SII Print Class Library for Android™
アプリケーションプログラマーズガイド

Rev.06

[対象製品]
DSP-A01シリーズ

セイコーインスツル株式会社

Rev.01	2019年	7月
Rev.02	2022年	3月
Rev.03	2023年	4月
Rev.04	2024年	3月
Rev.05	2025年	5月
Rev.06	2026年	6月

©セイコーインスツル株式会社　2019-2026
無断転載を禁じます。

Android™はGoogle LLC.の商標です。
Bluetooth® ワードマークおよびロゴは、Bluetooth SIG, Inc. が所有する登録商標です。
Oracle および Java® は、Oracle Corporation およびその関連会社の米国およびその他の国における登録商標です。
文中の社名、商品名等は各社の商標または登録商標である場合があります。

本書の内容は、断りなく変更することがあります。
本書及び本書に記載された製品の利用によって発生した
損害及びその回復に要する費用に対し、当社は一切の責任を負いかねます。
本書を第三者に無断で頒布することを禁じます。

はじめに

本書は、セイコーインスツル株式会社（以降：SII）が提供する「SII Print Class Library for Android™」（以降：SIIプリントクラスライブラリ）について説明します。

対象製品

本書の対象製品について下記に記載します。

製品	インターフェイス	本書での記載
DSP-A01-1	USB	ディスプレイ

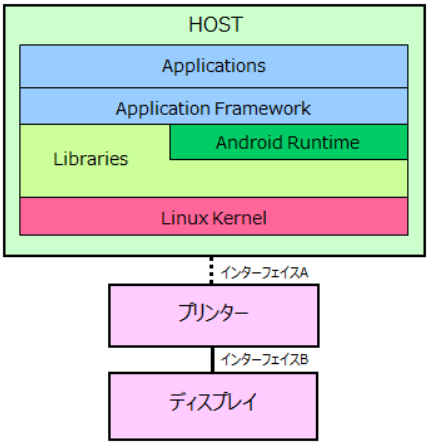
対象プリンター

SIIプリントクラスライブラリでサポートするプリンターを下記に記載します。

プリンター	インターフェイスA	ディスプレイ	インターフェイスB
RP-F10シリーズ	Bluetooth	DSP-A01シリーズ	USB
	USB		
	TCP/IP		

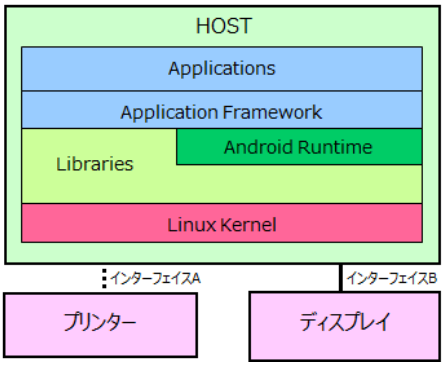
ディスプレイの接続方法は2種類あります。

(1)　ディスプレイを対象プリンターに接続して使用する場合（以降、”プリンター経由で使用”と記載）



構成イメージ

(2)　ディスプレイを単体で使用する場合（以降、”ディスプレイ単体で使用”と記載）



構成イメージ

用語一覧

本書で使用する用語について下記の通りに定義します。

プリンター

用語	内容
プリンター用技術説明書	下記の技術説明書。 ・RP-F10シリーズ サーマルプリンタ 技術説明書
プリンターコマンド	「プリンター用技術説明書」に記載されている、プリンターを制御するための命令。

ディスプレイ

用語	内容
ディスプレイ用技術説明書	下記の技術説明書。 ・DSP-A01シリーズ カスタマーディスプレイ 技術説明書
ディスプレイコマンド	「ディスプレイ用技術説明書」に記載されている、ディスプレイを制御するための命令。
スライド	画面サイズ（横480px×縦272px）のイメージデータ。 スタンバイ画面としてディスプレイへの表示、及び背景画面としてテンプレートと重ね合わせてディスプレイに表示する。
テンプレート	描画領域や配置位置などの属性を設定できる要素を持つ定型フォーム。要素にはtext要素(テキストデータ)、img要素(イメージデータ)、barcode要素(バーコードデータ)、及びqr要素(QRコードデータ)がある。 テンプレートを登録する場合、配置する要素にそれぞれマップIDを定義する。 登録済みのイメージデータやテキストデータ等は、テンプレートを選択し、そのマップIDを指定し、画面更新することでディスプレイに表示ができる。 テンプレートのデータはXMLファイル形式で指定する必要がある。 登録できるテンプレートの最大サイズは横480px×縦272px。
マップID	テンプレートを登録する際に位置や修飾情報等を保持した要素に定義するID。
マクロ	複数のAPIを実行する順に登録し、イベントが発生した際に自動的に実行される機能。
イベント	ディスプレイコマンド「イベント通知」によって定義されたイベント。

目次

1章　製品概要　1-1

1.1　SIIプリントクラスライブラリの提供機能 1-1
1.2　SIIプリントクラスライブラリの概要 1-1
1.2.1　SIIプリントクラスライブラリの構成 1-1
1.2.2　ライブラリの提供機能 1-2
1.2.3　ディスプレイの工場出荷時の登録データ 1-2

2章　製品仕様　2-1

2.1　動作環境 2-1

3章　ライブラリの利用方法　3-1

3.1　Androidアプリケーションの開発環境 3-1
3.2　提供ファイル 3-2
3.3　Android Studioのプロジェクトへの組み込み 3-3
3.4　作成したAndroidアプリケーションのAndroidデバイスでの利用 3-5
3.5　注意事項 3-5

4章　ライブラリの機能　4-1

4.1　ログファイル出力機能 4-1
4.1.1　ログ出力の設定方法 4-1
4.1.2　ログ出力の設定内容 4-1
4.1.3　ログファイル 4-2
4.2　APIリファレンス 4-3
4.2.1　PrinterManagerクラス 4-4
(1)　メソッド一覧 4-4
(2)　定数一覧 4-6
① デバイスモデル 4-6
② 国際文字セット 4-6
③ コードページ 4-7
④ ポート種別 4-7
⑤ ディスプレイ応答種別 4-8
(3)　列挙型定数一覧 4-8
① 強調印字(CharacterBold) 4-8
② アンダーライン(CharacterUnderline) 4-8
③ 文字フォント(CharacterFont) 4-8
④ 文字倍率(CharacterScale) 4-9
⑤ 位置揃え(PrintAlignment) 4-9
⑥ モジュールサイズ(ModuleSize) 4-10
⑦ エラー訂正レベル(ErrorCorrection) 4-10

⑧ メモリ領域(MemoryArea) 4-10
⑨ 登録フォント(RegisteredFont) 4-10
⑩ QRデータモード(QrDataMode) 4-11
⑪ QRクワイエットゾーン(QrQuietZone) 4-11
⑫ マクロ登録処理(MacroRegistrationFunction) 4-11
(4) メソッド詳細 4-12
PrinterManager コンストラクタ 4-12
connect プリンターとの通信開始(Bluetooth) 4-12
connect デバイスとの通信開始(USB) 4-13
connect プリンターとの通信開始(TCP/IP) 4-13
disconnect デバイスとの通信切断 4-14
sendBinary バイナリデータの送信 4-14
sendDataFile 指定ファイルの送信 4-15
startDiscoveryPrinter プリンターの探索開始(Bluetooth) 4-15
startDiscoveryPrinter プリンターの探索開始(USB) 4-16
startDiscoveryPrinter プリンターの探索開始(TCP/IP) 4-16
startDiscoveryDevice デバイスの探索開始(USB) 4-17
cancelDiscoveryPrinter プリンターの探索中断 4-17
cancelDiscoveryDevice デバイスの探索中断 4-17
getFoundPrinter 発見されたプリンター情報リストの取得 4-18
getFoundDevice 発見されたデバイス情報リストの取得 4-18
getSendTimeout 送信タイムアウト時間の取得 4-18
setSendTimeout 送信タイムアウト時間の設定 4-18
getReceiveTimeout 受信タイムアウト時間の取得 4-19
setReceiveTimeout 受信タイムアウト時間の設定 4-19
getInternationalCharacter 国際文字セットの取得 4-19
setInternationalCharacter 国際文字セットの設定 4-19
getCodePage コードページの取得 4-20
setCodePage コードページの設定 4-20
getPrinterModel デバイスモデルの取得 4-20
getPortType 接続中ポート種別の取得 4-21
isConnect デバイスとの通信状態の確認 4-21
getSocketKeepingTime ソケット維持時間の取得 4-21
setSocketKeepingTime ソケット維持時間の設定 4-21
getVersion SDKバージョンの取得 4-22
defragment メモリ領域の最適化 4-22

initializeMemoryArea
メモリ領域の初期化.....4-22
showTemplate テンプレートの表示.....4-23
showSlide スライドの表示.....4-24
enterStandbyMode スタンバイの表示.....4-25
executeMacro マクロの実行.....4-25
turnOnScreen 画面の表示オン・オフ.....4-26
selectTemplate テンプレートの選択.....4-26
setTemplateImageData
イメージデータの設定.....4-27
selectTemplateTextObject
text要素の選択.....4-28
setTemplateTextAlignment
テキストデータの位置揃え.....4-28
setTemplateTextLeftMargin
テキストデータの左マージンの設定.....4-29
setTemplateTextLineSpacing
テキストデータの改行量の設定.....4-30
setTemplateTextBold テキストデータの強調印字の設定.....4-30
setTemplateTextUnderline
テキストデータのアンダーラインの設定.....4-31
setTemplateTextSize テキストデータの文字サイズの設定.....4-31
setTemplateTextFont テキストデータの文字フォントの設定.....4-32
setTemplateTextRegisteredFont
テキストデータの登録フォントの設定.....4-33
setTemplateTextRightSpacing
テキストデータの右スペース量の設定.....4-33
setTemplateTextColor
テキストデータの文字色の設定.....4-34
setTemplateTextData テキストデータの入力.....4-34
setTemplateBarcodeData
バーコードデータの入力.....4-35
setTemplateQrCodeData
QRコードデータの入力.....4-36
registerTemplate テンプレートの登録.....4-38
unregisterTemplate テンプレートの削除.....4-39
registerImageData イメージデータの登録.....4-39
unregisterImageData イメージデータの削除.....4-41
registerSlideData スライドデータの登録.....4-41
unregisterSlideData スライドデータの削除.....4-43
registerUserDefinedCharacter
外字の登録.....4-43
unregisterUserDefinedCharacter
外字の削除.....4-44
registerOptionFont オプションフォントの登録.....4-44
unregisterOptionFont
オプションフォントの削除.....4-46

controlMacroRegistration
マクロ登録の開始・終了 4-46
getDisplayResponse ディスプレイからの各種応答取得 4-48
4.2.2 PrinterEventクラス 4-50
(1) メソッド一覧 4-50
(2) 終了イベント定数 4-50
(3) メソッド詳細 4-50
getEventType 終了イベントの取得 4-50
4.2.3 PrinterListenerインターフェイス 4-51
(1) メソッド一覧 4-51
(2) メソッド詳細 4-51
finishEvent デバイス探索の終了イベント 4-51
4.2.4 PrinterInfoクラス 4-52
(1) メソッド一覧 4-52
(2) メソッド詳細 4-52
getDevicePath デバイスパスの取得 4-52
4.2.5 PrinterExceptionクラス 4-53
(1) メソッド一覧 4-53
(2) 定数一覧 4-53
① エラーコード 4-53
(3) メソッド詳細 4-54
PrinterException コンストラクタ 4-54
getErrorCode エラーコードの取得 4-54

