

## Entrées-sorties

### Exercice 1

---

Vous devriez être maintenant familier avec la page de documentation de java. Dans la version avec les frames, elle contient la liste de toutes les classes que vous pouvez retrouver à cette adresse : <https://docs.oracle.com/javase/8/docs/api/allclasses-frame.html>

- combien de classes sont présentes dans la version 8 ?
- Donnez les noms de classes ou interfaces qui ont des doublons.

### Exercice 2

---

- Ecrivez un menu qui contient les options suivantes
  1. quittez l'application
  2. affichez le contenu du répertoire courant
  3. entrez dans un repertoire
  4. allez dans le repertoire parent
  5. affichez le contenu d'un fichier

Pour l'instant, créez juste le menu, on va coder les fonctionnalités dans ce qui suit. Le menu doit être robuste aux fautes de frappe de l'utilisateur. Le programme ne doit s'arrêter que si l'utilisateur tape 1.

On initialisera le répertoire courant avec le repertoire où vous lancez votre programme (.).

- Lorsque l'utilisateur tape 2, on va afficher les fichiers et les répertoires du répertoire courant. Pour cela, la méthode `newDirectoryStream` de la classe `Files` renvoie un itérable sur des `Path`. En affichant, vous pouvez indiquer à l'utilisateur s'il s'agit d'un fichier ou d'un répertoire (il existe une méthode dans la classe `Files` pour déterminer si un chemin correspond à un répertoire).
- Lorsque l'utilisateur tape 3, on lui demandera de choisir un répertoire et on mettra à jour le repertoire courant. Pour cela, regardez ce que fait la méthode `resolve` de la classe `Path`.
- Lorsque l'utilisateur tape 4, on mettra à jour le répertoire courant avec le répertoire père du répertoire courant. Une méthode de la classe `Path` semble particulièrement adaptée!
- La dernière fonctionnalité est d'afficher le contenu d'un fichier. Pour se faire, on pourra au préalable vérifier que le fichier existe bien et que le fichier n'est pas en fait un répertoire. Ici, des méthodes de la classe `Files` semblent particulièrement pertinentes.

## Exercice 3

---

Reprenez votre classe Box codée lors du premier TD.

1. écrivez une méthode **public void** `save(String filename)` qui sauvegarde l'instance de Box dans un fichier de nom `filename`.
2. écrivez une méthode **public static** `Box restore(String filename)` qui récupère une instance de la classe Box sauvegardée dans le fichier de nom `filename`.
3. écrivez une méthode `main` pour testez vos nouvelles fonctionnalités.