karty inteligentne

smart cards

bezpieczny nośnik informacji

dane abonenta, do płatności, bilety, klucz do podpisu, ...

niektóre można programować

komunikacja

przesyłanie danych pomiędzy aplikacją i kartą

po włożeniu karty

ATR

komendy i odpowiedzi

ISO 7816

APDU

Application Protocol Data Unit

RAPDU

Response Application Protocol Data Unit

Java Card

karty z maszyną wirtualną

ale nieco „inną”

kilkadziesiąt kB EEPROM i kilka kB RAM

Java Card Virtual Machine

brak typu int

i również long, float, double, char

zazwyczaj brak procesu garbage collection

dostępny na wyraźne życzenie

~~**wątki, dynamiczne ładowanie klas, wielowymiarowe struktury danych**~~

~~**klasa String, Security Manager, lambdy, strumienie**~~

API nakierowane na operacje na tablicach, kryptografię, atomowość i typowe zastosowania kart

aplet kartowy

pakiet kartowy

aplikacja w karcie, która odbiera APDU i odsyła RAPDU

javacard.framework.Applet

AID

identyfikator aplikacji (pakietu)

APDU, RAPDU, APDU, ...

javacard.framework.APDU

javax.smartcardio.CommandAPDU / ResponseAPDU

Card Manager

zarządza kartą

ale nie innymi apletami

dobre praktyki

im mniej obiektowości tym lepiej

im więcej private static final tym lepiej

new tylko w konstruktorze

zmienne robocze w RAM

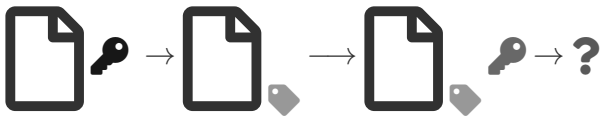
używanie API gdzie się tylko da

żadnych sekretnych wejść do aplikacji

wszystko w granicach rozsądku

dane mają być bezpieczne + karta ma działać szybko i niezawodnie

karta generuje losową parę kluczy: **prywatny** i **publiczny**



z karty wyciągamy tylko klucz **publiczny**

klucz **prywatny** **nigdy nie opuszcza karty**

Co dalej?

PC/SC

javax.smartcardio.*

można odczytać kartę czytnikiem NFC

Java Card Development Kit

przykłady + symulator cref

można zasymulować kartę „czytnikiem” NFC