

Programme

- Introduction à la programmation informatique et aux différents langages de programmation.
- Présentation des concepts de base de la programmation de haut niveau. Aspect algorithmique. Introduction au langage Java.
- Syntaxe de base du langage Java, la sémantique et le concept de la compilation.
- Les variables, types et opérations mathématiques
- Opérateurs, affectation,
- conditionnelles et boucles (1 et 2)
- Tableaux (accès aux composants, ...).
- Procédures et Fonctions.
- Présentations de quelques classes clés.
- Des algorithmes fondamentaux.
- Des exemples applicatifs de l'ensemble des notions de Java présentées.

Aujourd'hui conditionnelles et boucles

- Définition: Tableau
- Tableau à une dimension
- Les tableaux multidimensionnels
- Manipulation d'un tableau

Définition

- Un tableau est une variable un peu particulière.
- nous pouvons lui affecter plusieurs valeurs ordonnées séquentiellement.

$\text{Tab1} = \{5, 1, 8, 3, 4, 7, 6, 7, 8, 0\}$

- Nous appelons ces valeurs au moyen d'un indice.
 - $\text{Tab1} = \{5, 1, 8, 3, 4, 7, 6, 7, 8, 0\}$
 - $\text{indice} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\};$

Tableau à une dimension (1/4)

Type de tableau

- Il y a autant de types de tableaux que de types de variables.
- un tableau d'un type donné ne peut contenir que des éléments du même type:
une variable de type int ne peut pas recevoir une chaîne de caractères par exemple. Il en va de même pour les tableaux.

Tableau à une dimension (2/4)

Déclaration d'un tableau

<type du tableau> <nom du tableau> [] = { <contenu du tableau>;

Dans la pratique:

```
int tableauEntier[] = {0,1,2,3,4,5,6,7,8,9};
```

```
double tableauDouble[] = {0.0,1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0};
```

```
char tableauCaractere[] = {'a','b','c','d','e','f','g'};
```

```
String tableauChaine[] = {"chaine1", "chaine2", "chaine3" ,  
"chaine4"};
```

- Il faut utiliser des « ' ' » pour initialiser un tableau de caractères,
- Des « " " » pour initialiser un tableau de String

Tableau à une dimension (3/4)

Déclaration d'un tableau à une dimension

On peut déclarer un tableau vide: il doit impérativement contenir un nombre de cases bien défini.

Par exemple, la déclaration d'un tableau vide de six entiers :

```
int tableauEntier[] = new int[6];  
//Ou encore  
int[] tableauEntier2 = new int[6];
```

Exemple applicatif:

- Déclarer et afficher le tableau suivant : tab= {1,2,3,4}.
- Afficher le dernier élément du tableau.

Tableau à une dimension (4/4)

Exemple de solution:

```
int t[] = {1,2,3,4}; // Déclaration d'un tableau de 4 éléments  
  
// Afficher les éléments du tableau  
for(int i=0;i<4;i++)  
{System.out.println("L'élément t["+(i+1)+"] est :"+t[i]);}  
System.out.println("La valeur du dernier élément du tableau est  
:"+t[3]);
```

Les tableaux multidimensionnels (1/4)

Définition:

Un tableau multidimensionnel n'est rien d'autre qu'un tableau contenant au minimum deux tableaux.

Exemple:

Soit un tableau multidimensionnel (2 lignes), appelé premiersNombres:

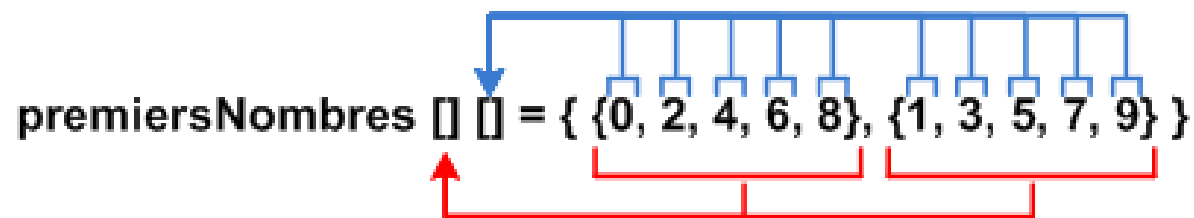
- la première contiendra les premiers nombres pairs,
- le deuxième contiendra les premiers nombres impairs.

