

كلية العلوم
ⵜⴰⵎⴷⴰⵏⵜ ⴰⵎⴰⵔⴰⵏⵜ
FACULTÉ DES SCIENCES



UNIVERSITE IBN ZOHR FACULTE DES SCIENCES

Atelier JavaCard : Calcul les Heures Travaillées d'un Employé

Java Card

Par : EN-NAHEL Aissam

Systèmes Informatiques Distribués & Big Data

Année universitaire 2022-2023

Table des matières

Table des matières	1
Chapitre 1 : Installation de l'environnement de travail	2
Chapitre 2 : Création du Project Eclipse	12
1. Le Code :	12
Employer_javacard.java	12
2. Explication des instructions :	14
CALCUL_HEURES	14
AFFICHE_HEURES	14
A_ZERO_NBR_H	14
START_TIME_H	14
START_TIME_MIN	14
STOP_TIME_H	14
STOP_TIME_MIN	14
A_ZERO	14
Chapitre 3 : Démo	15
Set Applet AID	15
Set Package AID	15
Generate Script	16
create-Employer_javacard.script	17
select-Employer_javacard.script	17
Enregistrement des variables en utilisation EEPROM	18
Déployer l'Applet	18
Utilisation de la carte	19

Chapitre 1 : Installation de l'environnement de travail

Pour configurer l'environnement de travail de mon projet JavaCard, je vais suivre ces étapes :

- 1- Tout d'abord, je vais télécharger le kit Java Card 2.2.2 pour Windows :
<https://www.oracle.com/java/technologies/java-archive-downloads-javame-downloads.html>

Java Card Development Kit 2.2.2

This software is licensed under the [Oracle License Agreement](#)

Product / File Description	File Size	Download
Java Card Development Kit 2.2.2 Linux	28.90 MB	java_card_kit-2_2_2-linux.zip
Java Card Development Kit 2.2.2 Solaris SPARC	28.79 MB	java_card_kit-2_2_2-solaris-sparc.zip
Java Card Development Kit 2.2.2 Windows	28.77 MB	java_card_kit-2_2_2-windows.zip

- 2- Ensuite, je vais télécharger Eclipse JCDE (Java Card Development Environment) :
<http://sourceforge.net/projects/eclipse-jcde/>

[Home](#) / [Browse Open Source](#) / [Desktop Environment](#) / EclipseJCDE



EclipseJCDE

Status: **Beta** Brought to you by: [amgad_s_youssef](#)



★★★★★ 1 Review

Downloads: 5 This Week

Last Update: 2013-03-22


[Download](#)  [Get Updates](#) [Share This](#)

[Windows](#) | [Mac](#) | [Linux](#) | [BSD](#) | [ChromeOS](#)

- 3- Je vais utiliser Eclipse JEE version 2022-09-R pour Windows que j'ai déjà téléchargée à partir du lien suivant :: <https://www.eclipse.org/downloads/packages/release/2022-09/r>


Eclipse IDE for Enterprise Java and Web Developers

502 MB 559,743 DOWNLOADS



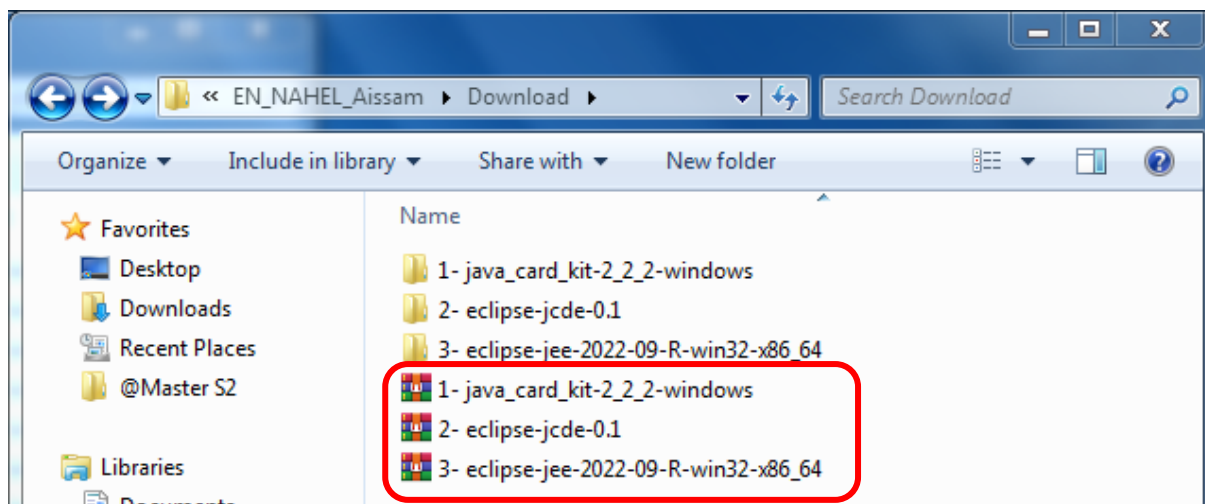
Tools for developers working with Java and Web applications, including a Java IDE, tools for JavaScript, TypeScript, JavaServer Pages and Faces, Yaml, Markdown, Web Services, JPA and Data Tools, Maven and Gradle, Git, and more.

Click [here](#) to open a bug report with the Eclipse Web Tools Platform.
Click [here](#) to raise an issue with the Eclipse Platform.
Click [here](#) to raise an issue with Maven integration for web projects.
Click [here](#) to raise an issue with Eclipse Wild Web Developer (incubating).

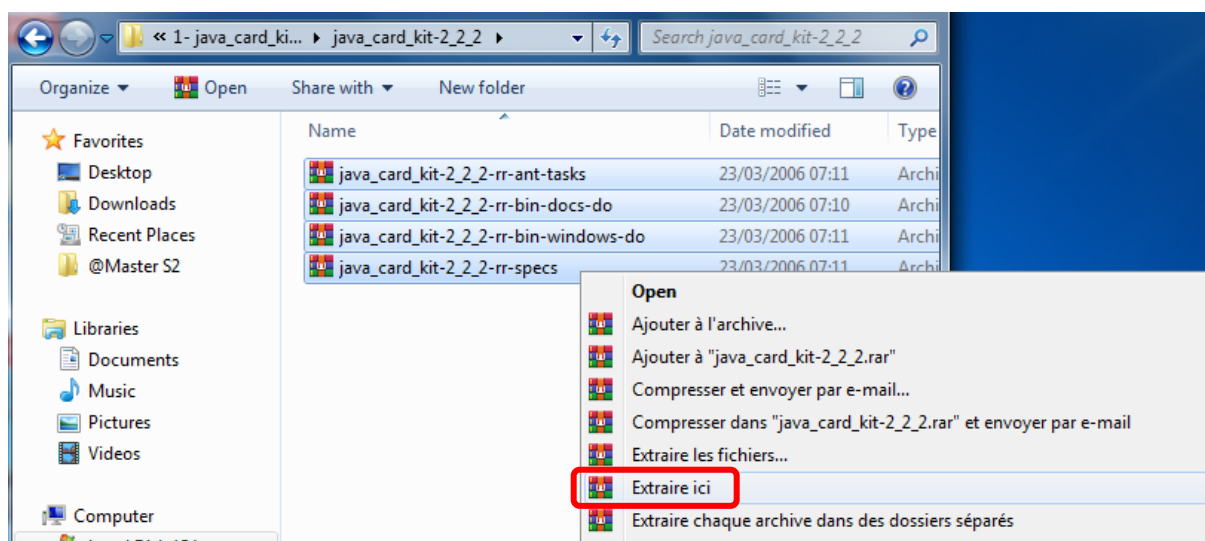


[Windows x86_64](#)
[macOS x86_64](#) | [AArch64](#)
[Linux x86_64](#) | [AArch64](#)

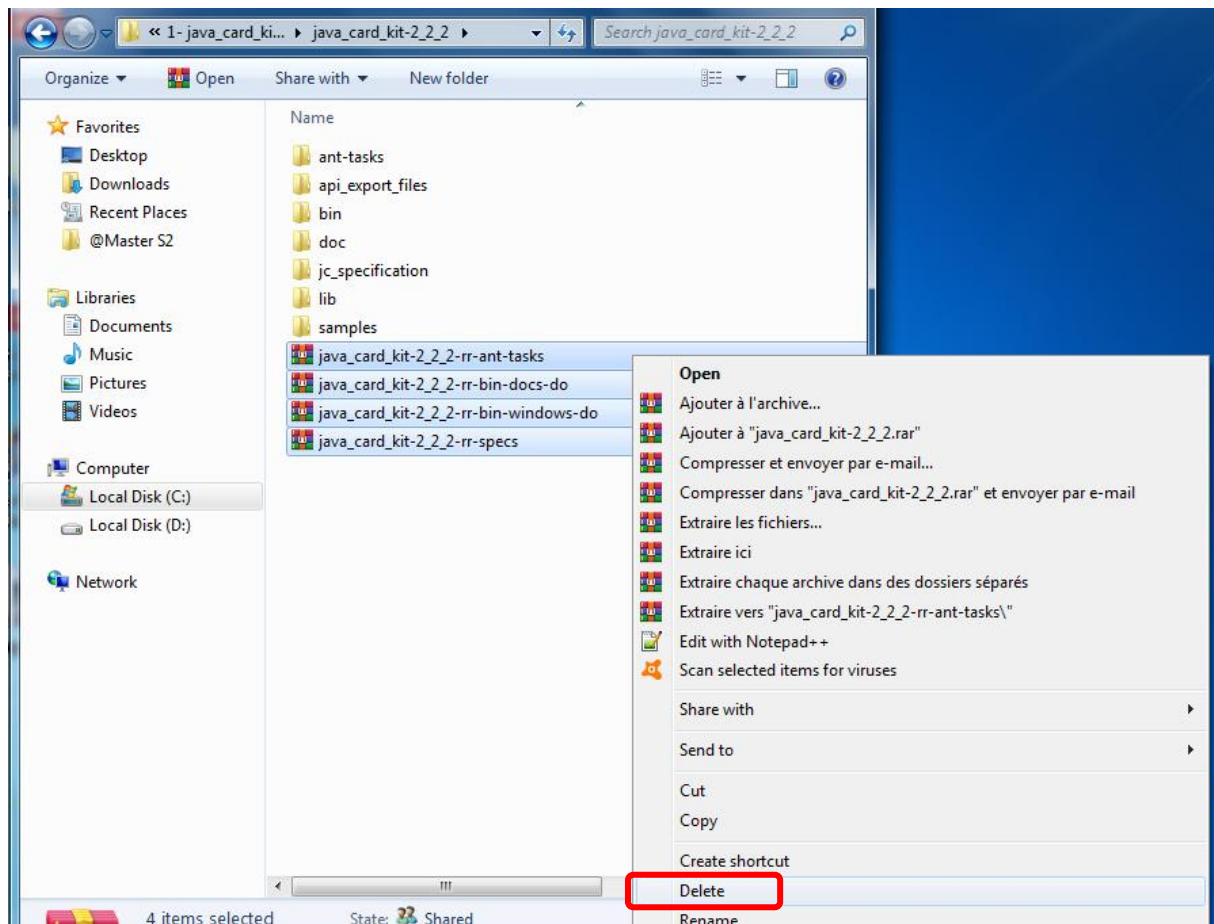
Voici tous les fichiers nécessaires :



