BIXOLON

API 레퍼런스 가이드

코도바 플러그인

Ver. 1.17

www.bixolon.co.kr

저작권

© BIXOLON Co., Ltd. 모든 권한을 소유합니다.

이 사용설명서와 제품에 사용된 저작물은 저작권법에 의해 보호되어 있습니다.
(주)빅솔론의 사전 서면 동의 없이 사용 설명서 및 제품에 사용된 저작물에 대한 일부 또는 전체를 무단으로 복제, 저장, 전송하는 것을 금합니다.
제공된 정보는 본 제품에만 해당되며 다른 제품에 대해서는 적용되지 않습니다.
또한 본 정보 사용으로 인해 발생하는 직/간접적 손해에 대해 책임지지 않습니다.

- 빅솔론 로고는 (주)빅솔론의 등록상표입니다.
- 모든 다른 상표 또는 제품 이름은 해당하는 회사 또는 조직의 상표입니다.

(주)빅솔론은 제품의 기능과 품질 향상을 위하여 지속적인 개선을 하고 있습니다.
이로 인하여 제품의 사양과 매뉴얼의 내용은 사전 통보 없이 변경될 수 있습니다.

사용 시 주의사항

프린터와 같은 전자 제품은 정전기에 의해 쉽게 훼손될 수 있습니다.
정전기로부터 프린터를 보호하기 위해서는 프린터 후면 부에 케이블을 연결하거나 제거하기 전에 반드시 프린터 전원을 끄십시오. 만약 프린터가 정전기로부터 손상을 입었을 경우에는 가까운 구입처에 문의하십시오.

목차

저작권 2

1. 매뉴얼 안내 7

1-1 지원 정보 7

1-2 지원 장치 8

2. 코도바 플러그인 정보 12

2-1 지원 플랫폼 12

2-2 의존성 플러그인 12

2-3 멀티캐스트 네트워크 권한 요청 12

3. MposControllerConfig 레퍼런스 13

3-1 개요 13

3-2 메소드 13

3-2-1 Select Interface 13
3-2-2 Select Command Mode 14
3-2-3 Open Service 14
3-2-4 Close Service 15
3-2-5 Search Devices 15
3-2-6 Get B-gate Serial Number 16
3-2-7 Get Serial Number 17

4. MPosControllerLookup 레퍼런스 18

4-1 개요 18

4-2 메소드 18

4-2-1 Refresh Deivces List 18
4-2-2 Get Device List 19

5. MPosControllerPrinter 레퍼런스 20

5-1 개요 20

5-2 메소드 20

5-2-1 Select Interface 20
5-2-2 Select Command Mode 21
5-2-3 Open Service 22
5-2-4 Close Service 23
5-2-5 Direct IO 23
5-2-6 Is Open 24

5-2-7 Get Device ID....24
5-2-8 Set Transaction....25
5-2-9 Set Text Encoding....26
5-2-10 Print Text....28
5-2-11 Set Character set....30
5-2-12 Set International Character set....32
5-2-13 Set Page Mode....34
5-2-14 Set Page Mode Print Area....35
5-2-15 Set Page Mode Direction....35
5-2-16 Set Page Mode Position....36
5-2-17 Print Bitmap File....37
5-2-18 Print Bitmap with Base64....38
5-2-19 Print 1D Barcode....39
5-2-20 Print QR Code....40
5-2-21 Print PDF417....41
5-2-22 Print Data Matrix....42
5-2-23 Check Batt Status....43
5-2-24 Check Printer Status....44
5-2-25 Asb Enable....45
5-2-26 Set Status Update Event....46
5-2-27 Open Drawer....46
5-2-28 Print PDF File....47

6 MPosControllerLabelPrinter 레퍼런스....48

6-1 개요....48

6-2 메소드....48

6-2-1 Select Interface....48
6-2-2 Select Command Mode....49
6-2-3 Open Service....50
6-2-4 Close Service....51
6-2-5 Direct IO....52
6-2-6 Is Open....53
6-2-7 Get Device ID....53
6-2-8 Set Transaction....54
6-2-9 Set Text Encoding....55
6-2-10 Set Character set....57
6-2-11 Set Printing Type....59
6-2-12 Set Margin....60
6-2-13 Set Back Feed Option....61
6-2-14 Set Length....62
6-2-15 Set Width....63
6-2-16 Set Buffer Mode....64
6-2-17 Set Speed....65
6-2-18 Set Density....65
6-2-19 Set Orientation....66
6-2-20 Set Offset....66
6-2-21 Set Cutting Position....67

6-2-22 Set Auto Cutter....68
6-2-23 Print Buffer....69
6-2-24 Check Printer Status....70
6-2-25 Draw Text Device Font....71
6-2-26 Draw Text Vector Font....73
6-2-27 Draw Circle....75
6-2-28 Draw Block....76
6-2-29 Draw Image File....77
6-2-30 Draw Image with Base64....78
6-2-31 Draw Barcode 1D....79
6-2-32 Draw Barcode Maxi Code....81
6-2-33 Draw Barcode PDF417....82
6-2-34 Draw Barcode QR Code....84
6-2-35 Draw Barcode Data Matrix....85
6-2-36 Draw Barcode Aztec....86
6-2-37 Draw Barcode Code49....87
6-2-38 Draw Barcode Coda Block....89
6-2-39 Draw Barcode Micro PDF....90
6-2-40 Draw Barcode IMB....91
6-2-41 Draw Barcode MSI....92
6-2-42 Draw Barcode Plessey....93
6-2-43 Draw Barcode TLC39....94
6-2-44 Draw Barcode RSS....95
6-2-45 Setup RFID....97
6-2-46 Set RFID Position....98
6-2-47 Set RFID Password....99
6-2-48 Set EPC Data Structure....100
6-2-49 Write RFID....101
6-2-50 Lock RFID....102
6-5-51 Draw PDF File....103

7. MPosControllerBCD 레퍼런스....104

7-1 개요....104

7-2 메소드....104

7-2-1 Display String....104
7-2-2 Clean Screen....107
7-2-3 Store Image File....108
7-2-4 Display Image....109
7-2-5 Clear Image....110
7-2-6 Show Cursor....111

8. MPosControllerDallasKey 레퍼런스....112

8-1 개요....112

9. MPosControllerHID 레퍼런스....113

9-1 개요.....113

10. MPosControllerMSR 레퍼런스 114

10-1 개요.....114

11. MPosControllerNFC 레퍼런스..... 115

11-1 개요.....115

12. MPosControllerRFID 레퍼런스..... 116

12-1 개요.....116

13. MPosControllerScale 레퍼런스..... 117

13-1 개요.....117

14. MPosControllerScanner 레퍼런스..... 118

14-1 개요.....118

15. MPosControllerUSBSerial 레퍼런스..... 119

15-1 개요.....119

1. 매뉴얼 안내

이 매뉴얼은 프린터 및 주변 기기를 제어할 수 있는 코도바 커스텀 플러그인(Cordova Custom Plugin)를 사용하여 어플리케이션을 개발하는 데 필요한 라이브러리에 관한 설명을 포함하고 있습니다.

1-1 지원 정보

• 개발 환경: Cordova 6.5.0 이상

- 설치된 코도바 버전은 아래 커맨드 라인 인터페이스(Command Line Interface)를 사용하여 확인할 수 있습니다.
  Cordova –version
- 커맨드 라인 인터페이스(Command Line Interface)에 대하여 추가적인 정보를 얻고자 한다면 아래 주소를 방문하십시오.
  https://cordova.apache.org/docs/en/3.1.0/guide/cli/index.html

1-2 지원 장치

영수증 프린터	해상도	지원 통신	최대 인쇄 너비(dot)
SPP-R200III	203 dpi	블루투스, 와이파이, USB	384 dots
SPP-R210	203 dpi		384 dots
SPP-R310	203 dpi		576 dots
SPP-R410	203 dpi		832 dots
SPP-C200	203 dpi	블루투스, USB	384 dots
SPP-C300	203 dpi	블루투스, USB	576 dots
SRP-350III	180 dpi	이더넷, USB	512 dots
SRP-352III	203 dpi		576 dots
SRP-330II	180 dpi		512 dots
SRP-332II	203 dpi		576 dots
SRP-350plusIII	180 dpi	블루투스, 와이파이, 이더넷, USB	512 dots
SRP-352plusIII	203 dpi		576 dots
SRP-380	180 dpi		512 dots
SRP-380plus	180 dpi		512 dots
SRP-382	203 dpi		576 dots
SRP-382plus	203 dpi		576 dots
SRP-383	300 dpi		864 dots
SRP-Q200	203 dpi		432 dots
SRP-Q300	180 dpi		512 dots
SRP-Q302	203 dpi		576 dots
SRP-Q300II	203 dpi		576 dots
SRP-F310II	180 dpi		512 dots
SRP-F312II	203 dpi		576 dots
SRP-F313II	203 dpi		640 dots
SRP-S300	203 dpi		576 dots
SRP-S300II	203 dpi		576 dots
SRP-S320	203 dpi		576 dots
SRP-S3000	203 dpi		576 dots
BGT-100P	180 dpi		512 dots
BGT-102P	203 dpi		576 dots
SRP-Q300H	180 dpi		512 dots
SRP-Q302H	203 dpi		576 dots
SRP-350plusV	180 dpi		512 dots
SRP-352plusV	203 dpi		576 dots
SRP-S200	203 dpi	이더넷, USB	432 dots
SRP-QE300	180 dpi		512 dots
SRP-QE302	203 dpi		576 dots
SRP-E300	180 dpi		512 dots

SRP-E302	203 dpi		576 dots
SRP-350V	180 dpi		512 dots
SRP-352V	203 dpi		576 dots
SRP-330III	180 dpi		512 dots
SRP-332III	203 dpi		576 dots
BGT-100P	180 dpi	블루투스, 와이파이, 이더넷, USB	512 dots
BGT-102P	203 dpi		576 dots
SRP-Q300H	180 dpi		512 dots
SRP-Q302H	203 dpi		576 dots
SRP-F310IIHi	180 dpi		512 dots
SRP-F312IIHi	203 dpi		576 dots
SRP-S300Hi	180 dpi		512 dots
SRP-S320Hi	203 dpi		576 dots

라벨 프린터	해상도	지원 통신	최대 인쇄 너비(dot)
SLP-DX220	203 dpi	블루투스, 와이파이, 이더넷, USB	432 dots
SLP-DX223	300 dpi		672 dots
SLP-TX220	203 dpi		432 dots
SLP-TX223	300 dpi		672 dots
SLP-DX420	203 dpi		864 dots
SLP-DX423	300 dpi		1248 dots
SLP-TX420	203 dpi		864 dots
SLP-TX423	300 dpi		1248 dots
SLP-TX400(RFID)	203 dpi		864 dots
SLP-TX403(RFID)	300 dpi		1248 dots
SLP-DL410	203 dpi		864 dots
SLP-DL413	300 dpi		1248 dots
SPP-L3000	203 dpi	블루투스, 와이파이, USB	576 dots
SPP-L310	203 dpi		576 dots
SPP-L410	203 dpi		832 dots
XM7-20	203 dpi		384 dots
XM7-30	203 dpi		576 dots
XM7-40(RFID)	203 dpi		832 dots
SRP-770III	203 dpi	블루투스, 와이파이, 이더넷, USB	832 dots
SRP-E770III	203 dpi	이더넷, USB	832 dots
XT5-40(RFID)	203 dpi	블루투스, 와이파이, 이더넷, USB	832 dots
XT5-43(RFID)	300 dpi		1248 dots
XT5-46(RFID)	600 dpi		2496 dots
XD5-40d	203 dpi		864 dots
XD5-43d	300 dpi		1248 dots
XD5-40t(RFID)	203 dpi		864 dots
XD5-43t(RFID)	300 dpi		1248 dots
XL5-40	203 dpi		864 dots
XL5-43	300 dpi		1248 dots
SRP-S3000_LABEL	203 dpi		576 dots
XT3-40	203 dpi		864 dots
XT3-43	300 dpi		1248 dots
XQ-840II	203 dpi	USB	864 dots
XQ-843II	300 dpi	USB	1248 dots
XD5-40IId	203 dpi	블루투스, 와이파이, 이더넷, USB	864 dots
XD5-43IId	300 dpi		1248 dots
XD5-40IIt	203 dpi		864 dots
XD5-43IIt	300 dpi		1248 dots
XT6-60	203 dpi	Bluetooth, LAN, WLAN, USB	1344 dots

XT6-63	300 dpi	Bluetooth, LAN, WLAN, USB	1984 dots
XM5-30	203 dpi	Bluetooth, WLAN, USB	576 dots

2. 코도바 플러그인 정보

2-1 지원 플랫폼

플랫폼	정보
Android	Android 4.0.3(API Level 15: ice cream sandwich) 이상
iOS	iOS 12.0 이상
UWP	Windows 10 version Build 14393 이상

2-2 의존성 플러그인

플러그인	정보
cordova-promise-polyfill	이 의존성 플러그인(dependency plugin)은 코도바에서 ES6 Promise 객체를 사용하기 위한 폴리필(polyfill) 입니다. 제공되는 샘플 프로그램에서 이 플러그인을 사용하여 작성된 코드가 있습니다. ES6 Promise 객체를 사용할 수 없는 환경이라면 이 플러그인을 설치하십시오.