**PremiersNombres: { {0,2,4,6,8},
{1,3,5,7,9} };**

Les tableaux multidimensionnels (2/4)

Déclaration d'un tableau multidimensionnels

```
int premiersNombres[][] = { {0,2,4,6,8},{1,3,5,7,9} };
```



Nous changeons de colonne par le biais de la première paire de crochets

Nous choisissons le terme d'un tableau grâce à la deuxième paire de crochets

Exemple applicatif:

- Déclarer et afficher le tableau suivant : `tab= {{1,2,3,4},{5,6,7,8}}`

Les tableaux multidimensionnels (3/4)

Exemple de solution:

```
public static void main(String[] args) {  
  
    int premiersNombres[][] = {{1,2,3,4},{5,6,7,8}};  
  
    for(int i = 0; i < 2; i++)  
    {  
        for(int j = 0; j < 4; j++)  
        {  
            System.out.print(premiersNombres[i][j]);  
        }  
        System.out.println("");  
    }  
}
```

Refaire le même exemple en utilisant cette fois-ci la commande
while

Les tableaux multidimensionnels (4/4)

Exemple de solution

```
int premiersNombres[][] = {{1,2,3,4},{5,6,7,8}},  i = 0, j = 0;

while (i < 2)
{
    j = 0;
    while(j < 4)
    {
        System.out.print(premiersNombres[i][j]);
        j++;
    }
    System.out.println("");
    i++;
}
```

Manipulation d'un tableau (1/10)

La taille d'un tableau:

On définit la taille d'un tableau grâce à l'instruction `length : <mon tableau>.length`

Exemple applicatif:

Déclarer un tableau contenant les données suivant: a, b, c, d, e, f, g et afficher son contenu en utilisant l'instruction `length`.

Manipulation d'un tableau (2/10)

Exemple de solution

```
public static void main(String[] args) {  
  
    char tableauCaractere[] = {'a', 'b', 'c', 'd', 'e', 'f', 'g'};  
  
    for(int i = 0; i < tableauCaractere.length; i++)  
    {  
        System.out.println("À l'emplacement " + i + " du tableau  
        nous avons = " + tableauCaractere[i]);  
    }  
}
```

Manipulation d'un tableau (3/10)

Opérateurs mathématiques

On peut utiliser les différents opérateurs arithmétiques (+, -, x, ...) et logique pour traiter des tableaux.

Copie d'un tableau

La copie intégrale d'un tableau dans un nouveau tableau peut donc se faire en utilisant la méthode `clone()`. La valeur retournée par cette méthode étant de type `Object`, il faut la convertir dans le type concerné.

Manipulation d'un tableau (4/10)

Exemple applicatif:

Ex1- Écrire un programme qui prend comme paramètre un tableau d'entiers int d'une dimension et qui renvoie la somme des entiers contenus dans le tableau

Ex2- Écrire un programme qui prend comme paramètre un tableau d'entiers int d'une dimension et qui renvoie la multiplication des entiers contenus dans le tableau.

Ex3 : Ecrire un programme qui copie un tableau dans un autre.

Manipulation d'un tableau (5/10)

Exemple de solution Ex1:

```
int tab[]= {1,2,3,4},  i = 0, sum = 0;

for (i = 0; i < tab.length; i++)
{
    sum += tab[i];
}

System.out.println("la valeur de sum est: " + sum);
```


Manipulation d'un tableau (6/10)

Exemple de solution Ex2:

```
int tab[]= {1,2,3,4},  i = 0, mult = 1;
```

```
    for (i = 0; i < tab.length; i++)  
    {  
        mult *= tab[i];  
    }
```

```
System.out.println("la valeur de sum est: " + mult);
```

Manipulation d'un tableau (7/10)

Exemple de solution Ex3.

```
int[] nombres = { 2, 3, 5, 7, 11 };  
int[] copie = (int[]) nombres.clone();  
int i=0;  
// tandis que copie contient toujours 2 3 5 7 11  
  
for (i = 0; i < copie.length; i++)  
{  
    System.out.println("la valeur de sum est: " + copie[i]);  
}
```