

TD / TP n°2

Objectifs:

Compréhension plus grande du modèle objet (visibilité, portée des variables, héritage). Le modèle objet doit être le plus détaché possible de l'implémentation, même s'il a une incidence dessus.

Notions à connaître:

Visibilité (publique, privée, etc...)

Héritage

Variables de classes et variables d'instance

Exercice 1:

Reprenons l'exemple du train, vu au TD précédent. Nous y parlions de gares, locomotives et wagons.

*- préciser la visibilité de chaque variable et méthode des classes vues dans le modèle.
Argumenter*

La composition d'un train dépend de la longueur du trajet (tant en kilomètres qu'en temps) qu'il aura à parcourir : un train de nuit n'aura qu'un seul wagon "normal", tous les autres étant remplacés par des wagons-couchettes. De plus, si le temps est supérieur à une heure, le train doit comporter un wagon-bar. La locomotive peut également changer en fonction du type de trajet envisagé : une locomotive de TGV ne sert à rien sur les petits axes, etc...

Ce genre de règles demande une gestion des wagons dans la gare.

- Rajouter dans le modèle les classes WagonBar, WagonLit, PetiteLocomotive, MoyenneLocomotive, GrosseLocomotive. Quelles relations impliquent t'elles avec les classes déjà existantes?

On remarque que l'existence des wagons- lit et wagons-bar ne remettent pas en question l'existence des wagons "normaux". Par contre, les sous-catégories de locomotives rendent la locomotive "normale" inutile.

- Comment modélise t'on une classe qui existe mais ne sera jamais utilisée?

Exercice 2:

Préparer le code correspondant aux classes de l'exercice précédent.

On peut imaginer dans un premier temps que la composition des trains ne dépend que de leur destination.

Ecrire une méthode "public void preparerTrain(String destination)" qui prépare les trains de la manière suivante:

- Sens -> 1 petite locomotive + 4 wagons normaux

- Troyes -> 1 moyenne locomotive + 4 wagons normaux + 1 wagon-bar + 4 wagons normaux
- Rome -> 1 grosse locomotive + 1 wagon normal + 1 wagon-bar + 4 wagons-lit + 1 wagon-bar + 4 wagons-lit

Exercice 3:

Modéliser le système solaire. On aura le soleil, les planètes, et leurs satellites. Comment peut-on intégrer le fait que tout, à part le Soleil, est un satellite pour autre chose? Comment peut-on intégrer dans le modèle que la Lune (qui est un satellite de la Terre) appartient au système solaire quand même?

A préparer pour le TP correspondant:

Nettoyer le pseudo-code en vue d'une implémentation sur machine.

A retenir pour le TD suivant:

Héritage, variables de classes et d'instance.