5章 サンプルプログラム 5-1

5.1 画面構成 5-1
5.1.1 メイン画面 5-1
5.1.2 [Settings]画面 5-3
5.2 注意事項 5-3

付録A 文字セット A-1

A.1 コードページ表(文字コードテーブル) A-1
A.2 国際文字セット A-11

1章　製品概要

本章では、SIIプリントクラスライブラリの製品概要について説明します。

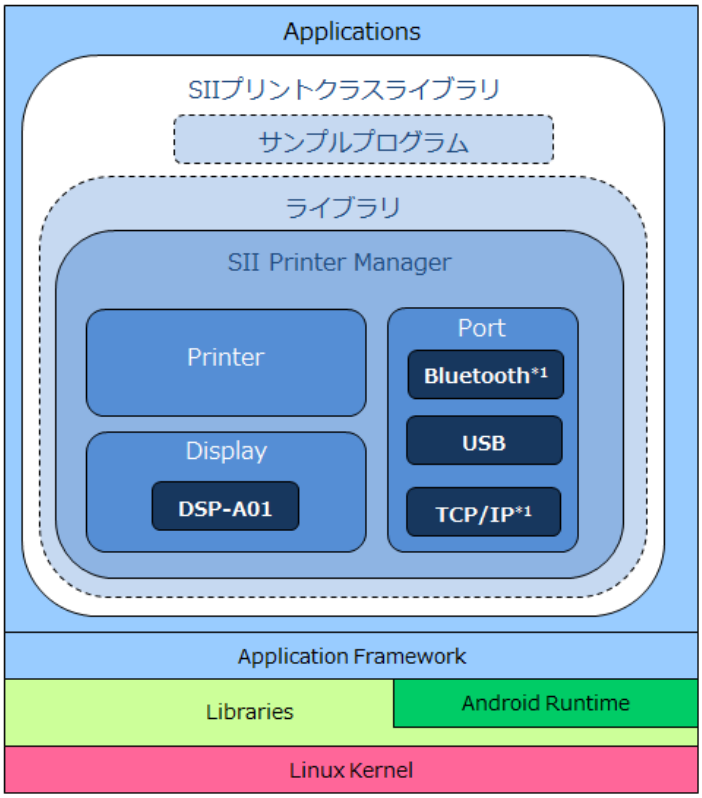
1.1　SIIプリントクラスライブラリの提供機能

ライブラリとサンプルプログラムを含むSIIプリントクラスライブラリは、Androidアプリケーションに「はじめに　対象製品」に記載のディスプレイを利用するための機能を提供します。
また、SIIプリントクラスライブラリは、ライブラリのサンプルプログラムをAndroid Studioのプロジェクトで提供します。

1.2　SIIプリントクラスライブラリの概要

1.2.1　SIIプリントクラスライブラリの構成

SIIプリントクラスライブラリに含まれるライブラリとサンプルプログラムの範囲を、下図に破線で示します。



*1：プリンター経由で使用する場合のみ有効

1.2.2　ライブラリの提供機能

ライブラリを使用することにより、Androidアプリケーションは、Androidデバイスの通信ポート（Bluetooth*1、USB、またはTCP/IP*1）を通じて、デバイスに表示データやディスプレイコマンドを容易に送信することができます。また、ディスプレイの応答を取得することができます。

ライブラリでは、下記の機能を提供します。

- デバイスとの接続及び切断
- デバイスへのデータ送信（表示データやディスプレイコマンド*2）
- デバイスへのデータファイル送信（表示データやディスプレイコマンド*2）
- ディスプレイからの各種応答取得
- ディスプレイの表示制御
- ログファイルの出力

*1:　プリンター経由で使用する場合のみ有効
*2:　ディスプレイから応答を読み取るには、**getDisplayResponse** を使用してください。

1.2.3　ディスプレイの工場出荷時の登録データ

プリンターにディスプレイを接続し、テンプレートを使用する場合、ディスプレイに工場出荷時に登録されるデータは、品質向上のため予告無く追加・変更される場合があります。
また、指定する言語設定、及び文字コードによっては適切なエンコードが指定されているテンプレートを利用する必要があります。工場出荷時に登録されるデータの詳細については、弊社Webサイトを参照してください。
https://www.sii-ps.com/dspa01/

2章　製品仕様

本章では、ライブラリの製品仕様について説明します。

2.1　動作環境

ライブラリの動作環境を下記に示します。

<table>
<tr><td rowspan="2">ディスプレイ</td><td colspan="2">モデル</td><td>DSP-A01</td></tr>
<tr><td colspan="2">通信インターフェイス</td><td>USB</td></tr>
<tr><td rowspan="15">Android
デバイス</td><td colspan="2">通信ポート</td><td>USB*1</td></tr>
<tr><td rowspan="14">OS</td><td>Android 7.0 (API 24)</td><td rowspan="14">対応</td></tr>
<tr><td>Android 7.1 (API 25)</td></tr>
<tr><td>Android 8.0 (API 26)</td></tr>
<tr><td>Android 8.1 (API 27)</td></tr>
<tr><td>Android 9.0 (API 28)</td></tr>
<tr><td>Android 10.0 (API 29)</td></tr>
<tr><td>Android 11.0 (API 30)</td></tr>
<tr><td>Android 12.0 (API 31)</td></tr>
<tr><td>Android 12.1 (API 32)</td></tr>
<tr><td>Android 13.0 (API 33)</td></tr>
<tr><td>Android 14.0 (API 34)</td></tr>
<tr><td>Android 15.0 (API 35)</td></tr>
<tr><td>Android 16.0 (API 36)</td></tr>
<tr><td colspan="3">サポート言語</td><td>日本語/英語</td></tr>
</table>

*1: AndroidデバイスがUSBホスト機能をサポートしている必要があります。

3章　ライブラリの利用方法

本章では、Androidアプリケーションの開発環境とライブラリの利用方法について説明します。

3.1　Androidアプリケーションの開発環境

Androidアプリケーションを開発するためには、下記のツールが必要です。詳細は各URLを参照してください。

- Android Studio
  https://developer.android.com/studio/index.html
- Windows用USB ドライバ（Windows環境で開発する場合）
  https://developer.android.com/studio/run/oem-usb.html

本書では、本章以降、各ツールを利用できる環境が整っていることを前提に説明します。

3.2 提供ファイル

SIIプリントクラスライブラリのファイル構成は、下記の通りです。

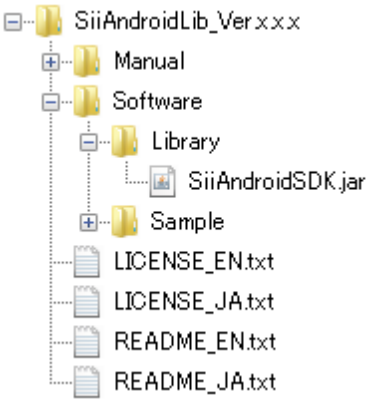


図 3-1

ライブラリは、JARファイル形式です。ライブラリのファイル名は、SiiAndroidSDK.jarです。

3.3 Android Studioのプロジェクトへの組み込み

SIIプリントクラスライブラリをAndroid Studioのプロジェクトに組込む手順を説明します。

SIIプリントクラスライブラリに含まれるサンプルプログラムについては、「5章 サンプルプログラム」を参照してください。

(1) Android Studioでプロジェクトを作成して、[libs]フォルダにライブラリファイル（SiiAndroidSDK.jar）をコピーしてください。[libs]フォルダが自動生成されない場合は、手動でフォルダを追加してください。
SIIプリントクラスライブラリに含まれるサンプルプログラムの場合は、フォルダが「¥Sample¥app¥libs」になります。

(2) ライブラリの追加が完了すると図 3-2の状態となります。

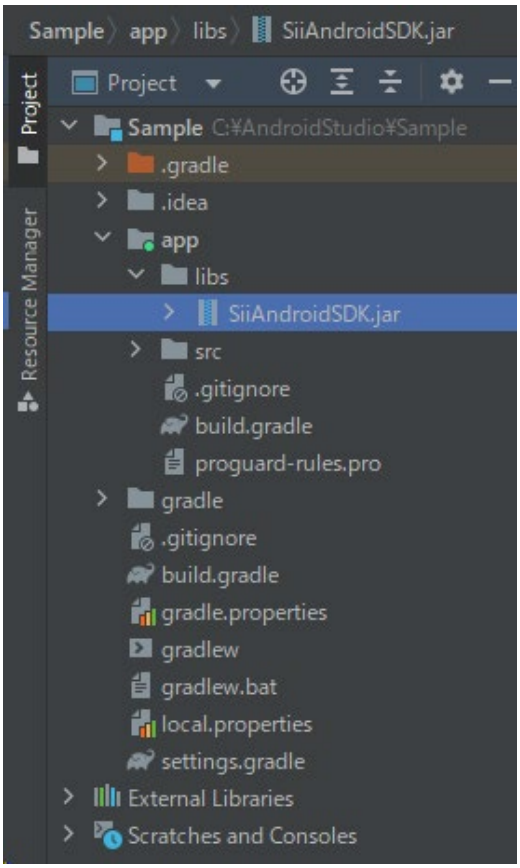


図 3-2

(3) アプリケーションのbuild.gradle(:app)のdependencies{}内に下記を追加してください。

```
implementation 'com.journeyapps:zxing-android-embedded:3.4.0@aar'
implementation 'com.google.zxing:core:3.4.1'
implementation files ('libs/SiiAndroidSDK.jar')
```