2-3 멀티캐스트 네트워크 권한 요청

iOS에서 빅솔론 Cordova Plugin는 브로드캐스트 IP 패킷을 전송하여 로컬 네트워크에서 프린터를 검색하는 API가 있습니다. iOS 14.5 이상부터 로컬 네트워크에 대한 정책이 변경되어 Apple의 허가 없이 API가 작동하지 않습니다. 아래 링크를 참조하세요.

1) https://developer.apple.com/news/?id=0oi77447
2) https://developer.apple.com/contact/request/networking-multicast

- 장치 검색용 통신 IP 주소: 255.255.255.255
- 장치 검색용 통신 포트번호: 48780, 48781, 9000, 3337

3. MposControllerConfig 레퍼런스

3-1 개요

• “MPosControllerConfig” 객체는 B-gate 장치에 추가적인 주변 장치를 등록 또는 제거하거나, 등록된 주변 장치의 통신 설정에 필요한 메소드를 제공합니다.

3-2 메소드

3-2-1 Select Interface

연결할 장치의 통신 인터페이스 유형과 주소를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
interfaceType	Number	통신 인터페이스 **1:** 와이파이 (Wi-Fi) **2:** 이더넷 (Ethernet) **4:** 블루투스 (Bluetooth) **16:** USB
address	String	IP 주소: 포트번호, MAC 주소, Bluetooth 시리얼 넘버 예) 192.168.1.2:9100 예) 74:F0:7D:E3:DF:EE

USB 통신은 Android에서만 사용 가능합니다.

3-2-2 Select Command Mode

장치와 직접 통신할지 “B-gate”를 통하여 통신할지 선택합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
commandMode	Number	0 (고정 값)

3-2-3 Open Service

장치와 통신 연결을 시작합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
id	Number	0 (고정 값)
timeout	Number	연결 시도에 대한 타임아웃 (초)

[예제]

```
return new Promise(function (returnResolve, returnReject) {
    new Promise(function (resolve, reject) { _Config.selectInterface(resolve, reject, _ConnectionInfo.interfaceType,0);})
     .then(function () { return new Promise(function (resolve, reject) { _Config.selectCommandMode(resolve, reject,
0);});});})
     .then(function () { return new Promise(function (resolve, reject) {
          _Config.openService(resolve, function(error){
                    returnReject();
               }, $('#deviceID').val(), _ConnectionInfo.timeout);
          });
     })
     .then(Function{
          returnResolve();
     });
});
```

3-2-4 Close Service

장치와 통신 연결을 종료합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
timeout	Number	연결 종료에 대한 타임아웃 (초)

3-2-5 Search Devices

B-gate에 연결되어 있는 장치 식별자(id) 목록 값을 반환합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
interfaceType	Number	장치 목록의 통신 유형 **1:** 와이파이 (Wi-Fi) **2:** 이더넷 (Ethernet) **4:** 블루투스 (Bluetooth)

[예제]

```
_Config.searchDevices(function(list){
    for(var i = 0; i < list.length; i++) {
        ...
    }
}, function(error){

});
```

3-2-6 Get B-gate Serial Number

B-gate 제품 시리얼 번호를 반환합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	제품 시리얼 넘버 값을 수신할 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수

[예제]

```
_Config.getSerialNumber(
  function{ console.log("getSerialNumber success"); },
  function{ console.log("getBgateSerialNumber failed");
  10/*POS Printer*/ }
)
```

3-2-7 Get Serial Number

프린터에 연결된 USB 장치의 시리얼번호를 확인합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	제품 시리얼 넘버 값을 수신할 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
deviceID	Number	연결할 장치의 식별자(ID) 값

[예제]

```
_Config.getBgateSerialNumber(
  function(bGateSN){ console.log("getBgateSerialNumber success"); },
  function{ console.log("getBgateSerialNumber failed"); }
)
```

4. MPosControllerLookup 레퍼런스

4-1 개요

• “MPosControllerLookup” 객체는 연결 가능한 장치 목록 정보와 관련된 메소드를 제공합니다.

4-2 메소드

4-2-1 Refresh Deivces List

연결 가능한 네트워크(Wi-Fi, Ethernet) 및 블루투스 장치 목록을 갱신합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
interfaceType	Number	갱신할 장치의 통신 유형 **1:** 네트워크 (Wi-Fi) **2:** 네트워크 (Ethernet) **4:** 블루투스
timeout	Number	장치 목록 갱신에 대한 타임아웃 (초)

[예제]

```
_Lookup.refreshDeivcesList(
  function{ console.log("refreshDeivcesList success"); },
  function{ console.log("refreshDeivcesList failed"); },
  4 /*Bluetooth*/, 2 /*Second*/
)
```

4-2-2 Get Device List

연결 가능한 네트워크(Wi-Fi, Ethernet) 및 블루투스 장치 목록을 JSON(JavaScript Object Notation) 형식으로 반환합니다.



이 메소드를 호출하기 전에 'refreshDevicesList' 메소드가 먼저 호출되어야 합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	연결 가능한 장치 목록 데이터를 받을 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
interfaceType	Number	장치 목록의 통신 유형 **1:** 네트워크 (Wi-Fi) **2:** 네트워크 (Ethernet) **4:** 블루투스

[예제]

```
_Lookup.getDeviceList(
 function(list){
  console.log("getDeviceList success");
  var jsonObjArray = JSON.parse(list);
  for(var i = 0; i < jsonObjArray.length; i++){
   // jsonObjArray[i].interfaceType
   // jsonObjArray[i].name
   // jsonObjArray[i].address
  }
 },
 function{ console.log("getDeviceList failed"); },
 4 /*Bluetooth*/,
)
```

5. MPosControllerPrinter 레퍼런스

5-1 개요

• “MPosControllerPrinter” 객체는 영수증 프린터 제어를 위한 메소드를 제공합니다.

5-2 메소드

5-2-1 Select Interface

연결할 장치(프린터)의 통신 인터페이스 유형과 주소를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
interfaceType	Number	통신 인터페이스 **1:** 와이파이 (Wi-Fi) **2:** 이더넷 (Ethernet) **4:** 블루투스 (Bluetooth) **16:** USB
address	String	IP 주소: 포트번호, MAC 주소, Bluetooth 시리얼 넘버 예) 192.168.1.2:9100 예) 74:F0:7D:E3:DF:EE

USB 통신은 Android에서만 사용 가능합니다.

5-2-2 Select Command Mode

장치와 직접 통신할지 “B-gate”를 통하여 통신할지 선택합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
commandMode	Number	연결할 장치와 통신 방식 **0:** B-gate를 통하여 장치와 통신 **1:** 장치와 직접 통신

5-2-3 Open Service

장치와 통신 연결을 시작합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
id	Number	연결할 장치의 식별자(ID) 값
timeout	Number	연결 시도에 대한 타임아웃 (초)



- 이 매뉴얼의 3-2-5에 기술된 “searchDevices”를 사용하면, 연결 가능한 장치 식별자(ID) 목록을 얻을 수 있습니다.
- B-gate를 이용한 통신이 아닌 경우, 장치 식별자(id)값은 -1로 설정하십시오.

  * 영수증 프린터의 장치 식별자(ID) 값 범위: 10 ~ 19

[예제]

```
return new Promise(function (returnResolve, returnReject) {
    new Promise(function (resolve, reject) { _Printer.selectInterface(resolve, reject, _ConnectionInfo.interfaceType,0);})
     .then(function () { return new Promise(function (resolve, reject) { _Printer.selectCommandMode(resolve, reject,
0);});})
     .then(function () { return new Promise(function (resolve, reject) {
          _Printer.openService(resolve, function(error){
                    returnReject();
               }, $('#deviceID').val(), _ConnectionInfo.timeout);
          });
     })
     .then(Function{
          returnResolve();
     });
});
```

5-2-4 Close Service

장치와 통신 연결을 종료합니다.



데이터 전송 중에 이 메소드를 호출하면 데이터 전송이 완료되지 않고 통신 연결이 종료될 수 있습니다. 인쇄 속도와 인쇄 길이에 따른 적절한 타임아웃 값을 설정하여 이 메소드를 호출하십시오.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
timeout	Number	연결 종료에 대한 타임아웃 (초)

5-2-5 Direct IO

장치에 사용자 정의 데이터를 전송합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	Object	사용자 정의 바이트 배열 (Byte Array)

[예제]

```
_Printer.directIO(function(result){
   ...
},function(result){
   ...
}, [0x1D, 0x56, 0x01] /* Cut */);
```

5-2-6 Is Open

장치의 사용 가능 상태를 반환합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 장치 개방 여부를 수신할 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수

[예제]

```
_Printer.isOpen(
 function(open){
  console.log("isOpen success");
  if(open) { ... }
  else { ... }
 },
 function{ console.log("isOpen failed"); }
)
```

5-2-7 Get Device ID

장치 식별자(ID) 값을 반환합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 장치 ID 값을 수신할 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수

[예제]

```
_Printer.getDeviceID(
 function(deviceId){ console.log("getDeviceID success"); },
 function { console.log("getDeviceID failed"); }
)
```

5-2-8 Set Transaction

트랜잭션(Transaction) 모드로 진입 또는 종료하여 버퍼에 있는 데이터를 장치로 전송합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
TransactionMode	Number	트랜잭션 모드 **0:** 트랜잭션 모드 종료 및 데이터 전송 **1:** 트랜잭션 모드 진입

[예제]

```
return new Promise(function (returnResolve, returnReject) {
    new Promise(function (resolve, reject) { _Printer.setTransaction(resolve, reject, 1);})
     /* add more operations as required */
     .then(Function { return new Promise(function (resolve, reject) { _Printer.setTransaction(resolve, reject, 0);});})
     .then(Function{
          returnResolve();
     });
});
```

5-2-9 Set Text Encoding

문자 인코딩에 사용할 코드 페이지(code page)를 설정합니다.



