

0.1 SDK Java 2 Standard Edition

0.1.1 1. Introduction :

Le Java Standard Development Kit, parfois appelé JDK, est le kit de développement basique proposé par Sun Microsystems. Ce Kit, en plus de contenir Java Runtime Environment (JRE), comprend aussi plusieurs outils :

- javac : le compilateur Java
- java : un interpréteur d'applications (machine virtuelle)
- applet viewer : un interpréteur d'applets
- jdb : un débogueur
- javap : un décompilateur, pour revenir du binaire (fichier compilé .class) au code source (fichier source .java)
- javadoc : un générateur de documentation
- jar : un compresseur de classes Java

Le compilateur javac :

C'est un compilateur qui transforme le code source en binaire interprétable par la machine virtuelle sur n'importe quelle plate-forme. Sa syntaxe est la suivante :

```
javac -g nom_du_fichier.java
```

L'option -g permet d'inclure des informations de débogage afin de pouvoir utiliser le débogueur jdb.

L'interpréteur :

L'interpréteur java est une machine virtuelle fonctionnant en mode texte. Sa syntaxe est la suivante :

```
java nom_du_fichier
```

L'interpréteur d'applets :

appletviewer permet de visualiser l'exécution d'un applet (il est aussi possible de la visualiser sur un navigateur ayant le plugin java). Sa syntaxe est la suivante :

```
jappletviewer nom_de_l_applet.class
```

Javadoc :

JavaDoc a pour but de créer une documentation au format HTML à partir du code, grâce aux commentaires prévus à cet effet (/**). (Un exemple du type de documentation qu'il peut produire est la spécification de l'API Java)

Le compresseur de classes :

Jar permet la compression des classes Java afin de réduire leurs tailles et de rendre leurs téléchargements plus rapides. Vous pouvez télécharger ce kit ici : <http://java.sun.com/j2se/1.4.2/download.html>. Vous y trouverez aussi la documentation de l'API. Le kit est proposé sous plusieurs formats d'installation différents :

- Linux GNUZIP Tar shell script, qui est un fichier binaire auto-extractible sous la forme `j2sdk-1_4_2_03-linux-i586.bin`, utilisé pour installer l'environnement java dans un répertoire choisi par l'utilisateur. Il est aussi disponible avec l'IDE Netbeans 3.5.1 sous la forme `j2sdk-1_4_2-nb-3_5_1-bin-linux.bin`. Nous ne traiterons pas ce dernier puisqu'il s'installe de la même façon que le premier et que nous installerons Netbeans séparément.
- Linux Red Hat shell script fichier (`j2sdk-1_4_2_03-linux-i586-rpm.bin`), qui est en fait un package RPM contenant l'environnement Java.

0.1.2 2. Installation du fichier binaire auto-extractible :

Après le téléchargement du fichier, il est nécessaire de créer le répertoire où sera installé le kit. Dans notre cas, il sera installé dans `/usr/java` :

```
# mkdir /usr/java
```

Copiez ensuite le fichier téléchargé (`j2sdk-1_4_2_03-linux-i586.bin`) dans ce répertoire. Ici, le fichier a été téléchargé dans le répertoire `/usr/src`.

```
# cp /usr/src/j2sdk-1_4_2_03-linux-i586.bin /usr/java
```

Passez dans le répertoire `/usr/java` :

```
# cd /usr/java
```

Il s'agit maintenant de lancer le script d'installation en utilisant les commandes suivantes :

```
# chmod a+x j2sdk-1_4_2_03-linux-i586.bin
```

```
# ./j2sdk-1_4_2_03-linux-i586.bin
```

Une fois la licence acceptée, le script va installer java dans un répertoire appelé `j2sdk1.4.2_03`

0.1.3 3. Installation du fichier RPM :

La procédure à suivre dans le cas du package RPM est identique à la première méthode. Mettez vous dans le répertoire où a été téléchargé J2SE SDK et lancez directement le script d'installation via les commandes :

```
# chmod a+x j2sdk-1_4_2_03-linux-i586-rpm.bin
```

```
# ./j2sdk-1_4_2_03-linux-i586-rpm.bin
```

Une fois l'accord de licence accepté le package RPM est installé automatiquement dans `/usr/java/j2sdk-1_4_2_03`. Déplacez-vous dans le répertoire `/usr/java/j2sdk1.4.2_03`, par :

```
# cd /usr/java/j2sdk-1_4_2_03
```

Utilisez la commande `rpm` pour installer le package comprenant l'environnement Java :

```
# rpm -ivh j2sdk-1_4_2_03-linux-i586.rpm
```

0.1.4 4. Ajouter Java dans le PATH :

Pour que les commandes Java soient accessibles par l'ensemble des applications, il est nécessaire de créer un script qui les exportera lorsqu'une personne se loguera sur la machine. Tout d'abord, il faut créer le script suivant :

```
export PATH=$PATH :/usr/java/j2sdk1.4.2_03/bin
export CLASSPATH=$CLASSPATH :/usr/java/j2sdk1.4.2_03/lib/tools.jar
export CLASSPATH=$CLASSPATH :/usr/java/j2sdk1.4.2_03/jre/lib/rt.jar :./
export JAVA_HOME=/usr/java/j2sdk1.4.2_03
export JDK_HOME=/usr/java/j2sdk1.4.2_03
```

Ensuite, il faut le rendre exécutable puis le copier dans /etc/profile.d :

```
# chmod a+x script.sh
# cp script.sh /etc/profile.d
```

Il suffit ensuite de se délogger puis de se logger à nouveau pour pouvoir utiliser Java.

0.1.5 5. Désinstaller J2SE SDK :

Dans le cas du fichier binaire, on peut l'effectuer de deux façons :

- Il suffit de supprimer le répertoire j2sdk1.4.2_03 et tous ce qu'il contient.

```
# rm -Rf /usr/java/j2sdk1.4.2_03
```

- Dans le cas d'un fichier rpm, il suffit de taper la commande :

```
# rpm -e j2sdk
```

Il ne reste plus qu'à supprimer le script permettant d'ajouter Java au PATH.

```
# rm -f /etc/profile.d/script.sh
```