

JPOS Quickstart

Information For User

● Usage

1. Ensure Java Runtime Environment is installed.
2. Edit jpos.xml, change the interface settings:
Locate the line start with `<prop name="port" value=` and change the prop value.

The value is determined by the scanner device's interface:

Interface	Value
RS232	COM
USB COM	COM
USB DataPipe	USB
HID POS	HID or HID-POS

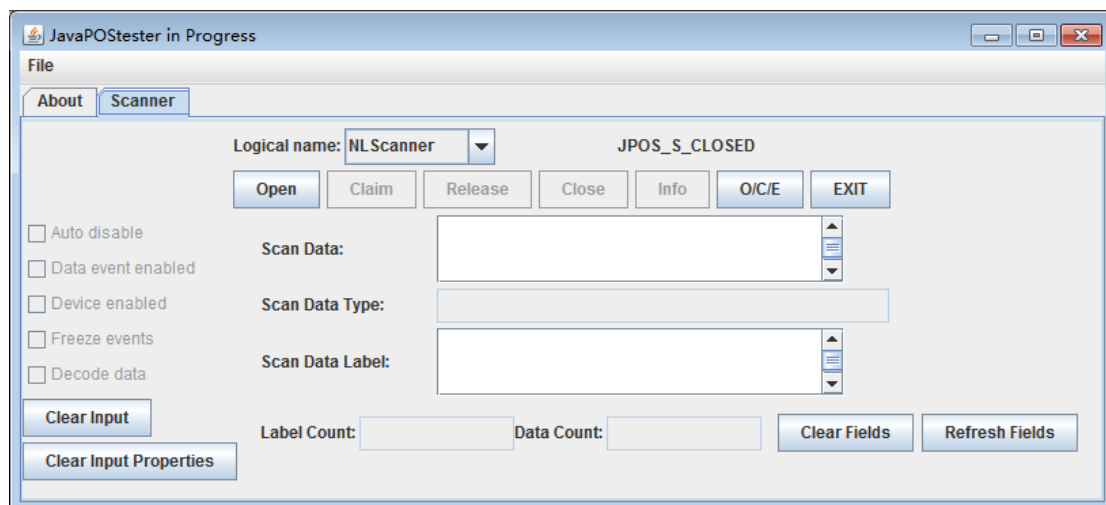
If the prop value is "COM", the prop value of "portno" should also be changed to the correct serial port number.

Example:

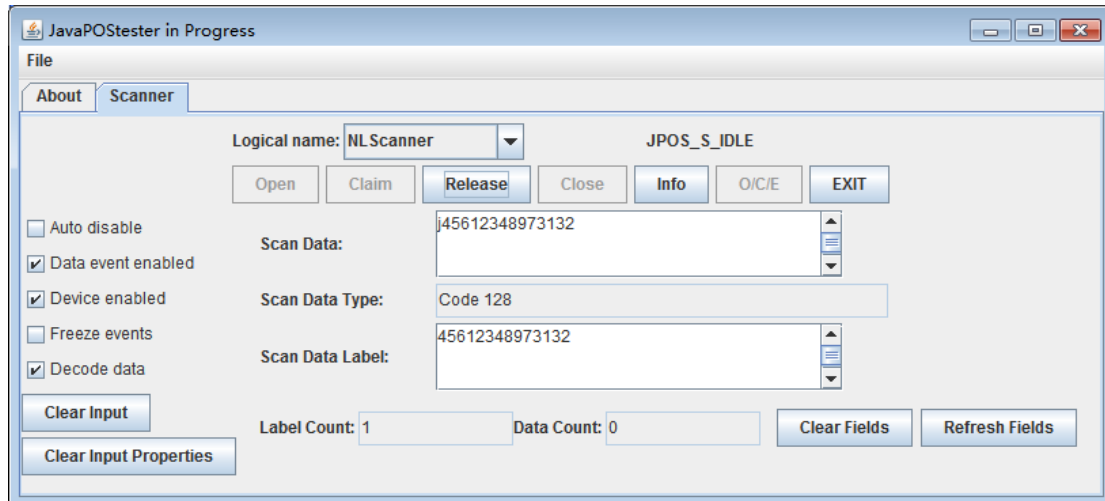
```
<prop name="port" value="COM"/>
```

```
<prop name="portno" value="1"/>
```