프린터는 미리 설정된 코드 페이지(문자 인코딩 테이블)를 참조하여 문자를 인쇄를 합니다. 설정된 코드 페이지 정보를 변경하지 않으면 정상적으로 문자가 출력되지 않을 수 있으므로 문자에 따라 적절한 코드 페이지를 설정하십시오.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
textEncoding	Number	문자열 인코딩에 사용할 코드 페이지 **437**: 영어 (PC437) **850**: 서유럽 (PC850) **860**: 포르투갈어 (PC860) **863**: 캐나다-프랑스어 (PC863) **865**: 노르웨이어 (PC865) **1252**: 라틴어1 (WPC1252) **866**: 키릴어 (PC866) **852**: 라틴어2 (PC852) **858**: 서유럽 (PC858) **862**: 히브리어 (PC862) **34**: 태국어 (THAI 11) **31**: 태국어 (THAI 14) **39**: 태국어 (THAI 16) **35**: 태국어 (THAI 18) **23**: 태국어 (THAI 42) **1253**: 그리스어 (WPC1253) **1254**: 터키어 (WPC1254) **1257**: 발틱어 (WPC1257)

		1251: 키릴어 (WPC1251) **27**: 페르시아어 (FARSI) **28**: 페르시아어 (FARSI) **737**: 그리스어 (PC737) **775**: 발틱어 (PC775) **1255**: 히브리어 (WPC1255) **855**: 키릴어 (PC855) **857**: 터키어 (PC857) **928**: 그리스어 (PC928) **1250**: 체코어 (WPC1250) **28605**: 라틴9 **1258**: 베트남어 (WPC1258) **49**: 베트남어 (TCVN3) **50**: 베트남어 (TCVN3 Capital) **51**: 베트남어 (VISCII) **864**: 아랍어 (PC864) **1256**: 아랍어 (WPC1256) **949**: 한국어 (KS5601) **932**: 일본어 (ShiftJIS) **950**: 중국어 번체 (BIG5) **936**: 중국어 간체 (GB2312)

[예제]

```
_Printer.setTextEncoding(
  function{ console.log("setTextEncoding success"); },
  function{ console.log("setTextEncoding failed"); },
  437
)
```

5-2-10 Print Text

문자의 다양한 글꼴 속성을 지정하여 문자를 인쇄합니다.



프린터에 설정된 기본 코드 페이지(code page) 값에 따라 문자가 '?' 또는 인쇄되지 않을 수 있습니다. 인쇄하려는 문자가 정상적으로 인쇄되지 않는다면 이 매뉴얼에 있는 'setTextEncoding' 또는 'setCharacterset' 메소드를 이용하여 코드페이지 값을 변경하십시오.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	인쇄될 문자열 데이터
textPrintProperty	Object	글꼴 속성을 위한 객체

<TextPrintProperty 객체>

```
var TextPrintProperty = function(codePage, fontType, width, height, underline,
alignment, bold, color, reverse){
    this.codePage = !!codePage ? codePage : 1252 /* WPC1252 */;
    this.fontType = !!fontType ? fontType : 0 /* Font A */;
    this.width = !!width ? width : 0;
    this.height = !!height ? height : 0;
    this.underline = !!underline ? underline : 0 /* No underline */;
    this.alignment = !!alignment ? alignment : 0 /* Left */;
    this.bold = !!bold ? bold : false;
    this.reverse = !!reverse ? reverse : false;
};
```

문자 인쇄 속성

- **codePage**: 문자 인코딩에 사용할 코드 페이지.
- **fontType**: 글꼴 타입 (Font A, Font B, Font C)
- **width**: 글꼴 너비 (0, 16, 32, 48, 64, 80, 96, 112)
- **height**: 글꼴 높이 (0 ~7)
- **underline**: 글꼴의 밑줄 속성 (0: 밑줄 없음, 1: 1dot 밑줄, 2: 2dot 밑줄)
- **alignment**: 정렬 (0: 좌측, 1: 중간, 2: 우측)
- **bold**: 굵게 인쇄
- **reverse**: 글꼴의 반전 인쇄

[예제]

```
_Printer.printText(
  function{ console.log("printText success"); },
  function{ console.log("printText failed"); },
  'Hello Cordova' + '\n', new TextPrintProperty(437, 1, 0, 0, 0,false, false, false)
)
```

5-2-11 Set Character set

문자 인코딩에 사용할 코드 페이지(code page)를 설정합니다.



프린터는 미리 설정된 코드 페이지(문자 인코딩 테이블)를 참조하여 문자를 인쇄를 합니다. 설정된 코드 페이지 정보를 변경하지 않으면 정상적으로 문자가 출력되지 않을 수 있으므로 문자에 따라 적절한 코드 페이지를 설정하십시오.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
codePage	Number	문자열 인코딩에 사용할 코드 페이지 **437**: 영어 (PC437) **850**: 서유럽 (PC850) **860**: 포르투갈어 (PC860) **863**: 캐나다-프랑스어 (PC863) **865**: 노르웨이어 (PC865) **1252**: 라틴어1 (WPC1252) **866**: 키릴어 (PC866) **852**: 라틴어2 (PC852) **858**: 서유럽 (PC858) **862**: 히브리어 (PC862) **34**: 태국어 (THAI 11) **31**: 태국어 (THAI 14) **39**: 태국어 (THAI 16) **35**: 태국어 (THAI 18) **23**: 태국어 (THAI 42) **1253**: 그리스어 (WPC1253) **1254**: 터키어 (WPC1254) **1257**: 발틱어 (WPC1257)

		1251: 키릴어 (WPC1251) **27**: 페르시아어 (FARSI) **28**: 페르시아어 (FARSI) **737**: 그리스어 (PC737) **775**: 발틱어 (PC775) **1255**: 히브리어 (WPC1255) **855**: 키릴어 (PC855) **857**: 터키어 (PC857) **928**: 그리스어 (PC928) **1250**: 체코어 (WPC1250) **28605**: 라틴9 **1258**: 베트남어 (WPC1258) **49**: 베트남어 (TCVN3) **50**: 베트남어 (TCVN3 Capital) **51**: 베트남어 (VISCII) **864**: 아랍어 (PC864) **1256**: 아랍어 (WPC1256) **949**: 한국어 (KS5601) **932**: 일본어 (ShiftJIS) **950**: 중국어 번체 (BIG5) **936**: 중국어 간체 (GB2312)

[예제]

```
_Printer.setCharacterset(
  function{ console.log("setCharacterset success"); },
  function{ console.log("setCharacterset failed"); },
  437 /*PC437*/
)
```

5-2-12 Set International Character set

국제 문자 인코딩에 사용할 문자 코드 테이블(character code table)를 설정합니다.

아래의 16진수 값으로 표현되는 14개의 문자는 'setCharacterset' 또는 'setTextEncoding' 메소드로 설정되는 코드 페이지와는 별개로 처리됩니다.

No	Country	ASCII code (Hexadecimal number)													
		23	24	25	26	40	5B	5C	5D	5E	60	7B	7C	7D	7E
0	U.S.A	#	$	%	&	@	[	\	]	^	`	{	|	}	~
1	France	#	$	%	&	à	°	ç	§	^	`	é	ù	è	¨
2	Germany	#	$	%	&	§	Ä	Ö	Ü	^	`	ä	ö	ü	β
3	U.K.	£	$	%	&	@	[	\	]	^	`	{	|	}	~
4	Denmark I	#	$	%	&	@	Æ	Ø	Å	^	`	æ	ø	å	~
5	Sweden	#	¤	%	&	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
6	Italy	#	$	%	&	@	°	\	é	^	ù	à	ò	è	ì
7	Spain I	Pt	$	%	&	@	¡	Ñ	¿	^	`	¨	ñ	}	~
8	Japan	#	$	%	&	@	[	¥	]	^	`	{	|	}	~
9	Norway	#	¤	%	&	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
10	Denmark II	#	$	%	&	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
11	Spain II	#	$	%	&	á	¡	Ñ	¿	é	`	í	ñ	ó	ú
12	Latin America	#	$	%	&	á	¡	Ñ	¿	é	ü	í	ñ	ó	ú
13	Korea	#	$	%	&	@	[	\	]	^		{	|	}	~

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
characterset	Number	국제 문자용 문자 집합(Character Set) **0:** 미국(USA) 코드 **1:** 프랑스(FRANCE) 코드 **2:** 독일(GERMANY) 코드 **3:** 영국(UK) 코드 **4:** 덴마크1(DENMARK1) 코드 **5:** 스웨덴(SWEDEN) 코드 **6:** 이탈리아(ITALY) 코드 **7:** 스페인(SPAIN) 코드 **8:** 일본(JAPAN) 코드

		9: 노르웨이(NORWAY) 코드 **10:** 덴마크2(DENMARK 2) 코드 **11:** 스페인2(SPAIN 2) 코드 **12:** 라틴 아메리카(LATIN AMERICA) 코드 **13:** 한국(KOREA) 코드 **14:** 슬로베니아(SLOVENIA) 코드 **15:** 중국(CHINA) 코드

[예제]

```
_Printer.setInternationalCharacterset(
  function{ console.log("setInternationalCharacterset success"); },
  function{ console.log("setInternationalCharacterset failed"); },
  0 /* USA */
)
```

5-2-13 Set Page Mode

페이지 모드 진입 또는 종료하여 버퍼에 있는 데이터를 장치로 전송합니다.



페이지 모드는 트랜잭션(Transaction) 모드와 같이 사용할 수 없습니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
pageMode	Number	페이지 모드 **0:** 페이지 모드 진입 **1:** 페이지 모드 종료 및 인쇄

[예제]

```
return new Promise(function (returnResolve, returnReject) {
    new Promise(function (resolve, reject) { _Printer. setPagemode(resolve, reject, 1);})
     .then(Function { return new Promise(function (resolve, reject) {
         _Printer.setPagemodePrintArea(resolve,reject,10, 10, 384, 2000);
     });})
     .then(Function { return new Promise(function (resolve, reject) {
         _Printer. setPagemodeDirection(resolve,reject, 0);
     });})
     .then(Function { return new Promise(function (resolve, reject) {
         _Printer. setPagemodePosition (resolve,reject, 100,100);
     });})
     .then(Function { return new Promise(function (resolve, reject) { _Printer. setPagemode(resolve, reject, 0);});})
     .then(Function{
          returnResolve();
     });
});
```

5-2-14 Set Page Mode Print Area

페이지 모드에서 인쇄 영역을 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
x	Number	인쇄 영역의 시작 X 좌표(dot)
y	Number	인쇄 영역의 시작 Y 좌표(dot)
width	Number	인쇄 너비(dot)
height	Number	인쇄 높이(dot)

5-2-15 Set Page Mode Direction

페이지 모드에서 인쇄 방향을 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
direction	Number	인쇄 방향 **0:** 좌측에서 우측 (시작 위치: 좌측 상단) **1:** 아래에서 위로 (시작 위치: 하단 좌측) **2:** 우측에서 좌측 (시작 위치: 하단 우측) **3:** 위에서 아래로 (시작 위치: 우측 상단)

5-2-16 Set Page Mode Position

페이지 모드에서 인쇄 좌표를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
x	Number	인쇄될 객체(문자, 바코드, 이미지 등)의 X좌표 (dot)
y	Number	인쇄될 객체(문자, 바코드, 이미지 등)의 Y좌표 (dot)

5-2-17 Print Bitmap File

HOST 기기에 존재하는 비트맵 파일을 인쇄합니다.

