

EXAMEN FINAL TP PYTHON 2024

Dans cet exercice on s'intéresse à créer des classes pour gérer les vols d'une compagnie aérienne locale qui organise des vols entre des villes. Plus précisément on s'intéressera aux plans de vol entre les différentes villes. C'est les vols disponibles ainsi que l'heure de départ.

1. Pour créer une classe `Vol_direct` qui représentera un vol direct entre deux villes (pas d'escale dans une ville intermédiaire), on doit :
Définir le constructeur de cette classe qui a les attributs : Un nom du vol, `dep`, `arr` : qui désignent respectivement la ville de départ et la ville d'arrivée,
`jour` : qui désigne le jour de la semaine (lundi, mardi, ...)
`heure` (un entier entre 0 et 24 qui représente l'heure de départ) et la distance entre les deux villes.
2. Redéfinir la méthode `__str__()` pour avoir une chaîne bien formatée de la forme :

«Ce vol part de 'Tunis' vers 'Djerba' le 'lundi' à 9 heure, la distance est de xxxxx miles »
3. Créer une classe `Vols` qui représentera tous les vols le long de la semaine en utilisant la classe `Vol_direct`.
Pour ce faire on doit définir le constructeur de cette classe avec un seul attribut qui est une liste de vols. Utiliser également celui de la superclasse.
4. Écrire une méthode `Liste_successeurs` qui retourne une liste contenant les villes arrivées d'une ville de départ passée comme paramètre.
5. Écrire une méthode `Appartient` qui vérifie si une ville appartient au plan du vol que ce soit comme ville d'arrivée ou de départ.
6. Écrire une méthode `Piloter(.....)` qui permet d'affecter un pilote à un vol. Un pilote a un nom, un prénom et une adresse
7. Modifier la méthode `__str__` pour avoir la chaîne le « Le pilote est Mohamed BEN Mohamed »
8. Écrire une méthode `Affiche` qui affiche tous les vols directs
9. Écrire un programme principal permettant de
 - a. Créer une liste LV d'objets `Vol_direct`
NB : on suppose avoir définie les 3 fonctions suivantes :
`Saisie_Jour` qui retourne un jour valide,
`Saisie_Heure()` qui retourne une heure valide et `Saisie_Ville()` qui retourne un nom de ville valide.
 - b. Créer un objet `Vol` nommé `V` à partir de la liste déjà créée
 - c. Afficher tous les vols
 - d. Saisir une ville qui doit appartenir au plan du vol puis déterminer et afficher la liste de ses successeurs