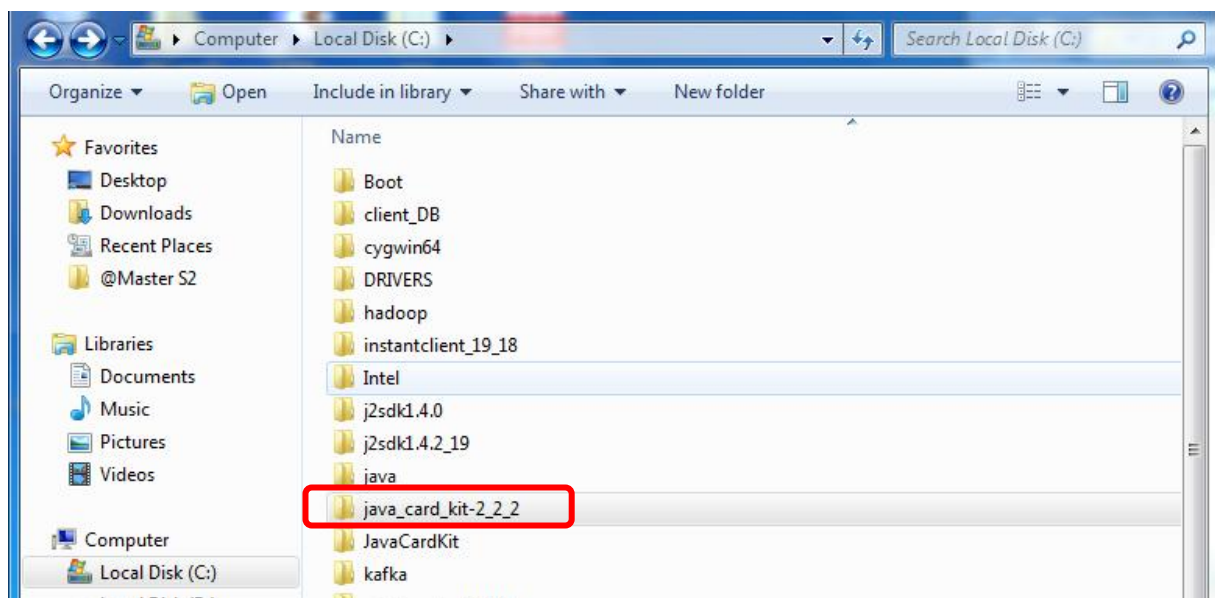
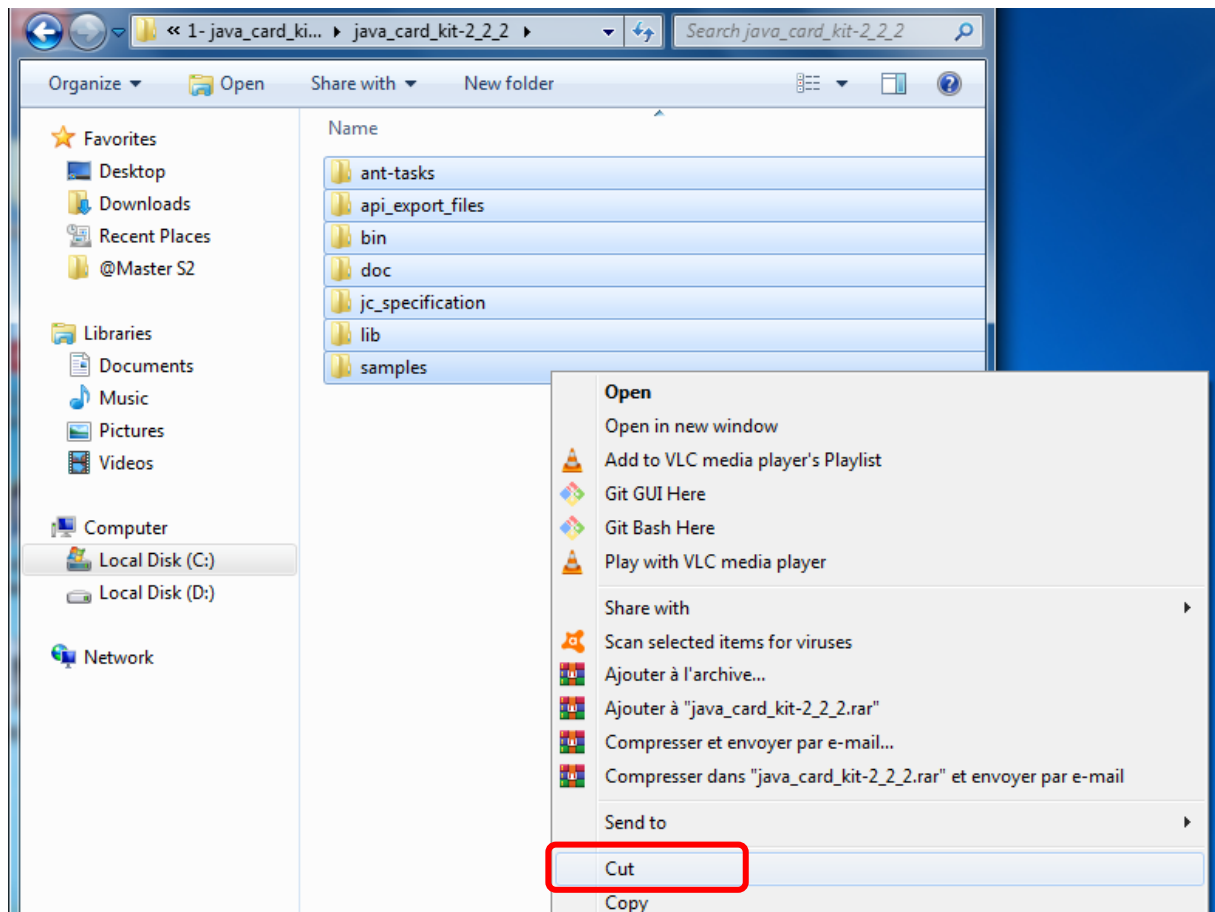
Une fois que j'aurai téléchargé ces fichiers, je commencerai par installer le Java Card kit 2.2.2
Je décompressai le fichier 1-java_card_kit-2_2_2-windows.zip
Ensuite, je décompressai les fichiers qui se trouvaient à l'intérieur du dossier 1-
java_card_kit-2_2_2-windows\java_card_kit-2_2_2\ :



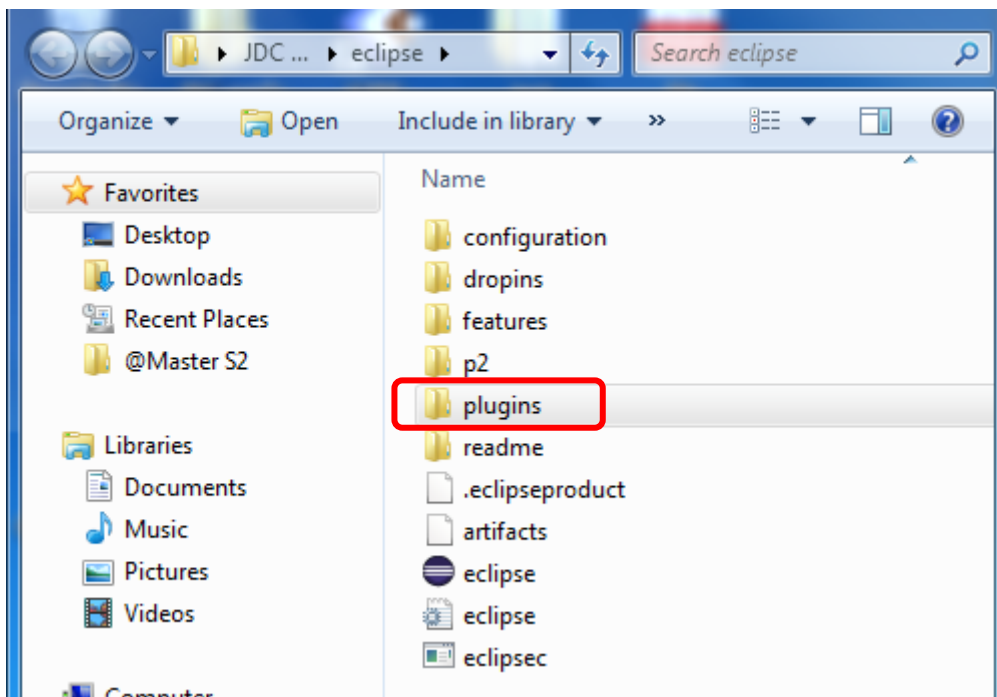
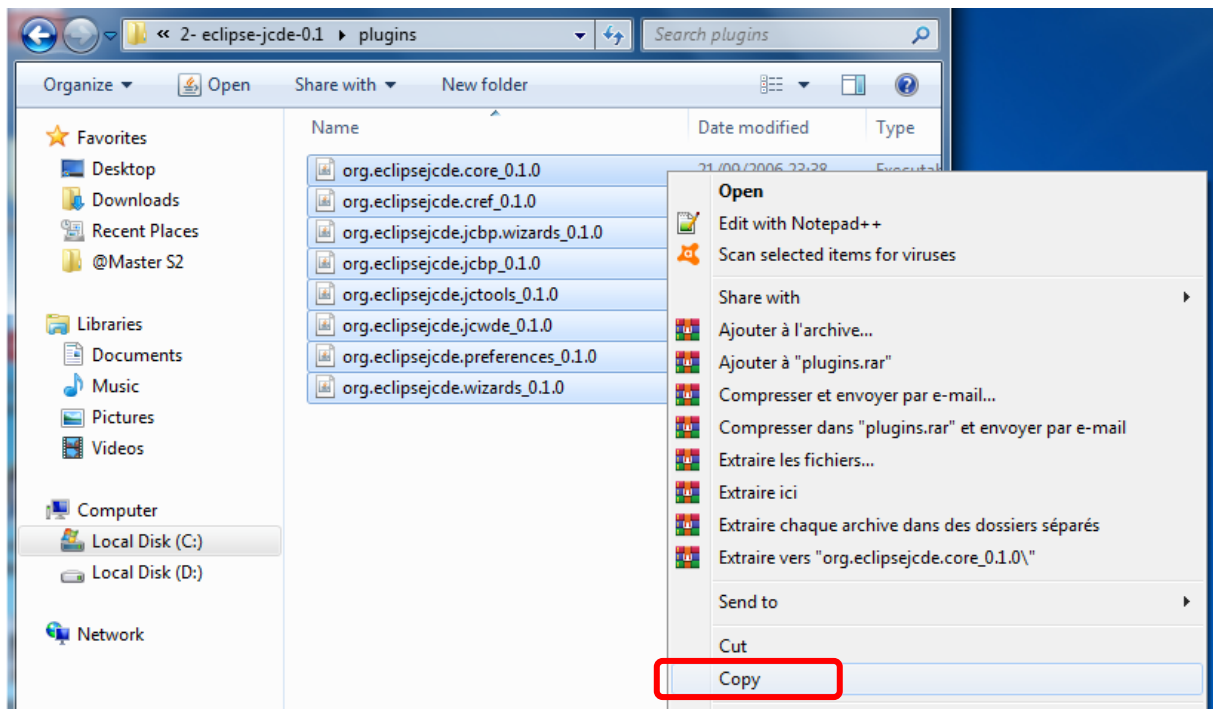
Maintenant, ces fichiers n'ont pas de rôles, il faut les supprimer :



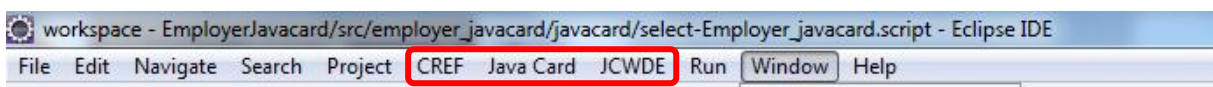
Je déplaçai les dossiers extraits vers le répertoire `C:\java_card_kit-2_2_2-windows\`