- Android 경우, SD카드에 있는 파일만 인쇄할 수 있습니다.
- 플랫폼(Android, iOS, UWP)의 버전에 따라 파일 접근 권한이 없을 경우 인쇄가 되지 않을 수 있습니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
path	String	이미지 파일 경로
width	Number	인쇄될 이미지 너비 또는 원본 크기로 인쇄 **-2**: 원본 크기로 인쇄
alignment	Number	정렬 **0:** 좌측 정렬 **1:** 중앙 정렬 **2:** 우측 정렬
brightness	Number	이미지 밝기 범위: 0 ≤ brightness ≤ 100 (Android, UWP) 범위: -100 ≤ brightness ≤ 100 (iOS)
isDithering	Boolean	디더링 알고리즘 적용 여부
isCompress	Boolean	데이터 압축 기능 적용 여부

[예제]

```
_Printer.printBitmapFile(
  function{ console.log("printBitmapFile success"); },
  function{ console.log("printBitmapFile failed"); },
  path, -2, 0, 50, true, true
)
```

5-2-18 Print Bitmap with Base64

Base64로 인코딩 된 그래픽 이미지를 인쇄합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
imageSource	String	Base64로 인코딩 된 이미지 데이터
width	Number	인쇄될 이미지 너비 또는 원본 크기로 인쇄 **-2:** 원본 크기로 인쇄
alignment	Number	정렬 **0:** 좌측 정렬 **1:** 중앙 정렬 **2:** 우측 정렬
brightness	Number	이미지 밝기 범위: 0 ≤ brightness ≤ 100 (Android, UWP) 범위: -100 ≤ brightness ≤ 100 (iOS)
isDithering	Boolean	디더링 알고리즘 적용 여부
isCompress	Boolean	데이터 압축 기능 적용 여부

[예제]

```
_Printer.printBitmapWithBase64(
  function{ console.log("printBitmapWithBase64 success"); },
  function{ console.log("printBitmapWithBase64 failed"); },
  new ImgInBase64Format().getImageData(0),-2, 0, 50, true, true
)
```

5-2-19 Print 1D Barcode

1차원 바코드를 인쇄합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
barcodeType	Number	바코드 유형 **101:** UPC-A **102:** UPC-E **103:** JAN-8 (EAN) **104:** JAN-13 (EAN) **106:** ITF **107:** CODABAR **108:** CODE39 **109:** CODE93 **110:** CODE128
barWidth	Number	바코드 너비 (범위: 2 ≤ barWidth ≤ 6)
height	Number	바코드 높이 (범위: 1 ≤ height ≤ 255)
alignment	Number	정렬 **0:** 좌측 정렬 **1:** 중앙 정렬 **2:** 우측 정렬
textPosition	Number	HRI (Human Readable Interface) 인쇄 위치 **0:** 인쇄 안함 **1:** 바코드 상단 **2:** 바코드 하단

[예제]

```
_Printer.print1DBarcode(
  function{ console.log("print1DBarcode success"); },
  function{ console.log("print1DBarcode failed"); },
  ‘1234567890’, 110 /*Code128*/, 3, 50, 0, 1
)
```

5-2-20 Print QR Code

QR 코드를 인쇄합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	QR 코드 데이터
model	Number	QR 코드 모델 **204:** 모델1 **205:** 모델2
alignment	Number	정렬 **0:** 좌측 정렬 **1:** 중앙 정렬 **2:** 우측 정렬
moduleSize	Number	QR 코드 크기 (범위: 1 ≤ moduleSize ≤ 8)
eccLevel	Number	에러 보정 레벨 **48:** 레벨 L **49:** 레벨 M **50:** 레벨 Q **51:** 레벨 H

[예제]

```
_Printer.printQRCode(
  function{ console.log("printQRCode success"); },
  function{ console.log("printQRCode failed"); },
  'QRCode', 204, 0, 5, 49
)
```

5-2-21 Print PDF417

PDF417 코드를 인쇄합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	PDF417 데이터
model	Number	PDF417 코드 유형 **201:** 표준 PDF417 **202:** 간소화된 PDF417
alignment	Number	정렬 **0:** 좌측 정렬 **1:** 중앙 정렬 **2:** 우측 정렬
columnNumber	Number	행(Columns) 개수 (범위: 0 ≤ columnNumber ≤ 30)
rowNumber	Number	열(Row) 개수 (범위: 3 ≤ rowNumber ≤ 90)
moduleWidth	Number	모듈 너비 (범위: 1 ≤ moduleWidth ≤ 4)
moduleHeight	Number	모듈 높이 (범위: 2 ≤ moduleHeight ≤ 8)
eccLevel	Number	에러 보정 레벨 **48:** 에러 보정 레벨 0 **49:** 에러 보정 레벨 1 **50:** 에러 보정 레벨 2 **51:** 에러 보정 레벨 3 **52:** 에러 보정 레벨 4 **53:** 에러 보정 레벨 5 **54:** 에러 보정 레벨 6 **55:** 에러 보정 레벨 7

[예제]

```
_Printer.printPDF417(
  function{ console.log("printPDF417 success"); },
  function{ console.log("printPDF417 failed"); },
  'Hello Cordova', 201 /* standard PDF417 */, 0, 0, 0, 3, 4, 48
)
```

5-2-22 Print Data Matrix

Data Matrix 코드를 인쇄합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	Data Matrix 바코드 데이터
alignment	Number	정렬 **0:** 좌측 정렬 **1:** 중앙 정렬 **2:** 우측 정렬
moduleSize	Number	모듈 크기(범위: 2 ≤ moduleSize ≤ 3)

[예제]

```
_Printer.printDataMatrix(
  function{ console.log("printDataMatrix success"); },
  function{ console.log("printDataMatrix failed"); },
  'DataMatrix', 0, 3
)
```

5-2-23 Check Batt Status

프린터의 배터리 상태를 체크합니다.



배터리가 장착된 모델에 한하여 동작합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	배터리 상태 정보를 수신할 콜백 함수 **0:** 배터리 완충 **16:** 배터리 잔량 높음 **32:** 배터리 잔량 중간 **64:** 배터리 잔량 낮음
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수

[예제]

```
_Printer.checkBattStatus(
 function(level){
    ...
 },
 Function{
    console.log("checkBattStatus failed");
 }
)
```

5-2-24 Check Printer Status

프린터의 현재 상태를 확인합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	프린터 상태 값을 수신할 콜백 함수 **0:** 정상 **1:** 프린터 커버 오픈 **2:** 용지 없음 **4:** 종이 부족 **8:** 에러 (오프라인) **128:** 현금 수납함(Cash Drawer)의 신호가 LOW **256:** 현금 수납함(Cash Drawer)의 신호가 HIGH
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수

[예제]

```
_Printer.checkPrinterStatus(
 function(status){ console.log("checkPrinterStatus success");
     // if (status & POS_CONST.PrinterStatus.COVEROPEN)
     // if (status & POS_CONST.PrinterStatus.PAPEREMPTY)
     // if (status & POS_CONST.PrinterStatus.PAPER_NEAREND)
     // if (status & POS_CONST.PrinterStatus.ERROR)
     // if (status & POS_CONST.PrinterStatus.DRAWER_LOW)
     // if (status & POS_CONST.PrinterStatus.DRAWER_HIGH)
     // if (status & POS_CONST.PrinterStatus.BATT_HIGH)
     // if (status & POS_CONST.PrinterStatus.BATT_MID)
     // if (status & POS_CONST.PrinterStatus.BATT_LOW)
 },
 function{ console.log("checkPrinterStatus failed"); }
)
```

5-2-25 Asb Enable

ASB (Automatic Status Back) 기능을 활성화 또는 비 활성화합니다.

활성화되면 프린터의 상태가 변경될 때 마다 지정된 콜백 함수로 프린터 상태 값이 전달됩니다.



프린터 상태 정보를 수신할 콜백 함수 설정은

'**setStatusUpdateEvent**' 메소드를 사용하십시오.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
enable	Boolean	프린터 현재 상태 통지 활성화 **true:** 프린터 상태 통지 시작 **false:** 프린터 상태 통지 중단

[예제]

```
var PrinterStatusEventHandler = function(status) {
    // if (status & POS_CONST.PrinterStatus.COVEROPEN)
    // if (status & POS_CONST.PrinterStatus.PAPEREMPTY)
    // if (status & POS_CONST.PrinterStatus.PAPER_NEAREND)
    // if (status & POS_CONST.PrinterStatus.ERROR)
    // if (status & POS_CONST.PrinterStatus.DRAWER_LOW)
    // if (status & POS_CONST.PrinterStatus.DRAWER_HIGH)
    // if (status & POS_CONST.PrinterStatus.BATT_HIGH)
    // if (status & POS_CONST.PrinterStatus.BATT_MID)
    // if (status & POS_CONST.PrinterStatus.BATT_LOW)
};

_Printer.setStatusUpdateEvent(Function{
    _Printer.asbEnable(function () {
      console.log("asbEnable success");
    },function(error){
      console.log("asbEnable failed");
    }, true /*Enable*/);
},function(error){
  console.log("setStatusUpdateEvent failed");
}, PrinterStatusEventHandler);
```

5-2-26 Set Status Update Event

프린터 상태 정보를 수신할 콜백 함수를 설정합니다.

대상 플랫폼이 UWP 경우, 프린터 상태 정보를 수신할 콜백 함수를 세 번째 매개변수 'handler'에 값으로 전달해야 합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	프린터 상태 값을 수신할 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
handler	Function	프린터 상태 값을 수신할 콜백 함수 *** Android, iOS 경우, 이 매개변수를 무시하십시오.**

5-2-27 Open Drawer

현금 수납함 열기 신호를 프린터로 전송합니다.

현금 수납함은 두 개의 PIN 번호(2번, 5번)가 있으며, 세 번째 매개 변수값에 따라 현금 수납함이 열리지 않을 수도 있으므로 유의하십시오.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
pinNubmer	Number	현금 수납함 PIN 번호 **0:** PIN 번호 2 **1:** PIN 번호 5

5-2-28 Print PDF File

PDF 파일을 인쇄합니다.



UWP는 이 메소드를 지원하지 않습니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
filePath	String	PDF 파일 경로
width	Number	인쇄될 이미지 너비 또는 원본 크기로 인쇄 **-2**: 원본 크기로 인쇄
alignment	Number	정렬 **0:** 좌측 정렬 **1:** 중앙 정렬 **2:** 우측 정렬
startPage	Number	인쇄할 시작 페이지 번호 * 첫 페이지는 '0' 부터 시작합니다.
endPage	Number	인쇄할 마지막 페이지 번호
threshold	Number	인쇄 기준 값 범위: 0 ≤ threshold ≤ 255
ditheringType	Number	디더링 유형 **0:** 디더링 안 함 **1:** 디더링
compressType	Number	압축 유형 **0:** 압축 안 함 **1:** RLE **2:** LZMA

6 MPosControllerLabelPrinter 레퍼런스

6-1 개요

• “MPosControllerLabelPrinter” 객체는 라벨 프린터 제어를 위한 메소드를 제공합니다.

6-2 메소드

6-2-1 Select Interface

연결할 장치(라벨 프린터)의 통신 인터페이스 유형과 주소를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
interfaceType	Number	통신 인터페이스 **1:** 와이파이 (Wi-Fi) **2:** 이더넷 (Ethernet) **4:** 블루투스 (Bluetooth) **16:** USB
address	String	IP 주소: 포트번호, MAC 주소, Bluetooth 시리얼 넘버 예) 192.168.1.2:9100 예) 74:F0:7D:E3:DF:EE

USB 통신은 Android에서만 사용 가능합니다.

6-2-2 Select Command Mode

장치와 직접 통신할지 “B-gate”를 통하여 통신할지 선택합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
commandMode	Number	연결할 장치와 통신 방식 **0**: B-gate를 통하여 장치와 통신 **1**: 장치와 직접 통신

6-2-3 Open Service

장치와 통신 연결을 시작합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
id	Number	연결할 장치의 식별자(ID) 값
timeout	Number	연결 시도에 대한 타임아웃 (초)



- 이 매뉴얼의 3-2-5에 기술된 “searchDevices”를 사용하면, 연결 가능한 장치 식별자(ID) 목록을 얻을 수 있습니다.
- B-gate를 이용한 통신이 아닌 경우, 장치 식별자(id)값은 -1로 설정하십시오.

  * 라벨 프린터의 장치 식별자(ID) 값 범위: 1 ~ 9

[예제]

```
return new Promise(function (returnResolve, returnReject) {
    new Promise(function (resolve, reject) { _LabelPrinter.selectInterface(resolve, reject, _ConnectionInfo.interfaceType,0);})
     .then(function () { return new Promise(function (resolve, reject) { _LabelPrinter.selectCommandMode(resolve, reject, 0);});})
     .then(function () { return new Promise(function (resolve, reject) {
          _LabelPrinter.openService(resolve, function(error){
                    returnReject();
               }, $('#deviceID').val(), _ConnectionInfo.timeout);
          });
     })
     .then(Function{
          returnResolve();
     });
});
```

6-2-4 Close Service

장치와 통신 연결을 종료합니다.



