

Durant cette séance, la majorité des groupes aura son premier Contrôle Continu de TD.

Il sera court, portant sur des questions de cours, les modalités peuvent varier légèrement d'un groupe à l'autre.

Le reste du temps de la séance sera occupé par ce TD.

Modélisation d'un site de vente

L'objectif général est de réaliser un site de vente sur internet qui propose des articles uniques. Lorsque plusieurs clients convoitent le même article, le site met en place une file d'attente virtuelle qui fonctionne sur le modèle du "premier arrivé, premier servi". Le premier client qui ajoute l'article dans son panier se place en tête. Les clients suivants qui font de même sont placés selon leur ordre d'arrivée. Si un client renonce au produit (c.à.d retire l'article de son panier), il est aussi retiré de la file d'attente correspondante.

Première partie

1. On définit une classe `Client` qui aura comme attributs **privés** :

- un attribut `nom` de type chaîne de caractères,
- un attribut `prenom` de type chaîne de caractères.

Écrivez cette classe en la munissant d'un constructeur. On veillera à ce que `nom` et `prenom` ne puissent pas être modifiés, et on écrira des méthodes `getNom` et `getPrenom` pour y accéder.

2. Définissez une classe `FileDAttente` avec un attribut `clientsPossibles` (un tableau) destinée à représenter la file d'attente de chaque produit. Puisqu'on a fait le choix d'utiliser un tableau, il vous faudra penser à introduire d'autres attributs pour en simplifier la future gestion.
3. Ajoutez un constructeur sans argument qui initialise une file d'attente.
4. Ajoutez une méthode `getTaille` qui renvoie le nombre d'élément qu'elle contient.
5. Écrivez une méthode `ajouterClient` qui prend garde d'éviter les doublons.
6. Écrivez une méthode `desiste` qui retire un client de la file d'attente.
7. Écrivez une méthode `extrairePremier` qui sort de la file, et retourne le premier client en attente
8. Écrivez une méthode `afficher` qui affiche les clients en précisant leur ordre d'attente, par exemple :
 1. Agnès Buzyn
 2. Olivier Véran
 3. Didier Raoult

Seconde partie

1. Écrivez une classe `Article` avec comme attributs :
 - `description` de type chaîne de caractères, invariable
 - `client` de type `Client`,
 - `file` de type `FileDAttente`,
 - `identifiant` de type entier.

Le champs `client` représente celui qui réserve actuellement l'article (éventuellement `null`), `file` représente tout ceux qui attendent. L'attribut `identifiant` est un numéro qui doit être unique pour chaque article. On écrira un constructeur qui prend en paramètre la désignation de l'article et on initialisera le reste des attributs.

2. Écrivez (dans la classe `Article`) une méthode `ajouterAuPanierDe` qui prend un objet `Client` en paramètre et tente de réserver pour lui. Si l'article est déjà réservé, on ajoutera le client à la file d'attente.
3. Écrire (dans la classe `Article`) une méthode `retirerDuPanierDe` qui prend un objet `Client` en paramètre :
 - si le client est celui qui réserve l'article, alors le client en tête de la file d'attente en est retiré et devient le nouveau réservant ;
 - si le client est dans la file d'attente, on le retire de la file d'attente ;

Troisième partie

On s'intéresse ici à la gestion de l'ensemble des articles d'un site.

Vous le modéliserez en interne à l'aide de deux tableaux, l'un correspondant au catalogue des articles encore en vente, et l'autre regroupant ceux qui ont été vendus.

Pour faciliter la lecture par les clients, le tableau correspondant au catalogue devra rester constamment ordonné par ordre alphabétique sur les descriptions.

Définissez et implémentez toutes les opérations de gestions utiles.