Popular Methods

- getBasicChannel ( )
- disconnect ( )

Related Classes

- java.io.BufferedReader
- java.io.FileReader
- java.security.NoSuchAlgorithmException
- java.util.concurrent.ExecutionException
- java.util.concurrent.Future
- java.util.ListIterator
- java.util.EnumMap
- java.security.Security
- java.security.AccessController
- java.security.PrivilegedAction

- org.apache.commons.codec.binary.Hex
- java.util.concurrent.ConcurrentSkipListMap
- javax.smartcardio.CardException
- javax.smartcardio.CardTerminal
- javax.smartcardio.CommandAPDU
- javax.smartcardio.TerminalFactory
- javax.smartcardio.ResponseAPDU
- javax.smartcardio.CardChannel
- javax.smartcardio.CardTerminals
- javax.smartcardio.ATR
- javax.smartcardio.CardNotPresentException
- com.github.devnied.emvnfccard.parser.EmvParser
- com.github.devnied.emvnfccard.exception.CommunicationException

Java Code Examples for javax.smartcardio.Card

The following are top voted examples for showing how to use javax.smartcardio.Card. These examples are extracted from open source projects. You can vote up the examples you like and your votes will be used in our system to generate more good examples.

Example 1

Project: *eid-applet* File: *PcscTest.java* View source code 7 votes

```
@Test
public void testReadPhoto() throws Exception {
        TerminalFactory terminalFactory = TerminalFactory.getDefault();
        CardTerminals cardTerminals = terminalFactory.terminals();
        CardTerminal cardTerminal = cardTerminals.list().get(0);
        Card card = cardTerminal.connect("T=0");
        CardChannel cardChannel = card.getBasicChannel();
        // select file
        cardChannel.transmit(
                        new CommandAPDU(0x00, 0xA4, 0x08, 0x0C, new byte[] { 0x3F, 0x00, (byte) 0xDF, 0x01, 0x40, 0x35 }));
        ByteArrayOutputStream baos = new ByteArrayOutputStream();
        int offset = 0;
        ResponseAPDU responseApdu;
        do {
                // read binary
                responseApdu = cardChannel.transmit(new CommandAPDU(0x00, 0xB0, offset >> 8, offset & 0xFF, 0xff));
                baos.write(responseApdu.getData());
                offset += responseApdu.getData().length;
        } while (responseApdu.getData().length == 0xff);
        BufferedImage photo = ImageIO.read(new ByteArrayInputStream(baos.toByteArray()));
        JOptionPane.showMessageDialog(null, new ImageIcon(photo));
}
```

Example 2

Project: *EMV-NFC-Paycard-Enrollment* File: *Main.java* View source code 7 votes

```
public static void main(final String[] args) throws CardException, CommunicationException {

                TerminalFactory factory = TerminalFactory.getDefault();
                List<CardTerminal> terminals = factory.terminals().list();
                LOGGER.info("Terminals: " + terminals);

                if (terminals != null && !terminals.isEmpty()) {
                        // Use the first terminal
                        CardTerminal terminal = terminals.get(0);

                        // Connect with the card
                        Card card = terminal.connect("*");
```

```
                        LOGGER.info("card: " + card);
                        CardChannel channel = card.getBasicChannel();

                        PcscProvider provider = new PcscProvider(channel);
                        EmvParser parser = new EmvParser(provider, false);
                        parser.readEmvCard();

                        // Disconnect the card
                        card.disconnect(false);
                } else {
                        LOGGER.error("No pcsc terminal found");
                }

        }
```

Example 3

Project: *dss* File: *MOCCASignatureTokenConnection.java* View source code 6 votes

```
private List<SignatureCard> getSignatureCards() {

                if (_signatureCards == null) {

                        _signatureCards = new ArrayList<SignatureCard>();
                        SmartCardIO io = new SmartCardIO();
                        SignatureCardFactory factory = SignatureCardFactory.getInstance();

                        for (Entry<CardTerminal, Card> entry : io.getCards().entrySet()) {
                                try {
                                        _signatureCards.add(factory.createSignatureCard(entry.getValue(), entry.getKey()));
                                } catch (CardNotSupportedException e) {
                                        // just log the error - MOCCA tries to connect to all cards and we may have an MSCAPI or PKCS1
                                        // inserted.
                                        LOG.info(e.getMessage());
                                }
                        }
                }
                return _signatureCards;
        }
```

