POO : JAVA

Pr Sara SEKKATE

Intelligence Artificielle & Génie Informatique

2022-2023

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Introduction au langage Java

Différents styles de programmation

- Style **applicatif** : Fondé sur l'évaluation d'expressions qui ne dépendent que de la valeur des arguments, et non de l'état de la mémoire
  - On parle aussi de programmation fonctionnelle
  - Proche des notations mathématiques, utilise beaucoup la récursivité
  - Accepte des arguments, produit un résultat (pas d'« effet de bord »)
  - Ex: Lisp, Caml, ML, Haskel
- Style **impératif** : Fondé sur l'exécution d'instructions qui modifient l'état de la mémoire
  - Utilise beaucoup les itérations et autres structures de contrôle
  - Les structures de données sont fondamentales
  - Ex: Fortran, C, Pascal

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Le style objet

- C'est un style de programmation où l'on considère que des composants autonomes (les objets) disposent de ressources et de moyens d'interactions entre eux.
- Ces objets représentent des données qui sont modélisées par des classes qui définissent des types.
- En plus de la manière dont sont structurés leurs objets, les classes définissent les actions qu'ils peuvent prendre en charge et la manière dont ces actions affectent leur état (méthodes).
- Java n'est pas le seul langage objet
- Python, Simula, C++,...

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Les avantages de la programmation objet

- Les caractéristiques de bases précédemment décrites peuvent être mises en oeuvre dans un style impératif, mais des fonctionnalités propres au style objet favorisent:
  - La modularité
  - L'abstraction
  - La sûreté
- L'objectif est de produire du code:
  - facile à développer, à maintenir, à faire évoluer,
  - réutilisable, tout ou en partie, sans avoir besoin de le dupliquer
  - qui facilite la conception, le maintien et l'exploitation de gros logiciels

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Les avantages de la programmation objet

- Les caractéristiques de bases précédemment décrites peuvent être mises en oeuvre dans un style impératif, mais des fonctionnalités propres au style objet favorisent:
  - La modularité : les objets forment des modules compacts regroupant des données et un ensemble d'opérations.
  - L'abstraction
  - La sûreté
- L'objectif est de produire du code:
  - facile à développer, à maintenir, à faire évoluer,
  - réutilisable, tout ou en partie, sans avoir besoin de le dupliquer
  - qui facilite la conception, le maintien et l'exploitation de gros logiciels

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Les avantages de la programmation objet

- Les caractéristiques de bases précédemment décrites peuvent être mises en oeuvre dans un style impératif, mais des fonctionnalités propres au style objet favorisent:
  - La modularité
  - L'abstraction : les entités objets de la POO sont proches de celles du monde réel. Les concepts utilisés sont donc proches des abstractions familières que nous exploitons.
  - La sûreté
- L'objectif est de produire du code:
  - facile à développer, à maintenir, à faire évoluer,
  - réutilisable, tout ou en partie, sans avoir besoin de le dupliquer
  - qui facilite la conception, le maintien et l'exploitation de gros logiciels

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Les avantages de la programmation objet

- Les caractéristiques de bases précédemment décrites peuvent être mises en oeuvre dans un style impératif, mais des fonctionnalités propres au style objet favorisent:
  - La modularité
  - L'abstraction
  - La sûreté : L'encapsulation et le typage des classes offrent une certaine robustesse aux applications.
- L'objectif est de produire du code:
  - facile à développer, à maintenir, à faire évoluer,
  - réutilisable, tout ou en partie, sans avoir besoin de le dupliquer
  - qui facilite la conception, le maintien et l'exploitation de gros logiciels

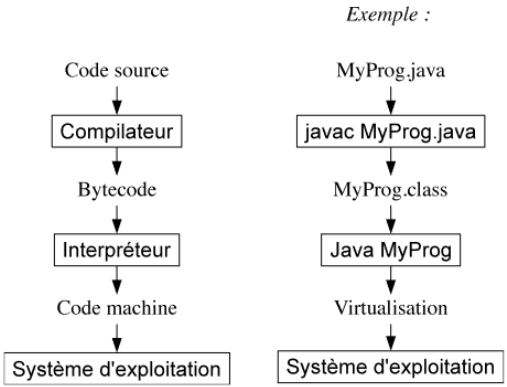
ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Environnement Java

