***JavaCard TP1***

Programmation des cartes à puce :

Environnement de développement JavaCard



***Plan :***

1. ***Introduction aux cartes à puce et le langage JavaCard***

***1.1 Une carte à puce c’est quoi ?***

***1.2*** ***C’est quoi le langage JavaCard ?***

1. ***Outlies et technologies utilisées***

***2.1 Outils logiciels nécessaires***

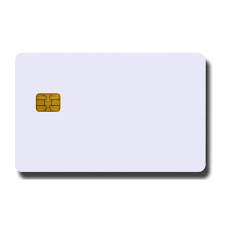
***2.2*** ***Instructions d'installation***

***But de ce TP :***

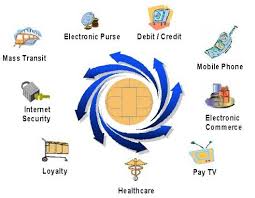
**Ce TP a pour but d’installer l’environnement de développement nécessaire pour programmer des applications JavaCard. Nous allons installer Eclipse l'IDE (Environnement de développement intégré), la plate-forme JavaCard 2.2.2 (kit de développement) ainsi que le plug-in Eclipse-JCDE (interface entre la plate-forme JavaCard et Eclipse).**

1. ***Introduction aux cartes à puce et le langage JavaCard :***

1.1 Une carte à puce c’est quoi ?

Une carte à puce est un rectangle en plastique , avec une épaisseur de un millimétré , qui porte un circuit intégré capable de mémorise de façon sécurisée une série d’information .

Ce circuit intégré s’appelle une puce



Le carte à puce est aujourd’hui un support

très répandu pour stocker des informations.

* Les cartes à puces souffre de ressources

très limitées pour assurer l’exécution

des application.

* C’est le moyen le plus sur pour assurer un niveau

de sécurité satisfaisant reste la carte à puce.

* 1. c’est quoi le langage JavaCard ?

Le **langage JavaCard** a été proposé par **Sun Microsystems Inc,** en 1996,

comme version épurée du langage Java s’adaptant aux environnements réduits

des cartes à puces intelligente.

* L’environnement minimum nécessaire au fonctionnement des API JavaCard :
* Un processeur 300kl,
* 12Ko ROM,
* 4Ko EEPROM ,
* 512 Octets de RAM.
* Un certain nombre de composants , par exemple les tableaux multidimensionnels,

Les Threads et le garbage collector … , **ont été enlevés** dans le première version du langage JavaCard.

* Les avantages du langage JavaCard

1. Langage de haut niveau orienté objets
2. Write Once , Run Anywhere
3. Plate-forme multi applicative
4. Partage de données entre applications
5. Sécurité des données
6. Souplesse
7. ***Outlies et technologies utilisées***

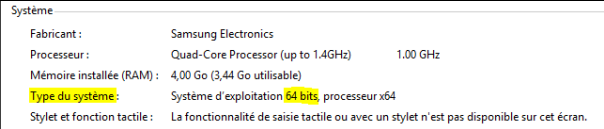
***2.1 Outils logiciels nécessaires*** :

Avant de commencer à télécharger les différentes ressources nécessaires, il est impératif de

