

Costa da Quinta, Joao Filipe

CLOO – TP7

Exo2:

Les deux classes suivantes sont suffisantes:

| article | facturation |
|---|---|
| <pre>-prix: int -categorie: String -style: String -photoNumerique: int -ID: int -dateRecuperation: String -etat: String -venduOuPas: bool</pre> | <pre>-IDitem: int -nomAcheteur: String -prenomAcheteur: String -adresseAcheteur: String</pre> |

En fait ça sera ensuite au logiciel de mettre tout en lien.

Faudra que article ait un constructeur comme suit:

article(this.categorie, this.style, this.photoNumerique, this.date, this.etat, this.provenance)

En effet il prend pas l'attribut prix, ni ID ni venduOuPas, le ID serait introduit automatiquement, en regardant le ID du dernier article crée (accès base de données). Comme ça chaque item peut être identifié par leur ID.

L'attribut venduOuPas est par défaut =False, mais quand on vent un article, alors on le change en =True.

Finalement, l'attribut prix sera ajouté manuellement lors de l'analyse du gérant.

Déroulement:

Le gérant arrive avec N nouveaux items, il indique alors au système qu'il en a N items, ainsi la date des prochains N items ainsi que la provenance sera la même (il la donne une fois plutôt que N).

Ensuite pour chaque item il doit introduire l'état, la catégorie, le style et finalement la photo.

Finalement, il fait appel à la fonctionnalité qui retourne une liste des items qui n'ont pas d'attribut prix, car c'est les items qui viennent d'être récupérés, pour chaque item on regarde leur style et la catégorie, et on cherche dans la liste de tous les items qui ont un prix, s'il y a des items qui ont le même style et catégorie, si on en trouve, on montre au gérant la photo, le prix, et le status de vente de l'item et la date d'acquisition. Il peut ainsi prendre une décision pour le prix de chaque item en fonction de l'état par rapport à celui qu'il a déjà, et le fait qu'il ait été vendu ou pas (s'il a déjà été vendu, alors le prix était adéquat/bas, si pas encore vendu, selon la date d'acquisition, faudrait peut-être baisser le prix).

Quand un article est acheté alors il crée un objet de la classe facturation qui prend en argument l'id de l'article vendu, et les coordonnées de l'acheteur, il peut ainsi enregistrer l'objet, et imprimer la facture. Faut aussi changer le status de vente de l'item vendu en =True.

Vu que un article qui a été vendu n'est pas supprimé de la base de données, pas besoin de mettre les infos de l'article dans la facturation, le ID qui est unique suffit.

EXO 3:

J'ai créé les classes qu'on a pas encore reçu, avec des noms de variables et méthodes complètes faux pour simuler mon code dans le package -> magasinComplement.

Dans le package magasin, j'ai créé magasin.class et ProduitDescription.class, il suffit de lancer le code dans magasin.class.