- Développé au départ pour être utilisé sur le web par Sun Microsystems en 1995.
- Java est un langage :
  - **compilé** : En bytecode
  - **interprété**: Un programme compilé n'est pas directement exécutable par le SE. Il doit être interprété par un autre programme, qu'on appelle **interpréteur**.
  - **fortement typé**:
    - Toute variable doit être déclarée avec un type
    - Le compilateur vérifie que les utilisations des variables sont compatibles avec leur type (notamment via un sous typage correct)
    - Les types sont d'une part fournis par le langage, mais également par la définition des classes

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Environnement Java



- Le code source est écrit sous forme de classes dans des fichiers *.java*.
- Il est alors compilé par le compilateur *javac* en un langage appelé *bytecode* et enregistre le résultat dans un fichier *.class*.
- Le bytecode doit être interprété par la machine virtuelle de Java qui transforme le code compilé en code machine compréhensible par le SE.
- Java est un langage **portable** : le bytecode reste le même quelque soit l'environnement d'exécution.

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Premier exemple

```
public class Welcome{
    public static void main(String[] args) {
        System.out.println("Hello IAGI1");
                    }
                }
```

- Toute classe publique doit être dans un fichier qui a le même nom que la classe
- Tout code doit être à l'intérieur d'une classe
- Comme il y a une méthode main, cette classe est « exécutable »

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Le langage Java

- Les variables, les opérateurs, les expressions, instructions, blocs, contrôle de flot sont très proches de ceux du C
  - Les exceptions sont une nouveauté
  - Les types primitifs ont une taille et une représentation normée
- Un style de nommage (très fortement) conseillé
  - Première majuscule pour les classes (class HelloWorld)
  - Première minuscule pour les variables/champs et les fonctions/méthodes (radius, getRadius())
  - Tout en majuscule pour les constantes (MAX_SIZE)

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Classes et objets

Une classe **ClasseA** représente plusieurs choses:

- Une **unité de compilation** : La compilation d'un programme qui contient une classe ClasseA produira un fichier ClasseA.class
- La définition du **type** ClasseA : Il peut servir à déclarer des variables comme ClasseA t;
- Un moule pour la création d'objets de type ClasseA
  - Cela nécessite en général la définition d'un ensemble de champs (fields) décrivant l'état d'un objet de ce type et d'un ensemble de méthodes définissant son comportement ou ses fonctionnalités
- Chaque objet de la classe ClasseA
  - Dispose de son propre état (la valeur de ses champs)
  - Répond au même comportement (via les méthodes de la classe)

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Structure d'une classe

- Une classe est définie par son nom et son package d'appartenance (ex: java.lang.String)
- En l'absence de directive, les classes sont dans un package dit « par défaut » (i.e., pas de package).
- Une classe peut contenir trois sortes de membres :
  - Des champs (fields) ou attributs
  - Des méthodes (methods) et constructeurs
  - Des classes internes
- Les membres statiques (static) sont dits membres de classe : Ils sont définis sur la classe et non sur les objets
- Les membres non statiques (ou d'instance) ne peuvent exister sans un objet

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

La syntaxe de Java

Éléments du langage Java

- Toute instruction se termine par ";"
- les commentaires se situent entre les symboles “/*” et “*/” ou commencent par le symbole “//” en se terminant à la fin de la ligne

```
int a ; // ce commentaire tient sur une ligne
int b ;
```

ou

```
/*Ce commentaire necessite
2 lignes*/
int a ;
```

- /** ... */ utilisé aussi pour la génération automatique de la documentation (javadoc)

