

Support de cours

Cours:

Initiation à la programmation (en Java)

Vidéo:

Init-JAVA-04-4-tableaux-affectation-pt3

Concepts (extraits des sous-titres générés automatiquement) :

Présentation des tableaux de taille. Nombre de valeurs de type. Type de boucle. Affichage du contenu d'un tableau. Saisie du contenu d'un tableau. Longueur du tableau. Fait de la situation. Ensemble des valeurs du tableau. Variable du même type. Saisie clavier classique. Éléments du tableau. Contenu du tableau. Différents éléments du tableau. Boucles. Indices du tableau.



[vers la recherche de séquences vidéo](#)
(dans Initiation à la programmation (en Java).)



[vers la vidéo](#)

Center for Digital Education. Plus de matériel de soutien pédagogique ici :

<https://www.epfl.ch/education/educational-initiatives/cede/educational-technologies-gallery/boocs-en/>
page 1/7

Tableaux : affectation et comparaison

(Partie 3)

Initiation à la programmation (Java)

Jamila Sam, Vincent Lepetit et Jean-Cédric Chappelier

...

notes

résumé


0m 0s



Pour terminer cette présentation des tableaux de taille fixe en Java, voici quelques petits exemples de manipulation routinière classique, comme par exemple l'affichage du contenu d'un tableau ou la saisie

résumé

0m 1s





Quelques exemples de manipulation de tableaux

Soit un tableau déclaré par :

```
double[] tab = new double[10]; ... tab[0] = 1.5;
```

Affichage du tableau :

- ▶ si l'on n'a pas besoin d'expliquer les indices :

```
System.out.print("Le tableau contient : ");
for(double element : tab) {
    System.out.print(element + " ");
}
System.out.println();
```

- ▶ si l'on veut expliciter les indices :

```
for(int i = 0; i < tab.length; ++i) {
    System.out.println("L'élément " + i + " vaut " + tab[i]);
}
```

du contenu d'un tableau. On suppose ici que l'on a au préalable déclaré une variable « table » qui peut contenir un certain nombre de valeurs de type « double ». On suppose également que, plus tard, le programme s'est chargé de remplir le tableau avec des valeurs que l'on estime adéquates, par exemple par des affectations de cette nature, et on s'intéresse à faire afficher le contenu du tableau. Il faut pour cela avoir recours à des itérations, des boucles « for », qui vont permettre d'itérer sur l'ensemble des valeurs du tableau pour les afficher une à une. Le type de boucle « for » à choisir, on a vu qu'il en existait deux, va dépendre en fait de la situation : est-ce qu'on aura besoin d'expliquer les indices ou pas. Dans le cas où il n'est pas nécessaire d'expliquer l'indice, on peut recourir à une boucle « for » de type « itération sur un ensemble de valeurs ». Nous avons vu que, pour ce type de boucle « for », il faut déclarer une variable du même type que les éléments du tableau. Cette variable va prendre tour à tour les différents éléments du tableau et on va pouvoir les afficher par cette simple tournure-là. Si l'on a besoin, par contre, d'expliquer les indices, alors on va avoir recours à une boucle « for » classique, où à ce moment-là il faut déclarer un indice, qui va varier de zéro jusqu'à la longueur du tableau. Nous avons vu que la longueur du tableau s'exprime par cette notation particulière. Sachant que les indices du tableau progressent entre zéro et taille -1, il faudra prendre la précaution de s'arrêter juste avant la longueur du tableau, et bien sûr, il faut faire progresser l'indice au fur et à mesure que l'on avance dans le tableau. Donc, à ce

notes

résumé

0m 13s



Quelques exemples de manipulation de tableaux

Soit un tableau déclaré par :

```
double[] tab = new double[10];
```

... $tab[0] = 1.5;$

Affichage du tableau :

- ▶ si l'on n'a pas besoin d'expliquer les indices :

```
System.out.print("Le tableau contient : ");
for(double element : tab) {
    System.out.print(element + " ");
}
System.out.println();
```

- ▶ si l'on veut expliciter les indices :

```
for(int i = 0; i < tab.length; ++i) {
    System.out.println("L'élément " + i + " vaut " + tab[i]);
}
```

moment-là, on peut faire afficher aussi bien l'indice,

notes

résumé

Saisie au clavier des éléments du tableau :

- Il est toujours nécessaire d'expliquer les indices :

```
→ for(int i = 0; i < tab.length; ++i) {  
    System.out.println("Entrez l'élément " + i + ":");  
    tab[i] = scanner.nextDouble();  
}
```

puisque'il est explicite, que la valeur qui lui correspond. Second exemple, celui de la saisie des éléments du tableau par une interaction clavier. Nous avons vu que la boucle « for » de type « itération sur un ensemble de valeurs » ne permet pas de modifier le contenu du tableau, et donc à ce moment-là nous n'avons pas le choix, il est absolument indispensable d'expliquer les indices en Java. Donc, nous aurons recours ici à une boucle « for » classique,

notes

résumé

1m 49s



itérant, comme tout à l'heure, de zéro jusqu'à la longueur du tableau en s'arrêtant à un indice juste avant, pour éviter le débordement, et à ce moment-là nous pouvons saisir chaque valeur du tableau séparément par une saisie clavier classique, comme nous avons déjà eu l'occasion de les pratiquer précédemment. de les pratiquer précédemment.

2m 13s