3. Double click Postest.bat



4. Click Open, Claim Button, then scan a barcode



5. Click Release, Close, Exit Button to quit

● Enable Logging

Edit log4j.properties

1. Change logging level

log4j.rootLogger=[LogLevel] [,A1] [,A2]

LogLevel can be: OFF, FATAL, ERROR, WARN, INFO, DEBUG, TRACE, ALL

- 1) Enable debug level logging to console and file:
log4j.rootLogger=DEBUG, A1, A2
- 2) Enable info level logging to console only:
log4j.rootLogger=INFO, A1
- 3) Enable warn level logging to file only:
log4j.rootLogger=WARN, A2
- 4) Disable logging:
log4j.rootLogger=OFF, A1, A2

2. Change the log file path

log4j.appender.A2.File=C:\\TEMP\\oposnls.log

Information For Developers

● Java POS XML

The configuration file is jpos.xml.

Prop name	Type	Notes	
port	string	The value is determined by the scanner device’s interface:	
		Interface	Value
		RS232	COM
		USB COM	COM
		USB DataPipe	USB
		HID POS	HID or HID-POS
portno	number	The Serial Port Number. Only valid when the prop value of "port" is "COM".	
ScannerType	number	The value must be 3.	
Suffix	number	The value must be 0.	

Example:

If the interface is RS232, and the serial port number is 1, change the following lines in jpos.xml:

```
<prop name="port" value="COM"/>
<prop name="portno" value="1"/>
```

● JavaPOS Scanner Properties

Name	Notes
autoDisable	
capCompareFirmwareVersion	
capPowerReporting	
capStatisticsReporting	
capUpdateFirmware	
capUpdateStatistics	
checkHealthText	Not implemented
claimed	
dataCount	
dataEventEnabled	
deviceEnabled	

freezeEvents	
outputID	Not required
powerNotify	Not implemented
powerState	Not implemented
state	
deviceControlDescription	Supplied by control object
deviceControlVersion	Supplied by control object
deviceServiceDescription	
deviceServiceVersion	
physicalDeviceDescription	
physicalDeviceName	
decodeData	
scanData	
scanDataLabel	
scanDataType	

● JavaPOS Scanner Methods

Name	Notes
open	
close	
claim	
release	
checkHealth	Not implemented
clearInput	
clearInputProperties	Not implemented
clearOutput	Not implemented
directIO	
compareFirmwareVersion	Not implemented
resetStatistics	Not implemented
retrieveStatistics	Not implemented
updateFirmware	Not implemented
updateStatistics	Not implemented

● Java Example

```
//create a JavaPOS Scanner control object
Scanner scanner = new Scanner;
```

```

try {
    // logicalName is the name of the jpos.xml file Key
    scanner.open(logicalName);

    // wait maximum timeout to get exclusive access to the device.
    int timeout = 100;
    scanner.claim(timeout);

    // enable device
    scanner.setDeviceEnabled(true);

    // enable Data Event
    scanner.setDataEventEnabled(true);

    // handle data event
    // .....

    // disable device
    scanner.setDeviceEnabled(false);

    // remove exclusive access
    scanner.release();

    //
    scanner.close();

} catch (jpos.JposException exception) {

}

```

● Useful Link

Document:

<http://javapos.com/docs.html>

Source Code:

<http://javapos.com/samplecode.html>

<https://sourceforge.net/projects/postest/files/postest/>

<https://sourceforge.net/projects/javapospostest2/files/JavaPOS%20P0Stest%20/>