Example 4

Project: *secure-element-gpdroid-nfc* File: *GPConnection.java* View source code 6 votes

```
/**
 * Method that acquires a CardChannel from the given terminal and performs
 * the given GPCommand on it
 *
 * @param keyset
 * @param channelSet
 * @param _cmd
 * @return
 */
public String performCommand(CardTerminal _term, GPKeyset keyset, GPChannelSet channelSet,
                GPCommand _cmd) {

        try {
                Card c = null;

                if (_term instanceof OpenMobileAPITerminal) {
                        ((OpenMobileAPITerminal) _term).setReader(_cmd.getSeekReader());
                }

                c = _term.connect("*");
                System.out.println("Found card in terminal: " + _term.getName());
                if (c.getATR() != null) {
                        System.out.println("ATR: "
                                        + GPUtil.byteArrayToString(c.getATR().getBytes()));
                }
                CardChannel channel = c.openLogicalChannel();
                return performCommand(channel, keyset, channelSet, _cmd);
        } catch (Exception e) {
                e.printStackTrace();
        }
        return null;
}
```

Example 5

Project: *commons-eid* File: *CCID.java* View source code 6 votes

```
private void getFeaturesUsingPPDU(final Card card) throws CardException {
        ResponseAPDU responseAPDU = card.getBasicChannel().transmit(
                        new CommandAPDU((byte) 0xff, (byte) 0xc2, 0x01, 0x00,
                                        new byte[]{}, 32));
        this.logger.debug("PPDU response: "
                        + Integer.toHexString(responseAPDU.getSW()));
        if (responseAPDU.getSW() == 0x9000) {
                byte[] featureBytes = responseAPDU.getData();
                this.logger
                                .debug("CCID FEATURES found using Pseudo-APDU Fallback Strategy");
                for (FEATURE feature : FEATURE.values()) {
                        Integer featureCode = findFeaturePPDU(feature.getTag(),
                                        featureBytes);
                        if (featureCode != null) {
                                this.features.put(feature, featureCode);
                                this.logger.debug("FEATURE " + feature.name() + " = "
                                                + Integer.toHexString(featureCode));
                        }
                }
                this.usesPPDU = true;
        } else {
                this.logger.error("CCID Features via PPDU Not Supported");
        }
}
```

Example 6

Project: *jdk8u-dev-jdk* File: *TestDirect.java* *View source code* 6 votes

```
public static void main(String[] args) throws Exception {
    TerminalFactory terminalFactory = TerminalFactory.getDefault();
    List<CardTerminal> cardTerminals = terminalFactory.terminals().list();
    System.out.println("Terminals: " + cardTerminals);
    if (cardTerminals.isEmpty()) {
        throw new Exception("No card terminals available");
    }
    CardTerminal cardTerminal = cardTerminals.get(0);
    Card card = cardTerminal.connect("DIRECT");
    card.disconnect(true);

    System.out.println("OK.");
}
```

Example 7

Project: *apdu4j* File: *JSONCardTerminal.java* *View source code* 6 votes

```
@Override
public Card connect(String protocol) throws CardException {
        try {
                HashMap<String, Object> m = new HashMap<>();
                m.put("cmd", "CONNECT");
                m.put("protocol", protocol);
                client.send(m);

                // Read back the response
                Map<String, Object> r = client.recv();
                if (r.containsKey("CONNECT") && ((String)r.get("CONNECT")).equals("OK")) {

                        if (r.containsKey("atr") && r.containsKey("protocol") && r.containsKey("reader")) {
                                terminalName = (String) r.get("reader");
                                atr = new ATR(HexUtils.stringToBin((String) r.get("atr")));
                                this.protocol = ((String) r.get("protocol"));
                                return new JSONCard(this);
                        }
                }
                throw new CardException("Could not connect to client");
        } catch (IOException e) {
                throw new CardException("Could not connect to client", e);
        }

}
```