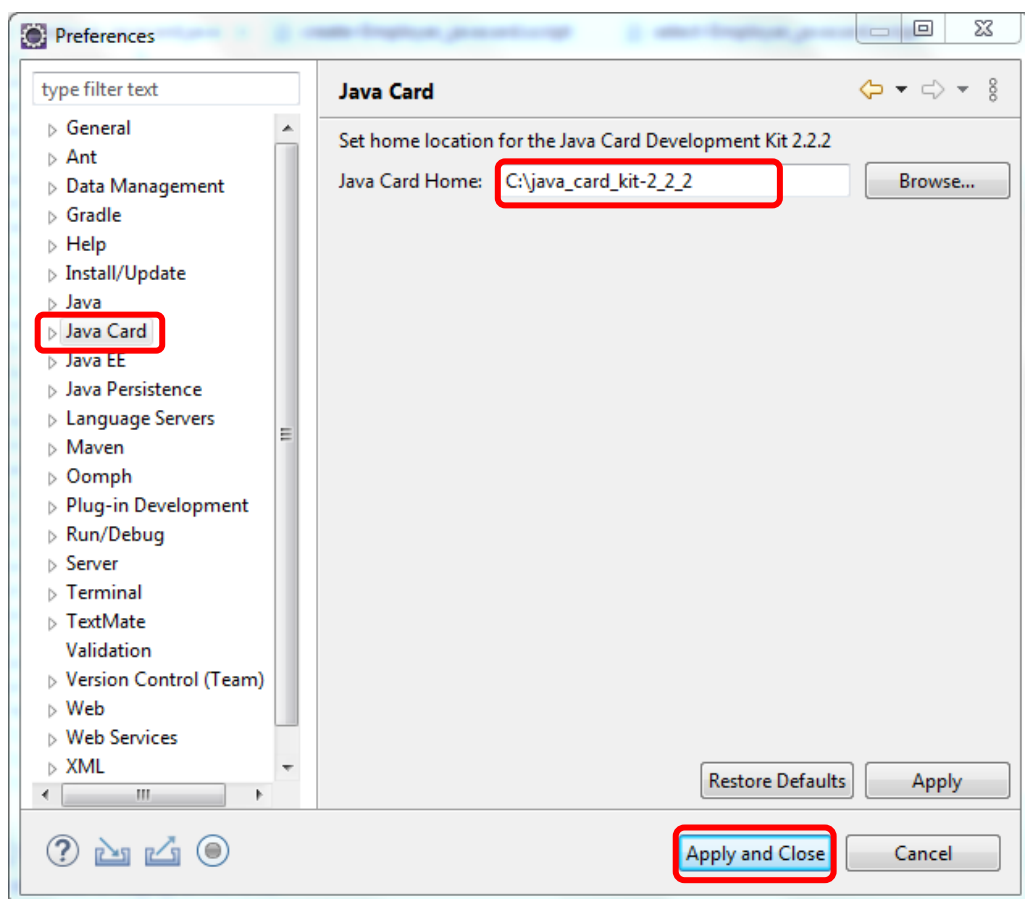
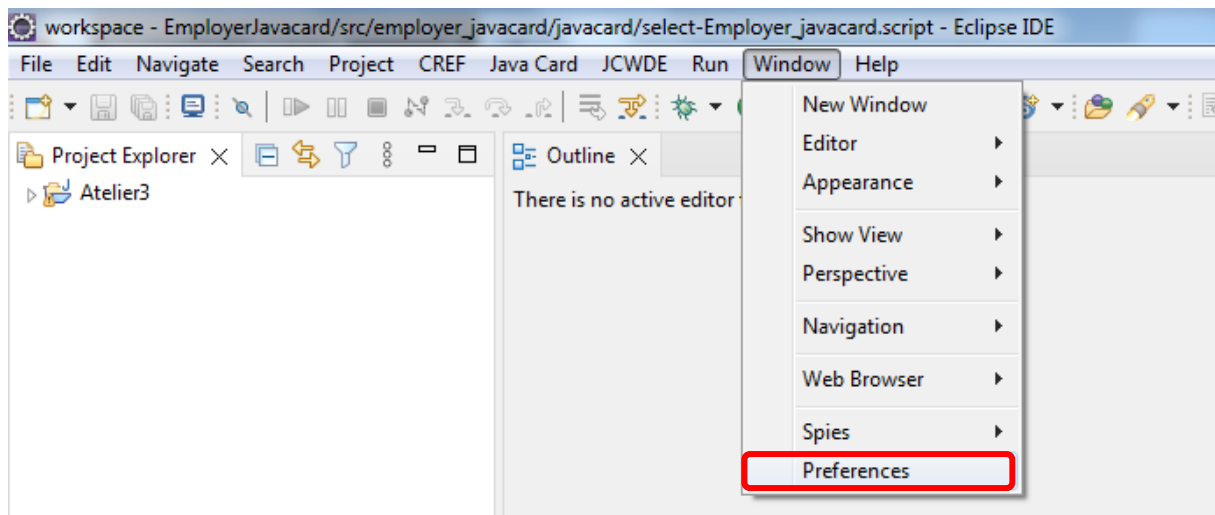
Puis j'ai configuré Eclipse : Après je décompressai le fichier 2-eclipse-jcde-0.1.zip et le fichier 3-eclipse-jee-2022-09-R-win32-x86_64.zip, j'ai copié les fichiers jars qui se trouvent dans 2-eclipse-jcde-0.1\plugins vers 3-eclipse-jee-2022-09-R-win32-x86_64\eclipse\plugins :



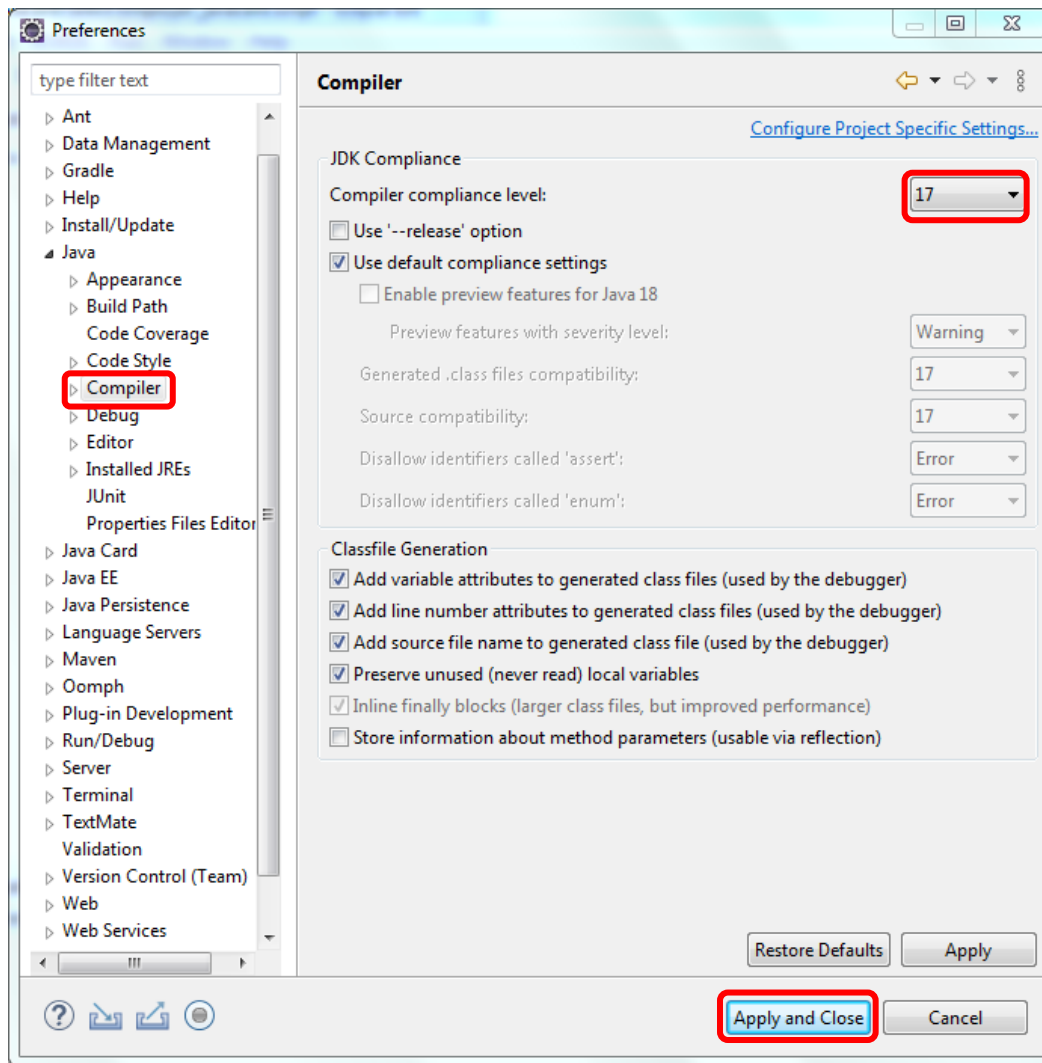
Maintenant, les options liées à Java Card sont activées dans Eclipse :



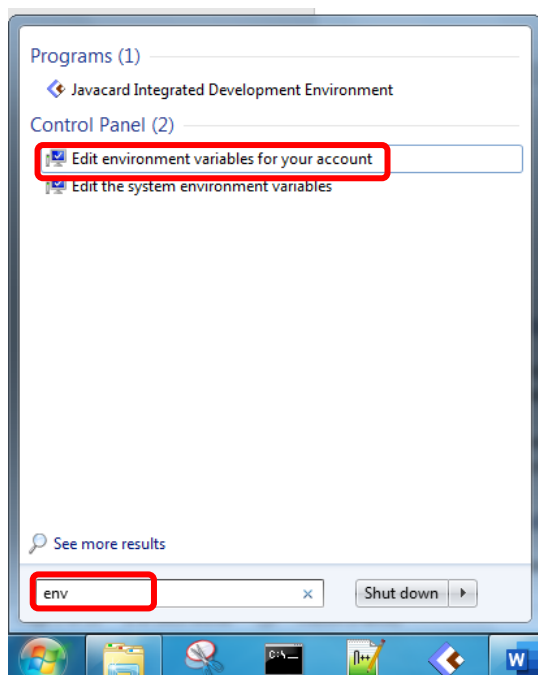
Il faut déterminer le chemin du Java Card kit 2.2.2 dans Eclipse :
Window → Preferences → Java Card



Et pour le Compiler JDK, J'utilisai 17 :

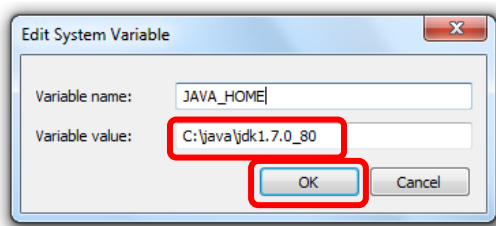
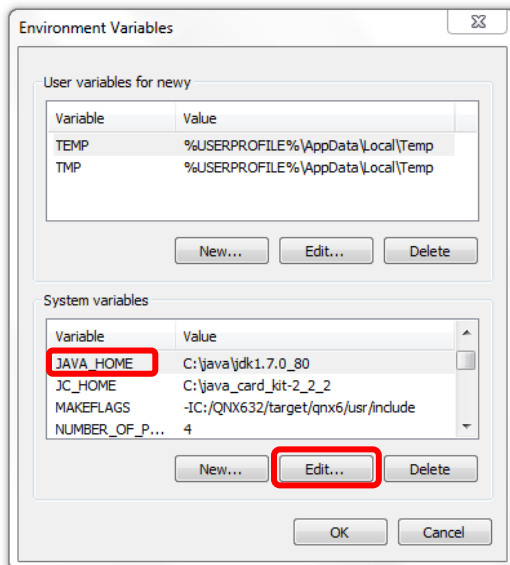


