode se fait dans un bloc.
- Les espaces, tabulations, sauts de ligne sont autorisés. Cela permet de présenter un code plus lisible.

# LES STRUCTURES DE CONTRÔLE

- Classiques du C/C++
  - if, else
  - switch, case, default, break
  - for
  - while
  - do, while

## EXEMPLE I

<b>int i=1;</b>	
<b>//if else</b> if (i<5){ System.out.println(i); } else{ System.out.println("Trop grand"); }	<b>//while</b> i=1; while(i<=5){ System.out.println(i); i++; }
<b>//for</b> for(i=1;i<=5;i++){ System.out.println(i); }	<b>//do while</b> i=1; do{ System.out.println(i); i++; } while(i<=5);

## EXEMPLE 2

### Structure de contrôle switch

```
switch (sexe) {  
    case 1:  
        System.out.println("Masculin");  
        break;  
    case 2:  
        System.out.println("Féminin");  
        break;  
    default:  
        System.out.println("Sexe indéterminé");  
}
```

# COMMENTAIRES

- `/*` commentaire sur une ou plusieurs lignes `*/`
  - Identiques à ceux existant dans le langage C
- `//` commentaire de fin de ligne
- `/**` commentaire d'explication `*/`
  - Les commentaires d'explication se placent généralement juste avant une déclaration (d'attribut ou de méthode)
  - Ils sont récupérés par l'utilitaire Javadoc et inclus dans la documentation ainsi générée.

# LES IDENTIFICATEURS

- Les identificateurs commencent par une lettre, « \_ » ou « \$ »
  - Attention : **Java est sensible à la casse** : il distingue les majuscules des minuscules
- Conventions sur les identificateurs
  - La première lettre est en majuscule pour les classes et les interfaces
    - MaClasse, UneJolieFenetre
  - La première lettre est minuscule pour les méthodes, les attributs et les variables.
    - setLongueur, i, uneFenetre
  - Les constantes sont entièrement en majuscules
    - LONGUEUR\_MAX
  - Si plusieurs mots sont accolés, mettre une majuscule à chacun des mots sauf le premier.
    - uneVariableEntiere



# LES MOTS RÉSERVÉS

<code>abstract</code>	<code>default</code>	<code>goto</code>	<code>null</code>	<code>synchronized</code>
<code>boolean</code>	<code>do</code>	<code>if</code>	<code>package</code>	<code>this</code>
<code>break</code>	<code>double</code>	<code>implements</code>	<code>private</code>	<code>throw</code>
<code>byte</code>	<code>else</code>	<code>import</code>	<code>protected</code>	<code>throws</code>
<code>case</code>	<code>extends</code>	<code>instanceof</code>	<code>public</code>	<code>transient</code>
<code>catch</code>	<code>false</code>	<code>int</code>	<code>return</code>	<code>true</code>
<code>char</code>	<code>final</code>	<code>interface</code>	<code>short</code>	<code>try</code>
<code>class</code>	<code>finally</code>	<code>long</code>	<code>static</code>	<code>void</code>
<code>continue</code>	<code>float</code>	<code>native</code>	<code>super</code>	<code>volatile</code>
<code>const</code>	<code>for</code>	<code>new</code>	<code>switch</code>	<code>while</code>

# LES TYPES DE DONNÉES

- Les nombres entiers
- Les nombres réels
- Les caractères et chaînes de caractères
- Les booléens
- Les classes

# LES TYPES PRIMITIFS

Type	Codage	Domaine
<i>char</i>	2 octets	valeur du jeu de caractères Unicode (65000 caractères possibles)
<i>int</i>	4 octets	-2 147 483 648 à 2 147 483 647
<i>long</i>	8 octets	-9223372036854775808 à 9223372036854775807
<i>byte</i>	1 octet	-128 à 127
<i>short</i>	2 octets	-32768 à 32767
<i>float</i>	4 octets	$-1.4 \times 10^{-45}$ à $3.4 \times 10^{38}$
<i>double</i>	8 octets	$4.9 \times 10^{-324}$ à $1.7 \times 10^{308}$
<i>boolean</i>	1 bit	0 ou 1 (en réalité, toute autre valeur que 0 est considérée égale à 1)

# LES CLASSES PRÉDÉFINIES

- String → chaîne de caractères
- Date → date
- Character → char
- Integer → int
- Long → long
- Byte → byte
- Float → float
- Double → double
- Boolean → boolean

Classes enveloppe

Package **java.lang**

Package **java.util**

# LES OPÉRATEURS

- Classiques : **+**, **-**, **\***, **/**
- Reste de la division entière : **%**
- Incrémentation et décrémentation: **++** et **--**
- Logiques :
  - De comparaison : **==**, **!=**, **<**, **>**, **<=**, **>=**
  - Autres : **||**, **&&**, **!**
- Class **Math** de Java : **max**, **min**, **pow**, **abs**, **cos**, **sin...**

Package **java.lang**

# LA DOCUMENTATION DE L'API JAVA

■ <https://docs.oracle.com/en/java/javase/15/docs/api/index.html>

# POINT D'ENTRÉE D'UN PROGRAMME

- Pour pouvoir faire un programme exécutable il faut toujours une classe qui contient une méthode particulière, la méthode « **main** »
- c'est le point d'entrée dans le programme : le microprocesseur sait qu'il va commencer à exécuter les instructions à partir de cet endroit.

## EXEMPLE

La méthode main reçoit toujours le paramètre args qui est un tableau de chaîne de caractères.

```
public class Bonjour
{
    public static void main(String args[ ])
    {
        System.out.println("bonjour");
    }
}
```



# COMPILATION ET EXÉCUTION 1/2

Fichier **Bonjour.java**

Le nom du fichier est nécessairement celui de la classe avec l'extension **.java**

**Attention** : Java est sensible à la casse des lettres.

```
public class Bonjour{  
    public static void main(String args[ ]) {  
        System.out.println("bonjour");  
    }  
}
```

# COMPILATION ET EXÉCUTION 2/2

- Dans une console DOS
  - `javac Bonjour.java`
    - Compilation en bytecode java
    - Indication des erreurs de syntaxe éventuelles
    - Génération d'un fichier `Bonjour.class` s'il n'y a pas d'erreurs
  - `java Bonjour`
    - Java est la machine virtuelle
    - Exécution du bytecode
    - Nécessite la méthode `main` qui est le point d'entrée dans le programme

# REFERENCES

- <https://docs.oracle.com/en/java/javase/15/docs/api/index.html>
- <https://docs.oracle.com/en/java/javase/15/docs/api/java.base/java/lang/package-summary.html>
- <https://docs.oracle.com/en/java/javase/15/docs/api/java.base/java/lang/Math.html>
- <https://web.maths.unsw.edu.au/~lafaye/CCM/java/javatype.htm>