데이터 전송 중에 이 메소드를 호출하면 데이터 전송이 완료되지 않고 통신 연결이 종료될 수 있습니다. 인쇄 속도와 인쇄 길이에 따른 적절한 타임아웃 값을 설정하여 이 메소드를 호출하십시오.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
timeout	Number	연결 종료에 대한 타임아웃 (초)

[예제]

```
_LabelPrinter.closeService(
  function{ console.log("closeService success"); },
  function{ console.log("closeService failed"); },
  3
)
```

6-2-5 Direct IO

장치에 사용자 정의 데이터를 전송합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	Object	사용자 정의 바이트 배열 (Byte Array)

[예제]

```
_LabelPrinter.directIO(
  function{ console.log("directIO success"); },
  function{ console.log("directIO failed"); },
  [0x0D, 0x0A]
)
```

6-2-6 Is Open

장치의 사용 가능 상태를 반환합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 장치 개방 여부를 수신할 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수

[예제]

```
_LabelPrinter.isOpen(
 function(open){
  if(open) { ... }
  else { ... }
 },
 function{ console.log("isOpen failed"); }
)
```

6-2-7 Get Device ID

장치 식별자(ID) 값을 반환합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 장치 ID 값을 수신할 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수

[예제]

```
_LabelPrinter.getDeviceID(
 function(deviceId){ console.log("getDeviceID success"); },
 function{ console.log("getDeviceID failed"); }
)
```

6-2-8 Set Transaction

트랜잭션(Transaction) 모드로 진입 또는 종료하여 버퍼에 있는 데이터를 장치로 전송합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
transactionMode	Number	트랜잭션 모드 **0:** 트랜잭션 모드 종료 및 데이터 전송 **1:** 트랜잭션 모드 진입

[예제]

```
return new Promise(function (returnResolve, returnReject) {
    new Promise(function (resolve, reject) { _LabelPrinter.setTransaction(resolve, reject, 1);})
     /* add more operations as required */
     .then(Function { return new Promise(function (resolve, reject) { _LabelPrinter.setTransaction(resolve, reject,
0);});});})
     .then(Function{
          returnResolve();
     });
});
```

6-2-9 Set Text Encoding

문자 인코딩에 사용할 코드 페이지(code page)를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
textEncoding	Number	문자열 인코딩에 사용할 코드 페이지 **437**: 영어 (PC437) **850**: 서유럽 (PC850) **852**: 라틴어2 (PC852) **860**: 포르투갈어 (PC860) **863**: 캐나다-프랑스어 (PC863) **865**: 노르웨이어 (PC865) **1252**: 라틴어1 (WPC1252) **857**: 터키어 (PC857) **737**: 그리스어 (PC737) **1250**: 체코어 (WPC1250) **1253**: 그리스어 (WPC1253) **1254**: 터키어 (WPC1254) **855**: 키릴어 (PC855) **862**: 히브리어 (PC862) **866**: 키릴어 (PC866) **1251**: 키릴어 (WPC1251) **1255**: 히브리어 (WPC1255) **928**: 그리스어 (PC928) **864**: 아랍어 (PC864) **775**: 발틱어 (PC775) **1257**: 발틱어 (WPC1257) **858**: 서유럽 (PC858) **949**: 한국어 (KS5601)

		932: 일본어 (ShiftJIS) **950**: 중국어 번체 (BIG5) **936**: 중국어 간체 (GB2312)

[예제]

```
_LabelPrinter.setTextEncoding(
  function{ console.log("setTextEncoding success"); },
  function{ console.log("setTextEncoding failed"); },
  437
)
```

6-2-10 Set Character set

문자 인코딩에 사용할 코드 페이지(code page)를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
characterset	Number	문자열 인코딩에 사용할 코드 페이지 **437**: 영어 (PC437) **850**: 서유럽 (PC850) **852**: 라틴어2 (PC852) **860**: 포르투갈어 (PC860) **863**: 캐나다-프랑스어 (PC863) **865**: 노르웨이어 (PC865) **1252**: 라틴어1 (WPC1252) **857**: 터키어 (PC857) **737**: 그리스어 (PC737) **1250**: 체코어 (WPC1250) **1253**: 그리스어 (WPC1253) **1254**: 터키어 (WPC1254) **855**: 키릴어 (PC855) **862**: 히브리어 (PC862) **866**: 키릴어 (PC866) **1251**: 키릴어 (WPC1251) **1255**: 히브리어 (WPC1255) **928**: 그리스어 (PC928) **864**: 아랍어 (PC864) **775**: 발틱어 (PC775) **1257**: 발틱어 (WPC1257) **858**: 서유럽 (PC858) **949**: 한국어 (KS5601)

		932: 일본어 (ShiftJIS) **950**: 중국어 번체 (BIG5) **936**: 중국어 간체 (GB2312)
internationalCharset	Number	국제 문자용 문자 집합(Character Set) **0:** 미국(USA) 코드 **1:** 프랑스(FRANCE) 코드 **2:** 독일(GERMANY) 코드 **3:** 영국(UK) 코드 **4:** 덴마크1(DENMARK1) 코드 **5:** 스웨덴(SWEDEN) 코드 **6:** 이탈리아(ITALY) 코드 **7:** 스페인(SPAIN) 코드 **8:** 일본(JAPAN) 코드 **9:** 노르웨이(NORWAY) 코드 **10:** 덴마크2(DENMARK 2) 코드 **11:** 스페인2(SPAIN 2) 코드 **12:** 라틴 아메리카(LATIN AMERICA) 코드 **13:** 한국(KOREA) 코드 **14:** 슬로베니아(SLOVENIA) 코드 **15:** 중국(CHINA) 코드

[예제]

```
_LabelPrinter.setCharacterset(
  function{ console.log("setCharacterset success"); },
  function{ console.log("setCharacterset failed"); },
  437, 1
)
```

6-2-11 Set Printing Type

라벨 프린터의 열전사 방식을 설정합니다.



라벨 프린터는 열전사 방식에는 “직접 열전사 방식”과 리본에 열전 전달하는 “간접 열전사” 방식이 있습니다. 리본이 장착된 모델인지 확인 후 적절한 방식을 선택하십시오.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
setPrintingType	String	인쇄 타입 설정 **“d”:** 직접 감열 **“t”:** 리본 감열

[예제]

```
_LabelPrinter.setPrintingType(
  function{ console.log("setPrintingType success"); },
  function{ console.log("setPrintingType failed"); },
  ‘d’ /* Direct Thermal */
)
```

6-2-12 Set Margin

인쇄 여백을 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
horizontalMargin	Number	수평 여백 (dot 단위)
verticalMargin	Number	수직 여백 (dot 단위)

[예제]

```
_LabelPrinter.setMargin(
  function{ console.log("setMargin success"); },
  function{ console.log("setMargin failed"); },
  0, 0
)
```

6-2-13 Set Back Feed Option

백 피드(Back feed)를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
isBackFeed	Boolean	백 피드 사용 여부
stepQuantity	Number	백 피드 스텝 수 **0**: 기본 값

[예제]

```
_LabelPrinter.setBackFeedOption(
  function{ console.log("setBackFeedOption success"); },
  function{ console.log("setBackFeedOption failed"); },
  true, 1
)
```

6-2-14 Set Length

라벨 길이, 라벨 간격(Gap) 길이, 라벨 용지 유형, 오프셋(offset) 길이를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
labelLength	Number	용지 길이 (dot 단위)
gapLength	Number	라벨 간 간격 길이 (dot 단위)
mediaType	String	용지 타입 **“G”:** 갭 용지 **“C”:** 연속 용지 **“B”:** 블랙마크 용지
offsetLength	Number	오프셋(offset) 길이 (dot 단위)

[예제]

```
_LabelPrinter.setLength(
  function{ console.log("setLength success"); },
  function{ console.log("setLength failed"); },
  1218, 24, ‘G’, 0
)
```

6-2-15 Set Width

인쇄 너비를 설정합니다.



최대 인쇄 너비는 이 매뉴얼의1-2 “지원 장치”를 참고하십시오.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
labelWidth	Number	인쇄 너비 (dot 단위)

[예제]

```
_LabelPrinter.setWidth(
  function{ console.log("setWidth success"); },
  function{ console.log("setWidth failed"); },
  832
)
```

6-2-16 Set Buffer Mode

이중 버퍼모드 사용 여부를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
doubleBuffer	Boolean	버퍼모드 설정 **true:** 이중 버퍼모드 사용 **false:** 이중 버퍼모드 사용 안 함

[예제]

```
_LabelPrinter.setBufferMode(
  function{ console.log("setBufferMode success"); },
  function{ console.log("setBufferMode failed"); },
  false
)
```

6-2-17 Set Speed

인쇄 속도를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
speed	Number	인쇄 속도 (inch per second)

[예제]

```
_LabelPrinter.setSpeed(
  function{ console.log("setSpeed success"); },
  function{ console.log("setSpeed failed"); },
  4
)
```

6-2-18 Set Density

인쇄 농도를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
density	Number	인쇄 농도 (범위: 1 ~ 20)

[예제]

```
_LabelPrinter.setDensity(
  function{ console.log("setDensity success"); },
  function{ console.log("setDensity failed"); },
  10
)
```

6-2-19 Set Orientation

라벨의 인쇄 방향을 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
orientation	String	인쇄 방향 **"T":** 상단에서 하단으로 인쇄 **"B":** 하단에서 상단으로 인쇄

[예제]

```
_LabelPrinter.setOrientation(
  function{ console.log("setOrientation success"); },
  function{ console.log("setOrientation failed"); },
  'T'
)
```

6-2-20 Set Offset

블랙마크(Black Mark) 또는 라벨 간격(Gap)과 절취선 간에 오프셋(offset) 길이를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
orientation	Number	오프셋 길이 (범위: -100 dot ~ 100 dot)

[예제]

```
_LabelPrinter.setOffset(
  function{ console.log("setOffset success"); },
  function{ console.log("setOffset failed"); },
  0
)
```

6-2-21 Set Cutting Position

라벨 프린터의 커팅 위치 또는 뜯어내기 위치를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
cuttingPosition	Number	offset 값 (범위: -100 dot ~ 100 dot)

[예제]

```
_LabelPrinter.setCuttingPosition(
  function{ console.log("setCuttingPosition success"); },
  function{ console.log("setCuttingPosition failed"); },
  0
)
```

6-2-22 Set Auto Cutter

오토 커터 사용 여부를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
isAutoCutter	Boolean	절단 또는 절단하지 않음 **true:** 절단 **false:** 절단하지 않음.
period	Number	커팅주기 설정. 예) 2로 설정할 경우 2장마다 용지를 커팅 합니다.

