

Les entrées/sorties et expresion régulière

Exercice 1 - Manipulation de fichier texte

On cherche à écrire un programme effectuant la copie des lignes paires d'un fichier texte.

Vous utiliserez pour cela les classes du paquetage `java.io`.

Le fichier est pris en temps que premier paramètre sur la ligne de commande et la sortie est effectuée sur la sortie standard.

- 1 Afficher dans un premier temps, toutes les lignes du fichiers une par une, en utilisant la méthode `readLine()` de `BufferedReader`.
- 2 Modifier le programme pour afficher seulement les lignes paires. Vous pourrez aussi remplacer le `BufferedReader` par un `LineNumberReader`.

Exercice 2 - Parsing de XML à la main

On cherche à écrire un petit parseur XML à la main permettant de lire le code suivant :

```
<toto>
  <titi>
  </titi>
  <tutu>
  </tutu>
</toto>
```

Pour chaque balise (markup ou element) ouvrante trouvée, vous appellerez la méthode `startElement()` et pour chaque balise fermante trouvée, la méthode `endElement()`.

```
public class MyXMLParser {
    public void startElement(String element) {
        ..
    }
    public void endElement(String element) {
        ...
    }
    public void parse(Reader reader) throws IOException {
        ...
    }
}
```

- 1 Compléter le code ci-dessus en écrivant la méthode `parse` en vous aidant du paquetage `java.util.regex`. On supposera qu'une balise ne peut se trouver sur plusieurs lignes en même temps.
- 2 Modifier votre programme pour qu'il prenne aussi en compte les balises suivante :

```
<tata/>
<tjtj hello="3" banane="true">
</tjtj>
```

Quelle méthode doit-on appeler pour `tata` et `tjtj` ?

Exercice 3 - Copie de fichiers

On cherche à écrire un programme effectuant la copie de fichiers binaire.

- 1 Écrire dans un premier temps, une copie de deux fichiers dont les noms sont pris sur la ligne de commande :

```
java Copy fichier.in fichier.out
```

On considerera les fichier comme contenant n'importe quoi donc à recopier en binaire.

- 2 Modifier le programme pour utiliser un tableau de 8192 octets pour le transfert.
- 3 Modifier le programme pour que si les fichiers ne sont pas spécifiés sur la ligne de commande le programme utilise l'entrée standard (System.in) et la sortie standard (System.out).

PS: ne faites pas de copier/coller.

Exercice 4 - Manipulation de fichier binaire et Serialisation

On cherche à lire/sauvegarder un ensemble de points de la classe `java.awt.Point` qui comporte deux champs entiers `x` et `y`.

- 1 Créer un programme permettant de sauvegarder les points en utilisant un `DataOutputStream`. Créer un programme permettant de lire le résultat obtenu.
- 2 Modifier les programmes précédents pour utiliser les classes `ObjectOutputStream` et `ObjectInputStream`.

Exercice 5 - Lister le contenu d'un répertoire

Le but est d'écrire un petit programme permettant de lister le contenu d'un répertoire comme `ls`. Vous ferez attention à écrire le code de façon "objet".

- 1 Écrire dans un premier temps, un programme qui liste le contenu du répertoire courant.
- 2 Modifier le programme pour qu'il puisse prendre optionnellement un fichier ou un répertoire en argument. Puis ajouter l'option `-a` qui permet de lister les fichiers cachés.
- 3 Ajouter l'option `-R` qui permet de parcourir récursivement les répertoires.
- 4 Ajouter l'option `-s` qui trie les fichiers.
- 5 Ajouter l'option `-r` regex qui affiche uniquement les fichiers/répertoires correspondant à l'expression régulière.
- 6 Ajouter l'option `-l` qui affiche en plus du nom
 - la taille
 - la date de dernière modification
 - les droits (lecture/écriture)