



Support de cours

Cours:

## Initiation à la programmation (en Java)

Vidéo:

### Init-JAVA-06-7-Surcharge-pt1

Concepts (extraits des sous-titres générés automatiquement) :

**Surcharge de méthodes. Types de ces paramètres. Traitements de même nature. Listes de paramètres. Signature de la fonction. Paramètre de type int. Type des paramètres. Première méthode. Exemple int. Fonctions de mêmes noms. Type de leurs paramètres. Deuxième méthode. Nombres de leurs paramètres. Type de retour. Paramètre de type double.**



[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/>

# Fonctions : surcharge

## (Partie 1)

### Initiation à la programmation (Java)

Jamila Sam, Vincent Lepetit et Jean-Cédric Chappelier

...

notes

résumé

0m 0s



appliquant à des entités de types différents.

$\text{int } f(\text{int } x)$	$f(\text{int } x)$	signature = nom + types des paramètres
$\text{int } f(\text{double } x)$	$f(\text{int } x)$	

notes

0m 1s



## La surcharge de méthodes : exemple

```

→ static void affiche(int x) {
→ System.out.println("entier : " + x);
→ }
→ static void affiche(double x) {
→ System.out.println("reel : " + x);
→ }
→ static void affiche(int x1, int x2) {
→ System.out.println("couple : " + x1 + "," + x2);
→ }

```

affiche(1), affiche(1.0) et affiche(1,1) produisent alors des affichages différents.

*int*  
entier : 1  
*double*  
reel : 1.0

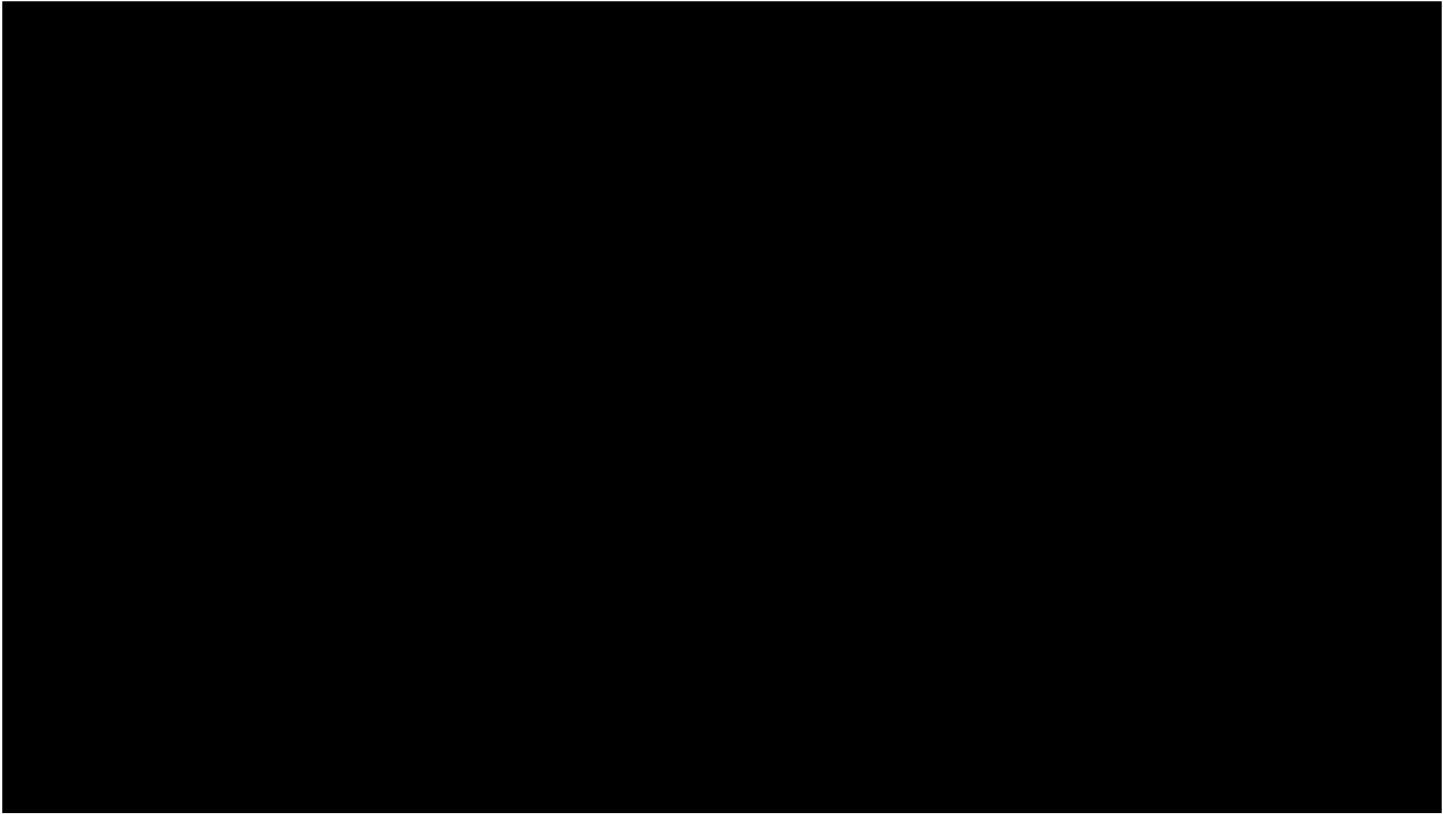
par exemple int pour la première, et double pour la seconde. Donc en résumé, en JAVA, on peut avoir des fonctions de mêmes noms, mais qui diffèrent dans le type de leurs paramètres. Par exemple je peux définir ces 3 méthodes qui s'appellent toutes les 3 affiche, et qui sont bien 3 méthodes différentes puisque leur liste de paramètres sont différentes. La première méthode a un paramètre de type int. La deuxième un paramètre de type double. Et la troisième 2 paramètres de type int. Quand j'exécute affiche(1), comme l'argument 1 est de type int, c'est la première méthode qui va être appelée et qui va m'afficher entier : 1. Quand j'exécute affiche(1.0), l'argument est de type double et c'est la deuxième méthode qui va être appelée ; qui va m'afficher réel : 1.0. Et quand j'exécute affiche(1,1) c'est la troisième méthode qui va être appelée.

notes

résumé

1m 37s





x1 va valoir 1, x2 : 1 également, et la méthode va m'afficher couple : 1,1 couple : 1,1

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

2m 49s



.....

.....

.....

.....

.....