[예제]

```
_LabelPrinter.setAutoCutter(
  function{ console.log("setAutoCutter success"); },
  function{ console.log("setAutoCutter failed"); },
  1
)
```

6-2-23 Print Buffer

프린터 버퍼에 저장된 인쇄 데이터를 이용하여 라벨 인쇄를 시작합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
numberOfCopies	Number	복사본 매수 (범위: 1 ~ 65535)

[예제]

```
_LabelPrinter.printBuffer(
  function{ console.log("printBuffer success"); },
  function{ console.log("printBuffer failed"); },
  1
)
```

6-2-24 Check Printer Status

프린터의 현재 상태를 반환합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	프린터 상태 정보를 수신할 콜백 함수 **0:** 정상 (에러 없음) **1:** 종이 없음 **2:** 커버 열림 **3:** 커터 에러 **4:** TPH 과열 **5:** GAP 센서 에러 **6:** 리본 없음 **7:** 라벨 버퍼 구성 중 **8:** 인쇄 중 **9:** 필러에 용지 붙음 **10**: 인쇄물이 걸려 있음 **11**: RFID 쓰기 실패
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수

[예제]

```
_LabelPrinter.checkPrinterStatus(
 function(status){ console.log("checkPrinterStatus success");
     // if (status & LBL_CONST.PrinterStatus.COVEROPEN)
     // if (status & LBL_CONST.PrinterStatus.PAPEREMPTY)
     // if (status & LBL_CONST.PrinterStatus.CUTTER_JAMMED)
     // if (status & LBL_CONST.PrinterStatus.TPH_OVERHEAT)
     // if (status & LBL_CONST.PrinterStatus.GAP_ERROR)
     // if (status & LBL_CONST.PrinterStatus.RIBBON_END)
     // if (status & LBL_CONST.PrinterStatus.BUILDING_LABEL_TO_BE_PRINTED)
     // if (status & LBL_CONST.PrinterStatus.PRINTING_LABEL)
     // if (status & LBL_CONST.PrinterStatus.LABEL_ISPAUSED)
 },
 function{ console.log("checkPrinterStatus failed"); }
)
```

6-2-25 Draw Text Device Font

비트맵 폰트를 이용하여 문자열 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	인쇄할 문자열 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
fontSelection	String	폰트 선택 **"0":** ASCII, 9 X 15 dots **"1":** ASCII, 12 X 20 dots **"2":** ASCII, 16 X 25 dots **"3":** ASCII, 19 X 30 dots **"4":** ASCII, 24 X 38 dots **"5":** ASCII, 32 X 40 dots **"6":** ASCII, 48 X 76 dots **"7":** ASCII, 22 X 34 dots **"8":** ASCII 28 X 44 dots **"9":** ASCII, 37 X 58 dots **"a":** 한국어 16 X 16 dots (숫자/알파벳은 9 X 15) **"b":** 한국어 24 X 24 dots (숫자/알파벳은 12 X 24) **"c":** 한국어 20 X 20 dots (숫자/알파벳은 12 X 20) **"d":** 한국어 26 X 26 dots (숫자/알파벳은 16 X 30) **"e":** 한국어 20 X 26 dots (숫자/알파벳은 16 X 30) **"f":** 한국어 38 X 38 dots (숫자/알파벳은 22 X 34) **"m":** 중문 간체 24 X 24 dots (숫자/알파벳은 12 X 24) **"m":** 중문 번체 24 X 24 dots (숫자/알파벳은 12 X 24) **"j":** 일본어 24 X 24 dots (숫자/알파벳은 12 X 24)
fontWidth	Number	가로 확대 (범위: 1 ~ 4)
fontHeight	Number	세로 확대 (범위: 1 ~ 4)
rightSpace	Number	오른쪽 자간 (기본 값: 0 dot)
rotation	Number	회전

		0: 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전
reverse	Boolean	역상 인쇄 여부
bold	Boolean	굵게 인쇄 여부
rightToLeft	Boolean	문자열 인쇄 방향 (좌측에서 우측, 우측에서 좌측)
alignment	Number	문자열 정렬 **0:** 왼쪽 정렬 **1:** 오른쪽 정렬

[예제]

```
_LabelPrinter.drawTextDeviceFont(
  function{ console.log("drawTextDeviceFont success"); },
  function{ console.log("drawTextDeviceFont failed"); },
  "Bitmap Font", 50, 0, '1', 1, 1, 0, 0, false, false, false, 0
)
```

6-2-26 Draw Text Vector Font

벡터 폰트를 이용하여 문자열 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	인쇄할 문자열 데이터
xPos	Number	X축 좌표
yPos	Number	Y축 좌표
fontSelection	String	폰트 선택 **"U":** ASCII **"K":** 한국어 (KS5601) **"B":** 중국어 번체 (BIG5) **"G":** 중국어 간체 (GB2312) **"J":** 일본어 (Shift-JIS) **"a":** OCR-A **"b":** OCR-B
fontWidth	Number	폰트 가로 너비(dot)
fontHeight	Number	폰트 세로 너비(dot)
rightSpace	Number	오른쪽 자간 (기본 값: 0 dot)
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전
reverse	Boolean	역상 인쇄 여부
bold	Boolean	굵게 인쇄 여부
italic	Boolean	기울임 인쇄 여부
rightToLeft	Boolean	문자열 인쇄 방향 (좌측에서 우측, 우측에서 좌측)
alignment	Number	문자열 정렬 **0:** 좌측 정렬

		1: 우측 정렬 **2:** 가운데 정렬

[예제]

```
_LabelPrinter.drawTextVectorFont(
  function{ console.log("drawTextVectorFont success"); },
  function{ console.log("drawTextVectorFont failed"); },
  "Vector Font", 50, 50, ‘U’, 24, 24, 3, 0, false, false, false, false, 0
)
```

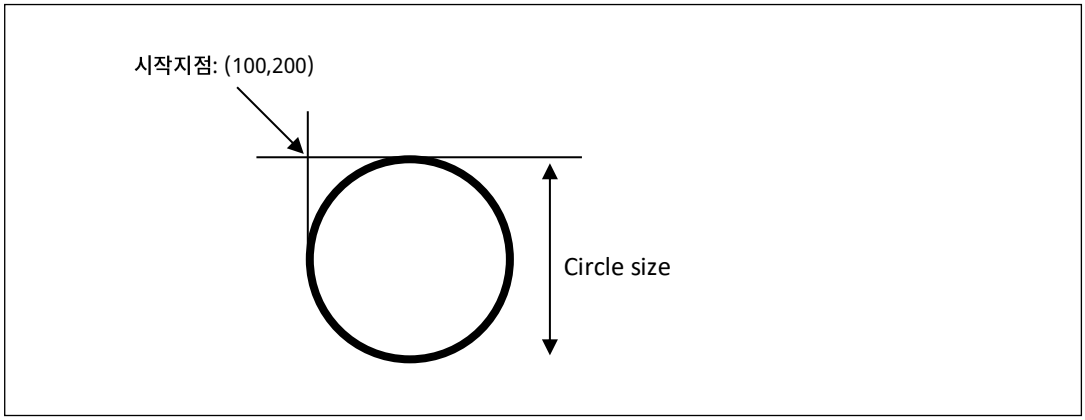
6-2-27 Draw Circle

원(Circle) 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
startPosX	Number	X축 좌표 (dot)
startPosY	Number	Y축 좌표 (dot)
sizeSelection	Number	원 크기 **1:** 40 × 40 dots (W × H) **2:** 56 × 56 dots (W × H) **3:** 72 × 72 dots (W × H) **4:** 88 × 88 dots (W × H) **5:** 104 × 104 dots (W × H) **6:** 168 × 168 dots (W × H)
multiplier	Number	확대 (범위: 1 ~ 4)

* 시작 지점: 좌측과 상단의 접선의 교점



[예제]

```
_LabelPrinter.drawCircle(
  function{ console.log("drawCircle success"); },
  function{ console.log("drawCircle failed"); },
  50, 10, 3 /* 72x72 */, 1
)
```

6-2-28 Draw Block

사각형(quadrangle) 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
startPosX	Number	좌측 상단 X축 좌표 (dot)
startPosY	Number	좌측 상단 Y축 좌표 (dot)
endPosX	Number	우측 하단 X축 좌표 (dot)
endPosY	Number	우측 하단 Y축 좌표 (dot)
option	String	그리기 옵션 **'O':** 선 겹치는 부분은 중복으로 인쇄 **'E':** 선 겹치는 부분은 인쇄 안 함 **'D':** 라인 삭제 **'S':** 사선 **'B':** 박스
thickness	Number	선 굵기 * 'option' 값이 'S' 또는 'B'인 경우에만 사용 가능

[예제]

```
_LabelPrinter.drawBlock(
  function{ console.log("drawBlock success"); },
  function{ console.log("drawBlock failed"); },
  50, 100, 100, 200, 'B', 4
)
```

6-2-29 Draw Image File

HOST 기기에 존재하는 비트맵 파일에 대한 인쇄 데이터를 프린터 버퍼에 저장합니다.



- Android 경우, SD카드에 있는 파일만 인쇄할 수 있습니다.
- 플랫폼(Android, iOS, UWP)의 버전에 따라 파일 접근 권한이 없을 경우 인쇄가 되지 않을 수 있습니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
path	String	이미지 파일 경로
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
width	Number	출력물의 이미지 너비
brightness	Number	이미지 밝기 범위: 0 ≤ brightness ≤ 100 (Android, UWP) 범위: -100 ≤ brightness ≤ 100 (iOS)
isDithering	Boolean	디더링 알고리즘 적용 여부
isCompress	Boolean	데이터 압축 전송 여부

[예제]

```
_LabelPrinter.drawImageFile(
  function{ console.log("drawImageFile success"); },
  function{ console.log("drawImageFile failed"); },
  path, 50, 50, 300, 50, true, true
)
```

6-2-30 Draw Image with Base64

Base64로 인코딩 된 그래픽 이미지 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
imageSource	String	Base64로 인코딩 된 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
width	Number	이미지 너비
brightness	Number	이미지 밝기 범위: 0 ≤ brightness ≤ 100 (Android, UWP) 범위: -100 ≤ brightness ≤ 100 (iOS)
isDithering	Boolean	디더링 알고리즘 적용 여부
isCompress	Boolean	데이터 압축 전송 여부

[예제]

```
_LabelPrinter.drawImageWithBase64(
  function{ console.log("drawImageWithBase64 success"); },
  function{ console.log("drawImageWithBase64 failed"); },
  imageSource, 50, 50, 300, 50, true, true
)
```

6-2-31 Draw Barcode 1D

1차원 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot 단위)
yPos	Number	Y축 좌표 (dot 단위)
barcodeType	Number	1차원 바코드 유형 **0:** Code39 **1:** Code128 **2:** Interleaved 2of 5 **3:** Codabar **4:** Code93 **5:** UPC-A **6:** UPC-E **7:** EAN13 **8:** EAN8 **9:** UCC/EAN128 **10:** Code11 **11:** Planet **12:** Industrial 2 of 5 **13:** Standard 2 of 5 **14:** Logmars **15:** UPC/EAN Extensions **16:** Postnet
widthNarrow	Number	좁은 바 (Narrow bar) 너비
widthWide	Number	넓은 바 (Wide bar) 너비
height	Number	바코드 높이
HRI	Number	HRI (Human Readable Interface) 인쇄 위치 **0:** 인쇄 안 함 **1:** 바코드 하단 (글꼴 크기 1) **2:** 바코드 상단 (글꼴 크기 1) **3:** 바코드 하단 (글꼴 크기 2)

		4: 바코드 상단 (글꼴 크기 2) **5:** 바코드 하단 (글꼴 크기 3) **6:** 바코드 상단 (글꼴 크기 3) **7:** 바코드 하단 (글꼴 크기 4) **8:** 바코드 상단 (글꼴 크기 4)
quietZoneWidth	Number	Quiet zone 너비 (범위: 0 ~ 20)
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcode1D(
  function{ console.log("drawBarcode1D success"); },
  function{ console.log("drawBarcode1D failed"); },
  "1234567890", 50, 50, 1/*Code128*/, 2, 2, 50, 5, 0, 0
)
```

6-2-32 Draw Barcode Maxi Code

Maxicode 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
mode	Number	Maxicode 모드 **0:** 모드 0 **2:** 모드 2 **3:** 모드 3 **4:** 모드 4

[예제]

```
_LabelPrinter.drawBarcodeMaxiCode(
  function{ console.log("drawBarcodeMaxiCode success"); },
  function{ console.log("drawBarcodeMaxiCode failed"); },
  "1234567890", 50, 50, 0
)
```

6-2-33 Draw Barcode PDF417

PDF417 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
Data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
maximumRowCount	Number	Row 개수 (범위: 3 ~ 90)
maximumColumnCount	Number	Column 개수 (범위: 1 ~ 30)
errorCorrectionLevel	Number	에러 교정 레벨 (범위: 0 ~ 8)
dataCompressionMethod	Number	데이터 압축 방법 **0:** 문자, codeword당 2글자 **1:** 숫자, codeword당 2.93 글자 **2:** 바이너리, codeword당 1.2 바이트
hri	Number	HRI (Human Readable Interface) 인쇄 위치 **0:** 인쇄 안 함 **2:** 바코드 하단
barcodeOriginPoint	Number	바코드의 시작지점 (기본값: 1) **0:** 바코드 중앙 **1:** 바코드 좌측 상단
moduleWidth	Number	바코드 모듈의 너비 (범위: 2 ~ 9)
barHeight	Number	바의 높이 (범위: 4 ~ 99)
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodePDF417(
  function{ console.log("drawBarcodePDF417 success"); },
  function{ console.log("drawBarcodePDF417 failed"); },
  "PDF417CODE", 50, 150, 30, 5, 0, 0, true, 1, 3, 10, 0
)
```