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITE HASSAN II DE CASABLANCA

Éléments du langage Java

- les identificateurs de variables ou de méthodes ne peuvent commencer que par une lettre {a..z}, {A..Z}, un caractère de soulignement _ ou un signe dollar $ et contenir des caractères alphanumériques ainsi que des symboles monétaires.
- Il faut évidemment que l'identificateur ne soit pas un mot réservé du langage

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

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Éléments du langage Java

- Types primitifs
  - entiers signés
    - byte (octet)
    - short (2 octets)
    - int (4 octets)
    - long (8 octets)
  - virgule flottante :
    - float (4 octets)
    - double (8 octets)
  - caractères : char (2 octets)
  - booléens : boolean (true ou false)

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Attention aux nombres à virgule flottante

- Ils ne sont que des approximations des valeurs
- Leur égalité au sens de l'opérateur == **n'a aucun sens**
- Il faut tester leur « proximité » modulo un epsilon donné

```
static final double EPSILON = 0.00001;
...
double d = 0.0;
double expected = 1.0;
while ( Math.abs(expected - d)> EPSILON )
    d += 0.1;
System.out.println(d); // 0.9999999999999999
```

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Éléments du langage Java

- Variables
  - Avant d'être utilisée, une variable doit être déclarée : Type var;
  - et initialisée : var=....;
- Constantes:
  - Variables introduites par le mot réservé "final"
  - On ne peut lui assigner une valeur qu'une **seule** fois

    final int CONST = 100;

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Éléments du langage Java

- Opérateurs
  - Arithmétiques +, -, *, /, %, ++, –, +=, -=; *=; /=, %=
  - Logiques ==, !=, <, >, <=, >=, &&, ||, !
  - Ternaires ? :
  - Bits &, |,», «
- Fonctions et constantes mathématiques

Math.sqrt(x)	Math.pow(x,a)	Math.exp(x)	Math.log(x)
Math.sin(x)	Math.cos(x)	Math.tan(x)	Math.atan(x)
Math.PI	Math.E		

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

- Énumérations

Exemple

```
enum Size{SMALL, MEDIUM, LARGE}; Size x = Size.MEDIUM;
```

  - Une variable de type énuméré ne peut prendre ses valeurs que parmi celles de sa déclaration ou *null*

- Contrôle de flux:
  - Instructions conditionnelles:

Syntaxe

```
if (condition) instruction;
if (condition) instruction1; else instruction2;
(condition)?(instruction1):(instruction2)
```

- switch :

Syntaxe

```
switch(choix){
case i1 : instruction1;break;
case i2;instruction2;break;
default : instruction3;break;
}
```

avec choix de type entier ou énuméré et in de type choix.

Éléments du langage Java

- Instructions itératives :
  - while (condition) instruction;
  - do instruction while (condition);
  - for(initialisation; condition; evaluation) instruction;
- Briser le flux :
  - **break** : utilisée pour sortir immédiatement d'un bloc d'instructions (sans traiter les instructions restantes dans ce bloc).
  - **break étiqueté** : fait ressortir d'une succession de blocs imbriqués.
  - **continue** : l'exécution du bloc est arrêtée mais pas celle de la boucle. Une nouvelle itération du bloc commence si la condition d'arrêt est toujours vraie.

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA

Break étiqueté

```
etiquette :
    while(...){
        for(...){...
            break etiquette;
            ...
        }
    }
    //execution apres le break
```

Tableaux et matrices

- Déclaration d'un tableau : type[] tab;
- Un tableau a toujours une taille fixe qui doit être précisée avant l'affectation de valeurs à ses indices: tab= new type[dimension]
- la taille de ce tableau est disponible dans une variable *length* appartenant au tableau et accessible par *tab.length*
- parcours d'un tableau avec boucle for-each:
  for(type variable:tableau) instruction;
- Remarque : Le parcours habituel est toujours valide.
- On peut également créer des matrices ou des tableaux à plusieurs dimensions en multipliant les crochets (ex : int[][] matrice;)

ENSAM
ÉCOLE NATIONALE SUPÉRIEURE D'ARTS ET MÉTIERS
UNIVERSITÉ HASSAN II DE CASABLANCA