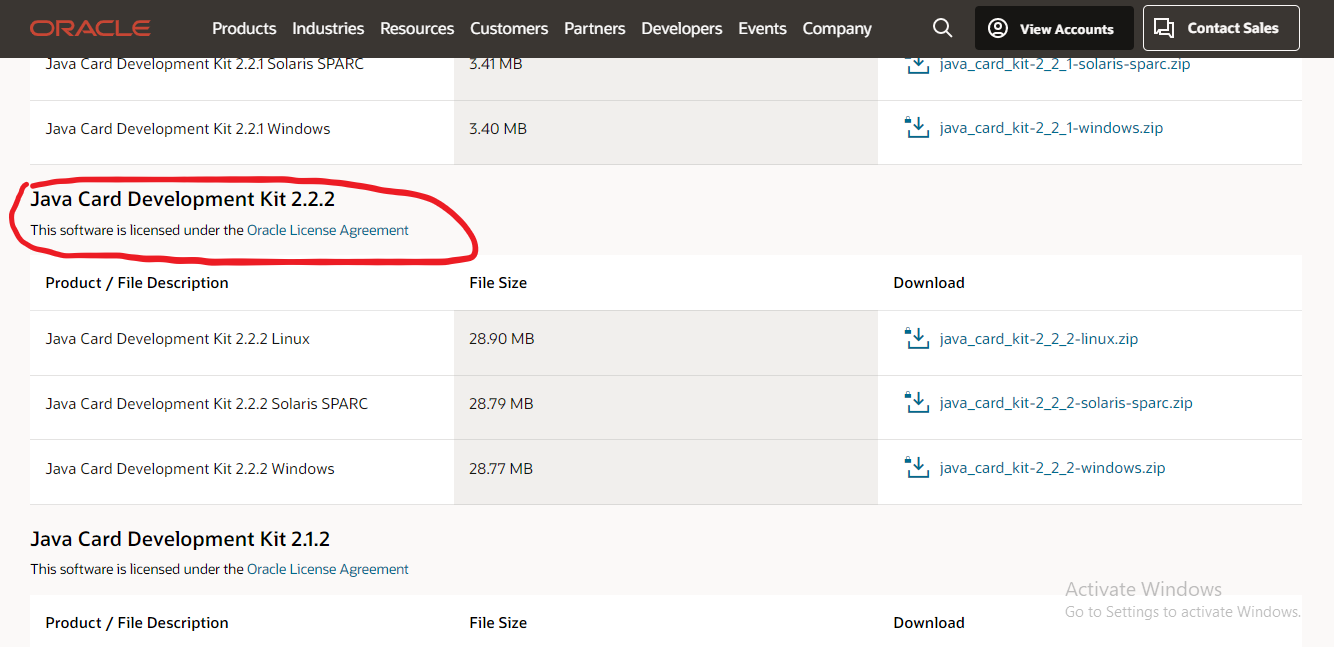
vérifier le type de votre système d’exploitation : click droit sur le poste de travail puis

sélectionner ‘Propriétés’.

**Q1.** *Identifiez votre* **système d’exploitation** *ainsi que son type***:**

****

**Q2***. Le Java Card Development Kit 2.2.2 :* **archive java\_card\_kit-2\_2\_2-windows.zip** *à partir du site d’Oracle* **:**

****[**http://www.oracle.com/technetwork/java/javasebusiness/downloads/java-archive-downloads-javame-419430.html**](http://www.oracle.com/technetwork/java/javasebusiness/downloads/java-archive-downloads-javame-419430.html)

**Q3.** *Télécharger un environnement standard édition de Java (Java SE) qui contient non seulement un Java Runtime Environnement (JRE) pour pouvoir tourner des applications java mais également un Java Developpement Kit pour pouvoir compiler vos codes sources. Nous proposons de télécharger Java SE 7u.* **link :** [**http://www.oracle.com/**](http://www.oracle.com/)

**Q4***. Suivant le type de votre système, télécharger un environnement de développement java pour pouvoir éditer votre code source. Nous proposons de télécharger* **ECLIPSE IDE***. link* **:**

[**http://www.eclipse.org/**](http://www.eclipse.org/)

**Q5.** *Le plugin d'intégration* **Eclipse-JCDE version 0.2** *:* **archive eclipse-jcde-0.2.zip** *à partir du site de téléchargement sourceforge* **:** [**http://sourceforge.net/projects/eclipse-jcde/**](http://sourceforge.net/projects/eclipse-jcde/)

****

Pour procéder à l'installation de l'environnement de développement, les outils logiciels suivant sont nécessaires :

* 1. ***Instructions d'installation :***
     1. ***Installation d’Eclipse sous windows :***

Pour centraliser nos fichiers, nous allons créer un seul et unique répertoire de travail directement sous la racine qu’on nommera C:\Eclipse.