(4) プリントクラスライブラリを利用するクラスファイルの先頭に下記を追加してください。
（xxxxは利用する機能に合わせてインポートしてください。）

```
import com.seikoinstruments.sdk.thermalprinter.PrinterManager;
import com.seikoinstruments.sdk.thermalprinter.xxxx;
```

(5) アプリケーションのマニフェスト(AndroidManifest.xml)に下記権限の宣言を追加してください。
また、アプリケーションに適切な権限をリクエストする処理(requestPermissions())を実装してください。

[Bluetoothを利用する場合]

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.BLUETOOTH"/>
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>
<uses-permission android:name="android.permission.BLUETOOTH_CONNECT"/>
<uses-permission android:name="android.permission.BLUETOOTH_SCAN"/>
```

[TCP/IPを利用する場合]

```
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
<uses-permission android:name="android.permission.INTERNET"/>
```

以上で、ライブラリの機能が利用可能になります。

3.4　作成したAndroidアプリケーションのAndroidデバイスでの利用

作成したAndroidアプリケーションをAndroidデバイス上で利用するためには、Androidデバイスで下記の設定を行ってください。

(1)　[設定] – [開発者向けオプション] – [USBデバッグ]をオンにしてください。(図 3-3)

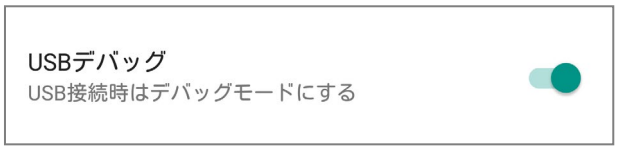


図 3-3

3.5　注意事項

- **対象範囲別ストレージについて**

  Android 10から導入された"対象範囲別ストレージ"ではアプリ固有のストレージとアプリ外部のストレージが区別されます。
  Android 10 (API 29)以降をアプリのターゲットとする場合、アプリ外部のストレージ内にあるメディアファイルに該当しないファイルを直接扱うことができません。メディアファイルに該当しないファイルは"ストレージアクセス フレームワーク"を利用すると扱うことができます。
  対象範囲別ストレージの詳細については、下記を参照してください。

  - データ ストレージとファイル ストレージの概要
    https://developer.android.com/training/data-storage

4章　ライブラリの機能

本章では、ライブラリに実装されている各クラスのAPIについて説明します。

4.1　ログファイル出力機能

ライブラリではログの取得とログファイルの出力ができます。

4.1.1　ログ出力の設定方法

ライブラリを組み込んだAndroidアプリケーションの固有のディレクトリ(例：内部共有ストレージ¥Android¥data¥<package name>¥files)に、下記内容のconfig.iniファイルを追加することで、ログの出力設定が可能になります。

config.ini