La dernière étape consiste à configurer les variables d'environnement :



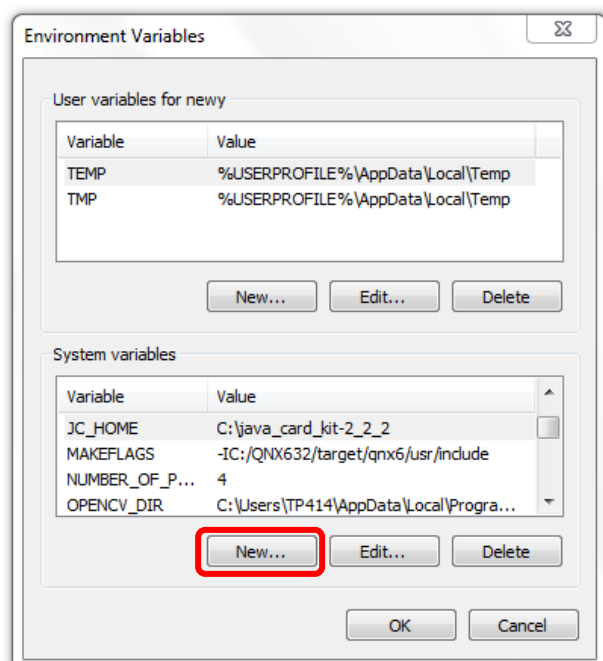
J'utilisai le JDK 1.7 :

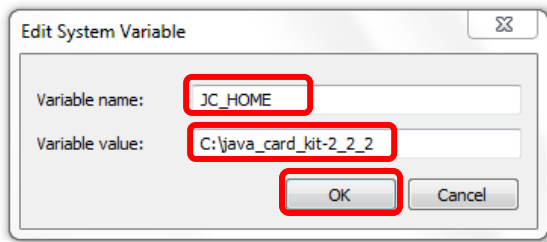
JAVA_HOME : C:\java\jdk1.7.0_80



Il faut ajouter nouvelle variable :

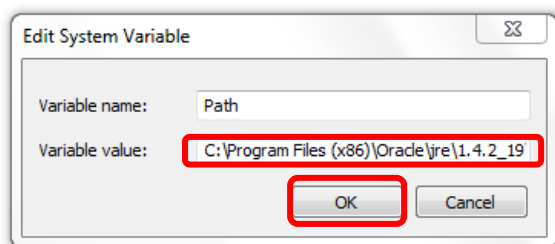
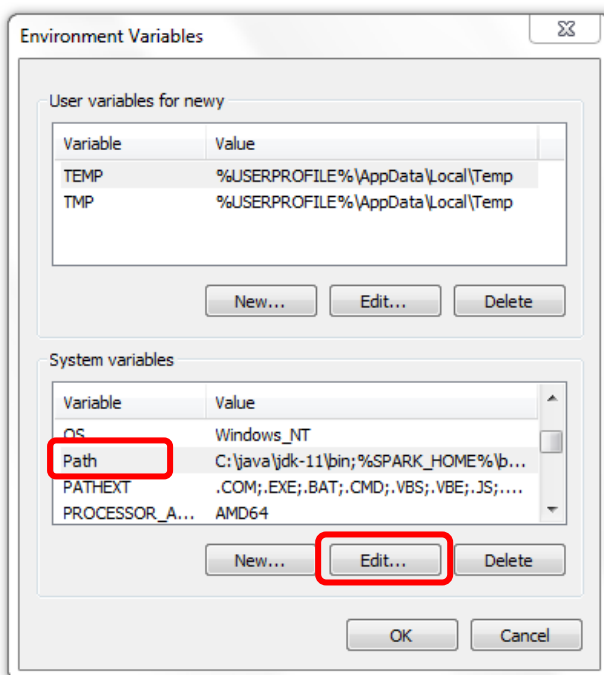
JC_HOME : C:\java_card_kit-2_2_2

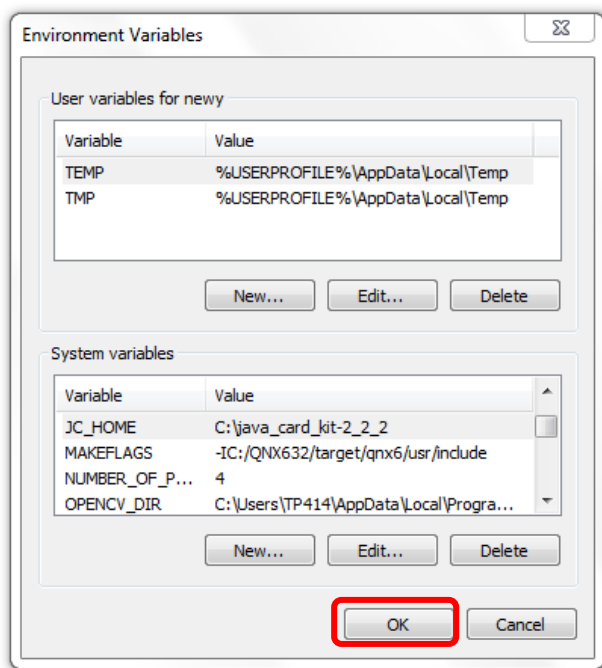




Et enfin, il faut ajouter le chemin de Java compatible avec Java Card à la variable d'environnement "**Path**". J'ai utilisé Java 1.4 :

Path : C:\Program Files (x86)\Oracle\jre\1.4.2_19\bin;





Pour vérifier la version de Java:

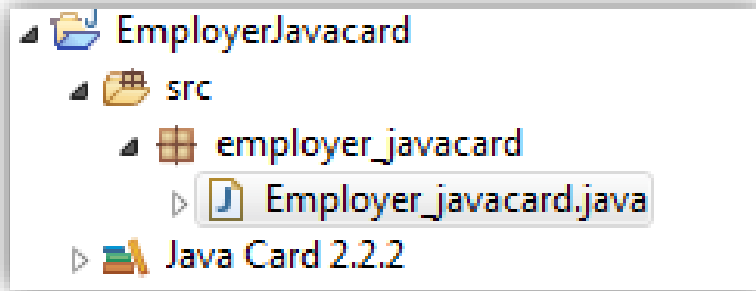
```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\EN_NAHEL_Aissam>java -version
java version "1.4.2"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.4.2-b28)
Java HotSpot(TM) Client VM (build 1.4.2-b28, mixed mode)

C:\Users\EN_NAHEL_Aissam>
```

Chapitre 2 : Création du Project Eclipse

Le projet Java Card "**EmployerJavacard**" est composé d'un seul fichier Java Card, à savoir "**Employer_javacard.java**" qui contient l'implémentation de l'applet Java Card qui gère le calcul des heures travaillées d'un employé.



1. Le Code :

Employer_javacard.java

```
package employer_javacard;

import javacard.framework.APDU;
import javacard.framework.Applet;
import javacard.framework.ISO7816;
import javacard.framework.ISOException;

public class Employer_javacard extends Applet {

    private byte nbr_hours = 0;
    private byte start_hour = 0;
    private byte start_min = 0;
    private byte stop_hour = 0;
    private byte stop_min = 0;

    private static final byte CLA = (byte) 0xB0;

    private static final byte CALCUL_HEURES = (byte) 0x01;
    private static final byte AFFICHE_HEURES = (byte) 0x02;

    private static final byte A_ZERO_NBR_H = (byte) 0x03;
    private static final byte A_ZERO = (byte) 0x04;

    private static final byte START_TIME_H = (byte) 0x05;
    private static final byte START_TIME_MIN = (byte) 0x06;
    private static final byte STOP_TIME_H = (byte) 0x07;
    private static final byte STOP_TIME_MIN = (byte) 0x08;

    private Employer_javacard() { }

    public static void install(byte bArray[], short bOffset, byte bLength)
    throws ISOException {
        new Employer_javacard().register();
    }
}
```

```

    }

    public void process(APDU apdu) throws ISOException {
        if (this.selectingApplet()) { return; }

        byte[] buffer = apdu.getBuffer();

        if(buffer[ISO7816.OFFSET_CLA] != CLA) {
            ISOException.throwIt(ISO7816.SW_CLA_NOT_SUPPORTED);
        }

        switch(buffer[ISO7816.OFFSET_INS]) {
            case CALCUL_HEURES:
                short diff_h = (short)(stop_hour - start_hour);
                diff_h = (short)(diff_h*60); //h est en minutes
                short diff_min = (short)(diff_h + (stop_min - start_min));

                nbr_hours = (byte) (diff_min / 60);
                if (diff_min % 60 >= 30) {
                    nbr_hours += 1;
                }

                break;
            case AFFICHE_HEURES:
                buffer[0] = nbr_hours;
                apdu.setOutgoingAndSend((short)0, (short)1);
                break;

            case A_ZERO_NBR_H:
                nbr_hours = 0;
                break;

            case A_ZERO:
                start_hour = 0;
                start_min = 0;
                stop_hour = 0;
                stop_min = 0;
                break;

            case START_TIME_H:
                apdu.setIncomingAndReceive();
                start_hour = buffer[ISO7816.OFFSET_CDATA];
                break;

            case START_TIME_MIN:
                apdu.setIncomingAndReceive();
                start_min = buffer[ISO7816.OFFSET_CDATA];
                break;

            case STOP_TIME_H:
                apdu.setIncomingAndReceive();
                stop_hour = buffer[ISO7816.OFFSET_CDATA];
                break;

            case STOP_TIME_MIN:
                apdu.setIncomingAndReceive();
                stop_min = buffer[ISO7816.OFFSET_CDATA];
                break;
        }
    }

```

```
}  
}
```

2. Explication des instructions :

Voici une description détaillée des différentes instructions utilisées dans le code :

CALCUL_HEURES

L'instruction **CALCUL_HEURES** (0x01) est utilisée pour calculer le nombre d'heures travaillées. Elle effectue la différence entre l'heure de début du travail et l'heure de fin du travail afin de déterminer la durée totale des heures travaillées.

AFFICHE_HEURES

L'instruction **AFFICHE_HEURES** (0x02) est utilisée pour afficher le nombre d'heures travaillées.

A_ZERO_NBR_H

L'instruction **A_ZERO_NBR_H** (0x03) est utilisée pour mettre le nombre d'heures travaillées à zéro. Cette instruction est exécutée une seule fois, lors de la première utilisation de l'applet, elle permet d'initialiser le compteur d'heures travaillées à zéro avant de commencer le suivi des heures.

START_TIME_H

L'instruction **START_TIME_H** a 0x05 est pour entrer l'heur de début de travail et l'enregistrer à le variable "**start_hour**" l'heur de début de travail.

START_TIME_MIN

L'instruction **START_TIME_MIN** (0x06) est utilisée pour entrer les minutes de l'heure de début du travail et les enregistrer dans la variable "**start_min**", elle permet de spécifier les minutes de l'heure de début du travail.

STOP_TIME_H

L'instruction **STOP_TIME_H** (0x07) est utilisée pour entrer l'heure de fin du travail et l'enregistrer dans la variable "**stop_hour**", elle permet de spécifier l'heure de fin du travail.

STOP_TIME_MIN

L'instruction **STOP_TIME_MIN** (0x08) est utilisée pour entrer les minutes de l'heure de fin du travail et les enregistrer dans la variable "**stop_min**", elle permet de spécifier les minutes de l'heure de fin du travail.