1. **Installation du JDK**

Eclipse ne contient pas par défaut un compilateur Java. Ainsi, pour développer des programmes en Java, il faut installer au préalable un kit de développement Java sur votre machine (**on utilisera JDK 1.7).**

Pour cela télécharger et installer l’exécutable **jdk-7u40-windows-i586.exe** à partir du lien

suivant : <http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html>

Remarque :

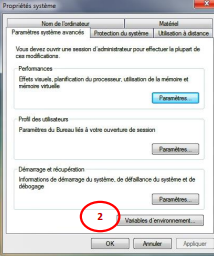
Il ne faut pas confondre le **JDK** (Java Development Kit) avec le **JRE** (Java Runtime Environment) :

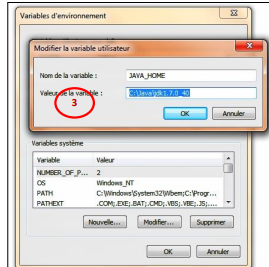
**✓** la **JRE** ne contient que les outils nécessaires pour exécuter des applications Java .

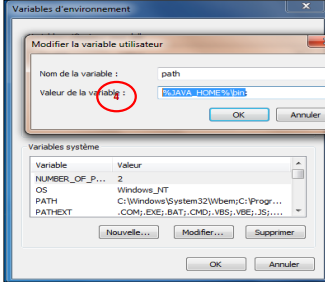
**✓** le **JDK** permet également d’exécuter des applications Java mais il permet en plus de **compiler** du code source java pour en faire des applications compréhensible par la **JRE**, le **JRE** est inclut dans le **JDK**

Une fois l'installation terminée, il faut ajouter la variable d’environnement **JAVA\_HOME**, contenant le chemin d'accès au **JDK**, dans la définition de la variable **Path**. Pour cela :

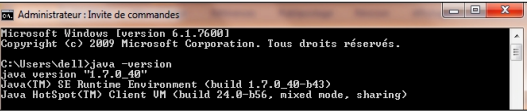
1. Créer un nouveau répertoire **C:\Java**, puis recopier tout le contenu du répertoire **C:\Program Files \Java\** sous ce répertoire **C:\Java**.
2. Cliquer avec le bouton droit sur l'icône du **Poste de travail**, puis **Propriétés > Paramètres système Avancés > Variables d'environnement** ;



1. Créer une nouvelle variable d’environnement (coté utilisateur) nommée **JAVA\_HOME** contenant le chemin **C:\ Java\jdk1.7.0\_40**
2. Modifier la variable Path (coté système) en ajoutant vers sa fin : **%JAVA\_HOME%\bin ;**

****

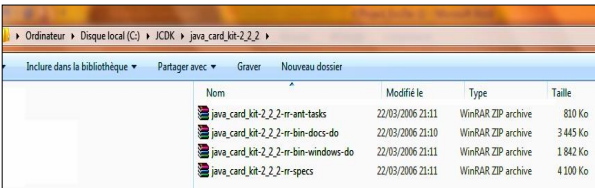
1. Pour vérifier le bon fonctionnement de votre JDK, il suffit de taper la commande « **java – version** » sous la console DOS :

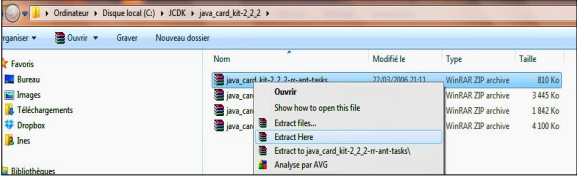
****

* + 1. ***Installation du Java Card Development Kit 2 .2.2 :***

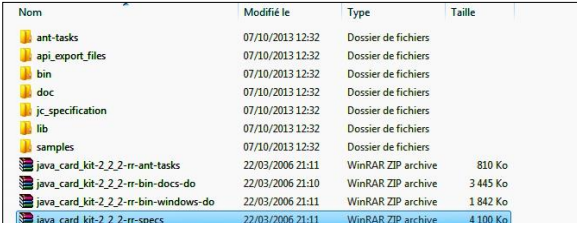
Tout d'abord, Il faut créer un nouveau répertoire **C:\JCDK** :

▪ Décompresser l’archive **java\_card\_kit-2\_2\_2-windows.zip** après l’avoir télécharger dans le répertoire **C:\JCDK**.

