# 23.09.2022

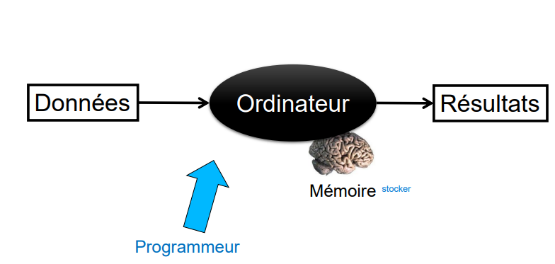
## Administratif :

Travaux asynchrones majoritairement fait sur place

On va apprendre JAVA

Eclipse – Faut créer un workspace. Un projet c’est comme un classeur 🡪 Pour créer un nouveau code faut créer une classe et toujours cocher static void.

## Théorie :

 L’ordinateur reçoit des données qu’il doit stocker (mémoire) et après les manipuler afin de nous fournir un résultat.

Exemple - Une calculatrice

* Point de vue utilisateur : taper un nombre, utiliser le x, taper un autre nombre, mettre égal (action) (= permet de donner un affichage)
* Point de vue développeur : donner des données à l’ordinateur, qui stock ces données dans sa mémoire (chiffre x chiffre) qui fait un traitement et qui affiche le résultat

Afin de développer des choses sur l’ordinateur, il faut voir du point de vue développeur

Entrées ordinateur : micro, caméra, clavier, souris…

Sorties ordinateur : imprimante, du son, vidéo…

Pour stocker des données, on utilise des variables. Une variable est composée de :

* Un nom 🡪 sert à repérer l’emplacement mémoire où est placée une information. Il y des règles de nommage à respecter.
  + Ne pas commencer par une majuscule (pas conseillé)
  + Pas d’espace
  + Majuscules « souvent » différentes des minuscules
  + Pas de caractères spéciaux (+ç%&…)
  + Pas de chiffre comme premier caractère
  + Eviter les accents
  + Mots composés : « nomPersonne » (monPremierSalaire mieux que mon\_premier\_salaire)
* Un type 🡪 détermine la façon dont est traduite la valeur en code binaire et la taille de l’emplacement mémoire. Il y a 4 types de base
  + Nombres entiers (byte, short, int, long)
    - Plus il y a d’octets, plus y a des nombres possibles. On va utiliser int la plupart du temps. Il faut utiliser celui qui est plus adapté car il sera + performant 🡪 années d’étude (byte)
  + Nombres flottants (float – 7 chiffres décimales, double - 30 chiffres décimales)
    - Quand c’est 8.76E4 🡪 8.76 \* 10^4
  + Caractères (char – un seul caractère, String – plusieurs caractères)
  + Booléens (oui/non – vrai/faux - …)

|  |
| --- |
| 0 |

Les informations sont organisées / stockées dans des octets (8 bits). Pour illustrer ici 1 bit = 1 case

|  |
| --- |
| 1 |

|  |  |
| --- | --- |
| 1 | 1 |

|  |  |
| --- | --- |
| 0 | 1 |

1 bit = 0 1 (1 case avec 0 et 1) (2 options)

|  |  |
| --- | --- |
| 0 | 0 |

|  |  |
| --- | --- |
| 1 | 0 |

2 bits = 0 1 (2 cases avec 0 et 1) (4 options)

Et ainsi de suite. Pour un octet on a 2^8 compositions différentes possibles. ENLEVER SIGNE 2^7 – revoir

## Pratique :

*Ressenti*

# 27.09.2022

# Théorie :

La création d’une variable = **la déclaration d’une variable**. Il faut déclarer une variable pour que l’ordinateur réserve l’espace mémoire.

Ex : int rayon ; (ça doit se terminer par un ; afin d’être exécuter en Java) 🡪 ; = fin d’une commande)

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

L’ordinateur réserve un espace de 4 byte.

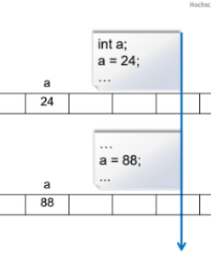
Une image contenant texte

Description générée automatiquementLe ; termine toute les instructions. La virgule sépare plusieurs variables du même type.