6-2-34 Draw Barcode QR Code

QRCode 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
size	Number	크기 (범위: 1 ~ 9)
model	Number	모델 **1:** 모델 1 **2:** 모델 2
errorCorrectionLevel	Number	에러 교정 레벨 **76:** 에러 교정률 7% **77:** 에러 교정률 15% **81:** 에러 교정률 25% **72:** 에러 교정률 35%
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodeQRCode(
  function{ console.log("drawBarcodeQRCode success"); },
  function{ console.log("drawBarcodeQRCode failed"); },
  "QRCODE", 50, 250, 4, 2 /* Model 2*/, 76, 0
)
```

6-2-35 Draw Barcode Data Matrix

DataMatrix 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
size	Number	DataMatrix 크기 (범위: 1 ~ 4)
reverse	Boolean	바코드 역상 또는 정상
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodeDataMatrix(
  function{ console.log("drawBarcodeDataMatrix success"); },
  function{ console.log("drawBarcodeDataMatrix failed"); },
  "DataMatrix", 50, 350, 4, false, 0
)
```

6-2-36 Draw Barcode Aztec

Aztec 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
size	Number	Aztec 크기 (범위: 1 ~ 10)
extendedChannel	Boolean	확장 채널 해석 코드
errorCorrectionLevel	Number	에러 교정 수준 **0:** 기본 에러 교정률 **1~99:** 에러 교정률 (1~99%) **101~104:** 1~4 레이어 컴팩트 심볼 **201~232:** 1~32 레이어 풀 레인지 심볼 **300:** 심플 Aztec “Rune”
menuSymbol	Boolean	메뉴 심볼 활성/비활성
numberOfSymbols	Number	Structured append 심볼 수 (범위: 1 ~ 26)
optionalID	String	Structured append 선택 ID 필드. (최대 24글자)
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodeAztec(
  function{ console.log("drawBarcodeAztec success"); },
  function{ console.log("drawBarcodeAztec failed"); },
  "Aztec", 50, 350, 4, 0, true, 1, “Aztec1”, 0
)
```

6-2-37 Draw Barcode Code49

Code49 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
widthNarrow	Number	좁은 바 (Narrow Bar)의 너비
widthWide	Number	넓은 바 (Wide Bar)의 너비
height	Number	바코드 높이
hri	Number	HRI(Human Readable Interface) 인쇄 위치. **0:** 인쇄 안 함 **1:** 바코드 상단 **2:** 바코드 하단
startingMode	Number	스타팅 모드 (Starting Mode) **0:** Regular Alphanumeric Mode **1:** Multiple Read Alphanumeric **2:** Regular numeric Mode **3:** Group Alphanumeric Mode **4:** Regular Alphanumeric Shift 1 **5:** Regular Alphanumeric Shift 2 **7:** Automatic Mode
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodeCode49(
  function{ console.log("drawBarcodeCode49 success"); },
  function{ console.log("drawBarcodeCode49 failed"); },
  "Code49", 50, 350, 1, 1, 300, 2, 0, 0
)
```

6-2-38 Draw Barcode Coda Block

CodaBlock 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
widthNarrow	Number	좁은 바 (Narrow Bar)의 너비
widthWide	Number	넓은 바 (Wide Bar)의 너비
height	Number	바코드 높이
securityLevel	Boolean	보안 레벨 활성화
numberOfCharactersPerRow	Number	라인 당 글자 수(범위: 2 ~ 62)
mode	String	Codablock 모드 **"A":** Code 39 문자 셋 사용 **"E":** Code 128 문자 셋 사용 **"F":** Code 128 문자 셋 사용, FNC1 추가
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodeCodaBlock(
  function{ console.log("drawBarcodeCodaBlock success"); },
  function{ console.log("drawBarcodeCodaBlock failed"); },
  "CodeBlock", 50, 350, 1, 1, 300, true, 2, 'A', 0
)
```

6-2-39 Draw Barcode Micro PDF

MicroPDF 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
moduleWidth	Number	모듈 너비 (범위: 2 ~ 8)
moduleHeight	Number	모듈 높이 (범위: 1 ~ 99)
mode	Number	모드 (범위: 0 ~ 33)
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodeMicroPDF(
  function{ console.log("drawBarcodeMicroPDF success"); },
  function{ console.log("drawBarcodeMicroPDF failed"); },
  "MicroPDF", 50, 350, 2, 50, 0, 0
)
```

6-2-40 Draw Barcode IMB

IMB 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
hri	Number	HRI(Human Readable Interface) 인쇄 위치 **0:** 인쇄 안 함 **1:** 바코드 하단
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodeIMB(
  function{ console.log("drawBarcodeIMB success"); },
  function{ console.log("drawBarcodeIMB failed"); },
  "BarcodeIMB", 50, 350, true, 0
)
```

6-2-41 Draw Barcode MSI

MSI 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
narrowWidth	Number	좁은 바 (Narrow Bar)의 너비
widthWide	Number	넓은 바 (Wide Bar)의 너비
height	Number	바코드의 높이
checkDigitSelection	Number	체크 디지트 (Check digit) **0:** 체크 디지트 사용 안 함 **1:** 1 mod 10 **2:** 2 mod 10 **3:** 1 mod 11, 1 mod 10
printCheckDigit	Boolean	체크 디지트 출력 여부
hri	Number	HRI (Human Readable Interface) 인쇄 위치 **0:** 인쇄 안 함 **1:** 바코드 상단 **2:** 바코드 하단
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodeMSI(
  function{ console.log("drawBarcodeMSI success"); },
  function{ console.log("drawBarcodeMSI failed"); },
  "BarcodeMSI", 50, 350, 2, 2, 50, 0, false, 2 , 0
)
```

6-2-42 Draw Barcode Plessey

Plessey 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
narrowWidth	Number	좁은 바(Narrow Bar)의 너비
widthWide	Number	넓은 바(Wide Bar)의 너비
height	Number	바코드의 높이
printCheckDigit	Boolean	HRI (Human Readable Interface)에 체크 디지트 출력 여부
hri	Number	HRI (Human Readable Interface) 인쇄 위치 **0:** 인쇄 안 함 **1:** 바코드 상단 **2:** 바코드 하단
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodePlessey(
  function{ console.log("drawBarcodePlessey success"); },
  function{ console.log("drawBarcodePlessey failed"); },
  "BarcodeMSI", 50, 350, 2, 2, 50, false, 2 , 0
)
```

6-2-43 Draw Barcode TLC39

TLC 39 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
narrowWidth	Number	좁은 바 (Narrow Bar)의 너비
widthWide	Number	넓은 바 (Wide Bar)의 너비
height	Number	바코드의 높이
rowHeightOfMicroPDF417	Number	Micro PDF417의 행의 높이 (범위: 1 ~ 255)
narrowWidthOfMicroPDF417	Number	Micro PDF417의 좁은 바 (Narrow bar) 너비 (범위: 1 ~ 10)
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodeTLC39(
  function{ console.log("drawBarcodeTLC39 success"); },
  function{ console.log("drawBarcodeTLC39 failed"); },
  " BarcodeTLC39", 50, 350, 2, 2, 50, 50, 2 , 0
)
```

6-2-44 Draw Barcode RSS

RSS 바코드 인쇄 데이터를 프린터 버퍼에 저장합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
data	String	바코드 데이터
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
barcodeType	Number	RSS 바코드 유형 **0:** RSS14 **1:** RSS14 Truncated **2:** RSS14 Stacked **3:** RSS14 Stacked Omnidirectional **4:** RSS Limited **5:** RSS Expanded **6:** UPC-A **7:** UPC-E **8:** EAN-13 **9:** EAN-8 **10:** UCC/EAN-128, CC-A/B **11:** UCC/EAN-128, CC-C
magnification	Number	확대 (범위: 1 ~ 10)
separatorHeight	Number	RSS 바코드 구분자의 높이 (범위: 0 ~ 22)
barcodeHeight	Number	바코드 높이
segmentWidth	Number	세그먼트 너비 (범위: 0 ~ 22 짝수만 가능)
rotation	Number	회전 **0:** 회전 안 함 **1:** 90도 회전 **2:** 180도 회전 **3:** 270도 회전

[예제]

```
_LabelPrinter.drawBarcodeTLC39(
  function{ console.log("drawBarcodeTLC39 success"); },
  function{ console.log("drawBarcodeTLC39 failed"); },
  "BarcodeRSS", 50, 350, 0, 1, 5, 50, 2, 0
)
```

6-2-45 Setup RFID

RFID 트랜스폰더 타입, 재시도 코딩 횟수, 재시도 라벨수량, 송수신 강도를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
rfidType	Number	RFID 트랜스폰더 Type **0:** none **1:** ISO 18000-6 Type A **2**: ISO 18000-6 Type B **3**: EPC Class 0 **4**: EPC Class 1 **5**: EPC Class 1 Generation 2
numberOfRetries	Number	코딩 실패 시 재시도 횟수 (범위: 1 ~ 10)
numberOfLabel	Number	코딩 실패 시 재시도 라벨 수량 (범위: 1 ~ 5)
radioPower	Number	송수신 강도 (범위: 0 ~ 30)

[예제]

```
_LabelPrinter.setupRFID(
  function{ console.log("setupRFID success"); },
  function{ console.log("setupRFID failed"); },
  5, 3, 2, 15
)
```

6-2-46 Set RFID Position

RFID 라벨의 코딩 위치를 직접 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
transPosition	Number	RFID 라벨 코딩의 위치 (y축 값)

[예제]

```
_LabelPrinter.setRFIDPosition(
  function{ console.log("setRFIDPosition success"); },
  function{ console.log("setRFIDPosition failed"); },
  320
)
```

6-2-47 Set RFID Password

RFID Access Password와 Kill Password를 설정합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
oldAccessPwd	String	현재 사용중인 Access Password
oldKillPwd	String	현재 사용중인 Kill Password
newAccessPwd	String	수정할 Access Password
newKillPwd	String	수정할 Kill Password

[예제]

```
_LabelPrinter.setRFIDPassword(
  function{ console.log("setRFIDPassword success"); },
  function{ console.log("setRFIDPassword failed"); },
  “00000000”, “00000000”, “33333333”, “33333333”
)
```

6-2-48 Set EPC Data Structure

EPC Data를 쓰기 위한 EPC Data 구조를 정의합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
totalSize	Number	필드의 총 비트량
fieldSizes	String	각 필드의 비트량(구분자는 콤마(‘,’)를 사용)

[예제]

```
_LabelPrinter.setEPCDataStructure(
  function{ console.log("setEPCDataStructure success"); },
  function{ console.log("setEPCDataStructure failed"); },
  64, “2,3,14,20,25”
)
```

6-2-49 Write RFID

RFID 라벨을 Write합니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
dataType	Number	data 타입 **65:** 아스키 **72:** 헥사데이터 **69**: EPC (반드시 setEPCDataStructure후 사용) **85**: 사용자 영역 헥사데이터
startingBlockNumber	Number	Write를 시작할 블록
dataLength	Number	Write할 바이트 수
data	String	dataType에 따라 입력 **65:** 아스키 형태로 데이터 입력 **72:** 헥사 형태로 데이터 입력 **69**: 10진수 입력 (설정된 필드당 비트 수) **85**: 헥사 행태로 데이터 입력

[예제]

```
_LabelPrinter.writeRFID(
 function{ console.log("writeRFID success"); },
 function{ console.log("writeRFID failed"); },
 65, 4, 12, '123456789012'
)
```

6-2-50 Lock RFID

Access Password로 Kill/Access/EPC Data를 Lock 시킵니다.



