

## Module : POO

### TD5- Classes abstraites et Interfaces

Un supermarché propose 3 types d'articles à vendre : des produits de grande consommation, des vêtements et des produits électroniques. Il offre des promotions sur certains articles pendant les vacances de printemps permettant des remises de 20% sur les produits électroniques et 30% sur les vêtements.

Soit une classe **Article** pour représenter les articles vendus dans ce supermarché. Cette classe possède les attributs protégés suivants :

- **reference** : un numéro qui caractérise l'article de manière unique.
- **libellé** : l'intitulé de l'article sous forme de texte.
- **prixHT** : le prix unitaire hors taxes de l'article.
- **qteStock** : le nombre d'unités disponibles de l'article, par défaut égale à 0.
- **TVA** : le pourcentage de la TVA qui est le même pour tous les articles et vaut 10%

1) Ecrire la classe **Article** en ajoutant les méthodes publiques suivantes :

- **Article(long ref, String lib, float p, int q)** : un constructeur.
- **void approvisionner(int nb)** : augmente la quantité disponible de l'article courant d'un nombre d'unités passé en paramètre.
- **void decrire()** : affiche la référence, et l'intitulé d'un article.
- **float calculPrixTTC()** : méthode abstraite qui retourne le prix TTC d'un article.
- **boolean appartientPromo()** : vérifie si l'article est un produit électronique ou un vêtement, sinon elle retourne faux.
- **boolean estDispo(int qteAchat)** : retourne vrai si la quantité demandée d'un article est disponible, sinon elle retourne faux.

2) Ecrire une interface **Promotion** ayant les éléments suivants :

- **debutPromo** : représente la date de début de la promotion fixée à "19/03/2022".
- **finPromo** : représente la date de début de la promotion fixée à "31/03/2022".
- **prixDeVente(String dateStr)** : méthode abstraite qui servira à retourner le prix d'un article avec ou sans remise selon la date d'achat.
- **estPeriodePromo(String dateStr)**: c'est une méthode default qui retourne vrai s'il s'agit bien d'une période de solde.
- **calculerRemise(double prixAvantRemise, double prixApresRemise)** : Méthode statique pour calculer la remise en pourcentage en appliquant la formule suivante :  $(\text{prixAvantRemise} - \text{prixApresRemise}) / \text{prixAvantRemise} * 100$ ;

Les attributs debutPromo et finPromo sont de type **LocalDate** du package **java.time.LocalDate** (voir annexe ci dessous)

3) Ecrire la classe **ProduitGC** qui décrit un article tel que le  $\text{prixTTC} = \text{prixHT} (1+\text{TVA})$ . Redéfinir la méthode *decrire()* pour afficher la classe de cet article.

4) Ecrire la classe **Vêtement** qui décrit un produit de grande consommation caractérisé aussi par sa **couleur** et sa **taille**. Redéfinir la méthode *decrire()* pour afficher aussi les attributs spécifiques. Cette classe implémente l'interface *Promotion* et définit la méthode *prixDeVente()*.

5) Ecrire la classe **ProduitElec** qui décrit un article ayant une **catégorie** (électroménager, TV, téléphones, ordinateurs) tel que le  $\text{prixTTC} = \text{prixHT} (1,08+\text{TVA})$ . Redéfinir la méthode *decrire()* pour afficher la classe et l'attribut spécifiques. Cette classe implémente l'interface *Promotion* et définit la méthode *prixDeVente()*.

6) Ecrire la classe **Achat** dans laquelle

- **supermarche** est un attribut statique qui décrit un tableau statique de 4 articles.
- La méthode *main* permet de remplir le tableau *supermarche* puis elle va décrire l'opération d'achat réalisée par un client le 21 Mars 2023 selon le tableau :

Référence	Libellé	Prix HT(DT)	Qté en stock	Qté acheté
123	Jupe bleue taille S	39,000	2	0
145	TV Led 80cm	1450,000	0	1
178	Pantalon noir taille M	42,000	5	1
191	Pâtes	0.410	18	3

Pour chaque article :

- Vérifiez que la quantité en stock est disponible
- Puis, affichez le produit acheté et son prix TTC à payer (calculez s'il y a une remise),
- Sinon, affichez que cet article est indisponible et approvisionnez-le.
- Ecrivez à la fin le montant total à payer.

## Annexe :

Pour manipuler les dates, Vous pouvez utiliser les méthodes suivantes de *LocalDate* :

- **public static LocalDate of(int year, int month, int dayOfMonth)** : retourne une instance de *LocalDate* représentant la date spécifiée par les paramètres.
- **public static DateTimeFormatter ofPattern(String pattern)** : retourne un objet *DateTimeFormatter* qui peut être utilisé pour formater des dates en chaînes de caractères. (dans notre cas, pattern= "dd/MM/yy")
- **public static LocalDate parse(dateStr, formatter)** est utilisée pour convertir une chaîne de caractères *dateStr* en un objet *LocalDate* en utilisant un *DateTimeFormatter* spécifié pour analyser la chaîne et interpréter sa représentation de date.