NFC Secure Element Example Java Card for Nokia6131

Warning: This example was created for the NFC implementation provided on Series 40 devices, and may include use of features not yet available for NFC on Symbian devices.

This is the code for the JAR file in order to compile a real Java Card CAP File and put into the secure element of the Nokia 6131 (make sure you have either G&D SmartC@fe, SUN's JavaCard Developer Kit for JavaCard or JCOP Tools at hand). Download appropriate Eclipse/JCOP Project: File:Ticket Applet.zip.

```
/**
 * Implementation of very simple wallet. The Wallet Container holds a
 * value (byte) which can be increase, decreased and read by sending
 * an appropriate APDU
 */
package congress;

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

/**
 * @author gmadlmay
 *
 */
public class Ticketing extends Applet {

 /**
  *  command set for communication between the
  *  J2ME application and Java Card Applet as well as
  *  the Java card Applet and the external reader
  */

 // Command (APDU INS) for increading the value in the wallet
 private final static byte INS_INC = 0x01;

 // Command (APDU INS) for decreading the value in the wallet
 private final static byte INS_DEC = 0x02;

 // command (APDU INS) for reading the value in the wallet
 private final static byte INS_READ = 0x03;

 // variable holding the amount stored in the wallet.
 private byte value = (byte) 0x20;

 public static void install(byte[] bArray, short bOffset, byte bLength) {
  // GP-compliant JavaCard applet registration
  new Ticketing()
    .register(bArray, (short) (bOffset + 1), bArray[bOffset]);
 }

 public void process(APDU apdu) {
  // Good practice: Return 9000 on SELECT
  if (selectingApplet()) {
   return;
  }

  // grab the byte-array of the APDU
  byte[] buf = apdu.getBuffer();

  // check the INS Byte of the APDU and take
  // the appropriate action
  switch (buf[ISO7816.OFFSET_INS]) {

  // increase the value in the wallet
  case (byte) INS_INC:
   if (value < 0xff)
    value++;

   // return no error, in case everything was okay
   ISOException.throwIt(ISO7816.SW_NO_ERROR);
   break;

  // increase the value in the wallet
  case (byte) INS_DEC:
   if (value > 0x00)
    value--;
   // return no error, in case everything was okay
   ISOException.throwIt(ISO7816.SW_NO_ERROR);
   break;

  // read the value of the wallet
  case (byte) INS_READ:

   // create a byte array out of the value (length: 1)
   byte[] outBuffer = new byte[1];
   outBuffer[0] = value;

   // now send back the data.
   apdu.setOutgoing();
   apdu.setOutgoingLength((byte) outBuffer.length);
   apdu.sendBytesLong(outBuffer, (short) 0, (byte) outBuffer.length);

   break;

  // all other INS
  default:
   // good practice: If you don't know the INStruction, say so:
   ISOException.throwIt(ISO7816.SW_INS_NOT_SUPPORTED);
  }
 }
}
```


(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.