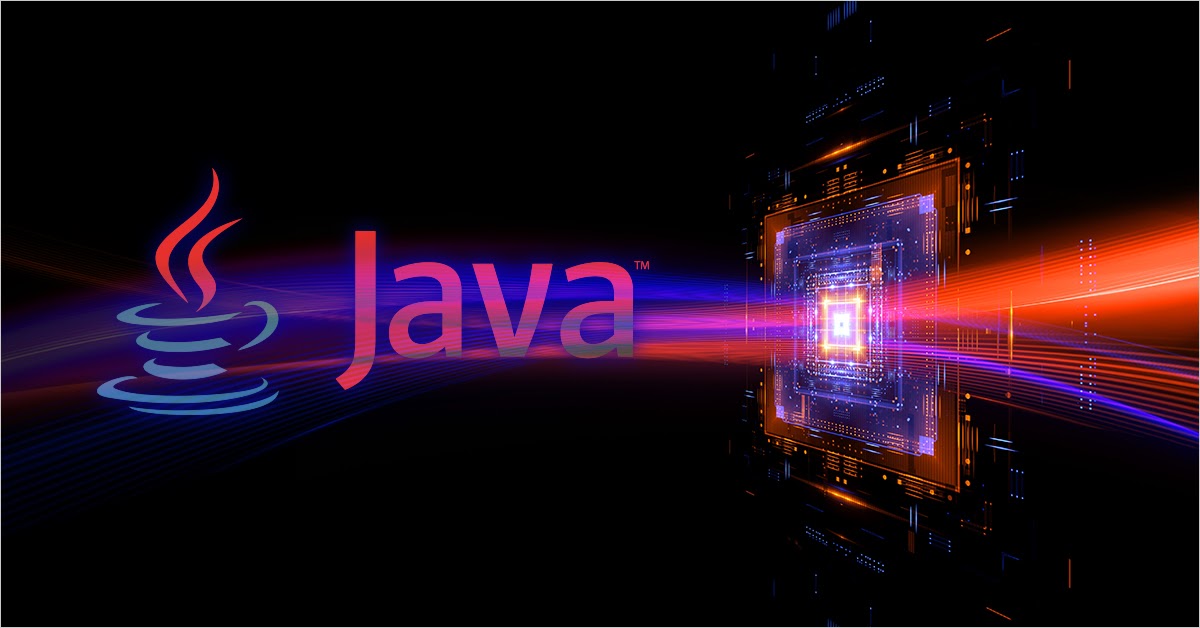
**Tutorials**

****

**HISTOIRE DE LANGAGE JAVA**

Le langage Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy, cofondateur de Sun Microsystems. Java a été officiellement présenté le 23 mai 1995 au SunWorld. La société Oracle racheta alors la société Sun en 2009, ce qui explique pourquoi ce langage appartient désormais à Oracle. La particularité et l’intérêt de Java réside dans sa portabilité entre les différents systèmes d’exploitations tels que Unix, Windows, ou MacOS. Un programme développé en langage Java, peut ainsi s’exécuter sur toutes les plateformes, grâce à ses frameworks associés visant à garantir cette portabilité.

**Les variables :**

Une variable est l'association d'une étiquette et d'une boîte pouvant contenir une donnée.

**Les Types des variables:**

**Primtive types :**

**boolean :** True/False

**byte :** numoccupes 8bits ( -128=<byte=<127 )

**short :** numoccupes 16bits ( -32768=<short=<32767 )

**int :** num occupes 32bits (-2\_147\_483\_648=<int=<2\_147\_483\_647) il faut qu’il soit réel

**long (ex: long myLong = 100L) :** num occupes 64bits

(-9\_223\_372\_036\_854\_775\_808=<long=<9\_223\_372\_036\_854\_775\_807 )

**float :** occupes 32bits utilisé pour les nombres avec une virgule

**double** (**float plus précisé):** occupes 64bits utilisé pour les

nombres avec une virgule

**char :** ex : (‘A’) un Caractère occupes 16bits  -unicode- (ex : char

copyrightSymb= ‘\u00A9’   )

**String : (**Normalement c’est **Class)**

Une chaîne de caractères, Donc il peut contenir plusieur caractère (

Techniquement   limiter par la mémoire ou ‘integer.MAX\_VALUE’(2.14B))

**Les Conditions:**

Une condition est une expression qui peut prendre l'une des deux valeurs suivantes **vrai** ou **faux**. On dit également que c'est une valeur de type **"logique"** ou **"booléen"**.

Les principaux opérateurs de comparaison que vous rencontrerez sont les suivants :

* égal à ( = en pseudo code)
* différent de ( != en pseudo code)
* strictement supérieur (**>**en pseudo code)
* strictement inférieur ( **<**en pseudo code)
* supérieur ou égal ( **> =** en pseudo code)
* inférieur ou égal (**< =** en pseudo code)

**Les Types des conditions:**

**If/else :**

 if(//condition)

{

  //Exécution des instructions si la condition est remplie

}else{

  //Exécution des instructions si la condition n'est pas remplie}

**Switch :**

 switch (/\*Variable\*/)

{

  case /\*Argument\*/:

    /\*Action\*/;

    break;

  default:

    /\*Action\*/;

**Condition ternaire :**

int x = 10, y = 20;

            int max = (x < y) ? y : x ;

**Les Fonctions:**

Une fonction est un bloc d’instructions qui retourne obligatoirement une et une seule valeur résultat à l’algorithme appelant. Une fonction n’affiche jamais la réponse à l’écran car elle la renvoie simplement à l’algorithme appelant.

**Les Types des Fonctions:**

**void:** void signifie que cette méthode n'a pas de valeur de retour (!return).

**int :** int retourne une numéro.

**String:** String retourne une chaine de caractéres.

**boolean:** retourne True ou False.

(exemples dans le breif)

**Les Collections:**

Java est un ensemble de classes et d'interfaces qui implémentent des structures de données de collection couramment réutilisables. Bien que référencé comme un framework, il fonctionne à la manière d'une bibliothèque

**Les Types des Collections:**

**List :**

  La classe **ArrayList** est un tableau redimensionnable, qui se trouve dans le package java.util. avec insertion a la fin .

parmi ses methodes: add/get/set/clear/remove…

**ex:** ArrayList<String> list= new ArrayList<String>();

La classe **LinkedList** est un tableau redimensionnable, qui se trouve dans le package java.util. avec insertion a au debut

parmi ses methodes: add/get/set/clear/remove…

**ex:** LinkedList<String> list= new LinkedList<String>();

**HashSet:**

Un HashSet est une collection d'éléments où chaqu’un de ces éléments est unique et il se trouve dans le package **java.util**.

parmi ses methodes: add/contain/size/clear/remove…

**ex:** HashSet<String> list= new HashSet<String>();

**HashMap :**

HashMap stocke les éléments dans des paires «key/value»(list.put("England", "London");), et vous pouvez y accéder par un index d'un autre type (par exemple, une chaîne). et il se trouve aussi  dans le package **java.util**.

parmi ses methodes: put/contain/size/clear/remove…

**ex:** HashMap<String, String> list= new HashMap<String, String>();