

Support de cours

Cours:

Introduction à la programmation orientée objet (en Java)

Vidéo:

W17-04-interface-JAVA-pt3

Concepts (extraits des sous-titres générés automatiquement) :

Lignes de codes. Mécanisme analogique. Lignes suivantes. Mécanisme double. Méthodes toString. Différents constructeurs. Méthode d'affichage. Code de cette partie. Petit programme principal. Heure explicite d'initialisation. Valeur de base. Mécanisme précis. Méthodes d'affichage. Modélisation de notre hiérarchie de classes. Modélisation des mécanismes.



[vers la recherche de séquences vidéo](#)

(dans Introduction à la programmation orientée objet (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/>

Etude de cas : modélisation des mécanismes

(Partie 3)

Introduction à la programmation orientée objet (en Java)

Jamila Sam, Jean-Cédric Chappelier et Vincent Lepetit

...

notes

résumé

0m 0s



```
// test de l'affichage des mécanismes
MecanismeAnalogique v1 = new MecanismeAnalogique(312.00, 20141212);
MecanismeDigital v2 = new MecanismeDigital(32.00, "11:45", "7:00");
MecanismeDouble v3 = new MecanismeDouble(543.00, "8:20", 20140328, "6:30");
System.out.println(v1);
System.out.println(v2);
System.out.println(v3);
// Test des montres
Montre m = new Montre(new MecanismeDouble(468.00, "9:15", 20140401, "7:00"));
m.ajouter(new Bracelet("cuir", 54.0));
m.ajouter(new Fermeoir("acier", 12.5));
m.ajouter(new Boitier("acier", 36.60));
m.ajouter(new Vitre("quartz", 44.80));
System.out.println('\n' + "Montre m :");
m.afficher();
```

Le code complet à ce stade (298 lignes) peut être téléchargé sur le site du cours :

<https://d396qusza40orc.cloudfront.net/java-fr/complements/Montres02.java>

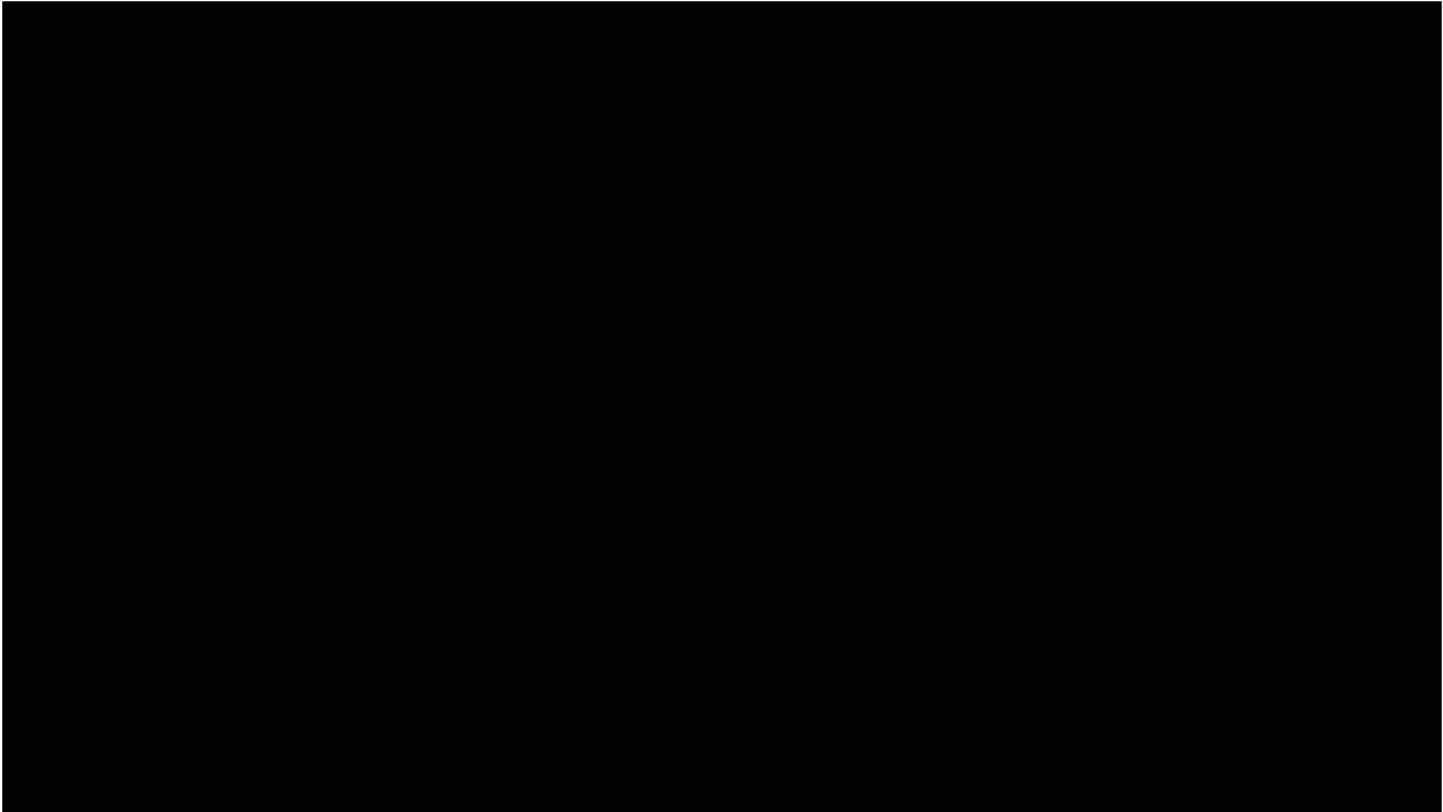
Voilà, nous en avons terminé avec la modélisation de notre hiérarchie de classes. Comme nous en avons désormais l'habitude, voici un petit programme principal permettant de tester les derniers développements. Ces lignes de codes permettront de tester nos différents constructeurs dans les sous-classes y compris ceux qui gèrent les valeurs par défaut non fournies. Ce constructeur construit un mécanisme analogique ayant pour valeur de base 312, ayant comme heure la valeur par défaut : c'est-à-dire 12.0, et ayant ceci comme date. Et cet appel construit un mécanisme double ayant pour valeur de base 543, ayant cette fois une heure explicite d'initialisation 8 heures 20, une date - puisqu'il s'agit d'un mécanisme double, il hérite du mécanisme analogique et donc a une date - et d'une heure de réveil. Les lignes suivantes permettront de vérifier que les méthodes toString, programmées aux différents niveaux, ont correctement été codées. Nous pouvons désormais construire une montre dotée d'un coeur qui correspond à un mécanisme précis et nous pouvons, comme nous le faisons précédemment, ajouter des accessoires à cette montre. Nous pouvons compléter la méthode d'affichage

notes

résumé

0m 1s





en permettant d'afficher aussi son coeur puisque désormais nous disposons de méthodes d'affichage pour les mécanismes et ceci nous permettra d'afficher une montre au complet. Le code de cette partie peut être téléchargé sur le site du cours. Et ceci conclut cette séquence sur la modélisation des mécanismes. sur la modélisation des mécanismes.

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

.....

.....

.....

.....

.....

1m 13s

