

Programmation Orientée Objet

Les classes abstraites et les interfaces

Formatrice : Lafhal Joairia

1.1 Exercice 1

- 1- Créer une Interface appelé Imprimable.
- 2- Ajouter une méthode appelé Imprimer().
- 3- Créer deux classes, Article et Magazine, qui implémentent l'interface Imprimable.
- 4- Les caractéristiques de ses classes sont :
 - i. Article :
 1. Champs : Nom de l'article et le nombre de pages.
 2. L'implémentation de la méthode Imprimer () qui affiche le résultat suivant : « 'nom de l'article' est un Article imprimable sur 'nombre de pages' pages ».
 - ii. Magazine :
 1. Champs : Nom du magazine et le nombre de pages.
 2. L'implémentation de la méthode Imprimer () qui affiche le résultat suivant : « 'nom du Magazine est un Magazine imprimable sur 'nombre de pages' pages ».
- 5- Exploiter les classes créées.

1.2 Exercice 2 :

On désire réaliser une application pour une société qui fabrique et commercialise des appareils électriques :

Classe AppElectrique : On considère qu'un appareil est caractérisé par une référence, une puissance (exprimée en watt), un poids et un prix.

1- Créer l'interface « **Comparable** » qui contient la méthode abstraite **CompareTo()**

a. Ecrire la classe « **AppElectrique** » qui implémente la classe AppElectrique permettant de modéliser ces objets.

Ajouter à cette classe un constructeur permettant d'instancier des objets de la classe

« **AppElectrique** » dont on précisera la référence, la puissance, le poids, et un constructeur sans paramètres.

b. Réécrire la méthode **__str__()** pour afficher les caractéristiques de l'appareil.

c. Redéfinir la méthode **CompareTo()** qui permet de comparer deux appareils par leur puissance

2- Ecrire la méthode **ClasseEnergetique()** : qui permet d'afficher la classe énergétique de l'appareil

selon sa puissance :

- Si la puissance est inférieure strictement à 300 watt c'est la classe A
- Si la puissance est comprise entre 300 watt et 1000 watt c'est la classe B
- Si la puissance est supérieure à 1000 watt c'est la classe C

3- **Classe Television** : une télévision est un appareil qui possède un type d'écran (LCD, LED ...), et une fréquence(en hertz) :

- Ecrire une classe « **Television** » héritant de la classe « **AppElectrique** ».
- Récrire le constructeur de cette classe afin d'initialiser, en plus, le type d'écran et la fréquence avec des valeurs passées en paramètre. **(2 pts)**
- Réécrire la méthode `__str__` affichant les caractéristiques de la télévision.
- Redéfinir la méthode **CompareTo()** qui permet de comparer deux appareils par leur fréquence

4) **Classe VeloElec** : un vélo électrique est un appareil qui a une autonomie et un kilométrage.

- Ecrire une classe **VeloElec** héritant de la classe « **AppElectrique** ».
- Récrire le constructeur de cette classe pour définir, en plus, l'autonomie (exprimée en Km) et le kilométrage.
- Ajouter les méthodes suivantes :
 - Rouler (float distance)** : permettant au vélo d'avancer et retournant le nouveau kilométrage.
 - Charger (int nbrminute)** : permettant de charger la batterie et renvoyer la nouvelle valeur de l'autonomie (on suppose qu'une heure de charge donne 10 km en autonomie).
- Réécrire la méthode `__str__()` donnant les caractéristiques d'un vélo électrique.
- Redéfinir la méthode **CompareTo()** qui permet de comparer deux appareils par leur autonomie

5) **Classe Program** : Tester ces trois classes dans un programme principal.

- Créer un appareil électrique de type télévision
- Afficher cet article et afficher sa classe énergétique.
- Créer un article de type vélo électrique
- Faites rouler ce vélo
- Charger ce vélo
- Afficher le vélo