

Support de cours

Cours:

Introduction à la programmation orientée objet (en C++)

Vidéo:

W17-02-affichage-CPP-pt1

Concepts (extraits des sous-titres générés automatiquement) :

Variable de type produit. Type montre. Hiérarchie de produits. Type réel de l'instance. Façon propre. Façon polymorphique. Étape précédente. Pointeur p. Étude de cas. Méthode d'affichage. Différents produits. Moyen de l'opérateur d'affichage. Affichage polymorphique des différents produits de notre hiérarchie. Propre façon. Informations spécifiques.



[vers la recherche de séquences vidéo](#)

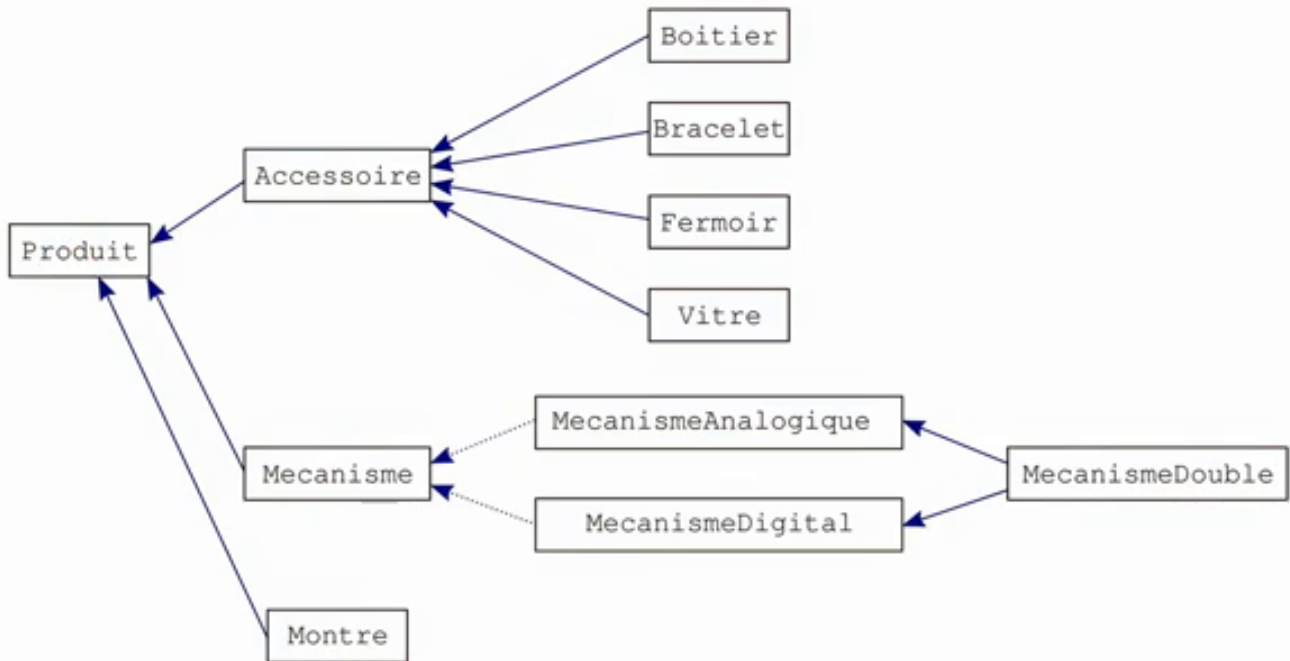
(dans Introduction à la programmation orientée objet (en C++).)



[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/>



Dans cette séquence, nous poursuivons notre étude de cas sur les montres, suisses ou pas, d'ailleurs et nous allons aborder l'affichage polymorphique des différents produits de notre hiérarchie. Pour rappel, donc, à l'étape précédente, nous avons ébauché une hiérarchie de produits et nous nous intéressons maintenant à comment faire en sorte que ces différents produits puissent s'afficher de façon polymorphique. Concrètement, un produit comme une montre, un mécanisme, ou un accessoire aura sa façon propre d'être affiché et nous voulons que cet affichage soit polymorphique. C'est à dire que si nous mettons un objet, par exemple, de type Montre dans une variable de type Produit

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

0m 0s



Affichage polymorphique

- Tous les produits sont « affichables », chacun à sa façon

☞ affichage polymorphique ?

```
Produit* p(new Montre (...)) ;
cout << *p << endl;
      ↘  montre !!
```

et que nous invoquons la méthode d'affichage sur la variable en question, l'affichage s'adapte automatiquement au type réel de l'instance stockée dans la variable. Chaque produit a donc sa propre façon de s'afficher et nous nous posons la question de comment procéder pour faire un affichage polymorphique. Nous aimerions bien que cet affichage puisse se faire au moyen de l'opérateur d'affichage. Donc, concrètement ici, si on imagine, par exemple que l'on déclare... un pointeur p sur un produit [Silence] on aimerait bien que "afficher le contenu pointé par p"... Affiche les informations spécifiques à la montre. Alors, question : est-il possible de programmer cet opérateur de sorte à avoir le comportement polymorphique souhaité ? le comportement polymorphique souhaité ?

notes

résumé

0m 37s