A_ZERO

L'instruction **A_ZERO** (0x04) est utilisée pour mettre les variables "**start_hour**", "**start_min**", "**stop_hour**" et "**stop_min**" à zéro, elle permet de réinitialiser les valeurs des heures et des minutes, en préparation pour un nouveau calcul des heures travaillées.

Chapitre 3 : Démo

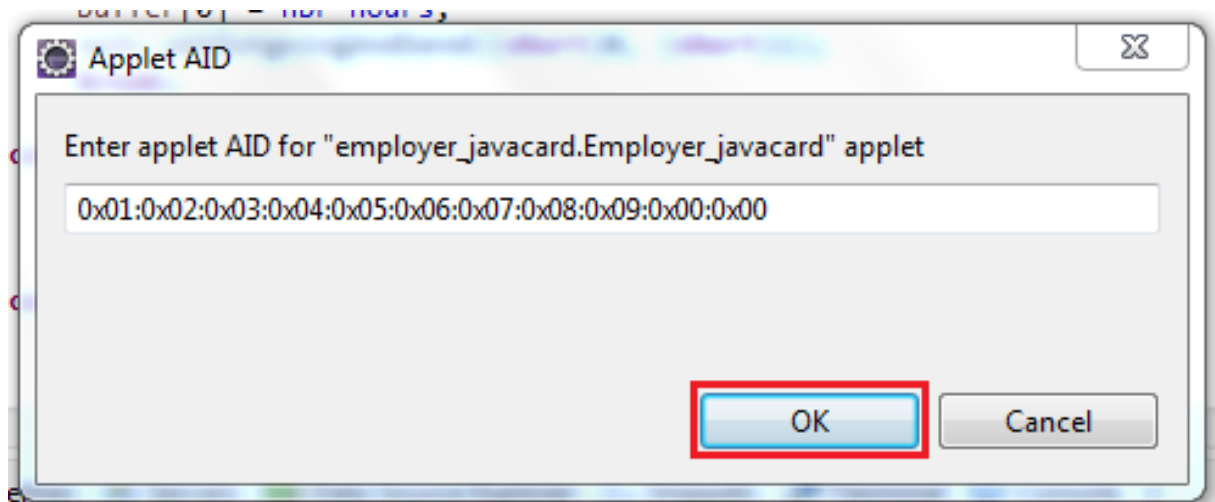
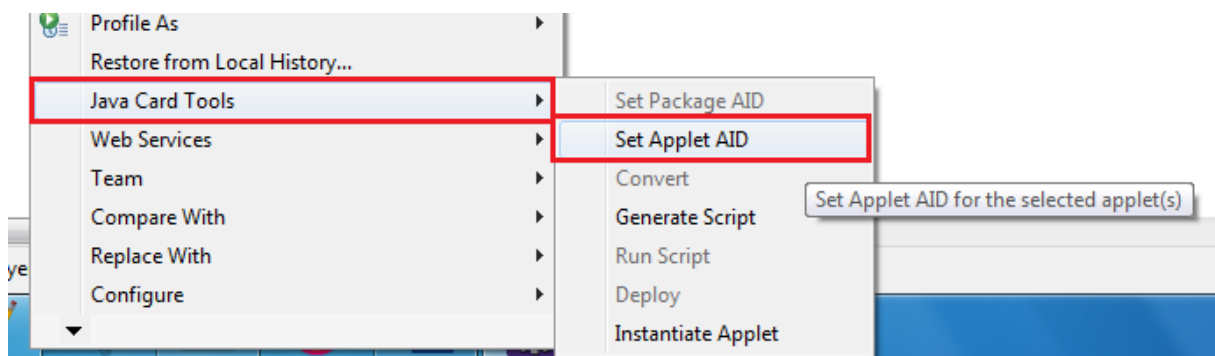
Dans ce chapitre, je vais vous présenter une démonstration de l'application de calcul des heures travaillées travers différentes étapes : la configuration de l'applet AID et du package AID, la génération du script, l'enregistrement des variables et l'utilisation de EEPROM, le déploiement de l'applet et l'utilisation l'applet.

Set Applet AID

La première étape consiste à configurer l'identifiant d'applet (AID). L'AID est une valeur unique qui identifie de manière unique notre applet.

Dans Eclipse je vais suivre :

eclipse --> clic droit sur le package --> Java Card Tool --> Set Applet AID

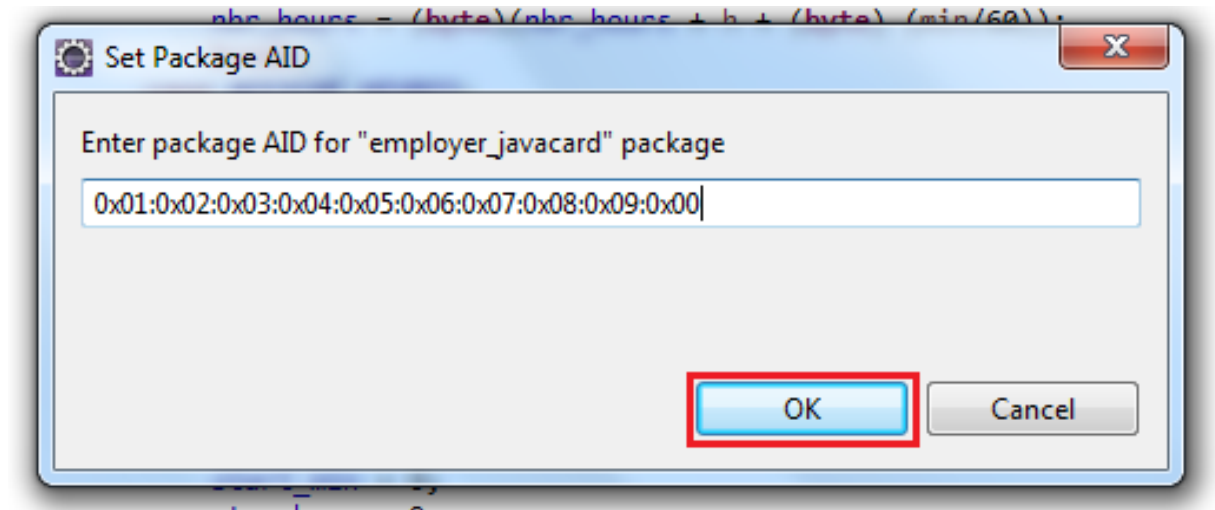
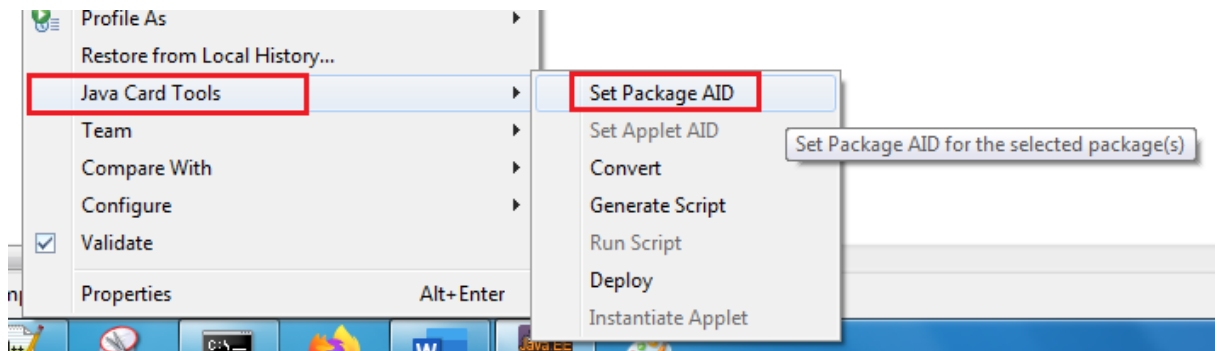


Set Package AID

En plus de l'AID, nous devons également configurer le package AID de l'applet. Le package AID est une autre valeur unique qui identifie le package auquel notre applet appartient.

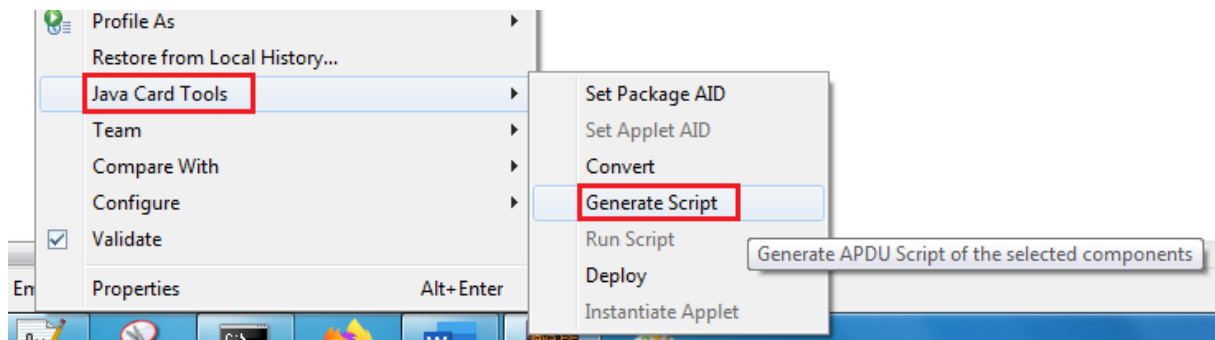
Dans Eclipse je vais suivre :

eclipse --> clic droit sur le package --> Java Card Tool --> Set Package AID

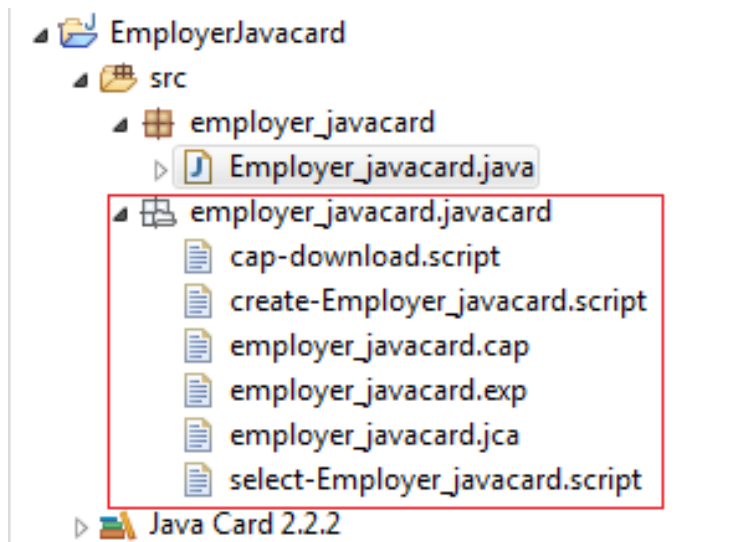


Generate Script

Je vais générer des scripts, dans Eclipse je vais suivre :
eclipse --> clic droit sur le package --> Java Card Tool --> Generate Script

