

PRG. POO. Exercices

Exercice 1 : une classe *Fraction*

1/ **Créer** une classe *Fraction* ayant ces caractéristiques :

- Le constructeur doit signaler une erreur si le dénominateur est nul. Il y aura un attribut *numérateur* et *denominateur*.
- La méthode *affiche(self)* doit renvoyer une chaîne de caractères de la forme « 12 / 35 » ou simplement de la forme « 12 » si le dénominateur vaut 1.
- La méthode *__eq__(self, frac)* (qui permet d'écrire *objet == frac*) avec une autre fraction en paramètre (appelé *frac* ici) doit renvoyer *True* s'il y a égalité de l'objet appelant la méthode et de *frac*.
- La méthode *__lt__(self, frac)* (qui permet d'écrire *objet < frac*) avec une autre fraction en paramètre (appelé *frac* ici) doit renvoyer *True* si l'objet appelant la méthode est strictement inférieur à *frac*.

2/ **Faire** un jeu de tests.

Rappels :

- Une fraction est un nombre rationnel s'écrivant sous la forme a/b : a est le numérateur et b le dénominateur. Ce sont tous les deux des nombres entiers, b devant être en plus non nul.
- Soit deux fractions a/b et c/d : $a/b = c/d$ si et seulement si $a \times d = b \times c$.
- Pour comparer deux fractions a/b et c/d , il faut évaluer $a \times d - b \times c$.
 - Si la différence est strictement positive, alors $a/b > c/d$.
 - Si la différence est nulle, il s'agit d'une égalité (cas précédent).
 - Si la différence est strictement négative, alors $a/b < c/d$.

Remarque : en cas de dénominateur nul, il faudrait « lever une exception » et probablement quitter le programme. Cela sera étudié avec la partie avancée de la POO (héritage).

Exercice 2 : une classe *Date*

1/ **Créer** une classe *Date* ayant ces caractéristiques :

- Elle doit représenter un *jour*, un *mois* et une *année*.
- La méthode *affiche(self)* doit renvoyer une chaîne de caractères de la forme « 1 juin 1980 ».
- La méthode *__lt__(self, dat)* avec une autre date en paramètre (appelé *dat* ici) doit renvoyer *True* si l'objet appelant la méthode est strictement inférieur à *dat*.

Aide : il pourrait être judicieux d'utiliser un unique dictionnaire disponible pour tout objet (sans être un attribut) pour organiser les douze mois de l'années notamment pour pouvoir comparer deux dates. Il faudra donc le créer hors de la classe.

2/ **Faire** un jeu de tests.

A noter : ce système « de données disponibles pour toute instance sans être un attribut » est appelé *variable statique* dans certains langages à orientation objet comme Java, C++ et bien sûr Python. Elle est appelée directement à partir de la classe et non à partir de ses instances.

Ceci est hors-programme mais le cours propose un lien vers 3 vidéos très bien faites expliquant ce principe (et d'autres tout aussi hors-programme !) et son application en Python.

Important : il faut s'attendre à une épreuve pratique personnelle notée sur la création d'une classe.