* + - Aller dans le répertoire **C:\JCDK \java\_card\_kit-2\_2\_2**, de cette manière vous devriez voir apparaître les fichiers suivants
    - Décompresser toutes les fichiers du répertoire courant **C:\JCDK \java\_card\_kit-2\_2\_2** :



* De cette manière vous devriez voir apparaître les fichiers suivants :

▪

Remarque :

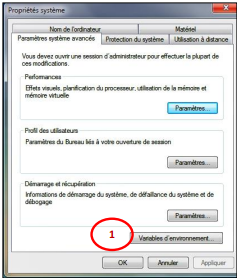
- **C:\JCDK\java\_card\_kit-2\_2\_2\bin** : contient les outils du JCDK (simulateur, ...)

- **C:\JCDK\java\_card\_kit-2\_2\_2\lib** : contient les librairies du JCDK

- **C:\JCDK\java\_card\_kit-2\_2\_2\jc\_specifications\specs\api\html** : contient l'API du JCDK (très utile)

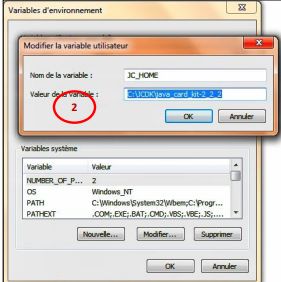
Une fois l'installation terminée, il faut ajouter la variable d’environnement **JC\_HOME** contenant le chemin d'accès au JCDK dans la définition de la variable **Path**. Pour cela :

1. Cliquer avec le bouton droit sur l'icône du Poste de travail, **puis Propriétés > Paramètres système Avancés > Variables d'environnement** .

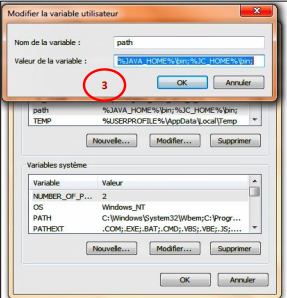


1. Créer une nouvelle variable d’environnement **JC\_HOME** contenant

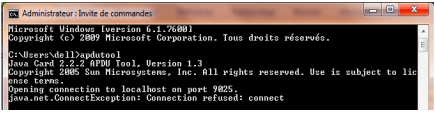
le chemin **C:\JCDK\java\_card\_kit-2\_2\_2**

****

1. Sélectionner la ligne Path puis faire Modifier la variable Path : **%JC\_HOME%\bin**;

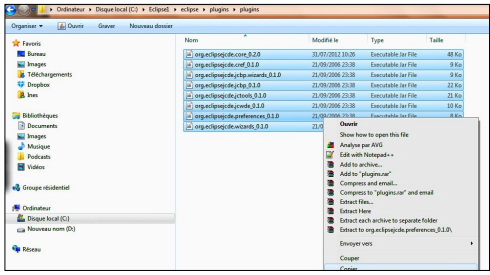


1. Vérifiez la réussite de configuration en tapant « **apdutool** » dans une console de commandes :

****

* + 1. ***Mise à jour des plugins Eclipse-JCDE :***

L'installation consiste à décompresser l’archive **eclipse-jcde-0.2.zip** après l’avoir téléchargée dans le répertoire **C:\Eclipse\eclipse\plugins**. Un sous-dossier "**plugins** " sera créer, il faut simplement recopier tout son contenu directement sous **C:\Eclipse\eclipse\plugins**.



**Conclusion**

A ce stade, nous disposons d'un environnement de développement complet permettant de créer des Applet Javacard, de les simuler et de créer des applications clientes.