```
LOGLEVEL=x
LOGSIZEMAX=xMB
LOGOUTPUT=x
```

参考　　xの設定内容については、「4.1.2 ログ出力の設定内容」を参照してください。

4.1.2　ログ出力の設定内容

項目	説明	設定内容
LOGLEVEL	ログレベル	0 ： ログを記録しません。 1 ： **PrinterException**スロー時のエラーログを記録します。 2 ： APIの実行履歴を記録します。
LOGSIZEMAX	ログファイル最大サイズ	1MB　： ログファイル最大サイズは1MB 5MB　： ログファイル最大サイズは5MB 10MB ： ログファイル最大サイズは10MB 50MB ： ログファイル最大サイズは50MB
LOGOUTPUT	Logcat出力有効・無効	0 ： Logcat出力無効 1 ： Logcat出力有効

4.1.3　ログファイル

ログファイルは、ライブラリを組み込んだAndroidアプリケーションのローカルファイルとして保存されます。

ログファイル名　：　PrinterManagerX.log (Xの範囲は、0～4)

最初のログファイルはPrinterManager0.logで作成されます。ログファイルの最大サイズを超えた場合、ファイル名をPrinterManager1.logに変更し、新たにPrinterManager0.logを作成します。

作成できるログファイルは最大5つです。

4.2 APIリファレンス

ライブラリのパッケージは、**com.seikoinstruments.sdk.thermalprinter**です。
com.seikoinstruments.sdk.thermalprinterには、下記のクラスが含まれます。

クラス名	説明	サポート*1
PrinterManager	デバイスとの通信、表示を行うためのAPIを提供します。 「**4.2.1 PrinterManagerクラス**」を参照してください。	✓
PrinterEvent	**startDiscoveryPrinter**、または**startDiscoveryDevice**が終了した際の終了イベントを取得するAPIを提供します。 「**4.2.2 PrinterEventクラス**」を参照してください。	✓
PrinterListener	**startDiscoveryPrinter**、または**startDiscoveryDevice**が終了した際の終了イベントを取得するためのインターフェイスです。 「**4.2.3 PrinterListenerインターフェイス**」を参照してください。	✓
PrinterInfo	**startDiscoveryPrinter**、または**startDiscoveryDevice**で発見されたデバイス情報を格納します。 「**4.2.4 PrinterInfoクラス**」を参照してください。	✓
PrinterException	API呼び出し時にスローされる例外クラスです。 「**4.2.5 PrinterExceptionクラス**」を参照してください。	✓

*1: ✓:サポート、−:DSP-A01は非サポート

4.2.1 PrinterManagerクラス

(1) メソッド一覧

PrinterManagerクラスで提供されるメソッドを下記に示します。

名前	説明	サポート[*1]
PrinterManager	コンストラクタ	✓
connect	プリンターとの通信開始(Bluetooth)	✓
connect	デバイスとの通信開始(USB)	✓
connect	プリンターとの通信開始(TCP/IP)	✓
disconnect	デバイスとの通信切断	✓
sendBinary	バイナリデータの送信	✓
sendDataFile	指定ファイルの送信	✓
startDiscoveryPrinter	プリンターの探索開始(Bluetooth)	✓
startDiscoveryPrinter	プリンターの探索開始(USB)	✓
startDiscoveryPrinter	プリンターの探索開始(TCP/IP)	✓
startDiscoveryDevice	デバイスの探索開始(USB)	✓
cancelDiscoveryPrinter	プリンターの探索中断	✓
cancelDiscoveryDevice	デバイスの探索中断	✓
getFoundPrinter	発見されたプリンター情報リストの取得	✓
getFoundDevice	発見されたデバイス情報リストの取得	✓
getSendTimeout	送信タイムアウト時間の取得	✓
setSendTimeout	送信タイムアウト時間の設定	✓
getReceiveTimeout	受信タイムアウト時間の取得	✓
setReceiveTimeout	受信タイムアウト時間の設定	✓
getInternationalCharacter	国際文字セットの取得	✓
setInternationalCharacter	国際文字セットの設定	✓
getCodePage	コードページの取得	✓
setCodePage	コードページの設定	✓
getPrinterModel	デバイスモデルの取得	✓
getPortType	接続ポートの種別取得	✓
isConnect	デバイスとの通信状態の確認	✓
getSocketKeepingTime	ソケット維持時間の取得	✓
setSocketKeepingTime	ソケット維持時間の設定	✓
getVersion	SDKバージョンの取得	✓
defragment	メモリ領域の最適化	✓
initializeMemoryArea	メモリ領域の初期化	✓
showTemplate	テンプレートの表示	✓
showSlide	スライドの表示	✓
enterStandbyMode	スタンバイの表示	✓
executeMacro	マクロの実行	✓
turnOnScreen	画面の表示オン・オフ	✓

名前	説明	サポート[*1]
selectTemplate	テンプレートの選択	✓
setTemplateImageData	イメージデータの設定	✓
selectTemplateTextObject	text要素の選択	✓
setTemplateTextAlignment	テキストデータの位置揃え	✓
setTemplateTextLeftMargin	テキストデータの左マージンの設定	✓
setTemplateTextLineSpacing	テキストデータの改行量の設定	✓
setTemplateTextBold	テキストデータの強調印字の設定	✓
setTemplateTextUnderline	テキストデータのアンダーラインの設定	✓
setTemplateTextSize	テキストデータの文字サイズの設定	✓
setTemplateTextFont	テキストデータの文字フォントの設定	✓
setTemplateTextRegisteredFont	テキストデータの登録フォントの設定	✓
setTemplateTextRightSpacing	テキストデータの右スペース量の設定	✓
setTemplateTextColor	テキストデータの文字色の設定	✓
setTemplateTextData	テキストデータの入力	✓
setTemplateBarcodeData	バーコードデータの入力	✓
setTemplateQrCodeData	QRコードデータの入力	✓
registerTemplate	テンプレートの登録	✓
unregisterTemplate	テンプレートの削除	✓
registerImageData	イメージデータの登録	✓
unregisterImageData	イメージデータの削除	✓
registerSlideData	スライドデータの登録	✓
unregisterSlideData	スライドデータの削除	✓
registerUserDefinedCharacter	外字の登録	✓
unregisterUserDefinedCharacter	外字の削除	✓
registerOptionFont	オプションフォントの登録	✓
unregisterOptionFont	オプションフォントの削除	✓
controlMacroRegistration	マクロ登録の開始・終了	✓
getDisplayResponse	ディスプレイからの各種応答取得	✓

*1: ✓:サポート、−:DSP-A01は非サポート

(2) 定数一覧

① デバイスモデル

デバイスとの通信開始と、デバイスモデル取得で利用する定数を下記に示します。

定数名	説明	値
PRINTER_MODEL_RP_FG10[*1]	RP-F10	301
PRINTER_MODEL_DSP_A01	DSP-A01	303
PRINTER_MODEL_DEFAULT	デバイスモデルの初期値	284

*1: プリンター経由で使用している場合のみ有効

② 国際文字セット

国際文字セットの設定/取得で利用する定数を下記に示します。

定数名	説明	値
COUNTRY_USA	アメリカ(USA)	0
COUNTRY_FRANCE	フランス(France)	1
COUNTRY_GERMANY	ドイツ(Germany)	2
COUNTRY_ENGLAND	イギリス(United Kingdom)	3
COUNTRY_DENMARK_1	デンマークⅠ(Denmark I)	4
COUNTRY_SWEDEN	スウェーデン(Sweden)	5
COUNTRY_ITALY	イタリア(Italy)	6
COUNTRY_SPAIN	スペインⅠ(Spain I)	7
COUNTRY_JAPAN	日本(Japan)	8
COUNTRY_NORWAY	ノルウェー(Norway)	9
COUNTRY_DENMARK_2	デンマークⅡ(Denmark II)	10
COUNTRY_SPAIN_2	スペインⅡ(Spain II)	11
COUNTRY_LATIN_AMERICA	ラテンアメリカ(Latin America)	12
COUNTRY_ARABIA	アラビア(Arabia)	17

③ コードページ

コードページの設定/取得で利用する定数を下記に示します。

定数名	説明	値
CODE_PAGE_437	USA, Standard Europe (Code Page437)	0
CODE_PAGE_KATAKANA	Katakana	1
CODE_PAGE_850	Multilingual (Code Page850)	2
CODE_PAGE_860	Portuguese (Code Page860)	3
CODE_PAGE_863	Canadian-French (Code Page863)	4
CODE_PAGE_865	Nordic (Code Page865)	5
CODE_PAGE_857[*1]	Turkish (Code Page857)	13
CODE_PAGE_737	Greek (Code Page737)	14
CODE_PAGE_1252	Latin (Code Page1252)	16
CODE_PAGE_866	Russian (Code Page866)	17
CODE_PAGE_852	Eastern Europe (Code Page852)	18
CODE_PAGE_858	Euro (Code Page858)	19
CODE_PAGE_855	Cyrillic (Code Page855)	34
CODE_PAGE_864[*1*2]	Arabic (Code Page864)	37
CODE_PAGE_1250	Central European (Code Page1250)	45
CODE_PAGE_1251	Cyrillic (Code Page1251)	46
CODE_PAGE_1253[*3]	Greek (Code Page1253)	47
CODE_PAGE_1254	Turkish (Code Page1254)	48

*1: Unicodeの20AChは表示できません。
*2: フォントBは表示できません。
*3 Unicodeの00AAhは表示できません。

④ ポート種別

デバイスとの通信開始と、接続ポート種別取得で利用する定数を下記に示します。

定数名	説明	値
PRINTER_TYPE_BLUETOOTH[*1]	Bluetooth	0
PRINTER_TYPE_USB	USB	1
PRINTER_TYPE_TCP[*1]	TCP/IP	2

*1: プリンター経由で使用している場合のみ有効

⑤ ディスプレイ応答種別

ディスプレイからの各種応答取得で利用する定数を下記に示します。

定数名	説明	値
DISPLAY_RESPONSE_REQUEST	実行応答リクエスト	0
DISPLAY_RESPONSE_USER_AREA	ユーザ領域の残り容量の送信	1
DISPLAY_RESPONSE_TEMPLATE_ID_LIST	テンプレートIDの送信	2
DISPLAY_RESPONSE_IMAGE_ID_LIST	イメージIDの送信	3
DISPLAY_RESPONSE_SLIDE_ID_LIST	スライドIDの送信	4
DISPLAY_RESPONSE_TEMPLATE_LABEL	テンプレート名の送信	5
DISPLAY_RESPONSE_IMAGE_LABEL	イメージ名の送信	6
DISPLAY_RESPONSE_SLIDE_LABEL	スライド名の送信	7

(3) 列挙型定数一覧

① 強調印字(CharacterBold)

強調印字で利用する列挙型定数を下記に示します。

定数名	説明
BOLD_CANCEL	強調印字を解除
BOLD	強調印字を指定

② アンダーライン(CharacterUnderline)

アンダーラインで利用する列挙型定数を下記に示します。

定数名	説明
UNDERLINE_CANCEL	アンダーラインを解除
UNDERLINE_1	1ドット幅アンダーラインを指定

③ 文字フォント(CharacterFont)

文字フォントで利用する列挙型定数を下記に示します。

定数名	説明
FONT_A	フォントA(24×12)
FONT_B	フォントB(16×8)

④ 文字倍率(CharacterScale)

文字倍率で利用する列挙型定数を下記に示します。

定数名	説明
VERTICAL_1_HORIZONTAL_1	縦1倍・横1倍
VERTICAL_1_HORIZONTAL_2	縦1倍・横2倍
VERTICAL_1_HORIZONTAL_3	縦1倍・横3倍
VERTICAL_1_HORIZONTAL_4	縦1倍・横4倍
VERTICAL_2_HORIZONTAL_1	縦2倍・横1倍
VERTICAL_2_HORIZONTAL_2	縦2倍・横2倍
VERTICAL_2_HORIZONTAL_3	縦2倍・横3倍
VERTICAL_2_HORIZONTAL_4	縦2倍・横4倍
VERTICAL_3_HORIZONTAL_1	縦3倍・横1倍
VERTICAL_3_HORIZONTAL_2	縦3倍・横2倍
VERTICAL_3_HORIZONTAL_3	縦3倍・横3倍
VERTICAL_3_HORIZONTAL_4	縦3倍・横4倍
VERTICAL_4_HORIZONTAL_1	縦4倍・横1倍
VERTICAL_4_HORIZONTAL_2	縦4倍・横2倍
VERTICAL_4_HORIZONTAL_3	縦4倍・横3倍
VERTICAL_4_HORIZONTAL_4	縦4倍・横4倍

⑤ 位置揃え(PrintAlignment)

位置揃えで利用する列挙型定数を下記に示します。
改行前のテキストデータに位置揃えを追加することはできません。

定数名	説明
ALIGNMENT_LEFT	左揃え
ALIGNMENT_CENTER	中央揃え
ALIGNMENT_RIGHT	右揃え

⑥ モジュールサイズ(ModuleSize)

QRコードのモジュールサイズで利用する列挙型定数を下記に示します。

定数名	説明	使用メソッド
QR_MODULE_SIZE_2	2ドット	**setTemplateQrCodeData**
QR_MODULE_SIZE_3	3ドット	
QR_MODULE_SIZE_4	4ドット	
QR_MODULE_SIZE_5	5ドット	
QR_MODULE_SIZE_6	6ドット	
QR_MODULE_SIZE_7	7ドット	
QR_MODULE_SIZE_8	8ドット	
QR_MODULE_SIZE_9	9ドット	
QR_MODULE_SIZE_10	10ドット	
QR_MODULE_SIZE_11	11ドット	

⑦ エラー訂正レベル(ErrorCorrection)

エラー訂正レベルで利用する列挙型定数を下記に示します。

定数名	説明	使用メソッド
QR_ERROR_CORRECTION_L	エラー訂正レベル L	**setTemplateQrCodeData**
QR_ERROR_CORRECTION_M	エラー訂正レベル M	
QR_ERROR_CORRECTION_H	エラー訂正レベル H	
QR_ERROR_CORRECTION_Q	エラー訂正レベル Q	

⑧ メモリ領域(MemoryArea)

メモリ領域の操作で利用する列挙型定数を下記に示します。

定数名	説明
MEMORY_DISPLAY_USERMEMORY	ユーザ領域

⑨ 登録フォント(RegisteredFont)

登録フォントで利用する列挙型定数を下記に示します。

定数名	説明
FONT_STANDARD	標準フォント
FONT_OPTION	オプションフォント

⑩ QRデータモード(QrDataMode)