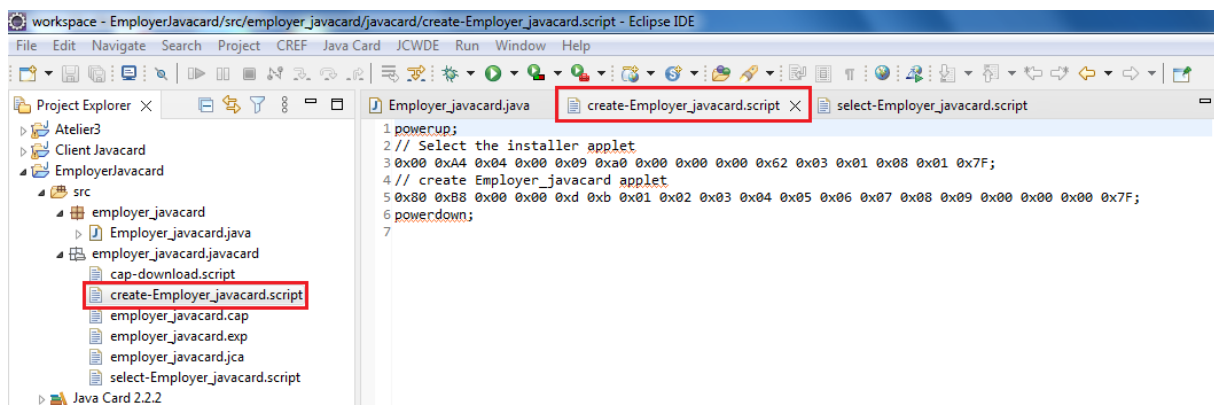


Les scripts générés sont :



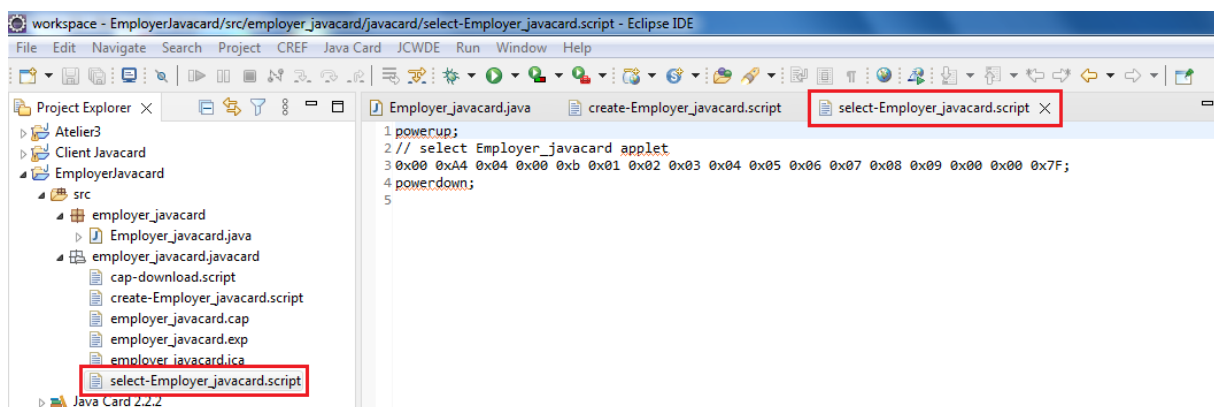
create-Employer_javacard.script

Le fichier **create-Employer_javacard.script** contient la commande de sélection de l'applet d'installation et la commande de création de l'applet Javacard Employeur.



select-Employer_javacard.script

Le fichier **select-Employer_javacard.script** contient la commande de sélection de l'applet.

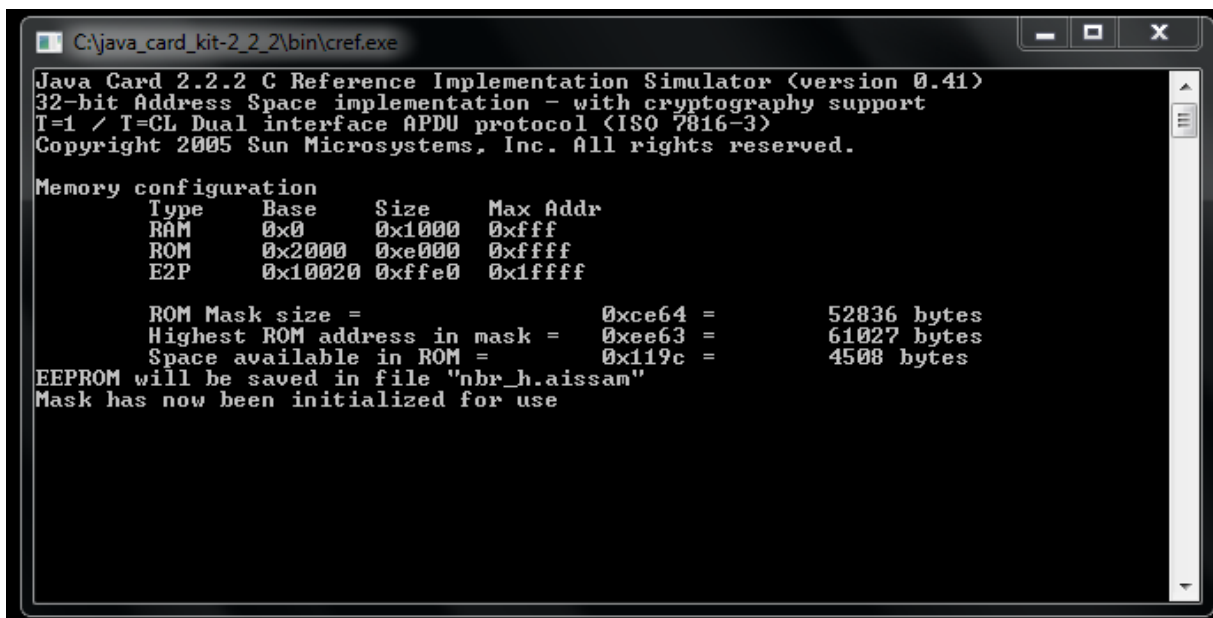


Enregistrement des variables en utilisation EEPROM

Avant de déployer l'applet, je dois enregistrer les variables nécessaires pour notre application. Cela peut inclure des variables telles que les heures de début et de fin du travail. Je vais utiliser le fichier **nbr_h.aissam** pour stocker les données.

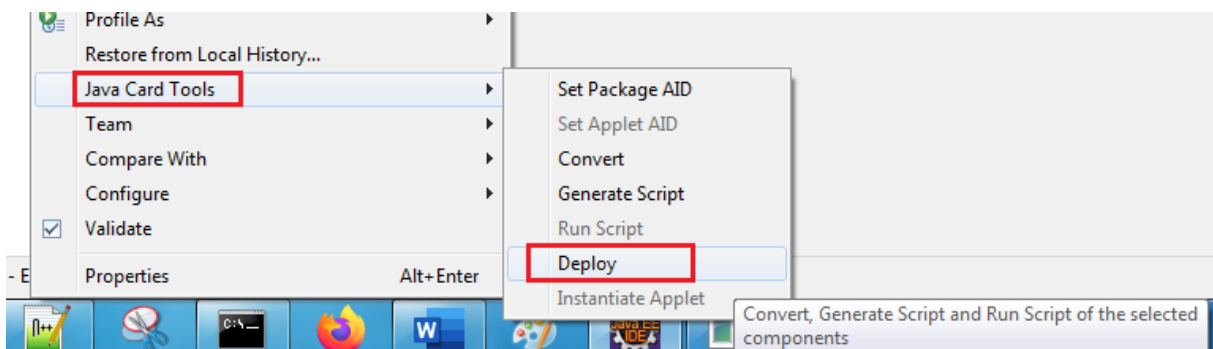
Dans cmd.exe :

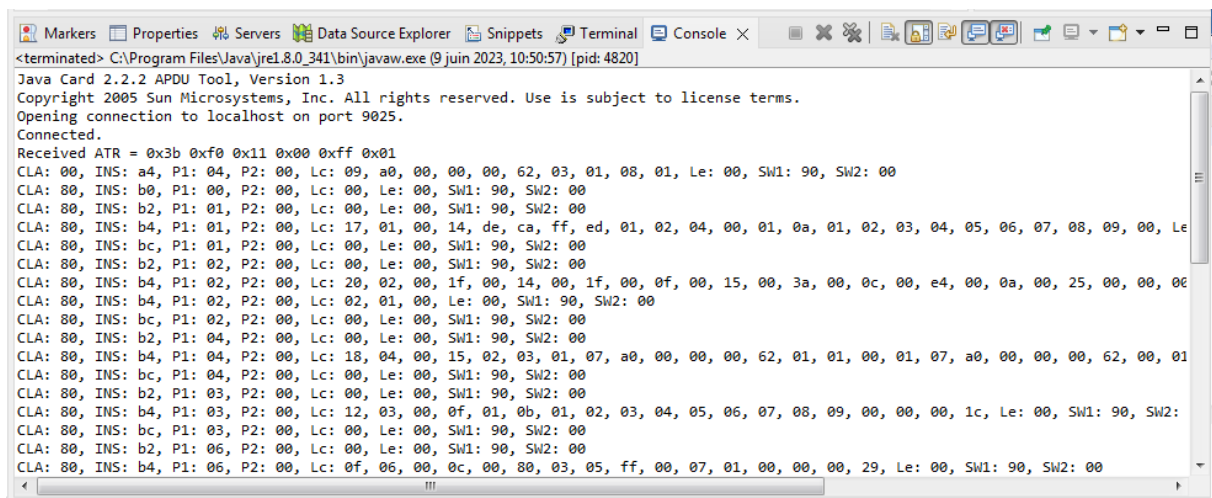
```
start cref -o nbr_h.aissam
```



Déployer l'Applet

Je suis prêt à déployer l'applet. Dans Eclipse je vais suivre :
eclipse --> clic droit sur le package --> Java Card Tool --> Deploy





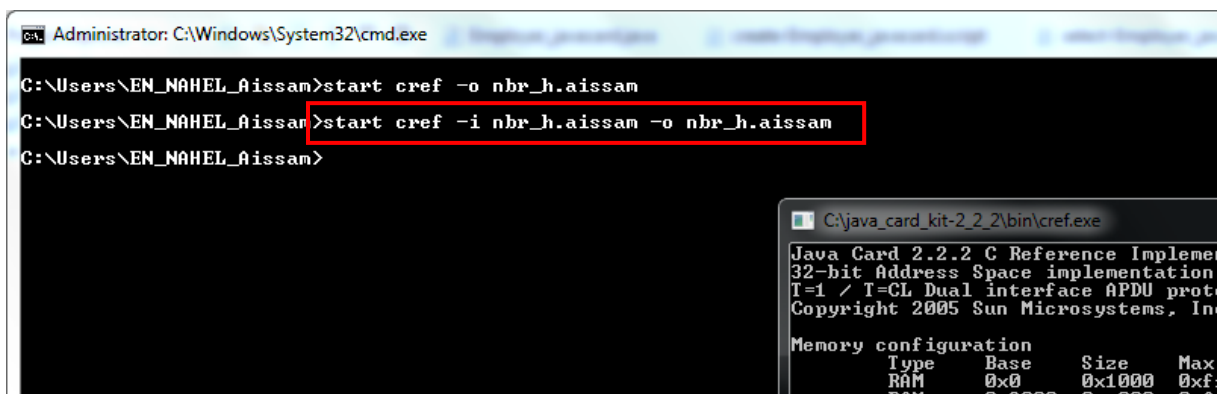
```
<terminated> C:\Program Files\Java\jre1.8.0_341\bin\javaw.exe (9 juin 2023, 10:50:57) [pid: 4820]
Java Card 2.2.2 APDU Tool, Version 1.3
Copyright 2005 Sun Microsystems, Inc. All rights reserved. Use is subject to license terms.
Opening connection to localhost on port 9025.
Connected.
Received ATR = 0x3b 0xf0 0x11 0x00 0xff 0x01
CLA: 00, INS: a4, P1: 04, P2: 00, Lc: 09, a0, 00, 00, 00, 62, 03, 01, 08, 01, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b0, P1: 00, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b2, P1: 01, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b4, P1: 01, P2: 00, Lc: 17, 01, 00, 14, de, ca, ff, ed, 01, 02, 04, 00, 01, 0a, 01, 02, 03, 04, 05, 06, 07, 08, 09, 00, Le
CLA: 80, INS: bc, P1: 01, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b2, P1: 02, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b4, P1: 02, P2: 00, Lc: 20, 02, 00, 1f, 00, 14, 00, 1f, 00, 0f, 00, 15, 00, 3a, 00, 0c, 00, e4, 00, 0a, 00, 25, 00, 00, 00
CLA: 80, INS: b4, P1: 02, P2: 00, Lc: 02, 01, 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: bc, P1: 02, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b2, P1: 04, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b4, P1: 04, P2: 00, Lc: 18, 04, 00, 15, 02, 03, 01, 07, a0, 00, 00, 00, 62, 01, 01, 00, 01, 07, a0, 00, 00, 00, 62, 00, 01
CLA: 80, INS: bc, P1: 04, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b2, P1: 03, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b4, P1: 03, P2: 00, Lc: 12, 03, 00, 0f, 01, 0b, 01, 02, 03, 04, 05, 06, 07, 08, 09, 00, 00, 00, 00, 1c, Le: 00, SW1: 90, SW2:
CLA: 80, INS: bc, P1: 03, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b2, P1: 06, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
CLA: 80, INS: b4, P1: 06, P2: 00, Lc: 0f, 06, 00, 0c, 00, 80, 03, 05, ff, 00, 07, 01, 00, 00, 00, 29, Le: 00, SW1: 90, SW2: 00
```

