|  |  |  |
| --- | --- | --- |
|  | **Prosit 5: Programmation orientée objet(JAVA)** |  |

**« TuniProd » Gestion Catégories de Produits**

Les propriétaires des différents magasins voulaient gérer leurs magasins d’une façon qu’ils comportent divers types de produits.

En effet la gestion journalière des différents magasins consiste en l’entrée et la sortie de divers types de produits alimentaires (produit laitiers et produits agricoles…), on va s’intéresser aux produits agricoles qui peuvent être soit des légumes, soit des fruits ;

Tous les produits possèdent des caractéristiques définis auparavant. A chaque produit agricole est associé un autre attribut pour indiquer la saison de récolte de ce produit.

On veut manipuler toutes les catégories des produits à travers de la même représentation : celle d’un Produit.

1. Définissez les classes ProduitFruit **et ProduitLegume.**
2. Créer les produits suivants :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Produit | Id | Type | libellé | quantité | saison |
| P1 | 1254 | Fruit | Fraise | 12.3 | Mars |
| P2 | 1224 | Fruit | Pastèque | 50 | Juin |
| P3 | 7896 | Fruit | Mandarine | 25.6 | Décembre |
| P4 | 8521 | Légumes | Artichauts | 14 | Janvier |

1. Utiliser la méthode ***« ajouterProduit(Produit p) »*** pour ajouter ces produits à un magasin
2. Pour chaque produit de type **Fruit** on souhaite calculer la quantité totale de produit dans le magasin, pour cela on se propose d’ajouter la méthode **« float *calculStock()* »** dans la classe **Magasin**
3. Créer une nouvelle interface nommée **« Critere »** qui contient la méthode ***estFrais(String saison)*** *qui se comporte comme suit :*
   1. **Pour un ProduitFruit :** retourne **true** si la saison passée en paramètre est identique à la saison de récolte du produit
   2. **Pour un ProduitLegume :** retourne **true** si la saison passée en paramètre est ne dépasse pas la saison de récolte du produit de plus que 1 mois