Example 8

Project: *GlobalPlatformPro* File: *SEAccessControlUtility.java* *View source code* 6 votes

```
/**
 * Send Access Control rule GET DATA.
 *
 * @param card
 * @param P1
 * @return
 * @throws CardException
 * @throws GPException
 */
private static ResponseAPDU sendAcrGetData(final Card card, final byte P1) throws CardException, GPException {
        CommandAPDU list = new CommandAPDU(GlobalPlatform.CLA_GP, GlobalPlatform.INS_GET_DATA, 0xFF, P1, 256);

        ResponseAPDU response = card.getBasicChannel().transmit(list);

        try {
                GPException.check(response, "ACR GET DATA failed");
        } catch (GPException e) {
                if (SEAccessControl.ACR_GET_DATA_ERROR.containsKey(e.sw)) {
```

```
                        System.out.println("[SW] " + SEAccessControl.ACR_GET_DATA_ERROR.get(e.sw));
                }
                throw e;
        }
        return response;
}
```

Example 9

Project: *GlobalPlatformPro* File: *SEAccessControlUtility.java* View source code 6 votes

```
/**
 * List access rules.
 *
 * @param card
 * @throws CardException
 * @throws GPException
 */
public static void acrList(final GlobalPlatform gp, final Card card) throws CardException, GPException {
        try {
                gp.select(SEAccessControl.ACR_AID);
                ResponseAPDU response = sendAcrGetData(card, SEAccessControl.ACR_GET_DATA_ALL);
                SEAccessControl.BerTlvData temp = SEAccessControl.AcrListResponse.getAcrListData(null, response.getData());

                while (temp.getCurrentIndex() < temp.getLength()) {
                        response = sendAcrGetData(card, SEAccessControl.ACR_GET_DATA_NEXT);
                        temp = SEAccessControl.AcrListResponse.getAcrListData(temp, response.getData());
                }

                SEAccessControl.AcrListResponse resp = SEAccessControl.AcrListResponse.fromBytes(temp.getLength(), temp.getData());
                SEAccessControl.printList(resp.acrList);
        } catch (GPException e) {
                throw new GPException("Could not read " + SEAccessControl.ACR_AID);
        }
}
```

Example 10

Project: *irma_future_id* File: *SCTerminal.java* View source code 6 votes

```
void updateTerminal() {
if (! isCardPresent()) {
    scCard = null;
} else {
    try {
        if (scCard != null) {
            // check if it is the same card, else remove
            Card newCard = terminal.connect("*");
            if (! scCard.equalCardObj(newCard)) {
                scCard = null;
            }
        }
    } catch (CardException ex) {
        // error means delete it anyways
        scCard = null;
    }
}
   }
```

Example 11

Project: *irma_future_id* File: *NFCCardTerminal.java* View source code 6 votes

```
@Override
   public synchronized Card connect(String arg0) throws CardException {
if (nfcCard == null || this.nfcCard.isodep == null) {
    logger.warn("No tag present.");
    throw new CardNotPresentException("No tag present");
}
try {
    if (! nfcCard.isodep.isConnected()) {
        nfcCard.isodep.setTimeout(3000);
        nfcCard.isodep.connect();
    }
} catch (IOException e) {
    nfcCard = null;
    throw new CardException("No connection could be established", e);
}
return nfcCard;
   }
```

Example 12

Project: *ts-cards* File: *SmartcardIoTransmitter.java* View source code 5 votes

```
public SmartcardIoTransmitter(Card card) {
    this.card = card;
}
```