QRデータモードで利用する列挙型定数を下記に示します。

定数名	説明
QRDATAMODE_NUMERIC	数字モード
QRDATAMODE_ALPHANUMERIC	英数字モード
QRDATAMODE_8BITBYTE	8ビットバイトモード
QRDATAMODE_KANJI	漢字モード
QRDATAMODE_MIXTURE	混在モード

⑪ QRクワイエットゾーン(QrQuietZone)

QRクワイエットゾーンで利用する列挙型定数を下記に示します。

定数名	説明
QRQUIETZONE_EXIST	QRクワイエットゾーンあり
QRQUIETZONE_NONE	QRクワイエットゾーンなし

⑫ マクロ登録処理(MacroRegistrationFunction)

マクロ登録処理で利用する列挙型定数を下記に示します。

定数名	説明
MACRO_REGISTRATION_CLEAR	マクロ登録処理の中止
MACRO_REGISTRATION_START	マクロ登録処理の開始
MACRO_REGISTRATION_REGIST	マクロ登録とマクロ登録処理の終了

(4) メソッド詳細

PrinterManager　コンストラクタ

com.seikoinstruments.sdk.thermalprinter.PrinterManagerクラスのコンストラクタです。

構文　public **PrinterManager**(Context *context*)

パラメータ　*context*　本メソッドを呼び出すアプリケーションのコンテキストを指定してください。
例： **MainActivity.this**

connect　プリンターとの通信開始(Bluetooth)

Bluetooth接続でプリンターとの通信を開始します。
構文(a)は、常にセキュアなモードでプリンターと通信を行います。
構文(b)は、セキュアなモード、またはインセキュアなモードを指定してプリンターとの通信を行います。

構文　(a) public void **connect**(int *printerModel*, String *address*) throws **PrinterException**

(b) public void **connect**(int *printerModel*, String *address*, boolean *secure*) throws **PrinterException**

パラメータ　*printerModel*　Bluetooth接続対象のデバイスモデル定数
利用可能な定数は、「4.2.1(2)① デバイスモデル」を参照してください。

address　Bluetoothアドレス
例： ″00:11:22:AA:BB:CC″

secure　true　セキュアなモードでプリンターと通信
false　インセキュアなモードでプリンターと通信
通常はセキュアなモードでの通信を推奨します。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーの内容は、「**4.2.5 PrinterExceptionクラス**」を参照してください。

説明　本メソッドはプリンター経由で使用している場合のみ有効です。

本メソッドは、他の**PrinterManager**クラスのメソッドを使用する前に呼び出してください。

*address*で指定したBluetoothアドレスに対し、*printerModel*で指定したプリンターを接続します。
また、指定した*printerModel*を基に、接続時にプリンターの初期設定を行います。

connect　デバイスとの通信開始(USB)

USB接続でデバイスとの通信を開始します。
構文(a)は、指定した機種のデバイスと通信を行います。
構文(b)は、指定した機種、及びパスのデバイスと通信を行います。

構文　(a) public void **connect**(int *printerModel*) throws **PrinterException**

(b) public void **connect**(int *printerModel*, String *address*) throws **PrinterException**

パラメータ　*printerModel*　USB接続対象のデバイスモデル定数
利用可能な定数は、「4.2.1(2)① デバイスモデル」を参照してください。

address　USBデバイスファイルのパス
例: "/dev/bus/usb/001/002"

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーの内容は、「**4.2.5 PrinterExceptionクラス**」を参照してください。

説明　本メソッドは、他の**PrinterManager**クラスのメソッドを使用する前に呼び出してください。

printerModel、及び、*address*で指定したデバイスに接続します。
また、指定した*printerModel*を基に、接続時にデバイスの初期設定を行います。

connect　プリンターとの通信開始(TCP/IP)

TCP/IP接続でプリンターとの通信を開始します。

構文　public void **connect**(int *printerModel*, String *address*) throws **PrinterException**

パラメータ　*printerModel*　イーサネット接続対象のデバイスモデル定数
利用可能な定数は、「4.2.1(2)① デバイスモデル」を参照してください。

address　IPアドレス
例: "192.168.0.190"

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーの内容は、「**4.2.5 PrinterExceptionクラス**」を参照してください。

説明　本メソッドはプリンター経由で使用している場合のみ有効です。

本メソッドは、他の**PrinterManager**クラスのメソッドを使用する前に呼び出してください。

Androidデバイスと同じネットワークに接続されたプリンターと、TCP/IP接続で通信を開始します。
*address*で指定されたIPアドレスに対して接続を行います。通信にはTCPポート9100番、及び26100番を使用します。また、指定した*printerModel*を基に、接続時にプリンターの初期設定を行います。

- **ライブラリのTCP/IP接続中のソケットの作成・破棄について**

  ライブラリは**connect**後、**disconnect**するまで作成したソケットを維持します。また、**disconnect**するまで別のアプリケーションから同じプリンターに接続することはできません。

  プリンターへのデータ送信完了時を基準とし、**setSocketKeepingTime**で設定するソケット維持時間経過後、ソケットをいったん破棄します。その後すぐに新規のソケットを作成し、次の接続に使用します。

disconnect　デバイスとの通信切断

デバイスとの通信を終了します。

構文　public void **disconnect**() throws **PrinterException**

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーの内容は、「**4.2.5 PrinterExceptionクラス**」を参照してください。

注意　本メソッドの直前に、**getDisplayResponse**の**DISPLAY_RESPONSE_REQUEST**（実行応答リクエスト）により実行応答を取得することを推奨します。取得しない場合は、Androidデバイスのデータが全てデバイスに送信される前に通信が切断され、データの一部が消失する場合があります。
getDisplayResponseを実行しない場合は、お客様のプログラムにおいて、問題が無いことを十分評価してからご使用ください。

sendBinary　バイナリデータの送信

バイナリデータをデバイスに送信します。

構文　public void **sendBinary**(byte [] *binary*) throws **PrinterException**

パラメータ　*binary*　デバイスに送信するバイナリデータ
一度に指定可能なデータサイズは3MB(3145728バイト)です。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドは、指定されたバイナリデータを変換せずにデバイスに送信します。

本メソッドでディスプレイコマンドをバイナリデータとして送信することにより、ライブラリではサポートされていないディスプレイ機能を利用できます。ただし、本メソッドはディスプレイから応答を取得するコマンドには対応しておりません。

sendDataFile　指定ファイルの送信

ファイルデータを送信します。

構文　public void **sendDataFile**(String *fileName*) throws **PrinterException**

パラメータ　*fileName*　デバイスに送信するデータファイル名
入力可能な形式について下記に説明します。

- Javaの標準的なクラス「java.io.File」が扱う絶対パス文字列
  Android 10 (API 29)以降をアプリのターゲットとする場合には、一部のファイルが直接扱えなくなりますのでご注意ください。
  詳細については、「3.5 注意事項 – 対象範囲別ストレージについて」を参照してください。

- Androidに用意されたクラス「android.net.Uri」が扱う下記スキーム名のURI文字列
  ・file://
  ・content://
  本パラメータには"ストレージ アクセス フレームワーク"から取得したURI文字列を指定する必要があります。"ストレージ アクセス フレームワーク"から取得せずに作成されたURIはファイルを開けない可能性がありますのでご注意ください。

指定可能な最大ファイルサイズは、16MB(16777216バイト)です。
送信可能なファイル拡張子とファイルの送信について下記に説明します。

- .bin、.dat
  バイナリデータとして変換せずにデバイスに送信します。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

startDiscoveryPrinter　プリンターの探索開始(Bluetooth)

Bluetooth接続のプリンターを探索します。発見されたプリンターのプリンター情報は、**PrinterInfo**クラスに格納されます。

構文　public void **startDiscoveryPrinter**(PrinterListener *listener*) throws **PrinterException**

パラメータ　*listener*　**PrinterListener**のインスタンス
本メソッドの終了や**cancelDiscoveryPrinter**による中断は*listener*に設定されたインスタンスを経由して、**finishEvent**により、終了イベントとしてユーザアプリケーションに通知されます。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。

説明　本メソッドでは、SII製プリンター以外も発見される場合があります。
また、すでにライブラリや他のアプリケーションでBluetooth接続が確立されているプリンターは発見されません。

本メソッドはアプリケーションのメインスレッドからは呼び出さないでください。

startDiscoveryPrinter　プリンターの探索開始(USB)

USB接続のプリンターを探索します。発見されたプリンターのプリンター情報は、**PrinterInfo**クラスに格納されます。

構文　public void **startDiscoveryPrinter**(PrinterListener *listener*, int *deviceType*)
throws **PrinterException**

パラメータ　*listener*　**PrinterListener**のインスタンス
本メソッドの終了や**cancelDiscoveryPrinter**による中断は*listener*に設定されたインスタンスを経由して、**finishEvent**により、終了イベントとしてユーザアプリケーションに通知されます。

deviceType　ポート種別
PRINTER_TYPE_USBを指定してください。

エラー　本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。

説明　本メソッドは、SII製デバイスを探索します。発見したデバイスのデバイス情報は、後述する**PrinterInfo**クラスに格納されます。

startDiscoveryPrinter　プリンターの探索開始(TCP/IP)

SII製プリンターを探索します。

構文　public void **startDiscoveryPrinter**(PrinterListener *listener*, int *retry*, int *timeout*)
throws **PrinterException**

パラメータ　*listener*　**PrinterListener**のインスタンス
本メソッドの終了や**cancelDiscoveryPrinter**による中断は*listener*に設定されたインスタンスを経由して、**finishEvent**により、終了イベントとしてユーザアプリケーションに通知されます。

retry　リトライ回数(回)
*retry*で設定した回数だけローカルブロードキャストパケットを送信します。
有効範囲は、1～5です。
1未満の値を指定した場合は1回に設定されます。
5を超える値を指定した場合は5回に設定されます。

timeout　探索タイムアウト時間（ミリ秒：ms）
探索1回あたりのタイムアウト時間を設定します。ローカルブロードキャストパケットの送信毎に、*timeout*で指定した時間が経過するまでプリンターからの応答を待ちます。
有効範囲は、3000～60000です。
3000未満の値を指定した場合は3000msに設定されます。
60000を超える値を指定した場合は60000msに設定されます。

エラー　本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。

説明　本メソッドは、SII製プリンターを探索します。発見したプリンターのプリンター情報は、後述する**PrinterInfo**クラスに格納されます。

startDiscoveryDevice　デバイスの探索開始(USB)

SII製デバイスを探索します。

構文　public void **startDiscoveryDevice**(PrinterListener *listener*, int *deviceType*) throws **PrinterException**

パラメータ　*listener*　**PrinterListener**のインスタンス
本メソッドの終了や**cancelDiscoveryDevice**による中断は*listener*に設定されたインスタンスを経由して、**finishEvent**により、終了イベントとしてユーザアプリケーションに通知されます。

deviceType　ポート種別
PRINTER_TYPE_USBを指定してください。

エラー　本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。

説明　本メソッドは、SII製デバイスを探索します。発見したデバイスのデバイス情報は、後述する**PrinterInfo**クラスに格納されます。

cancelDiscoveryPrinter　プリンターの探索中断

実行中の**startDiscoveryPrinter**を中断します。

構文　public void **cancelDiscoveryPrinter**()

説明　本メソッドによる中断は、**startDiscoveryPrinter**の*listener*に設定されたインスタンスを経由して、**finishEvent**により、終了イベントとしてユーザアプリケーションに通知されます。

cancelDiscoveryDevice　デバイスの探索中断

実行中の**startDiscoveryDevice**を中断します。

構文　public void **cancelDiscoveryDevice**()

説明　本メソッドによる中断は、**startDiscoveryDevice**の*listener*に設定されたインスタンスを経由して、**finishEvent**により、終了イベントとしてユーザアプリケーションに通知されます。

getFoundPrinter　発見されたプリンター情報リストの取得

startDiscoveryPrinterで発見されたプリンターのプリンター情報を、格納先の**PrinterInfo**クラスからArrayListで取得します。