**L’affectation** = action de mettre quelque chose/ stocker une information dans un emplacement mémoire = Donner une valeur à la variable.

Ex : Rayon 🡨 4 (on met 4 dans la valeur rayon et pas l’inverse)

En informatique quand on rajoute quelque chose dans une variable l’autre est détruite. On ne peut qu’avoir une seule chose dans une variable.

Ici 88 détruit le 24

Le A vaut 88 à la fin

Car une variable ne peut que contenir une seule valeur.

Rôle de l’utilisateur : entre des infos (clavier…) et lit les résultats

Rôle du développeur : va lire les infos de l’utilisateur et va afficher/livrer des infos à l’utilisateur

**Opérateurs arithmétiques**

Exemple CFF 🡪 Comment faire pour changer prix du billet selon choix utilisateur

Une image contenant texte, tableau blanc

Description générée automatiquementOpérateurs arithmétiques : addition +, soustraction -, multiplication\*, division/ et modulo %

Modulo 🡪

**PRIORITE DES OPERATEURS**

Calcul 🡪 4+2\*5

Calculatrice basique – 30 / Calculatrice scientifique – 14

Le résultat juste est 14. La basique fait les calculs de gauche à droite, la scientifique applique les priorités.

Une image contenant texte

Description générée automatiquementUne image contenant table

Description générée automatiquement

Conversion

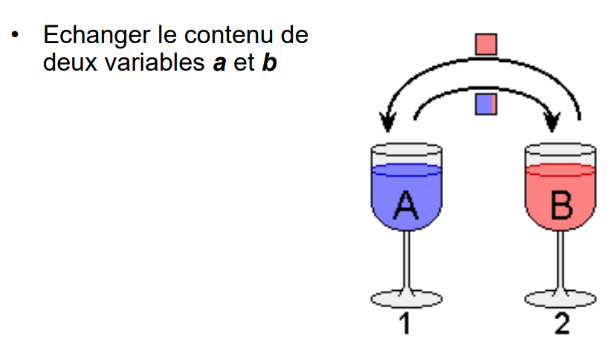
Conversion : ce que je stocke dans un byte je peux le mettre dans une grandeur plus grande mais attention pas inversement.

## Pratique

*Exercice*

Un verre de vin de la cave Bonvin. Un de la cave Briguet

On remplit malheureusement le verre de la cave Bonvin avec l’autre vin. Et inversement

Comment faire pour y remédier sans perdre le contenu ?

Etape 1 : On prend un 3ème verre.

Etape 2 : vider un verre dans le 3ème verrre.

Etape 3 : vider l’autre verre dans le deuxième

Etape 4 : remettre le troisième verre dans le premier.

Si j’imagine une variable int nb 1 = 4 ; int nb2 = 5 ;

Et que je veux inverser les valeurs des variables

1. Créer int temporaire ;
2. Prendre contenu nb2 et mettre temp : temp = nb2 ; (affectation)
3. Prendre contenu nb1 et mettre dans vide : nb2 = nb1
4. Nb2 = temp

*En java l’opération de droite à gauche se fait avec le signe égalité*

**Exercices sur Eclipse :**

Dans l’image ci-dessous on a appliqué l’exercice précédent sur Java.

Une image contenant texte

Description générée automatiquement

2ème exercice : On désire écrire un programme qui, étant donné une somme entière d’argent, calcule le nombre de billets de 1000, 200, 100, 50, 20, 10 que pourrait fournir un distribuer.

Une image contenant texte

Description générée automatiquement

*Petits Tips Eclipse :*

\*Dès qu’on travaille avec la variable on doit plus la déclarer (int)

\*Différence entre println et print : Println ça ramène à la ligne

\*System.in (entrée)   
\*Différence entre le next et le nextln : 🡪 ça ne met pas tout à la ligne

\*Scanner – permet que l’utilisateur mette des données

* Indiquer ce que l’utilisateur va devoir saisir (ici montant) = maSaisie.nextInt() ;

*Chapitre suivant : ne pas montrer les billets qu’il n’existe pas*

*Prochaine fois (avec Nicole) : refaire petits exemples qui sont donner dans le support*