Example 13

Project: *ts-cards* File: *SmartcardIoTransmitter.java* View source code 5 votes

```
public static SmartcardIoTransmitter create() throws CardException {
    TerminalFactory tf = TerminalFactory.getDefault();

    CardTerminal cardTerminal = null;

    if (tf.terminals().list().size() == 1) {
        cardTerminal = tf.terminals().list().get(0);
    } else {
        for (CardTerminal terminal : tf.terminals().list()) {
            logger.info("Checking for card: {}", terminal.getName());
            if (terminal.isCardPresent()) {
                cardTerminal = terminal;
            }
        }
    }

    if (cardTerminal != null) {
        cardTerminal.waitForCardPresent(0);
    } else {
        throw new RuntimeException("No card found!");
    }

    logger.info("Using -> " + cardTerminal.getName());

    Card card = cardTerminal.connect("*");

    return new SmartcardIoTransmitter(card);

}
```

Example 14

Project: *JCMathLib* File: *CardManager.java* View source code 5 votes

```
private Card waitForCard(CardTerminals terminals)
        throws CardException {
    while (true) {
        for (CardTerminal ct : terminals
                .list(CardTerminals.State.CARD_INSERTION)) {

            return ct.connect("*");
        }
        terminals.waitForChange();
    }
}
```

Example 15

Project: *jdk8u-jdk* File: *TestDirect.java* View source code 5 votes

```
public static void main(String[] args) throws Exception {
    TerminalFactory terminalFactory = TerminalFactory.getDefault();
    List<CardTerminal> cardTerminals = terminalFactory.terminals().list();
    if (cardTerminals.isEmpty()) {
        System.out.println("Skipping the test: " +
                "no card terminals available");
        return;
    }
    System.out.println("Terminals: " + cardTerminals);
    CardTerminal cardTerminal = cardTerminals.get(0);
    Card card = cardTerminal.connect("DIRECT");
    card.disconnect(true);

    System.out.println("OK.");
}
```

Example 16

Project: *openjdk-jdk10* File: *TestDirect.java* View source code 5 votes

```
public static void main(String[] args) throws Exception {
    TerminalFactory terminalFactory = TerminalFactory.getDefault();
    List<CardTerminal> cardTerminals = terminalFactory.terminals().list();
    if (cardTerminals.isEmpty()) {
        System.out.println("Skipping the test: " +
                "no card terminals available");
        return;
    }
    System.out.println("Terminals: " + cardTerminals);
    CardTerminal cardTerminal = cardTerminals.get(0);
    Card card = cardTerminal.connect("DIRECT");
    card.disconnect(true);

    System.out.println("OK.");
}
```

Example 17

Project: *Myst* File: *CardManagement.java* View source code 5 votes

```
private static void connectToAllCardsByTerminalFactory(TerminalFactory factory, byte[] appAID, ArrayList<CardChannel> cardsList) throw
    ArrayList<CardTerminal> terminals = new ArrayList<>();

    Card card = null;
    try {
        for (CardTerminal t : factory.terminals().list()) {
            terminals.add(t);
            if (t.isCardPresent()) {
                System.out.print("Connecting...");
                card = t.connect("*"); // Connect with the card

                System.out.println(" Done.");

                System.out.print("Establishing channel...");
                CardChannel channel = card.getBasicChannel();

                System.out.println(" Done.");

                // Select applet (mpcapplet)
                System.out.println("Smartcard: Selecting applet...");
                CommandAPDU cmd = new CommandAPDU(appAID);
                ResponseAPDU response = transmit(channel, cmd);

                if (response.getSW() == (ISO7816.SW_NO_ERROR & 0xffff)) {
                    cardsList.add(channel);
                }
            }
        }
    } catch (Exception e) {
        System.out.println("Failed.");
    }

    System.out.println("MPC cards found: " + cardsList.size());
}
```

Example 18

Project: *openjdk9* File: *TestDirect.java* *View source code* 5 votes

```
public static void main(String[] args) throws Exception {
    TerminalFactory terminalFactory = TerminalFactory.getDefault();
    List<CardTerminal> cardTerminals = terminalFactory.terminals().list();
    if (cardTerminals.isEmpty()) {
        System.out.println("Skipping the test: " +
                "no card terminals available");
        return;
    }
    System.out.println("Terminals: " + cardTerminals);
    CardTerminal cardTerminal = cardTerminals.get(0);
    Card card = cardTerminal.connect("DIRECT");
    card.disconnect(true);

    System.out.println("OK.");
}
```