Utilisation de la carte

Une fois l'applet déployé, je vais l'utiliser pour simuler le comptage des heures travaillées. Je vais envoyer les commandes appropriées et observer les résultats affichés.

Tout d'abord, j'ai lancé la commande "start cref -i nbr_h.aissam -o nbr_h.aissam" dans le cmd.exe afin d'utiliser le fichier dans lequel j'ai enregistré les données des heures travaillées:

```
start cref -i nbr_h.aissam -o nbr_h.aissam
```



```
Administrator: C:\Windows\System32\cmd.exe
C:\Users\EN_NAHEL_Aissam>start cref -o nbr_h.aissam
C:\Users\EN_NAHEL_Aissam>start cref -i nbr_h.aissam -o nbr_h.aissam
C:\Users\EN_NAHEL_Aissam>

C:\java_card_kit-2_2_2\bin\cref.exe
Java Card 2.2.2 C Reference Implementation
32-bit Address Space implementation
T=1 / T=CL Dual interface APDU protocol
Copyright 2005 Sun Microsystems, Inc.

Memory configuration
Type      Base      Size      Max
RAM       0x0      0x10000   0xf
ROM       0x20000  0x30000   0xf
```

```

C:\java_card_kit-2_2_2\bin\cref.exe
Java Card 2.2.2 C Reference Implementation Simulator (version 0.41)
32-bit Address Space implementation - with cryptography support
T=1 / T=CL Dual interface APDU protocol (ISO 7816-3)
Copyright 2005 Sun Microsystems, Inc. All rights reserved.

Memory configuration
  Type      Base      Size      Max Addr
  RAM       0x0       0x1000   0xffff
  ROM       0x2000   0xe000   0xffff
  E2P       0x10020   0xffe0   0x1ffff

  ROM Mask size =          0xce64 =      52836 bytes
  Highest ROM address in mask = 0xee63 =    61027 bytes
  Space available in ROM =    0x119c =     4508 bytes
EEPROM (0xffe0 bytes) restored from file "nbr_h.aissam"
Using a pre-initialized Mask

```

Ensuite, j'ai utilisé la commande "**apdutool**" pour établir une connexion avec l'applet. J'ai sélectionné l'installation de l'applet et la création de l'applet JavaCard Employeur en utilisant les commandes du fichier "**create-Employer_javacard.script**". Ces deux dernières commandes, ne sont effectuées qu'une seule fois.

```

Employer_javacard.java  create-Employer_javacard.script  select-Employer_javacard.script
1 powerup;
2 // Select the installer applet
3 0x00 0xA4 0x04 0x00 0x09 0xA0 0x00 0x00 0x00 0x62 0x03 0x01 0x08 0x01 0x7F;
4 // create Employer_javacard applet
5 0x80 0xB8 0x00 0x00 0xD 0xB 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00 0x00 0x00 0x7F;
6 powerdown;
7

```

```

Employer_javacard.java  create-Employer_javacard.script  select-Employer_javacard.script
1 powerup;
2 // Select the installer applet
3 0x00 0xA4 0x04 0x00 0x09 0xA0 0x00 0x00 0x00 0x62 0x03 0x01 0x08 0x01 0x7F;
4 // create Employer_javacard applet
5 0x80 0xB8 0x00 0x00 0xD 0xB 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00 0x00 0x00 0x7F;
6 powerdown;
7

```

Ensuite, j'ai exécuté la commande de sélection de l'applet en utilisant la commande du fichier "select-Employer_javacard.script". Cette commande permet de sélectionner l'applet Employeur spécifique pour l'utilisation.

```

1 powerup;
2 // select Employer_javacard applet
3 0x00 0xA4 0x04 0x00 0xb 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00 0x00 0x7F;
4 powerdown;
5

```

apdu tool

powerup;

0x00 0xA4 0x04 0x00 0x09 0xa0 0x00 0x00 0x00 0x62 0x03 0x01 0x08 0x01 0x7F;

0x80 0xB8 0x00 0x00 0xd 0xb 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00
0x00 0x00 0x7F;

0x00 0xA4 0x04 0x00 0xb 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00 0x00
0x7F;

```

Administrator: C:\Windows\System32\cmd.exe - apdu tool

C:\Users\EM_NAHEL_aissam>start cref -o nbr_h.aissam
C:\Users\EM_NAHEL_aissam>start cref -i nbr_h.aissam -o nbr_h.aissam

C:\Users\EM_NAHEL_aissam>apdu tool
Java Card 2.2.2 APDU Tool, Version 1.3
Copyright 2005 Sun Microsystems, Inc. All rights reserved. Use is subject to license terms.
Opening connection to localhost on port 9025.
Connected.

powerup:
Received ATR = 0x3b 0xf0 0x11 0x00 0xff 0x01

0x00 0xA4 0x04 0x00 0x09 0xa0 0x00 0x00 0x00 0x62 0x03 0x01 0x08 0x01 0x7F;
CLA: 00, INS: a4, P1: 04, P2: 00, Lc: 09, a0, 00, 00, 00, 62, 03, 01, 08, 01, Le: 00, SW1: 90, SW2: 00

0x80 0xB8 0x00 0x00 0xd 0xb 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00 0x00 0x7F;
CLA: 80, INS: b8, P1: 00, P2: 00, Lc: 0d, 0b, 01, 02, 03, 04, 05, 06, 07, 08, 09, 00, 00, 00, Le: 0b, 01, 02, 03, 04, 05, 06, 07, 08, 09, 00, 00, SW1: 90, SW2: 00

0x00 0xA4 0x04 0x00 0xb 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00 0x00 0x7F;
CLA: 00, INS: a4, P1: 04, P2: 00, Lc: 0b, 01, 02, 03, 04, 05, 06, 07, 08, 09, 00, 00, Le: 00, SW1: 90, SW2: 00

```

Les commandes suivantes sont utilisées pour interagir avec l'applet, elles permettent de configurer :

- l'heure de début du travail : 11h:00min par exemple, **11** en hexadécimal est **0x0B**, et **00** en hexadécimal est **0x00**.
- l'heure de fin du travail : 12h:45min par exemple, **12** en hexadécimal est **0x0C**, et **45** en hexadécimal est **0x2D**.

Et elles permettent de calculer les heures travaillées et d'afficher les résultats.

```

0xB0 0x02 0x00 0x00 0x00 0x00; // (AFFICHE_HEURES)

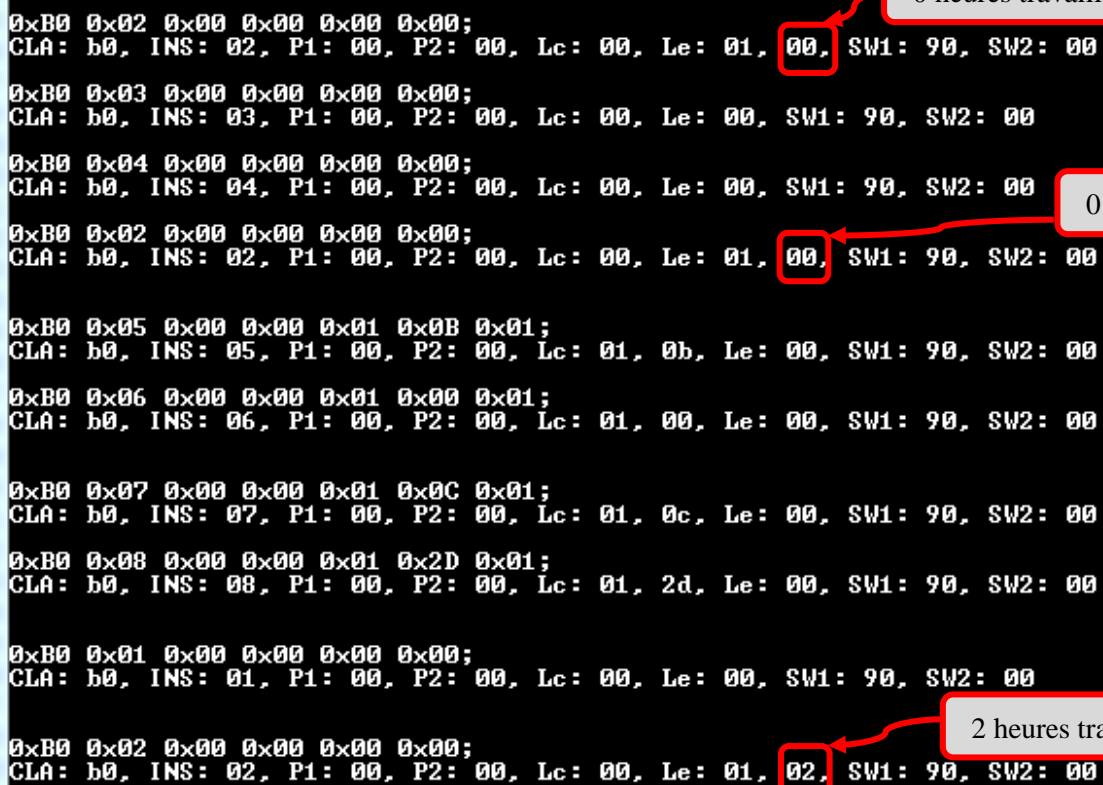
0xB0 0x03 0x00 0x00 0x00 0x00; // (A_ZERO_NBR_H)
0xB0 0x04 0x00 0x00 0x00 0x00; // (A_ZERO)
0xB0 0x02 0x00 0x00 0x00 0x00; // (AFFICHE_HEURES)

//exemple de l'heure de debut de travail : 11h:00min
0xB0 0x05 0x00 0x00 0x01 0x0B 0x01; // (START_TIME_H) (11h:)
0xB0 0x06 0x00 0x00 0x01 0x00 0x01; // (START_TIME_MIN) (:00min)

//exemple de l'heure de debut de travail : 12h:45min
0xB0 0x07 0x00 0x00 0x01 0x0C 0x01; // (STOP_TIME_H) (12h:)
0xB0 0x08 0x00 0x00 0x01 0x2D 0x01; // (STOP_TIME_MIN) (:45min)

0xB0 0x01 0x00 0x00 0x00 0x00; // (CALCUL_HEURES)
0xB0 0x02 0x00 0x00 0x00 0x00; // (AFFICHE_HEURES)