# 04.10.2022

*Alternative simple*

**If-else**

* **Une image contenant table

  Description générée automatiquementUne image contenant texte

  Description générée automatiquement**Permet de programmer un choix

**Opérateurs relationnels**

On n’est pas forcément obligé de remplir le « else » (facultatif)  
*Exercices eclipses :*

exercicevaleuraleatoire

* Tips
  + 2 manières de générer des numéros aléatoires : 

exercicecinema

* Une image contenant texte

  Description générée automatiquement

exercicedivisiblepar9

Une image contenant texte

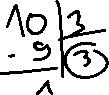
Description générée automatiquement

*Théorie sur le casting :*

On peut mettre un int dans un double mais pas l’inverse. Pour mettre un double dans un int il faut caster.

* Exemple : 3.2+4 = il va d’abord convertir le 4 en 4.0 = 3.2+4.0= 7.2
* Exercices opérateurs arithmétiques

**Ce que java fait 🡪 division entière – combien de fois 3 peux aller dans 10**



# 11.10.2022

Opérateurs arithmétiques : +, -, \*, /, %

Opérateurs relationnels : ==, <, >, …

Opérateurs logiques :

* && 🡪 ET
* || 🡪 OU
* Une image contenant texte

  Description générée automatiquement ! 🡪 NON

La machine fait plusieurs tests dépendant comment on écrit le programme :

* Ici il y a 3 SI, Java va contrôler les chiffres données dans

les 3 SI.

**Un if – else complet c’est une instruction complète**

**Choses à faire attention**

Une image contenant texte

Description générée automatiquementUne image contenant texte, personne, capture d’écran

Description générée automatiquementLes accolades sont nécessaires s’il y a deux instructions :

Une image contenant texte

Description générée automatiquement

* Visuellement, ça aide si c’est bien aligné.

Pour vérifier des if-else imbriqués on a deux outils :

* Une image contenant texte, tableau blanc

  Description générée automatiquementUne image contenant texte, tableau blanc

  Description générée automatiquementArbre à choix - Traces

Comment utiliser opérateurs logique :

**1er condition OPERATEUR LOGIQUE 2ème condition**

Une image contenant texte

Description générée automatiquementExemple : age<65 && age>=16

Exercice opérateurs logiques :

Si 2=5 et 5>3 🡪 FAUX (faux et vrai – faux)

Si 5<3 et 2=5 🡪 FAUX (faux et faux – faux)

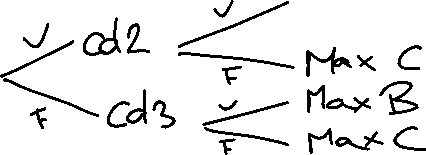
Si 5=2 ou 5>3 🡪 VRAI (faux ou vrai – vrai)

Si non 2 <= 5 🡪 FAUX (non vrai – faux)

Exercice :

Arbre de choix avec 3 numéros, trouver lequel est le max :

Condition 1 – A > B | Condition 2 – A > C | Condition 3 – B > C



# 18.10.2022

Les boucles

Les entiers ont a juste un problème de taille.

Combien de réels entre 0 et 1 ? une infinité

1 = 0.999999999999999 (c’est quand même très proche de 1)

C’est pour ça qu’on appelle des pseudos-réels. On ne peut pas tester des doubles entre eux !!!

C’est pour ça qu’on ne peux pas utiliser des doubles avec la commande Switch

\_

Les boucles

La boucle While

Une image contenant texte

Description générée automatiquement

While = tant que

Est-ce que ma somme (0) est plus petite que 100 ? 🡪 oui

Donc ça va écrire « Entrez un nombre : »

Somme **+=** 🡪 exemple si on fait 6+1 ça fait 7 🡪 et la somme vaudra 7

Façons de l’écrire

A = a+1 ;

A += 1 ;

A ++ ; (pas a voir mtn)

Prendre le contenu que j’ai dans 1, j’additionne quelque chose et le résultat je remet dans a

* A += nombre
* A = somme + nombre…

\_

Si on met 50

Le nombre vaudra 0 + 50 = 50

Ça remettra 50 dans somme

La différence avec if c’est qu’on reteste les conditions (boucle)

Est-ce que 50 est plus petit que 100 ? oui donc on va mettre à l’écran entrez un nombre

Imagine on entre 40

Somme += nombre

Ça fait 90 – on refait la condition

Est-ce que 90 est plus petit que 100 ? oui donc on rentre dans la boucle.