Example 19

Project: *jdk8u_jdk* File: *TestDirect.java* *View source code* 5 votes

```
public static void main(String[] args) throws Exception {
    TerminalFactory terminalFactory = TerminalFactory.getDefault();
    List<CardTerminal> cardTerminals = terminalFactory.terminals().list();
    if (cardTerminals.isEmpty()) {
        System.out.println("Skipping the test: " +
                "no card terminals available");
        return;
    }
    System.out.println("Terminals: " + cardTerminals);
    CardTerminal cardTerminal = cardTerminals.get(0);
    Card card = cardTerminal.connect("DIRECT");
    card.disconnect(true);

    System.out.println("OK.");
}
```

Example 20

Project: *lookaside_java-1.8.0-openjdk* File: *TestDirect.java* *View source code* 5 votes

```
public static void main(String[] args) throws Exception {
    TerminalFactory terminalFactory = TerminalFactory.getDefault();
    List<CardTerminal> cardTerminals = terminalFactory.terminals().list();
    if (cardTerminals.isEmpty()) {
        System.out.println("Skipping the test: " +
                "no card terminals available");
        return;
    }
    System.out.println("Terminals: " + cardTerminals);
    CardTerminal cardTerminal = cardTerminals.get(0);
    Card card = cardTerminal.connect("DIRECT");
    card.disconnect(true);

    System.out.println("OK.");
}
```

Example 21

Project: *eid-applet* File: *PcscEid.java* View source code 5 votes

```
public ResponseAPDU transmit(byte[] command, Card card, CardChannel cardChannel) throws CardException {
        if (this.ioctl == null) {
                // PPDU
                return cardChannel.transmit(new CommandAPDU(0xff, 0xc2, 0x01, this.feature, command));
        } else {
                byte[] result = card.transmitControlCommand(this.ioctl, command);
                ResponseAPDU responseApdu = new ResponseAPDU(result);
                return responseApdu;
        }
}
```

Example 22

Project: *eid-applet* File: *PcscEid.java* View source code 5 votes

```
public byte[] transmitByteResponse(byte[] command, Card card, CardChannel cardChannel) throws CardException {
        if (this.ioctl == null) {
                // PPDU
                return cardChannel.transmit(new CommandAPDU(0xff, 0xc2, 0x01, this.feature, command)).getData();
        } else {
                byte[] result = card.transmitControlCommand(this.ioctl, command);
                return result;
        }
}
```

Example 23

Project: *eid-applet* File: *PcscEid.java* View source code 5 votes

```
public void logoff(String readerName) throws Exception {
        this.view.addDetailMessage("logoff from reader: \"" + readerName + "\"");
        TerminalFactory factory = TerminalFactory.getDefault();
        CardTerminals cardTerminals = factory.terminals();
        CardTerminal cardTerminal = cardTerminals.getTerminal(readerName);
        if (null == cardTerminal) {
                this.view.addDetailMessage("logoff: card reader not found: " + readerName);
                List<String> readerList = getReaderList();
                this.view.addDetailMessage("reader list: " + readerList);
                // throw new RuntimeException("card reader not found: " +
                // readerName);
                // we won't fail in this case...
                return;
        }
        Card card = cardTerminal.connect("T=0");
        try {
                CardChannel cardChannel = card.getBasicChannel();
                CommandAPDU logoffApdu = new CommandAPDU(0x80, 0xE6, 0x00, 0x00);
                ResponseAPDU responseApdu = cardChannel.transmit(logoffApdu);
                this.view.addDetailMessage("logoff... " + readerName);
                if (0x9000 != responseApdu.getSW()) {
                        this.view.addDetailMessage("logoff status word: " + Integer.toHexString(responseApdu.getSW()));
                        if (0x6e00 == responseApdu.getSW()) {
                                // BUD001 smart card reader work-around
                                return;
                        }
                        throw new RuntimeException("logoff failed");
                }
        } finally {
                card.disconnect(true);
        }
}
```