構文　public ArrayList<**PrinterInfo**> **getFoundPrinter**()

戻り値　**PrinterInfo**クラスのArrayList

getFoundDevice　発見されたデバイス情報リストの取得

startDiscoveryDeviceで発見されたデバイスのデバイス情報を、格納先の**PrinterInfo**クラスからArrayListで取得します。

構文　public ArrayList<**PrinterInfo**> **getFoundDevice**()

戻り値　**PrinterInfo**クラスのArrayList

getSendTimeout　送信タイムアウト時間の取得

データ送信時のタイムアウト時間を取得します。

構文　public int **getSendTimeout**()

戻り値　送信タイムアウト時間（ミリ秒：ms）

説明　本メソッドは**isConnect**がtrue/falseに関わらず取得が可能です。

setSendTimeout　送信タイムアウト時間の設定

データ送信時のタイムアウト時間を設定します。

構文　public void **setSendTimeout**(int *sendTimeout*)

パラメータ　*sendTimeout*　送信タイムアウト時間（ミリ秒：ms）
有効範囲は、100～90000です。
有効範囲外の値を指定した場合は10000msに設定されます。

説明　本メソッドで送信タイムアウト時間の設定を行っていない場合の値は10000となります。

本メソッドは**isConnect**がtrue/falseに関わらず設定が可能です。

設定したタイムアウト時間が有効になるのは、次回データ送信時となります。

getReceiveTimeout　受信タイムアウト時間の取得

データ受信時のタイムアウト時間を取得します。

構文　public int **getReceiveTimeout**()

戻り値　受信タイムアウト時間（ミリ秒：ms）

説明　本メソッドは**isConnect**がtrue/falseに関わらず取得が可能です。

setReceiveTimeout　受信タイムアウト時間の設定

データ受信時のタイムアウト時間を設定します。

構文　public void **setReceiveTimeout**(int *receiveTimeout*)

パラメータ　*receiveTimeout*　受信タイムアウト時間（ミリ秒：ms）
有効範囲は、100～90000です。
有効範囲外の値を指定した場合は10000msに設定されます。

説明　本メソッドで受信タイムアウト時間の設定を行っていない場合の値は10000となります。

本メソッドは**isConnect**がtrue/falseに関わらず設定が可能です。

設定したタイムアウト時間が有効になるのは、次回データ受信時となります。

getInternationalCharacter　国際文字セットの取得

国際文字セットの値を取得します。

構文　public int **getInternationalCharacter**()

戻り値　値については、「4.2.1(2)②　国際文字セット」を参照してください。

説明　**sendDataFile**、または**setTemplateTextData**にてテキストデータを送信した場合は、下記に示す文字コードの表示結果が異なります。表示される文字に関しては、「付録A　文字セット」を参照してください。

国際文字セット設定により表示結果が異なる文字コード
0x23、0x24、0x40、0x5B、0x5C、0x5D、0x5E、0x60、0x7B、0x7C、0x7D、0x7E

setInternationalCharacter　国際文字セットの設定

国際文字セットの値を設定します。

構文　public void **setInternationalCharacter**(int *internationalCharacter*)

パラメータ　*internationalCharacter*　国際文字セット定数
設定可能な値は「4.2.1(2)②　国際文字セット」を参照してください。
無効な値を指定した場合は無視されます。

説明　本メソッドで国際文字セットの設定を行っていない場合は、Androidデバイスの言語設定により下記の国際文字セットとなります。

Androidデバイスの言語設定が日本の場合:　**COUNTRY_JAPAN**
Androidデバイスの言語設定が日本以外の場合:**COUNTRY_USA**

getCodePage　コードページの取得

コードページの値を取得します。

構文　public int **getCodePage**()

戻り値　値については、「4.2.1(2)③ コードページ」を参照してください。

setCodePage　コードページの設定

コードページの値を設定します。

構文　public void **setCodePage**(int *codePage*)

パラメータ　*codePage*　コードページ定数
設定可能な値は、「4.2.1(2)③ コードページ」を参照してください。
無効な値を指定した場合は無視されます。

説明　本メソッドでコードページの設定を行っていない場合は、Androidデバイスの言語設定により下記のコードページとなります。

Androidデバイスの言語設定が日本の場合:　**CODE_PAGE_KATAKANA**
Androidデバイスの言語設定が日本以外の場合:**CODE_PAGE_1252**

sendDataFile、または**setTemplateTextData**でテキストデータを送信した場合は、本プロパティの設定により利用するエンコードが変更されます。

getPrinterModel　デバイスモデルの取得

通信中のデバイスモデルの値を取得します。

構文　public int **getPrinterModel**()

戻り値　値については、「4.2.1(2)① デバイスモデル」を参照してください。
isConnectがfalseの場合は、**PRINTER_MODEL_DEFAULT**が返ります。

説明　デバイスが未接続でも、一度**connect**が成功している場合は、前回接続に成功したデバイスモデル値が返ります。

getPortType　接続中ポート種別の取得

デバイスとの接続に利用しているポート種別の値を取得します。

構文　public int **getPortType**()

戻り値　値については、「4.2.1(2)④ ポート種別」を参照してください。
isConnectがfalseの場合は、**PRINTER_TYPE_BLUETOOTH**が返ります。

説明　デバイスが未接続でも、一度**connect**が成功している場合は、前回接続に成功したポート種別の値が返ります。

isConnect　デバイスとの通信状態の確認

デバイスとの通信状態を確認します。

構文　public boolean **isConnect**()

戻り値　true　デバイスと通信中
false　デバイスと未接続

説明　データ送信に失敗すると、デバイスとの通信を終了し、本メソッドはfalseになります。falseの場合は、**connect**で再度通信を開始してください。

getSocketKeepingTime　ソケット維持時間の取得

ソケット維持時間を取得します。

構文　public int **getSocketKeepingTime**()

戻り値　ソケット維持時間（ミリ秒：ms）

説明　本メソッドはプリンター経由で使用している場合のみ有効です。
本メソッドは**isConnect**がtrue/falseに関わらず取得が可能です。

setSocketKeepingTime　ソケット維持時間の設定

ソケット維持時間を設定します。

構文　public void **setSocketKeepingTime**(int *socketKeepingTime*)

有効範囲　60000～300000(ミリ秒：ms)
60000未満の値を指定した場合は60000msに設定されます。
300000を超える値を指定した場合は300000msに設定されます。

初期値　300000

説明　本メソッドはプリンター経由で使用している場合のみ有効です。
本メソッドは**isConnect**がtrue/falseに関わらず設定が可能です。

ソケット維持時間には、接続するプリンターの受信タイムアウト時間設定と等しい時間を指定してください。受信タイムアウト時間設定の詳細は「プリンター用技術説明書」に記載されている、プリンターコマンド「無線LAN通信の設定」を参照してください。

設定したソケット維持時間が有効になるのは、次回**connect**実行時となります。

getVersion　SDKバージョンの取得

SDKのバージョンを文字列で取得します。

構文　public String **getVersion**()

戻り値　SDKバージョン文字列（例：SDKバージョンがVer.1.0.0の場合、戻り値は”1.0.0”となります）

説明　本メソッドは**isConnect**がtrue/falseに関わらず取得が可能です。

defragment　メモリ領域の最適化

メモリ領域を最適化します。

構文　public void **defragment**(MemoryArea *memoryArea*) throws **PrinterException**

パラメータ　*memoryArea*　メモリ領域
利用可能な定数は、「4.2.1(3)⑧ メモリ領域(MemoryArea)」を参照してください。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。
最適化には数分かかることがあります。
最適化中にデバイスの電源をオフにしないでください。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

initializeMemoryArea　メモリ領域の初期化

メモリ領域を初期化します。

構文　public void **initializeMemoryArea**(MemoryArea *memoryArea*) throws **PrinterException**

パラメータ　*memoryArea*　メモリ領域
利用可能な定数は、「4.2.1(3)⑧ メモリ領域(MemoryArea)」を参照してください。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。
初期化には数分かかることがあります。
初期化中にデバイスの電源をオフにしないでください。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

注意　**MEMORY_DISPLAY_USERMEMORY**を指定して初期化すると、下記のメソッドで登録したデータは削除されます。
- **registerTemplate**
- **registerImageData**
- **controlMacroRegistration**
- **registerUserDefinedCharacter**
- **registerOptionFont**

工場出荷時に登録されるシステムで利用する一部のデータも削除されます。
それにより、プリンターでエラーなどが発生した際にガイド画面が表示できなくなります。
使用済みのメモリ領域を再度利用する場合は、**defragment**をご利用ください。

showTemplate　テンプレートの表示

画面をテンプレート表示にします。

構文　public void **showTemplate**(int *time_ms*) throws **PrinterException**

パラメータ　*time_ms*　表示時間（ms：ミリ秒）
画面に表示する時間を*time_ms* (ms)で指定してください。
有効範囲は0～25500です。
0を超える値、かつ100未満の値を指定した場合は100msに設定されます。
25500を超える値を指定した場合は25500msに設定されます。

マクロ登録の場合：
*time_ms*に0を指定した場合は、テンプレート表示を続けます。
*time_ms*に0以外を指定した場合は、表示時間経過後は次のテンプレートに移ります。

マクロ登録以外の場合：
*time_ms*に0を指定した場合は、テンプレート表示を続けます。
*time_ms*に0以外を指定した場合は、表示時間経過後は直前のテンプレートに戻ります。直前に表示時間に0以外を指定して画面更新を行っていた場合は、0を指定して画面更新を行ったテンプレートまで戻ります。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　画面を更新し、下記のメソッドで指定されたデータを表示します。
- **selectTemplate**
- **setTemplateImageData**
- **selectTemplateTextObject**
- **setTemplateTextAlignment**
- **setTemplateTextLeftMargin**
- **setTemplateTextLineSpacing**
- **setTemplateTextBold**
- **setTemplateTextUnderline**
- **setTemplateTextSize**
- **setTemplateTextFont**
- **setTemplateTextRightSpacing**
- **setTemplateTextColor**
- **setTemplateTextData**
- **setTemplateBarcodeData**
- **setTemplateQrCodeData**

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

showSlide　スライドの表示

画面をスライド表示にします。

構文　public void **showSlide**(int *slideID,* int *time_ms*) throws **PrinterException**

パラメータ　*slideID*　スライドID
registerSlideDataで登録したスライドデータのスライドIDを指定してください。
有効範囲は0～91です。
指定したスライドIDにスライドデータが登録されていない場合は無視されます。

time_ms　表示時間（ms：ミリ秒）
画面に表示する時間を*time_ms* (ms)で指定してください。
有効範囲は0～25500です。
0を超える値、かつ100未満の値を指定した場合は100msに設定されます。
25500を超える値を指定した場合は25500msに設定されます。

マクロ登録の場合：
*time_ms*に0を指定した場合は、スライド表示を続けます。
*time_ms*に0以外を指定した場合は、表示時間経過後は次のスライドに移ります。

マクロ登録以外の場合：
*time_ms*に0を指定した場合は、スライド表示を続けます。
*time_ms*に0以外を指定した場合は、表示時間経過後は直前のスライドに戻ります。直前に表示時間に0以外を指定して画面更新を行っていた場合は、0を指定して画面更新を行ったスライドまで戻ります。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　選択中のテンプレートは解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

enterStandbyMode　スタンバイの表示

ディスプレイをスタンバイモードにします。

構文　public void **enterStandbyMode**() throws **PrinterException**

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドはスタンバイモード中、及びガイドモード中は無視されます。
選択中のテンプレートは選択が解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

executeMacro　マクロの実行

マクロを実行します。

構文　public void **executeMacro**(int *macroID,* int *repeatCount*) throws **PrinterException**

パラメータ　*macroID*　マクロID
controlMacroRegistrationで登録したマクロのマクロIDを指定してください。
有効範囲は0～127です。
指定したマクロIDにマクロが登録されていない場合は無視されます。

repeatCount　実行回数
マクロを実行する回数を指定してください。
有効範囲は0～255です。
0を指定した場合は、繰り返しを継続します。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　選択中のテンプレートは解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

