

Billetterie Salle de concert

Introduction

Une société d'organisation événementielle désire gérer les concerts grâce à un programme.

Le programme **main** gère un ensemble de salles et d'évènements, il permet de :

1. Lister les salles de concert (nom, capacité de la salle, capacité de la scène), triée par ordre alphabétique de nom ainsi que les évènements liés à cette salle triés par date
2. Lister les évènements triés par date
3. Créer des événements (une date, une salle, un titre)
 - a. Pour chaque événement, on peut
 - i. ajouter des Humains (`addPublic()`)
 - ii. ajouter des Instrumentistes (`addArtiste()`)

Un bloc statique créera un ensemble de salles.

Création d'un événement

1. On doit donner une date, un nom et une salle de concert à un événement
 - a. L'utilisateur doit encoder la date dans le format 25/02/16 (dd/MM/yyyy)
 - i. Pour vérifier la date vous pouvez utiliser la classe **`java.text.SimpleDateFormat`** avec la méthode **`parse(String source)`**
 - b. Cette date doit être supérieure à la date actuelle + 2 jours
 - i. Vous pouvez utiliser la classe **`java.util.GregorianCalendar`** pour configurer la date actuelle + 2 jours
 - ii. NB : Les dates sont directement calculées en fonction du nombre de millisecondes écoulées depuis le 1er janvier 1970. Vous pouvez récupérer ce nombre (que cela soit sur un objet de type `Date` ou `Calendar`) et faire ainsi des comparaisons
 - c. L'utilisateur encode ensuite l'heure dans le format 20:00 (HH/mm)
 - i. Vérifiez que l'heure est correcte
2. On peut ajouter du public en fonction de la capacité de la salle.
 - a. Si l'ajout fonctionne, on reçoit une chaîne sous forme de : "Billet pour le spectacle *nom du spectacle* qui aura lieu le *jour/mois/année* à *heure :minute* en la salle *nom de la salle*. Numéro du billet : *numéro du billet*"
 - b. Si l'ajout ne fonctionne pas, une exception **`PlusDePlaceDisponibleException`** sera soulevée
3. On peut ajouter des artistes (Instrumentiste), aucune valeur ne sera retournée, si l'ajout ne fonctionne pas, une exception **`PlusDePlaceDisponibleException`** sera soulevée
4. On doit pouvoir connaître le nombre de places libres
5. On doit pouvoir récupérer la salle de concert

6. On doit pouvoir récupérer la date de l'événement
7. On doit pouvoir récupérer le titre de l'événement

NB : Le but est d'utiliser les collections. Vous devez stocker la liste des salles et des événements dans une collection ainsi que les événements liés à une salle.