

Projet

Vous gérez les résultats d'un marathon. Le responsable du chronométrage vous remet un fichier contenant le résultat de la course. Le fichier est un fichier texte où chaque ligne contient le résultat d'un concurrent. Sur une ligne, on trouve 5 champs séparés par un point virgule. Les champs sont les suivants :

- le prénom
- le nom
- le code de la nationalité (3 lettres)
- la catégorie du concurrent. Les catégories sont Msen, M35, M40, M45, M50, M55, M60, M65 et FSen, F35, F40, F45, F50, F55, F60, F65. Le M signifie que le concurrent est un homme, le F signifie que le concurrent est une femme. Toutes les catégories ne sont pas forcément présentes.
- le temps au format hh:mm:ss où hh représentent les heures, mm les minutes et ss les secondes.

Développer un programme qui va

- lire le fichier résultat (on donne plus loin un fragment de code dont vous pouvez vous inspirer)
- faire un menu et demander à l'utilisateur ce qu'il veut faire. La question est posée dans la console et l'utilisateur tape un nombre dans la console pour indiquer son choix. Les choix possibles sont les suivants :
 1. quitter le programme
 2. demander le classement général (toutes catégories confondues)
 3. demander le classement pour les hommes
 4. demander le classement pour les femmes
 5. demander le classement pour une catégorie
 6. le chronométrage pouvant avoir des ratés, il faut pouvoir ajouter à la main un résultat. Dans ce cas, on demandera toutes les informations à l'utilisateurs et on ajoutera le concurrent (attention, le responsable du chronométrage n'a peut être pas bien regardé et il se peut qu'en fait, le concurrent soit bien dans le système, dans ce cas, on conservera le résultat initial).
 7. demander la position d'un concurrent au classement général. Dans ce cas, on demandera ensuite le nom du concurrent. Si plusieurs concurrents ont le même nom, on les afficheras tous.
 8. disqualifier un concurrent. Dans ce cas, on demandera ensuite le nom du concurrent. Si plusieurs concurrents ont le même nom, on demandera alors le prénom.
 9. Dans quelque mois, l'organisateur de la course organise un relais mixte avec deux hommes et deux femmes. Le but de l'organisateur est d'avoir des équipes de niveau semblable pour augmenter le spectacle. En utilisant les résultats dont vous disposez, proposez 5 équipes à inviter.

Pour faire le menu, vous pourrez utiliser un objet Scanner qui accède à ce que l'utilisateur tape. Dans l'exemple suivant, le programme va attendre que l'utilisateur tape un nombre suivi de la touche entrée. Une fois que l'utilisateur a effectué cette opération, le nombre entré sera stocké dans la variable `choix`.

```
1 Scanner questionConsole = new Scanner(System.in);
2 int choice;
3 System.out.println("Que voulez vous faire?");
4 choice = questionConsole.nextInt();
```

Pour tester votre programme, vous trouverez un fichier nommé “result.csv” sur la page du cours. On donne dans la suite un fragment de code pour lire ce fichier.

```
1 try{
2     Scanner scanLine = new Scanner(new File("test.csv"));
3     while (scanLine.hasNextLine()){
4         String line = scanLine.nextLine();
5         System.out.println(line);
6         Scanner scan = new Scanner(line);
7         scan.useDelimiter(";");
8         String lastname = scan.next();
9         String timeS = scan.next();
10        Scanner ts = new Scanner(timeS);
11        ts.useDelimiter(":");
12        int hours = ts.nextInt();
13        int min = ts.nextInt();
14        int sec = ts.nextInt();
15        Duration duration = Duration.ZERO;
16        duration = duration.plusHours(hours);
17        duration = duration.plusMinutes(min);
18        duration = duration.plusSeconds(sec);
19        // do something with this data
20    }
21    scanLine.close();
22 }
23 catch(FileNotFoundException e){
24     System.out.println("Results file not found!");
25 }
26 catch(IOException e){
27     System.err.println(e.getStackTrace());
28     System.err.println(e);
29 }
```

Notez qu’on va utiliser un objet de type `Duration` pour représenter une durée. Cette classe a été introduite dans la version 8 de Java (dernière version en date). L’affichage utilise une norme qui n’est pas très lisible. Ecrivez une méthode pour écrire le temps en adoptant le format de l’exemple suivant : `1h28'06"`.

Ce projet est à faire en binôme. Il existe plusieurs architectures pour répondre à ce problème. Votre solution doit utiliser aux mieux les atouts d’un langage orienté objet.

documents à rendre :

- tous les fichiers sources commentés (i.e. les fichiers `.java`)
- un texte court qui motive vos choix d’architecture et qui indique les principales difficultés rencontrées.

remise du projet : par email à `stephane.airiau@dauphine.fr` le lundi 10 Octobre (ou avant !)