turnOnScreen　画面の表示オン・オフ

画面のバックライトのオン・オフを設定します。

構文　public void **turnOnScreen**(boolean *isOn*) throws **PrinterException**

パラメータ　*isOn*　画面状態
画面状態を下記から指定してください。
true　バックライトオン
false　バックライトオフ

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。

説明　プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

selectTemplate　テンプレートの選択

ディスプレイで表示するテンプレートを選択します。
構文(a)は、テンプレートとテンプレートの背景に使用するスライドデータを選択できます。
構文(b)は、テンプレートを選択できます。

構文　(a) public void **selectTemplate**(int *templateID,* int *slideID*) throws **PrinterException**

(b) public void **selectTemplate**(int *templateID*) throws **PrinterException**

パラメータ　*templateID*　テンプレートID
選択するテンプレートのテンプレートIDを指定してください。
有効範囲は0～127です。
指定したテンプレートIDにテンプレートが登録されていない場合は無視されます。

slideID　スライドID
テンプレートの背景に使用するスライドデータのスライドIDを指定してください。
有効範囲は0～91です。
指定したスライドIDにスライドデータが登録されていない場合は無視されます。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドを実行するとテンプレート上のデータはクリアされます。
選択したテンプレートは**showTemplate**を実行すると画面に表示されます。

選択中のテンプレートは**showSlide**、**enterStandbyMode**、**executeMacro**で選択が解除されます。
選択中のテンプレートは**showTemplate**で時間指定表示を実行すると選択が解除されます。

setTemplateTextDataで20h～7EhのASCII文字以外を入力する場合、**codePage**の値に応じて下記のテンプレートを使用してください。
codePageが**CODE_PAGE_KATAKANA**の場合：
　エンコード指定がShift_JISのテンプレートを使用してください。
codePageが**CODE_PAGE_KATAKANA**以外：
　エンコード指定がISO-2022-JPのテンプレートを使用してください。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateImageData　イメージデータの設定

選択中のテンプレートに表示するイメージデータを設定します。

構文　public void **setTemplateImageData**(int *mapID,* int *imageID*) throws **PrinterException**

パラメータ　*mapID*　マップID
有効範囲は0～63です。
指定したマップIDがテンプレートに定義されていない場合は無視されます。

imageID　イメージID
registerImageDataで登録したイメージデータのイメージIDを指定してください。
有効範囲は0～63です。
指定したイメージIDにイメージデータが登録されていない場合は無視されます。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　選択中のテンプレートのマップIDを本メソッドで指定してから、配置するイメージIDを指定してください。
指定したイメージデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

テンプレートが未選択の場合は無視されます。
プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

selectTemplateTextObject　text要素の選択

編集を開始するtext要素を選択します。

構文　public void **selectTemplateTextObject**(int *mapID*) throws **PrinterException**

パラメータ　*mapID*　マップID
有効範囲は0～63です。
指定したマップIDがテンプレートに定義されていない場合は無視されます。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　選択中のテンプレートのマップIDを本メソッドで指定してから、文字の編集を開始してください。

指定したマップIDのtext要素にスクロールを設定している場合は、**showTemplate**の後に本メソッドを実行した場合、スクロールを実行します。
指定したマップIDのtext要素にスクロールを設定していない場合は、**showTemplate**の後に本メソッドを実行した場合、入力済みのテキストデータを破棄します。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**を実行した場合

テンプレートが未選択の場合は無視されます。
プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextAlignment　テキストデータの位置揃え

ディスプレイで表示するテキストデータに位置揃えを設定します。

構文　public void **setTemplateTextAlignment**(PrintAlignment *alignment*) throws **PrinterException**

パラメータ　*alignment*　位置揃え
利用可能な定数は、「4.2.1(3)⑤ 位置揃え(PrintAlignment)」を参照してください。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　位置揃えは下記の場合のみ有効です。
・**selectTemplateTextObject**で指定したマップIDにテキストデータが未入力時
・**selectTemplateTextObject**で指定したマップIDに入力したテキストデータが改行直後

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextLeftMargin　テキストデータの左マージンの設定

ディスプレイで表示するテキストデータに左マージンを設定します。

構文　public void **setTemplateTextLeftMargin**(int *margin*) throws **PrinterException**

パラメータ　*margin*　左マージン（ピクセル：px）
有効範囲は0～479です。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　左マージンの設定は下記の場合のみ有効です。
・**selectTemplateTextObject**で指定したマップIDにテキストデータが未入力時
・**selectTemplateTextObject**で指定したマップIDに入力したテキストデータが改行直後

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextLineSpacing　テキストデータの改行量の設定

ディスプレイで表示するテキストデータに改行量を設定します。

構文　public void **setTemplateTextLineSpacing**(int *spacing*) throws **PrinterException**

パラメータ　*spacing*　改行量(ピクセル:px)
有効範囲は0～255です。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　改行量の設定は下記の場合のみ有効です。
・**selectTemplateTextObject**で指定したマップIDにテキストデータが未入力時
・**selectTemplateTextObject**で指定したマップIDに入力したテキストデータが改行直後

指定したマップIDのtext要素にスクロールを設定している場合は、本メソッドの設定は反映されません。

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextBold　テキストデータの強調印字の設定

ディスプレイで表示するテキストデータに強調印字を設定します。

構文　public void **setTemplateTextBold**(CharacterBold *bold*) throws **PrinterException**

パラメータ　*bold*　強調印字
利用可能な定数は、「4.2.1(3)① 強調印字(CharacterBold)」を参照してください。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッド実行後のテキストデータから強調印字は適用されます。1文字ずつ強調印字の設定ができます。

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextUnderline　テキストデータのアンダーラインの設定

ディスプレイで表示するテキストデータにアンダーラインを設定します。

構文　public void **setTemplateTextUnderline**(CharacterUnderline *underline*) throws **PrinterException**

パラメータ　*underline*　アンダーライン
利用可能な定数は、「4.2.1(3)② アンダーライン(CharacterUnderline)」を参照してください。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッド実行後のテキストデータからアンダーラインは適用されます。1文字ずつアンダーラインの設定ができます。

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextSize　テキストデータの文字サイズの設定

ディスプレイで表示するテキストデータに文字サイズを設定します。

構文　public void **setTemplateTextSize**(CharacterScale *scale*) throws **PrinterException**

パラメータ　*scale*　文字倍率
利用可能な定数は、「4.2.1(3)④ 文字倍率(CharacterScale)」を参照してください。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッド実行後のテキストデータから文字サイズは適用されます。1文字ずつ文字サイズの設定ができます。

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextFont　テキストデータの文字フォントの設定

ディスプレイで表示するテキストデータに文字フォントを設定します。

構文　public void **setTemplateTextFont**(CharacterFont *font*) throws **PrinterException**

パラメータ　*font*　文字フォント
利用可能な定数は、「4.2.1(3)③ 文字フォント(CharacterFont)」を参照してください。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッド実行後のテキストデータから文字フォントは適用されます。1文字ずつ文字フォントの設定ができます。

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextRegisteredFont　テキストデータの登録フォントの設定

ディスプレイで表示するテキストデータで使用する登録フォントを設定します。

構文　public void **setTemplateTextRegisteredFont**(RegisteredFont *font*) throws **PrinterException**

パラメータ　*font*　登録フォント
利用可能な定数は、「4.2.1(3)⑨ 登録フォント(RegisteredFont)」を参照してください。
オプションフォントが登録されていない場合は無視されます。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッド実行後のテキストデータから登録フォントは適用されます。1文字ずつ登録フォントの設定ができます。

本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextRightSpacing　テキストデータの右スペース量の設定

ディスプレイで表示するテキストデータに右スペース量を設定します。

構文　public void **setTemplateTextRightSpacing**(int *spacing*) throws **PrinterException**

パラメータ　*spacing*　文字の右スペース量（ピクセル：px）
有効範囲は0～255です。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッド実行後のテキストデータから文字の右スペース量は適用されます。1文字ずつ右スペース量の設定ができます。

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextColor　テキストデータの文字色の設定

ディスプレイで表示するテキストデータで使用する文字色を設定します。

構文　public void **setTemplateTextColor**(int *color*) throws **PrinterException**

パラメータ　*color*　文字色
有効範囲は0～0xffffffです。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドでテキストデータに文字色を設定できます。
文字色はRGB24bitカラーで設定できます。設定した文字色はRGB555の16bitカラーに減色して表示されます。

本メソッド実行後のテキストデータから文字色は適用されます。1文字ずつ文字色の設定ができます。

本メソッドを実行する前に**selectTemplateTextObject**でマップIDを指定してください。
本メソッドを実行した後に**setTemplateTextData**でテキストデータを入力してください。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドの設定は下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateTextData　テキストデータの入力

ディスプレイで表示するテキストデータを入力します。

構文　public void **setTemplateTextData**(String *text*) throws **PrinterException**

パラメータ　*text*　ディスプレイで表示するテキストデータ
一度に指定可能なデータサイズは1～1020バイトです。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドは、入力されたテキストデータを、**internationalCharacter**、及び**codePage**の設定を基に表示可能なテキストデータにエンコードして画面に表示します。

本メソッドを実行する前に**selectTemplate**でテンプレートを選択してください。
選択中のテンプレートのマップIDを**selectTemplateTextObject**で指定してから、本メソッドでテキストデータを入力してください。
selectTemplateTextObjectでマップIDが指定されていない場合は、無視されます。
入力したテキストデータは**showTemplate**を実行すると画面に表示されます。

本メソッドで入力したテキストデータは下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateBarcodeData　バーコードデータの入力

選択中のテンプレートのbarcode要素のマップIDを指定し、バーコードデータを入力します。
構文(a)は、表示するバーコードにデータを文字列で入力できます。
構文(b)は、表示するバーコードにデータをバイト配列で入力できます。

