

TP n°3.1

L'extension de la classe Date

Nous avons une classe Utile qui possède deux méthodes utiles permettant :

- de créer une Date depuis une chaîne de caractères (ou trois entiers),
- de formater une chaîne de caractère depuis une Date

Cette technique de créer une classe utile n'est pas une si bonne pratique que ça d'un point de vue programmation orientée objet. Car cette classe n'a pas pour but d'être instanciée.

Du coup à quoi sert cette classe, si on ne crée pas d'objet de celle-ci ?

D'un point de vue de la POO une classe sert à créer des objets.

Notre classe Utile ne possède que des méthodes "static" afin de pouvoir les appeler partout dans le programme. De ce point de vue, c'est très pratique et ça marche très bien. Mais on est clairement dans le procédurale et non plus dans la programmation orientée objet.

Du coup d'un point de vue de la POO ce n'est pas une bonne pratique.

Le langage java est orienté objet on va trouver un moyen de répondre tout en respectant ce paradigme.

1° - Créer une classe MaDate

1° - Votre classe MaDate va hériter de la classe java.util.Date.

2° - Ajouter un constructeur ou deux constructeurs afin de donner la possibilité de construire un objet de type MaDate. Réfléchissez aux paramètres (n'oubliez pas la doc).

3° - Surcharger la méthode toString() avec le comportement que vous souhaitez.

4° - Testez votre classe en l'utilisant dans votre code.

2° - Halala cette classe Date

Si vous doutiez encore de cette classe, ouvrez les sources et prenez le temps de regarder les auteurs de celle-ci.

Alors oui, on pourrait râler tout ce qu'on veut sur cette classe si elle avait été défini "final" (non héritable) mais il ne me semble pas que ça soit le cas. Ha, sacré James...