Je mets 60

60+90 = 150

Est-ce que 150 est plus petit que 100 ?

NON – réponse faux donc on ne retourne pas dans la boucle

MOTS-CLES :

Condition – se fait à l’entrée de la boucle. Elle est variable (ça change, on ne teste pas avec le même chiffre)

Il y a 3 types de boucle en JAVA. Mais on peut tout écrire avec la boucle While

Si la valeur initiale était de + de 100 on ne devrait pas passer dans la boucle

\_

# 25.10.2022

3 boucles : for, while, do/while

Boucle a une initialisation, condition et incrémentation

Boucle for 🡪 tout dans la même ligne

Une image contenant texte

Description générée automatiquement

Do.while 🡪 n’importe le test qu’on fait on passe une fois dans la boucle (même si 0)

* Syntaxe do/while (point virgule !!!!)
* Une image contenant texte

  Description générée automatiquement

Qu’a-t-on vu jusqu’à maintenant en programmation ?

* Types de base et variables
* Entrées (system.out.print/println/printf) et sorties (Scanner)
* Opérateurs arithmétiques, relationnels et logiques
* Alternatives simples et imbriquées
* Alternative multiple en java : Switch
* Boucles (while, for, do…while)

A voir la semaine prochaine (15.11.2022) :

* Les tableaux
* Les méthodes (ou fonctions) (par exemple math.random)

Version 2 Marienbad – ne pas accepter 0 !!!

**Boucles imbriquées**

Une image contenant texte

Description générée automatiquement

Exsuppboucles – ex1 faire a la maison

Devoirs : faire exercices imbriquées sauf 9

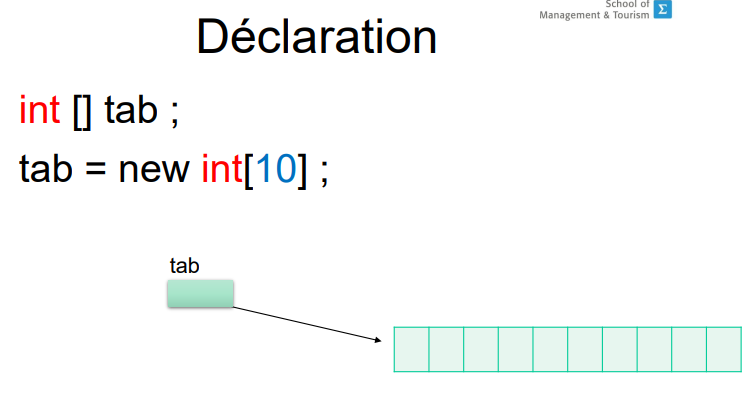
# 15.11.2022

Une image contenant texte

Description générée automatiquementLes tableaux :

On aimerait avoir un programme qui stocke tout mais qui est facile à exécuter, solution : les tableaux

Tableau = structure de données homogènes

* Regroupement de variables de même type sous un même nom

Si on veut déclarer :

Un tableau de 5 variables de type int : int [] tab ; // tab = new int [5]

On peut aussi faire double [] tab = new double [30]

Etc…

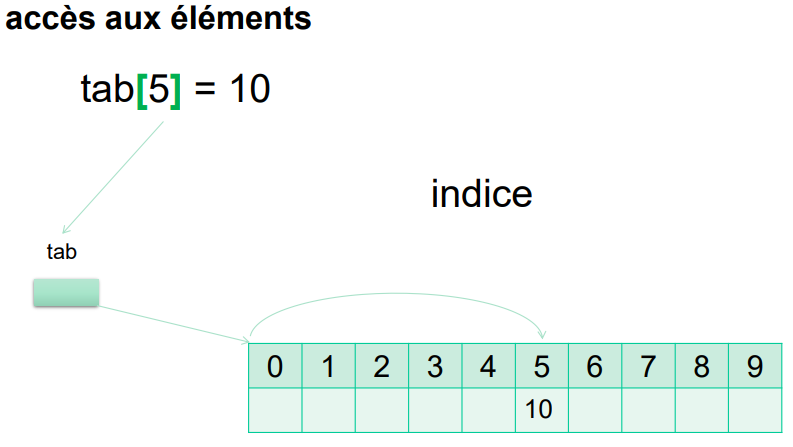
**Remarques :**

* + La taille du tableau est fixée par l’opérateur new
  + Il n’est plus possible de la modifier en cours d’exécution
  + Mais la taille peut être une variable

