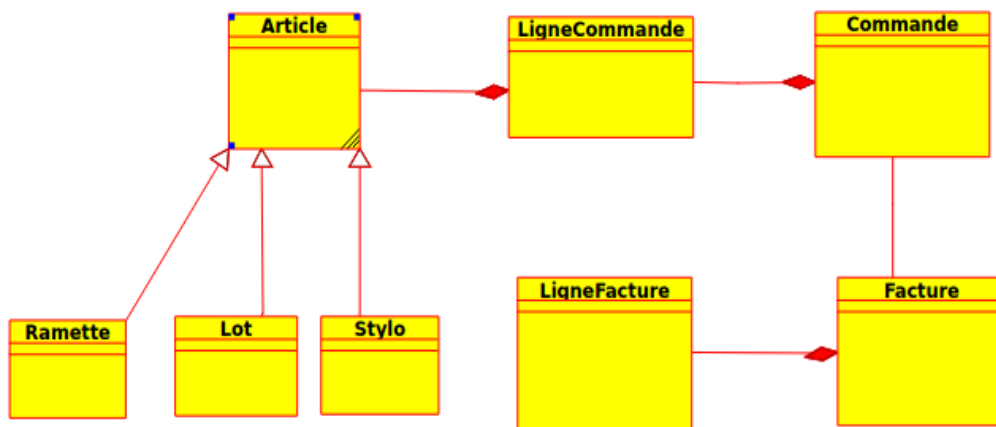


TP n°4

Entrées/sorties, sérialisation et réflexion

Système de facturation

L'objectif ici sera de mettre en place un système de facturation pour une entreprise. Cette entreprise établit des factures numérotées (en incrémentant partant de 1) et datées, comprenant le nom du client, le prix total de la commande et, pour chaque article commandé, sa référence, son prix unitaire, le nombre d'articles commandés et le prix total pour cet article. On adoptera le diagramme de classes suivant.



L'application affichera une facture pour

- 10 stylos (choisissez ses caractéristiques)
- 15 ramettes de papier
- 2 lots de 5 stylos (différents des 10 stylos)

et une autre facture pour

- 3 stylos
- 2 ramettes

pour 2 clients différents.

Une facture s'affichera sur le modèle suivant :

Facture 12345 Client: Toto Date: 10 novembre 2012

Quant.	Ref.	Nom	Marque	PU	PT
10	s1	Stylo Jade	Watertruc	76.0	760.0
15	r1	Ramette haute qualité	Clairefont	14.0	210.0
2	l1	Lot de 5 Stylo	Marker	686.0	1372.0

Prix total facture : 2342.0 euros

1) Mettez en œuvre une architecture MVC et identifiez le contenu de chaque partie de ce modèle (modèle, vue et contrôleur). Implémentez votre solution (en utilisant les classes fournies). Cette solution doit par exemple donner la main à l'utilisateur pour demander

l'affichage d'une facture en saisissant son numéro ainsi que le mode d'affichage voulu (à l'écran ou dans un fichier texte).

2) Dans ce système de facturation, on souhaite ajouter une fonctionnalité : on souhaite sauvegarder des commandes mise en instance ou suspendues par l'utilisateur. Autrement dit, les commandes non validées qui n'aboutissent pas encore à la création d'une facture seront sauvegardées. Proposez une première solution simple.

3) Maintenant on souhaite pouvoir visualiser par ailleurs ces commandes non validées pour inspection. Elles seront analysées par le service commercial. Proposez une deuxième solution pour rendre cela possible (rendre lisibles les enregistrements des commandes).

4) Votre application doit maintenant pouvoir récupérer une commande non validée à partir de son enregistrement en vue d'une facturation (validation). Mettez en place la solution qu'il faut pour rendre cela possible.

5) Échangez maintenant quelques articles sérialisés (fichiers XML). L'objectif de cette question est de pouvoir explorer un article que vous désérialisez dont vous ignorez l'origine. Utilisez la réflexion pour mettre en œuvre ce mécanisme.

6) On souhaite maintenant créer un article similaire à celui que vous avez récupéré suite à la désérialisation, et ce sans passer par l'opérateur new car vous n'êtes pas sensés connaître le type de cet article. Contrairement à l'article d'origine, cet article sera mis en promotion en positionnant le flag (de la classe article) « promotion » à 1. Faites-vous aider par la documentation officielle et utilisez les instructions suivantes pour mettre en place ce service :

```
Class c1 = a.getClass();  
Object o = c1.newInstance();  
Class[] types = new Class[]{};  
Constructor ct = c1.getConstructor(types);  
Object o2 = ct.newInstance();
```