Example 24

Project: *javaemvreader* File: *SmartcardioTerminalProviderImpl.java* View source code 5 votes

```
@Override
public CardConnection connectTerminal(String name) throws TerminalException {
    try {
        CardTerminal smartCardIOTerminal = terminals.getTerminal(name);
        Card _card = smartCardIOTerminal.connect("*");
        return new SmartcardioCardConnection(_card, smartCardIOTerminal);
    } catch (CardException ex) {
        throw new TerminalException(ex);
    }
}
```

Example 25

Project: *animamea* File: *Smart.java* View source code 5 votes

```
public static void main(String[] args) {
                try {
                        // show the list of available terminals
                        TerminalFactory factory = TerminalFactory.getDefault();
                        List<CardTerminal> terminals = factory.terminals().list();
```

```
                        // get the first terminal
                        if (terminals.isEmpty()) {
                                System.out.println("No terminals found!");
                        } else {
                                System.out.println("Terminals: " + terminals);
                                CardTerminal terminal = terminals.get(0);
                                // establish a connection with the card
                                // Card card = terminal.connect("T=1");
                                Card card = terminal.connect("DIRECT");
                                System.out.println("card: " + card);
                                byte[] ccidResp = card.transmitControlCommand(
                                                CM_IOCTL_GET_FEATURE_REQUEST, new byte[] {});
                                System.out.println(HexString.bufferToHex(ccidResp));
                                CardChannel channel = card.getBasicChannel();
//                              channel.transmit(new CommandAPDU(new byte[]{0,1,2,3,4,5,6}));
                                // disconnect
                                card.disconnect(false);
                        }
                } catch (Exception e) {
                        System.err.println(e.getLocalizedMessage());
                }
        }
```

Example 26

Project: *commons-eid* File: *MixedDetectionExamples.java* View source code 5 votes

```
@Override
public void cardInserted(final CardTerminal cardTerminal, final Card card) {
        if (card != null) {
                System.out.println("Other Card ["
                                + String.format("%x", new BigInteger(1, card.getATR()
                                                .getBytes())) + "] Inserted Into Terminal ["
                                + cardTerminal.getName() + "]");
        } else {
                System.out.println("Other Card Inserted Into Terminal ["
                                + cardTerminal.getName() + "] but failed to connect()");
        }
}
```

Example 27

Project: *commons-eid* File: *SimulatedCardTerminal.java* View source code 5 votes

```
@Override
public Card connect(final String protocol) throws CardException {
        if (!isCardPresent()) {
                throw new CardException("No Card Present");
        }
        return this.card;
}
```

Example 28

Project: *commons-eid* File: *CardAndTerminalManagerExercises.java* View source code 5 votes

```
@Override
public void cardInserted(final CardTerminal cardTerminal, final Card card) {
        if (card != null) {
                LOGGER.debug("Card [{}] Inserted Into Terminal [{}]", StringUtils.atrToString(card.getATR()),
                                cardTerminal.getName() + "]");
        } else {
                LOGGER.debug("Card present but failed to connect()");
        }
}
```

Example 29

Project: *commons-eid* File: *BeIDCardManagerExercise.java* View source code 5 votes

```
@Override
public void cardInserted(final CardTerminal cardTerminal, final Card card) {
        if (card != null) {
                System.out.println("Other Card ["
                                + String.format("%x", new BigInteger(1, card.getATR()
                                                .getBytes())) + "] Inserted Into Terminal ["
                                + cardTerminal.getName() + "]");
        } else {
                System.out.println("Other Card Inserted Into Terminal ["
                                + cardTerminal.getName() + "] but failed to connect()");
        }
}
```

Example 30

Project: *commons-eid* File: *CardAndTerminalManagerTests.java* View source code 5 votes

```
@Override
public synchronized void cardInserted(final CardTerminal cardTerminal,
```

```
                final Card card) {
        if (this.recordedState.containsKey(cardTerminal)) {
                throw new IllegalStateException(
                                "Cannot Insert 2 Cards in 1 CardTerminal");
        }
        this.recordedState.put(cardTerminal, card);
}
```

