

PARTIEL 2

Seuls sont autorisés, à titre de documents, les listings comprenant explicitement le nom imprimé de l'étudiant (et celui-ci seulement) et les notes manuscrites comportant le nom de l'étudiant.

Les deux parties sont indépendantes mais elles doivent être présentées dans l'ordre (prévoir deux pages par exercice).

Le but est de réaliser un serveur et un client permettant de transmettre des images sur le réseau informatique.

Exercice 1.- (Serveur d'actualités)

Dans le répertoire de l'application serveur se trouve un sous-répertoire de nom "images" qui contient des images de format .jpg contenant des actualités telles que meteo.jpg pour la carte météorologique actualisée toutes les heures ou tv.jpg pour le programme de télévision, actualisé au moins une fois par jour mais également en cas de reprogrammation.

Écrire une application Java serveur (sur le port 8888) qui reçoit une ligne de caractères d'un client :

- "QUIT" pour terminer la connexion.
- "METEO" pour envoyer la carte météo du jour.
- "TV" pour envoyer le programme TV.

[*Pour simplifier, on ne vérifiera pas complètement que le nom de l'image .jpg demandé par le client. On enverra le programme TV par défaut.*

Il vaut mieux utiliser les méthodes permettant de lire et d'écrire un caractère à la fois dans un fichier, et non ligne par ligne.]

Exercice 2.- (Client graphique d'actualités)

Écrire une application Java client qui affiche un cadre fermant dans lequel apparaît :

- une étiquette "Actualités" ;
- une liste déroulante dans laquelle apparaissent les choix possible (meteo, tv, ...) ;
- un bouton "ok" (qui permet de demander le chargement après avoir choisi le thème d'actualités voulu) ;
- l'image désirée lorsque celle-ci est reçue depuis le serveur.

[Comme d'habitude en Java, on ne cherchera pas à bien placer les widgets les uns par rapport aux autres.

On se servira d'un fichier `temp.jpg` pour y entreposer le fichier rapatrié.

L'organisation fonctionnelle devra être très clairement indiquée par des commentaires séparant chaque partie.

Il vaut mieux utiliser les méthodes permettant de lire et d'écrire un caractère à la fois dans un fichier, et non ligne par ligne.

Le client a intérêt à établir une connexion pour chaque image d'actualité demandée.]

Documentation

1 Liste déroulante

La classe :

```
Choice
```

du paquetage :

```
java.awt
```

est un composant (widget) qui permet de mettre en place une liste déroulante.

Instantiée par le constructeur par défaut, la méthode :

```
void add(String)
```

permet d'ajouter un item qui apparaîtra dans la liste déroulante.

La méthode :

```
String getSelectedItem()
```

permet de récupérer le nom de l'item sélectionné par l'utilisateur.

2 fichiers

La classe `InputStream` (dont une classe dérivée est la classe `FileInputStream`) possède la méthode :

```
public int read()
```

noindent qui renvoie l'octet suivant à lire dans le fichier, ou -1 si on est arrivé à la fin de fichier.

La classe `FileOutputStream` possède la méthode :

```
public void write(int)
```

qui permet d'écrire un octet dans le fichier (malgré le type).