반드시 setRFIDPassword를 실행 후 사용하십시오.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수

[예제]

```
_LabelPrinter.writeRFID(
  function{ console.log("lockRFID success"); },
  function{ console.log("lockRFID failed"); }
)
```

6-5-51 Draw PDF File

PDF 파일에 대한 인쇄 데이터를 프린터 버퍼에 저장합니다.



UWP는 이 메소드를 지원하지 않습니다.

[매개 변수]

매개 변수	데이터 타입	설명
successCallback	Function	메소드 성공 시, 호출되는 콜백 함수
errorCallback	Function	메소드 실패 시, 호출되는 콜백 함수
filePath	String	PDF 파일 경로
xPos	Number	X축 좌표 (dot)
yPos	Number	Y축 좌표 (dot)
width	Number	인쇄될 이미지 너비 또는 원본 크기로 인쇄 **-2**: 원본 크기로 인쇄
pageNumber	Number	인쇄할 페이지 번호 * 첫 페이지는 '0' 부터 시작합니다.
threshold	Number	인쇄 기준 값 범위: 0 ≤ threshold ≤ 255
ditheringType	Number	디더링 유형 **0:** 디더링 안 함 **1:** 디더링
compressType	Number	압축 유형 **0:** 압축 안 함 **1:** RLE **2:** LZMA

7. MPosControllerBCD 레퍼런스

7-1 개요

• MPosControllerBCD 인터페이스는 프린터에 연결된 디스플레이 장치를 제어하기 위한 메소드를 제공합니다.

7-2 메소드

7-2-1 Display String

디스플레이 장치에 문자열을 출력합니다.

[매개변수]

• string data: 출력할 문자

• int codePage: 문자 인코딩용 코드페이지

상수명	값	설명
MPOS_CODEPAGE_PC437	0	영어 (PC437)
MPOS_CODEPAGE_KATAKANA	1	일본어 (카타카나)
MPOS_CODEPAGE_PC850	2	서유럽 (PC850)
MPOS_CODEPAGE_PC860	3	포르투갈어 (PC860)
MPOS_CODEPAGE_PC863	4	캐나다-프랑스어 (PC863)
MPOS_CODEPAGE_PC865	5	노르웨이어 (PC865)
MPOS_CODEPAGE_WPC1252	16	라틴어1 (WPC1252)
MPOS_CODEPAGE_PC866	17	키릴어 (PC866)
MPOS_CODEPAGE_PC852	18	라틴어2 (PC852)
MPOS_CODEPAGE_PC858	19	서유럽 (PC858)
MPOS_CODEPAGE_PC862	21	히브리어 (PC862)
MPOS_CODEPAGE_PC864	22	아랍어 (PC864)
MPOS_CODEPAGE_WPC1253	24	그리스어 (WPC1253)
MPOS_CODEPAGE_WPC1254	25	터키어 (WPC1254)
MPOS_CODEPAGE_WPC1257	26	발틱어 (WPC1257)

MPOS_CODEPAGE_FARSI	27	페르시아어
MPOS_CODEPAGE_WPC1251	28	키릴어 (WPC1251)
MPOS_CODEPAGE_PC737	29	그리스어 (PC737)
MPOS_CODEPAGE_PC775	30	발틱어 (PC775)
MPOS_CODEPAGE_HEBREW_OLD	32	히브리어 (Hebrew Old)
MPOS_CODEPAGE_WPC1255	33	히브리어 (WPC1255)
MPOS_CODEPAGE_PC855	36	키릴어 (PC855)
MPOS_CODEPAGE_PC857	37	터키어 (PC857)
MPOS_CODEPAGE_PC928	38	그리스어 (PC928)
MPOS_CODEPAGE_WPC1256	40	아랍어 (WPC1256)
MPOS_CODEPAGE_WPC1258	41	베트남어 (WPC1258)
MPOS_CODEPAGE_WPC1250	47	체코어 (WPC1250)
MPOS_CODEPAGE_LATIN9	48	라틴어 9 (Latin 9)
MPOS_CODEPAGE_TCVN3	49	베트남어 (TCVN3)
MPOS_CODEPAGE_TCVN3_CAPTIAL	50	베트남어 (TCVN3 Capital)
MPOS_CODEPAGE_VISCII	51	베트남어 (VISCII)

• int internationalCharacterSet: 출력할 문자

상수명	값	설명
MPOS_ICS_USA	0	미국(USA) 코드
MPOS_ICS_FRANCE	1	프랑스(FRANCE) 코드
MPOS_ICS_GERMANY	2	독일(GERMANY) 코드
MPOS_ICS_UK	3	영국(UK) 코드
MPOS_ICS_DENMARK_I	4	덴마크1(DENMARK1) 코드
MPOS_ICS_SWEDEN	5	스웨덴(SWEDEN) 코드
MPOS_ICS_ITALY	6	이탈리아(ITALY) 코드
MPOS_ICS_SPAIN_I	7	스페인(SPAIN) 코드
MPOS_ICS_JAPAN	8	일본(JAPAN) 코드
MPOS_ICS_NORWAY	9	노르웨이(NORWAY) 코드
MPOS_ICS_DENMARK_II	10	덴마크2(DENMARK 2) 코드
MPOS_ICS_SPAIN_II	11	스페인2(SPAIN 2) 코드
MPOS_ICS_LATIN_AMERICA	12	라틴 아메리카(LATIN AMERICA) 코드
MPOS_ICS_KOREA	13	한국(KOREA) 코드
MPOS_ICS_SLOVENIA_CROATIA	14	슬로베니아(SLOVENIA) 코드
MPOS_ICS_CHINA	15	중국(CHINA) 코드

• int line: 출력할 라인 번호

1 ≤ line ≤ 4 (BCD-2000)
1 ≤ line ≤ 2 (BCD-3000)

[예제]

```
_Bcd.displayString(
  function{ console.log("displayString success"); },
  function{ console.log("displayString failed"); },
  'Hello Cordova' + '\n', 0, 0, 1
)
```

7-2-2 Clean Screen

디스플레이 장치에 표시된 모든 내용을 지웁니다.

[예제]

```
_Bcd.cleanScreen (
  function{ console.log("cleanScreen success"); },
  function{ console.log("cleanScreen failed");}
)
```

7-2-3 Store Image File

디스플레이 장치에 이미지를 다운로드 합니다.

[매개변수]

• string fileName: 이미지 경로

• int width: 이미지 너비

-1 또는 1 ≤ width ≤ 240 (BCD-2000)

-1 또는 1 ≤ width ≤ 160 (BCD-3000)

• int imageNumber: 이미지 저장용 인덱스

1 ≤ imageNumber ≤ 5

[예제]

```
_Bcd.storeImageFile (
  function{ console.log("storeImageFile success"); },
  function{ console.log("storeImageFile failed"); } ,
  “fileName”, 160, 1
)
```

7-2-4 Display Image

디스플레이 장치에 이미지를 출력합니다.

[매개변수]

• int imageNumber: 출력할 이미지 인덱스

1 ≤ imageNumber ≤ 5

• int xPos: 출력 X좌표

0 ≤ xPos < 240 (BCD-2000)
0 ≤ xPos < 160 (BCD-3000)

• int yPos: 출력 Y좌표

0 ≤ yPos < 64 (BCD-2000)
0 ≤ yPos < 32 (BCD-3000)

[예제]

```
_Bcd.displayImage (
  function{ console.log("displayImage success"); },
  function{ console.log("displayImage failed"); } ,
  1, 0, 0
)
```

7-2-5 Clear Image

디스플레이 장치에 다운로드 된 이미지를 삭제합니다.

[매개변수]

• int imageNumber: 출력할 이미지 인덱스

1 ≤ imageNumber ≤ 5

[예제]

```
_Bcd.clearImage (
  function{ console.log("clearImage success"); },
  function{ console.log("clearImage failed"); } ,
  1
)
```

7-2-6 Show Cursor

디스플레이 장치에 커서를 표시하거나 감출 수 있습니다.

[매개변수]

• int cursor: **커서 표시 여부**

상수명	값	설명
MPOS_CURSOR_DISABLE	0	**커서 감추기**
MPOS_CURSOR_ENABLE	1	**커서 표시**

[예제]

```
_Bcd.showCursor (
  function{ console.log("showCursor success"); },
  function{ console.log("showCursor failed"); } ,
  1
)
```

8. MPosControllerDallasKey 레퍼런스

8-1 개요

• MPosControllerDallasKey 인터페이스는 프린터에 연결된 Dallas Key를 제어하기 위한 클래스입니다. MPosControllerDevices의 DataEvent를 상속받아 사용 가능합니다.

9. MPosControllerHID 레퍼런스

9-1 개요

• MPosControllerHID 인터페이스는 프린터에 연결된 HID를 제어하기 위한 클래스입니다.
MPosControllerDevices의 DataEvent를 상속받아 사용 가능합니다.

10. MPosControllerMSR 레퍼런스

10-1 개요

• MPosControllerMSR 인터페이스는 프린터에 연결된 MSR 장치를 제어하기 위한 클래스입니다. MPosControllerDevices의 DataEvent를 상속받아 사용 가능합니다.

11. MPosControllerNFC 레퍼런스

11-1 개요

• MPosControllerNFC 인터페이스는 프린터에 연결된 NFC를 제어하기 위한 클래스입니다.
MPosControllerDevices의 DataEvent를 상속받아 사용 가능합니다.

12. MPosControllerRFID 레퍼런스

12-1 개요

• MPosControllerRFID 인터페이스는 프린터에 연결된 RFID를 제어하기 위한 클래스입니다.
MPosControllerDevices의 DataEvent를 상속받아 사용 가능합니다.

13. MPosControllerScale 레퍼런스

13-1 개요

• MPosControllerScale 인터페이스는 프린터에 연결된 저울을 제어하기 위한 클래스입니다.
MPosControllerDevices의 DataEvent를 상속받아 사용 가능합니다.

14. MPosControllerScanner 레퍼런스

14-1 개요

• MPosControllerScanner 인터페이스는 프린터에 연결된 바코드 스캐너를 제어하기 위한 클래스입니다. MPosControllerDevices의 DataEvent를 상속받아 사용 가능합니다.

15. MPosControllerUSBSerial 레퍼런스

15-1 개요

• MPosControllerUSBSerial 인터페이스는 프린터에 연결된 USB to Serial 장치를 제어하기 위한 클래스입니다. MPosControllerDevices의 DataEvent를 상속받아 사용 가능합니다.

개정 이력

Ver.	날짜	설명
1.00	2019-09-20	- 신규제작
1.01	2020-03-11	- 지원 모델 추가 - 잘못 표기된 해상도 수정
1.02	2020-09-17	- 지원 모델 추가 - 프린터 상태 값 추가 - RFID 관련 API 추가
1.03	2020-10-30	- 지원 모델 추가 - Windows 지원 버전 정보 변경 - printText 파라미터 내용 수정
1.04	2021-02-09	- 지원 모델 추가
1.05	2021-08-19	- 지원 모델 추가 - 멀티캐스트 네트워크 권한 요청 설명 추가
1.06	2021-10-06	- 지원 모델 추가
1.07	2022-03-30	- 지원 모델 추가 - 지원 메소드 추가 (PDF 인쇄관련 메소드)
1.08	2022-04-26	- 지원 모델 추가
1.09	2022-06-23	- 지원 모델 추가
1.10	2022-07-04	- 지원 메소드 수정 (setPdfLicenseKey 메소드 삭제)
1.11	2022-07-18	- 지원 모델 추가
1.12	2023-12-27	- MPosControllerConfig API 추가 - USB 주변기기 레퍼런스 추가
1.13	2024-01-26	- XQ-840II, XQ-843II 모델추가
1.14	2024-08-14	- SRP-S300II 모델 추가
1.15	2024-10-07	- XD5-40IId/43IId/40IIt/43IIt 모델 추가
1.16	2024-12-26	- SRP-380II -> SRP-380plus 모델명 변경 - SRP-382II -> SRP-382plus 모델명 변경
1.17	2025-03-07	- XT6-60, 63, XM5-30 모델 추가