Example 31

Project: *commons-eid* File: *CardAndTerminalManager.java* View source code 5 votes

```
private void listenersTerminalsWithCardsInserted(
                final Set<CardTerminal> inserted) {
        if (!inserted.isEmpty()) {
                Set<CardEventsListener> copyOfListeners;

                synchronized (this.cardEventsListeners) {
                        copyOfListeners = new HashSet<CardEventsListener>(
                                        this.cardEventsListeners);
                }

                for (CardTerminal terminal : inserted) {
                        Card card = null;

                        if (this.autoconnect) {
                                try {
                                        card = terminal.connect(this.protocol.getProtocol());
                                } catch (final CardException cex) {
                                        this.logger.debug("terminal.connect("
                                                        + this.protocol.getProtocol() + ") failed. "
                                                        + cex.getMessage());
                                }
                        }

                        for (CardEventsListener listener : copyOfListeners) {
                                try {
                                        listener.cardInserted(terminal, card);
                                } catch (final Exception thrownInListener) {
                                        this.logger
                                                        .error("Exception thrown in CardEventsListener.cardInserted:"
                                                                        + thrownInListener.getMessage());
                                }

                        }
                }
        }
}
```

Example 32

Project: *commons-eid* File: *CCID.java* View source code 5 votes

```
public CCID(final Card card, final CardTerminal cardTerminal,
                final Logger logger) {
        this.card = card;
        this.logger = logger;
        this.features = new EnumMap<FEATURE, Integer>(FEATURE.class);
        this.usesPPDU = false;

        final boolean onMSWindows = (System.getProperty("os.name") != null && System
                        .getProperty("os.name").startsWith("Windows"));

        this.logger
                        .debug("Getting CCID FEATURES using standard control command");
        try {
                getFeaturesUsingControlChannel(card, onMSWindows);
                if (this.features.isEmpty() && onMSWindows
                                && isPPDUCardTerminal(cardTerminal.getName())) {
                        // Windows 10 work-around
                        getFeaturesUsingPPDU(card);
                }
        } catch (final CardException cexInNormal) {
                this.logger
                                .debug("GET_FEATURES over standard control command failed: "
                                                + cexInNormal.getMessage());
                if (onMSWindows && isPPDUCardTerminal(cardTerminal.getName())) {
                        this.logger
                                        .debug("Attempting To get CCID FEATURES using Pseudo-APDU Fallback Strategy");
                        try {
                                getFeaturesUsingPPDU(card);
                        } catch (CardException cexInPseudo) {
                                this.logger
                                                .error("Pseudo-APDU Fallback strategy failed as well: "
                                                                + cexInPseudo.getMessage());
                        }
                } else {
                        this.logger
                                        .debug("Not risking PPDU Fallback strategy for CardTerminal ["
                                                        + cardTerminal.getName() + "] on this platform");
                }
        }
}
```

Example 33

Project: *commons-eid* File: *CCID.java* View source code 5 votes

```
private void getFeaturesUsingControlChannel(final Card card,
                final boolean onMSWindows) throws CardException {
        byte[] featureBytes = card.transmitControlCommand(onMSWindows
                        ? GET_FEATURES_MICROSOFT
                        : GET_FEATURES, new byte[0]);
        this.logger.debug("CCID FEATURES found using standard control command");
        for (FEATURE feature : FEATURE.values()) {
                Integer featureCode = findFeatureTLV(feature.getTag(), featureBytes);
                if (featureCode != null) {
                        this.features.put(feature, featureCode);
                        this.logger.debug("FEATURE " + feature.name() + " = "
                                        + Integer.toHexString(featureCode));
                }
        }
}
```

