

TP n°1

Introduction Eclipse

Pour chaque classe, générer le constructeur, les accesseurs et les modifieurs en utilisant Eclipse.

Exercice 1 Créer une classe `Arrondissement` qui contient un entier entre 1 et 20. On lance une exception `ArrondissementException` lorsqu'on veut créer un objet de type `Arrondissement` pour un entier supérieur à 20 et inférieur à 0.

Exercice 2 Créer une classe `Temps` qui contient 5 entiers correspondant à
jour/mois/annee heure :minute.

On lance une exception `TempsException` lorsque les paramètres des constructeurs et modifieurs ne sont pas corrects.

Exercice 3 Créer une classe `Evenement` dont les objets sont caractérisés par :

- le titre,
- la liste des dates auxquelles l'événement se produit,
- sa durée,
- l'arrondissement où il se produit.

Utiliser l'insertion automatique de `try...catch` pour traiter les exceptions.

Ajouter à la classe `Evenement` une méthode `afficher` qui devrait afficher le titre et les dates d'un événement. Tester cette méthode en utilisant l'approche décrite dans le cours (qui n'utilise pas une méthode `main`).

Créer les classes `Theatre`, `Film`, et `SpectacleEnfants` qui héritent la classe `Evenement`.

Un objet de la classe `Theatre` est caractérisé par : la liste d'acteurs, l'auteur, la salle et le genre.

Un objet de la classe `Film` est caractérisé par : une liste de paires (salle de cinema, liste de dates), le producteur, le genre, le site web, et la bande annoncée.

Un objet de la classe `SpectacleEnfants` est caractérisé par : l'âge des enfants qui peuvent participer.

Pour représenter le genre, créer une nouvelle classe `Genre` qui contient une chaîne de caractères avec des valeurs dans un ensemble fixé : comédie, tragédie, drame, thriller, action, histoire.

Surcharger la méthode `afficher` dans les classes `Theatre`, `Film`, et `SpectacleEnfants`.

Exercice 4 - Jouer avec les perspectives et les vues.

- Enlever/rajouter la hiérarchie des types et tester la sur les différents types de données.
- Pour la perspective `Java Browsing`, éliminer toutes les fenêtres dont vous ne savez pas à quoi elles correspondent ; faite la même chose pour la perspective `Java`.

Exercice 5 Créer une classe `Pariscope` qui contient la liste des événements qui se passe sur Paris. Implanter des méthodes pour ajouter ou supprimer des événements. Implanter deux méthodes de recherche avec le même nom, `recherche`, une qui renvoie la liste des événements qui se passe dans un arrondissement donné et une qui renvoie la liste de tous les théâtres qui se passe à une date donnée.

Exercice 6 Créer une classe `UtilisationPariscopes` qui contient :

- un attribut de type `Pariscopes`,
- une méthode `initialisation` qui ajoute des événements au `Pariscopes`,
- une méthode `main` dont les arguments sont utilisés pour faire des recherches dans le `Pariscopes`. Par exemple, si `args[] = {12, 21032011}` on appelle la méthode `recherche` paramétrée par l'arrondissement 12 et la méthode `recherche` paramétrée par la date 21-03-2011.

Exercice 7 Exécuter en ligne de commande votre projet.

Exercice 8 Annoter votre programme et générer le Javadoc.