```



```

0xB0 0x02 0x00 0x00 0x00 0x00;
CLA: b0, INS: 02, P1: 00, P2: 00, Lc: 00, Le: 01, 00, SW1: 90, SW2: 00

0xB0 0x03 0x00 0x00 0x00 0x00;
CLA: b0, INS: 03, P1: 00, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00

0xB0 0x04 0x00 0x00 0x00 0x00;
CLA: b0, INS: 04, P1: 00, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00

0xB0 0x02 0x00 0x00 0x00 0x00;
CLA: b0, INS: 02, P1: 00, P2: 00, Lc: 00, Le: 01, 00, SW1: 90, SW2: 00

0xB0 0x05 0x00 0x00 0x01 0x0B 0x01;
CLA: b0, INS: 05, P1: 00, P2: 00, Lc: 01, 0b, Le: 00, SW1: 90, SW2: 00

0xB0 0x06 0x00 0x00 0x01 0x00 0x01;
CLA: b0, INS: 06, P1: 00, P2: 00, Lc: 01, 00, Le: 00, SW1: 90, SW2: 00

0xB0 0x07 0x00 0x00 0x01 0x0C 0x01;
CLA: b0, INS: 07, P1: 00, P2: 00, Lc: 01, 0c, Le: 00, SW1: 90, SW2: 00

0xB0 0x08 0x00 0x00 0x01 0x2D 0x01;
CLA: b0, INS: 08, P1: 00, P2: 00, Lc: 01, 2d, Le: 00, SW1: 90, SW2: 00

0xB0 0x01 0x00 0x00 0x00 0x00;
CLA: b0, INS: 01, P1: 00, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00

0xB0 0x02 0x00 0x00 0x00 0x00;
CLA: b0, INS: 02, P1: 00, P2: 00, Lc: 00, Le: 01, 02, SW1: 90, SW2: 00

```

On termine la connection avec l'applet en utilisant la commande **powerdown;**

```
powerdown;
```



```
powerdown;
```

```
Terminate batch job (Y/N)?  
Terminate batch job (Y/N)? y
```

```
C:\Users\EN_NAHEL_Aissam>
```

Pour tester si les données sont correctement enregistrées et effectuer une nouvelle simulation, j'ai à nouveau exécuté la commande "start cref -i nbr_h.aissam -o nbr_h.aissam" dans le **cmd.exe**, cette commande permet d'utiliser le fichier "**nbr_h.aissam**" dans lequel j'ai précédemment enregistré les données des heures travaillées.

```
start cref -i nbr_h.aissam -o nbr_h.aissam
```

```
powerdown;
```

```
Terminate batch job (Y/N)?  
Terminate batch job (Y/N)? y
```

```
C:\Users\EN_NAHEL_Aissam>start cref -i nbr_h.aissam -o nbr_h.aissam
```

```
C:\Users\EN_NAHEL_Aissam>
```

```
C:\java_card_kit-2_2_2\bin\cref.exe
```

```
Java Card 2.2.2 C Reference Implemen  
32-bit Address Space implementation  
T=1 / T=CL Dual interface APDU proto  
Copyright 2005 Sun Microsystems, Inc.
```

```
C:\java_card_kit-2_2_2\bin\cref.exe
```

```
Java Card 2.2.2 C Reference Implementation Simulator (version 0.41)  
32-bit Address Space implementation - with cryptography support  
T=1 / T=CL Dual interface APDU protocol (ISO 7816-3)  
Copyright 2005 Sun Microsystems, Inc. All rights reserved.
```

```
Memory configuration
```

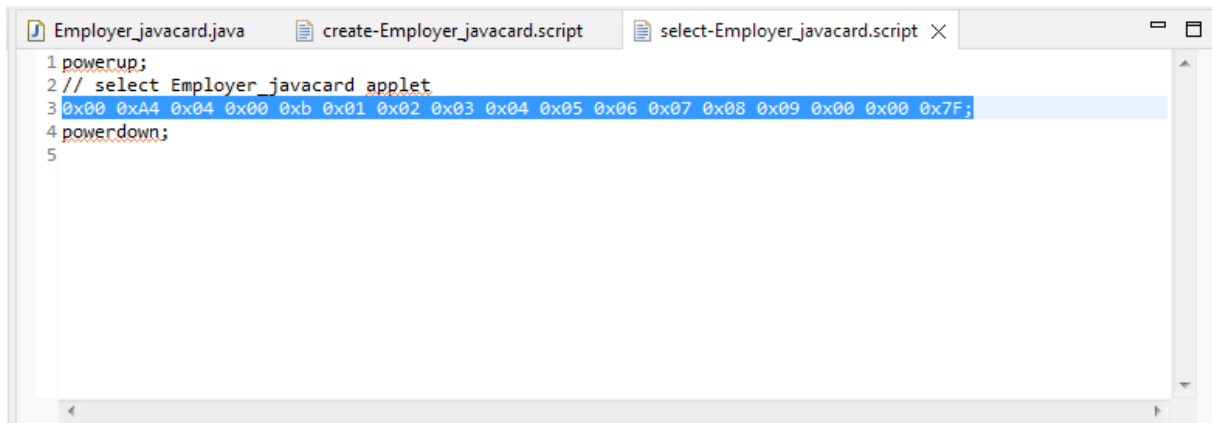
Type	Base	Size	Max Addr
RAM	0x0	0x1000	0xfff
ROM	0x2000	0xe000	0xffff
E2P	0x10020	0xffe0	0x1ffff

```
ROM Mask size = 0xce64 = 52836 bytes  
Highest ROM address in mask = 0xee63 = 61027 bytes  
Space available in ROM = 0x119c = 4508 bytes
```

```
EEPROM (0xffe0 bytes) restored from file "nbr_h.aissam"  
Using a pre-initialized Mask
```

J'établir la connexion avec l'applet par "**apdutool**", puis j'ai sélectionné l'applet.

```
apdutool
```



Lors de cette deuxième interaction avec l'applet, j'ai utilisé les commandes suivantes, qui permettent de configurer :

- l'heure de début du travail : 09h:45min par exemple, **09** en hexadécimal est **0x09**, et **45** en hexadécimal est **0x2D**.

- l'heure de fin du travail : 12h:45min par exemple, **14** en hexadécimal est **0x0E**, et **15** en hexadécimal est **0x0F**.

Et elles permettent de calculer les heures travaillées et d'afficher les résultats.

```
powerup;

0x00 0xA4 0x04 0x00 0xb 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00 0x00
0x7F;

0xB0 0x02 0x00 0x00 0x00 0x00; // (AFFICHE_HEURES)

//exemple de l'heure de début de travail: 09h:45min
0xB0 0x05 0x00 0x00 0x01 0x09 0x01; // (START_TIME_H) (09h:)
0xB0 0x06 0x00 0x00 0x01 0x2D 0x01; // (START_TIME_MIN) (:45min)

//exemple de l'heure de fin de travail : 14h:15min
0xB0 0x07 0x00 0x00 0x01 0x0E 0x01; // (STOP_TIME_H) (14h:)
0xB0 0x08 0x00 0x00 0x01 0x0F 0x01; // (STOP_TIME_MIN) (:15min)

0xB0 0x01 0x00 0x00 0x00 0x00; // (CALCUL_HEURES)
0xB0 0x02 0x00 0x00 0x00 0x00; // (AFFICHE_HEURES)

powerdown;
```

```
Administrator: C:\Windows\System32\cmd.exe
Terminate batch job (Y/N)? y
C:\Users\EN_NAHEL_Aissam>start cref -i nbr_h.aissam -o nbr_h.aissam
C:\Users\EN_NAHEL_Aissam>apdutool
Java Card 2.2.2 APDU Tool, Version 1.3
Copyright 2005 Sun Microsystems, Inc. All rights reserved. Use is subject to license terms.
Opening connection to localhost on port 9025.
Connected.

powerup;
Received ATR = 0x3b 0xf0 0x11 0x00 0xff 0x01
0x00 0xA4 0x04 0x00 0xb 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00 0x00 0x7F;
CLA: 00, INS: a4, P1: 04, P2: 00, Lc: 0b, 01, 02, 03, 04, 05, 06, 07, 08, 09, 00, 00, Le: 00, SW1: 90, SW2: 00
0xB0 0x02 0x00 0x00 0x00 0x00;
CLA: b0, INS: 02, P1: 00, P2: 00, Lc: 00, Le: 01, 02, SW1: 90, SW2: 00
2 heures travaillées

0xB0 0x05 0x00 0x00 0x01 0x09 0x01;
CLA: b0, INS: 05, P1: 00, P2: 00, Lc: 01, 09, Le: 00, SW1: 90, SW2: 00
0xB0 0x06 0x00 0x00 0x01 0x2D 0x01;
CLA: b0, INS: 06, P1: 00, P2: 00, Lc: 01, 2d, Le: 00, SW1: 90, SW2: 00
0xB0 0x07 0x00 0x00 0x01 0x0E 0x01;
CLA: b0, INS: 07, P1: 00, P2: 00, Lc: 01, 0e, Le: 00, SW1: 90, SW2: 00
0xB0 0x08 0x00 0x00 0x01 0x0F 0x01;
CLA: b0, INS: 08, P1: 00, P2: 00, Lc: 01, 0f, Le: 00, SW1: 90, SW2: 00

0xB0 0x01 0x00 0x00 0x00 0x00;
CLA: b0, INS: 01, P1: 00, P2: 00, Lc: 00, Le: 00, SW1: 90, SW2: 00
7 heures travaillées
0xB0 0x02 0x00 0x00 0x00 0x00;
CLA: b0, INS: 02, P1: 00, P2: 00, Lc: 00, Le: 01, 07, SW1: 90, SW2: 00
powerdown;
Terminate batch job (Y/N)? y
C:\Users\EN_NAHEL_Aissam>
```

Pour vérifier le nombre d'heures travaillées

```
start cref -i nbr_h.aissam -o nbr_h.aissam
```

```
apdutool
```

```
powerup;
```

```
0x00 0xA4 0x04 0x00 0xb 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00 0x00
0x7F; // (Sélectionner l'Applet)
```

```
0xB0 0x02 0x00 0x00 0x00 0x00; // (AFFICHE_HEURES)
```

```
powerdown;
```

```
Administrator: C:\Windows\System32\cmd.exe
Terminate batch job (Y/N)? y
C:\Users\EN_NAHIEL_Aissam>start cref -i nbr_h.aissam -o nbr_h.aissam
C:\Users\EN_NAHIEL_Aissam>apdutool
Java Card 2.2.2 APDU Tool, Version 1.3
Copyright 2005 Sun Microsystems, Inc. All rights reserved. Use is subject to license terms.
Opening connection to localhost on port 9025.
Connected.

powerup;
Received ATR = 0x3b 0xf0 0x11 0x00 0xff 0x01
0x00 0xA4 0x04 0x00 0xb 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x09 0x00 0x00 0x7F;
CLA: 00, INS: a4, P1: 04, P2: 00, Lc: 0b, 01, 02, 03, 04, 05, 06, 07, 08, 09, 00, 00, Le: 00, SW1: 90, SW2: 00
0xB0 0x02 0x00 0x00 0x00 0x00;
CLA: b0, INS: 02, P1: 00, P2: 00, Lc: 00, Le: 01, 07, SW1: 90, SW2: 00
powerdown;
Terminate batch job (Y/N)? y
C:\Users\EN_NAHIEL_Aissam>
```

7 heures travaillées