Example 34

Project: *infobip-open-jdk-8* File: *TestDirect.java* View source code 5 votes

```
public static void main(String[] args) throws Exception {
    TerminalFactory terminalFactory = TerminalFactory.getDefault();
    List<CardTerminal> cardTerminals = terminalFactory.terminals().list();
    System.out.println("Terminals: " + cardTerminals);
    if (cardTerminals.isEmpty()) {
        throw new Exception("No card terminals available");
    }
    CardTerminal cardTerminal = cardTerminals.get(0);
    Card card = cardTerminal.connect("DIRECT");
    card.disconnect(true);

    System.out.println("OK.");
}
```

Example 35

Project: *esteidhacker* File: *CardDelegate.java* View source code 5 votes

```
public static CardDelegate any(boolean sign, NotificationInterface console) throws CardException {
    CardTerminal t = EstEID.getTerminal();
    if (t == null) {
        return null;
    }
    t = LoggingCardTerminal.getInstance(t);
    Card c = t.connect("*");
    return new CardDelegate(c.getBasicChannel(), sign, console == null ? CONSOLE : console);
}
```

Example 36

Project: *smartcard-io* File: *Connect.java* View source code 5 votes

```
protected void connect(CardTerminal cardTerminal) throws Exception {
        System.out.println("" + cardTerminal + " connect");
        Card card = cardTerminal.connect("*");
        try {
                System.out.println("" + card + " begin transaction");
                card.beginExclusive();
                System.out.println("" + card + " end transaction");
                card.endExclusive();
        } finally {
                System.out.println("" + card + " disconnect");
                card.disconnect(false);
        }
}
```

Example 37

Project: *irma_future_id* File: *SeekTerminal.java* View source code 5 votes

```
@Override
   public Card connect(String arg0) throws CardException {
try {
    return new SeekCard(reader.openSession());
} catch (IOException e) {
    throw new CardException(e);
}
   }
```

Example 38

Project: *irma_future_id* File: *TerminalImpl.java* View source code 5 votes

```
public synchronized Card connect(String protocol) throws CardException {
SecurityManager sm = System.getSecurityManager();
if (sm != null) {
```

```
    sm.checkPermission(new CardPermission(name, "connect"));
}
if (card != null) {
    if (card.isValid()) {
        String cardProto = card.getProtocol();
        if (protocol.equals("*") || protocol.equalsIgnoreCase(cardProto)) {
            return card;
        } else {
            throw new CardException("Cannot connect using " + protocol
                    + ", connection already established using " + cardProto);
        }
    } else {
        card = null;
    }
}
try {
    card = new CardImpl(this, protocol);
    return card;
} catch (PCSCException e) {
    if (e.code == SCARD_W_REMOVED_CARD) {
        throw new CardNotPresentException("No card present", e);
    } else {
        throw new CardException("connect() failed", e);
    }
}
   }
```

Example 39

Project: *jnasmartcardio* File: *WinscardReaderTestWithCardPresent.java* *View source code* 5 votes

```
@Test public void testGetAtr() throws CardException {
        List<CardTerminal> terminalList = terminals.list();
        CardTerminal terminal = terminalList.get(0);
        Card connection = terminal.connect("*");
        ATR atr = connection.getATR();
        byte[] atrBytes = atr.getBytes();
        assertNotSame(0, atrBytes.length);
        boolean hasNonZeroAtr = false;
        for (int i = 0; i < atrBytes.length; i++) {
                hasNonZeroAtr = hasNonZeroAtr || atrBytes[i] != 0;
        }
        assertTrue(hasNonZeroAtr);
}
```

Example 40

Project: *jnasmartcardio* File: *WinscardReaderTestWithCardPresent.java* *View source code* 5 votes

```
@Test public void testTransmit() throws CardException {
        List<CardTerminal> terminalList = terminals.list();
        CardTerminal terminal = terminalList.get(0);
        Card connection = terminal.connect("*");
        CardChannel channel = connection.getBasicChannel();
        assertEquals(0, channel.getChannelNumber());
        ResponseAPDU response = channel.transmit(new CommandAPDU(0x00, 0xa4, 0x00, 0x00, 0));
        int sw = response.getSW();
        assertEquals(String.format("got response 0x%04x instead of 0x9000, %s", sw, response), 0x9000, sw);
}
```