

## TP05 DÉFINIR DES CLASSES

### Exercice 1 : classe Nombre

Écrire la classe Nombre vu en cours .

Ajouter les méthodes suivantes :

- `int nombreDiviseurs` qui retourne le nombre de diviseurs d'un nombre ( exemple le nombre 16 a comme diviseur 8, 4, 2, 1 soit 4
- `estPremier` qui retourne un booléen si c'est un nombre premier c'est à dire si le nombre est divisible uniquement par 1 et par lui-même
- `estParfait` qui retourne un booléen si le nombre est parfait  
( voir : [http://fr.wikipedia.org/wiki/Nombre\\_parfait](http://fr.wikipedia.org/wiki/Nombre_parfait))  
les « int » parfaits sont très rares , par mesure de simplification vous pouvez initialiser un tableau dans lequel vous ferez figurer les 5 premiers nombres parfaits.
- `estAmi(Nombre nb)` qui reçoit comme paramètre un objet Nombre et qui retourne un booléen si c'est le Nombre nb est un nombre ami.

Écrire une classe TestNombre pour tester la classe nombre et les différentes méthodes

### EXERCICE 2 : CLASSE LOCATION

| Location   |
|--|
| <code>- nomClient: String</code><br><code>- categorie: char</code><br><code>- nbKm: double</code><br><code>- nbJour: int</code>  |
| <code>+ montantJour(): double</code><br><code>+ montantLocation(): double</code><br><code>+ kmSupplementaire(): int</code><br><code>+ prixKmSupplementaire(): double</code><br><code>+ montantTotal(): double</code> |

Définissez la classe Location

1 - Déclarez les attributs :

- `categorie` accepte les caractères suivants : E(Economique), C(Confort), L(Luxe)

2 - Implémentez 2 constructeurs

3 – Implémentez les getters et setters

4 - Implémentez les différentes méthodes :

- `montantJour` retourne le prix journalier de la location en fonction de la grille tarifaire ci-dessous :

| <i><b>Catégorie</b></i> | <i><b>E</b></i> | <i><b>C</b></i> | <i><b>L</b></i> |
|-------------------------|-----------------|-----------------|-----------------|
| <b>Prix / jour</b>      | <b>30,00 €</b>  | <b>50,00 €</b>  | <b>75,00 €</b>  |

- `montantLocation` retourne le montant de la location pour le nombre de jours de location
- `kmSupplementaire` retourne le nombre de km effectués dépassant ceux inclus dans la location.
  - Pour une location d'un jour, 100 km sont offerts.
  - Pour une location d'une semaine, 1000 km sont offerts. ( si une personne loue 10j elle a droit a 1000km + 3\*100km)
  - Pour une location de 30j, 5000km sont offerts. Il n'y a pas de location supérieur à 1 mois. (Si il n'y a pas de km supp la méthode retourne 0)
- `pxKmSupplementaire` retourne le montant engendré par les km supplémentaires (0,5€/km)
- `mtTotal` retourne le montant total

4 - Faites un programme pour tester votre classe :

- Créer un objet `Location` à partir des informations saisies par l'utilisateur.
- D'afficher le montant détaillé, si nécessaire résultant de cet objet.

### EXERCICE 3 : CLASSE DVD

Vous avez à votre disposition la classe suivante ( vous ne devez donc pas la programmer ) .

| <i>Class Date</i>  |
|--|
| <i>Constructor Summary</i>   |
| <b>Date</b> ( ) crée un nouvel objet date avec la date du système : date du jour |

1. Donnez les instructions pour créer, puis afficher la date du jour.
2. Donnez les instructions pour créer un Objet `GregorianCalendar` avec comme date le 10 février 2013.

|   |
|---|
| <b>GregorianCalendar</b> (int year, int month, int date)<br><small>Constructs a GregorianCalendar with the given date set in the default time zone with the default locale.</small> |
|---|

3. Donnez les instructions permettant de calculer la différence en nombre de jours entre la date du jour et une instance de la classe `GregorianCalendar`. (<http://java.mesexemples.com/date-et-heure/calculer-la-difference-entre-deux-dates/>)

#### 4. Définissez la classe DVD :

- a) Définissez les attributs privés
- b) Définissez 2 constructeurs.
- c) Définissez les getters et les setters .
- d) Définissez la méthode prixDVD() : double. Elle renvoie le prix d'un objet DVD sachant que celui-ci dépend de la couleur du DVD, un DVD rouge coûte 3 € la journée alors qu'un DVD vert ne coûte que 2 € la journée. Remarque : les valeurs possibles pour l'attribut couleur sont R pour rouge, V pour vert. En cas d'erreur sur la couleur, retournez 0 .
- e) Définissez la méthode isNouveaute () : boolean. Elle renvoie true si la sortie du DVD date de moins de 3 mois par rapport à la date du jour, renvoie false sinon.

#### 5. Utilisez la classe DVD :

- a) Créez un objet DVD avec les informations suivantes : « la porte » sorti le 05/02/06 étiqueté rouge
- b) Modifiez le titre du DVD : « la porte » => « la porte des secrets »
- c) Affichez son prix de location
- d) Affichez un message pour indiquer s'il s'agit d'une nouveauté ou non.

| <i><b>DVD</b></i>  |
|--|
| - titre : String<br>- couleur : char<br>- dateSortie : <b>GregorianCalendar</b><br>(Objet <b>GregorianCalendar</b> ) |
| + toString() : String<br>+ prixDVD() : double<br>+ isNouveaute() : boolean   |