Une image contenant texte

Description générée automatiquement

* + Les tableaux sont des objets
  + nomTableau.length nous fournit la taille du tableau



Exercices :

On n’a pas le choix que de faire une boucle pour afficher les tableaux

Exercice 4 PowerPoint 🡪 le faire à la maison

Loterie 🡪 faire à la maison aussi

Problème 🡪 IndexOutOfBandException :

Si on a un tableau de 8 cases et qu’on en saisit 9 par exemple

+ EX STATS DES

* Voir réponses cyberlearn

# 21.11.2022

Méthodes

Math. 🡪 c’est une bibliothèque de méthodes

Math.random, math.pi, math.absolu…

On met un argument entre les 2 paranthèses.

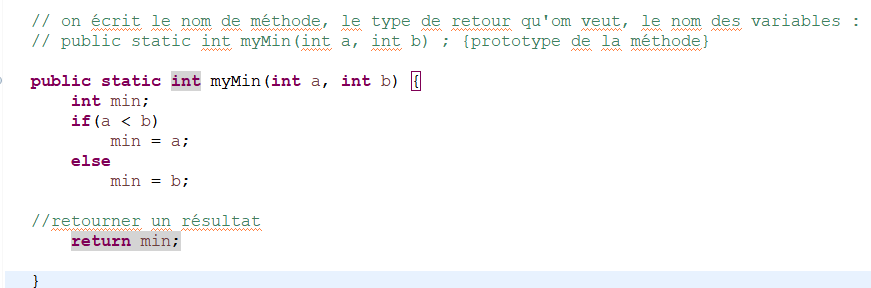
3 noms importants (exemple math.abs(-3) 🡪 3.0) :

* Nom de la méthode (abs)
* Argument (-3) 🡪 0, 1, 2 ou ?
* Retour (3.0) 🡪 0 ou 1

Est-ce que . ou , sur eclipse ? demander

Nous on appelle une méthode, mais il y a quelqu’un qui a écrit un code derrière cette méthode.

Ex : écrire sa propre méthode math.min



Paramètres formelles 🡪 (int x, int y) nom qu’on veut

Une image contenant texte

Description générée automatiquement

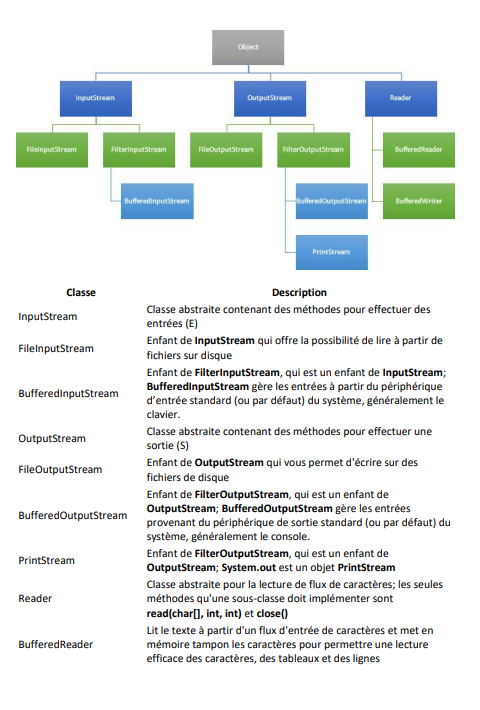
Cette méthode n’a pas de retour

Les méthodes commencent par des minuscules !!!

// factorielle – n !

Probabilités

5 ! =5\*4\*3\*2\*1

FICHIERS

[File Operations in Java - Javatpoint](https://www.javatpoint.com/file-operations-in-java)