構文　(a) public void **setTemplateBarcodeData**(int *mapID,* String *text*) throws **PrinterException**

(b) public void **setTemplateBarcodeData**(int *mapID,* byte [] *data*) throws **PrinterException**

パラメータ　*mapID*　マップID
有効範囲は0～7です。
指定したマップIDが選択中のテンプレートに定義されていない場合は無視されます。

text　バーコードデータ
入力可能な文字は、英数字('0'～'9'、'A'～'Z'、'a'～'z')などのASCII文字コード20h(スペース)～7Eh(チルダ)の文字です。
入力可能な文字数は1～150文字です。
バーコードの仕様に沿わないバーコードデータは無視されます。

data　バーコードデータ
入力可能な値は00h～7Fhです。
入力可能なデータ数は1～150です。
バーコードの仕様に沿わないバーコードデータは無視されます。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　選択中のテンプレートのマップIDを本メソッドで指定してから、バーコードデータを入力してください。
入力したバーコードデータは**showTemplate**を実行すると画面に表示されます。

本メソッドで入力したバーコードデータは下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

テンプレートが未選択の場合は無視されます。
プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

setTemplateQrCodeData　QRコードデータの入力

選択中のテンプレートのqr要素のマップIDを指定し、QRコードデータを入力します。
構文(a)は、QRコードデータを文字列で入力できます。
構文(b)は、QRコードデータを文字列で入力できます。*moduleSize*、*errorCorrection*、*mode*、及び*qrQuietZone*は選択中のテンプレートの設定が反映されます。
構文(c)は、QRコードデータをバイト配列で入力できます。
構文(d)は、QRコードデータをバイト配列で入力できます。*moduleSize*、*errorCorrection*、*mode*、及び*qrQuietZone*は選択中のテンプレートの設定が反映されます。

構文　(a) public void **setTemplateQrCodeData**(int *mapID,*
ModuleSize *moduleSize,*
ErrorCorrection *errorCorrection,*
QrDataMode *mode,*
QrQuietZone *qrQuietZone,*
String *text*) throws **PrinterException**

(b) public void **setTemplateQrCodeData**(int *mapID,* String *text*) throws **PrinterException**

(c) public void **setTemplateQrCodeData**(int *mapID,*
ModuleSize *moduleSize,*
ErrorCorrection *errorCorrection,*
QrDataMode *mode,*
QrQuietZone *qrQuietZone,*
byte [] *data*) throws **PrinterException**

(d) public void **setTemplateQrCodeData**(int *mapID,* byte [] *data*) throws **PrinterException**

パラメータ　*mapID*　マップID
有効範囲は0～7です。
指定したマップIDが選択中のテンプレートに定義されていない場合は無視されます。

moduleSize　　モジュールサイズ
利用可能な定数は、「4.2.1(3)⑥ モジュールサイズ(ModuleSize)」を参照してください。

errorCorrection　　エラー訂正レベル
利用可能な定数は、「4.2.1(3)⑦ エラー訂正レベル(ErrorCorrection)」を参照してください。

mode　　データモード
利用可能な定数は、「4.2.1(3)⑩ QRデータモード(QrDataMode)」を参照してください。

qrQuietZone　　クワイエットゾーン
利用可能な定数は、「4.2.1(3)⑪ QRクワイエットゾーン(QrQuietZone)」を参照してください。

text　　QRコードデータ
入力可能な文字は下記の通りです。
・英数字('0'～'9'、'A'～'Z'、'a'～'z')などのASCII文字コード
20h(スペース)～7Eh(チルダ)の文字
・JIS X 0201に基づく8ビットラテン/カナ文字
・JIS X 0208に基づくシフトJISコード
入力可能なデータサイズは1～3909バイトです。
QRコードの仕様に沿わないQRコードデータは無視されます。

data　　QRコードデータ
ディスプレイで表示するQRコードデータ
入力可能な値は00h～FFhです。
入力可能なデータ数は1～3909です。
QRコードの仕様に沿わないQRコードデータは無視されます。

例外　　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　　選択中のテンプレートのマップIDを本メソッドで指定してから、QRコードデータを入力してください。
入力したQRコードデータは**showTemplate**を実行すると画面に表示されます。

本メソッドで入力したQRコードデータは下記の条件でクリアされます。
・**selectTemplate**を実行した場合
・**showTemplate**の*time_ms*で0以外を指定し、指定した表示時間を経過した場合
・**executeMacro**に登録されている**showTemplate**を実行した場合

テンプレートが未選択の場合は無視されます。
プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

registerTemplate　テンプレートの登録

テンプレートを登録します。

構文　public void **registerTemplate**(int *templateID,*
String *label,*
String *filePath*) throws **PrinterException**

パラメータ　*templateID*　テンプレートID
登録するテンプレートのテンプレートIDを指定してください。
有効範囲は0～127です。
テンプレートID=127はシステムで利用していますので、指定しないでください。

label　テンプレート名
登録するテンプレートに識別用の名前を付けることができます。
指定可能な文字は、英数字('0'～'9'、'A'　～'Z'、'a'～'z')などの
ASCII文字コード20h(スペース)～7Eh(チルダ)の文字です。
ユニコードの00A5h ('¥')は用いないでください。
指定可能な文字数は0～32文字です。
*label*は省略することも可能です。省略する場合はnullを指定してください。
指定したテンプレート名は**getDisplayResponse**で取得できます。

filePath　ディスプレイに登録するテンプレートデータのファイルパス
入力可能な形式について下記に説明します。

- Javaの標準的なクラス「java.io.File」が扱う絶対パス文字列
  Android 10 (API 29)以降をアプリのターゲットとする場合には、一部のファイルが直接扱えなくなりますのでご注意ください。
  詳細については、「3.5 注意事項 – 対象範囲別ストレージについて」を参照してください。
- Androidに用意されたクラス「android.net.Uri」が扱う下記スキーム名のURI文字列
  ・file://
  ・content://
  本パラメータには"ストレージ アクセス フレームワーク"から取得したURI文字列を指定する必要があります。"ストレージ アクセス フレームワーク"から取得せずに作成されたURIはファイルを開けない可能性がありますのでご注意ください。

サポートするファイルの拡張子は.xmlです。
登録可能な最大データサイズは8192バイトです。
登録するテンプレートデータの詳細については「ディスプレイ用技術説明書」を参照してください。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

注意　**工場出荷時に登録されるデータは、品質向上のため予告無く追加・変更される場合があります。**

unregisterTemplate　テンプレートの削除

登録済みのテンプレートを削除します。

構文　public void **unregisterTemplate**(int *templateID*) throws **PrinterException**

パラメータ　*templateID*　テンプレートID
削除するテンプレートのテンプレートIDを指定してください。
有効範囲は0～127です。
指定したテンプレートIDにテンプレートが登録されていない場合は無視されます。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

テンプレートを削除しても使用済みのメモリは解放されません。使用済みのメモリは、**defragment**を実行すると再度利用できます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

registerImageData　イメージデータの登録

イメージデータを登録します。

構文　public void **registerImageData**(int *imageID,*
String *label,*
String *filePath*) throws **PrinterException**

パラメータ　*imageID*　イメージID
有効範囲は0～63です。
イメージID=49～63はシステムで利用していますので、指定しないでください。

label　　イメージ名
登録するイメージデータに識別用の名前を付けることができます。
指定可能な文字は、英数字('0'～'9'、'A' ～'Z'、'a'～'z')などの
ASCII文字コード20h(スペース)～7Eh(チルダ)の文字です。
ユニコードの00A5h ('¥')は用いないでください。
指定可能な文字数は0～32文字です。
*label*は省略することも可能です。省略する場合はnullを指定してください。
指定したイメージ名は**getDisplayResponse**で取得できます。

filePath　　ファイルパス
入力可能な形式について下記に説明します。

- Javaの標準的なクラス「java.io.File」が扱う絶対パス文字列
  Android 10 (API 29)以降をアプリのターゲットとする場合には、一部のファイルが直接扱えなくなりますのでご注意ください。
  詳細については、「3.5 注意事項 – 対象範囲別ストレージについて」を参照してください。

- Androidに用意されたクラス「android.net.Uri」が扱う下記スキーム名のURI文字列
  ・file://
  ・content://
  本パラメータには"ストレージ アクセス フレームワーク"から取得したURI文字列を指定する必要があります。"ストレージ アクセス フレームワーク"から取得せずに作成されたURIはファイルを開けない可能性がありますのでご注意ください。

登録するイメージデータのファイル名を指定してください。
サポートするファイルの拡張子は.jpg、.jpeg、及び.pngです。
ただし、サポートする拡張子であっても、フォーマットによっては登録できない場合があります。
指定可能な最大ファイルサイズは786432バイトです。
登録できる最大データサイズは横480×縦272ピクセル(px)です。

例外　　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

注意　　**工場出荷時に登録されるデータは、品質向上のため予告無く追加・変更される場合があります。**

unregisterImageData　イメージデータの削除

登録済みのイメージデータを削除します。

構文　public void **unregisterImageData**(int *imageID*) throws **PrinterException**

パラメータ　*imageID*　イメージID
有効範囲は0～63です。
指定したイメージIDにイメージデータが登録されていない場合は無視されます。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

イメージデータを削除しても使用済みのメモリは解放されません。使用済みのメモリは、**defragment**を実行すると再度利用できます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

registerSlideData　スライドデータの登録

スライドデータを登録します。

構文　public void **registerSlideData**(int *slideID,*
String *label,*
String *filePath*) throws **PrinterException**

パラメータ　*slideID*　スライドID
有効範囲は0～91です。
スライドID=80～90はシステムで利用していますので、指定しないでください。

label　スライド名
登録するスライドデータに識別用の名前を付けることができます。
指定可能な文字は、英数字('0'～'9'、'A'～'Z'、'a'～'z')などのASCII文字コード20h(スペース)～7Eh(チルダ)の文字です。
ユニコードの00A5h('¥')は用いないでください。
指定可能な文字数は0～32文字です。
*label*は省略することも可能です。省略する場合はnullを指定してください。
指定したスライド名は**getDisplayResponse**で取得できます。

filePath　　ファイルパス
入力可能な形式について下記に説明します。

- Javaの標準的なクラス「java.io.File」が扱う絶対パス文字列
  Android 10 (API 29)以降をアプリのターゲットとする場合には、一部のファイルが直接扱えなくなりますのでご注意ください。
  詳細については、「3.5 注意事項 – 対象範囲別ストレージについて」を参照してください。

- Androidに用意されたクラス「android.net.Uri」が扱う下記スキーム名のURI文字列
  ・file://
  ・content://
  本パラメータには"ストレージ アクセス フレームワーク"から取得したURI文字列を指定する必要があります。"ストレージ アクセス フレームワーク"から取得せずに作成されたURIはファイルを開けない可能性がありますのでご注意ください。

登録するスライドデータのファイル名を指定してください。
サポートするファイルの拡張子は.jpg、.jpeg、及び.pngです。
ただし、サポートする拡張子であっても、フォーマットによっては登録できない場合があります。
指定可能な最大ファイルサイズは786432バイトです。
登録できるデータサイズは横480×縦272ピクセル(px)固定です。

例外　　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

登録したスライドデータを表示する場合は**showSlide**を実行してください。
登録したスライドデータをテンプレートの背景として使用する場合は**selectTemplate**でスライドIDを指定してください。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

注意　　**工場出荷時に登録されるデータは、品質向上のため予告無く追加・変更される場合があります。**

unregisterSlideData　スライドデータの削除

登録済みのスライドデータを削除します。

構文　public void **unregisterSlideData**(int *slideID*) throws **PrinterException**

パラメータ　*slideID*　スライドID
有効範囲は0～91です。
指定したスライドIDにスライドデータが登録されていない場合は無視されます。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

registerUserDefinedCharacter　外字の登録

ディスプレイに外字を登録します。

構文　public void **registerUserDefinedCharacter**(String *filePath*) throws **PrinterException**

パラメータ　*filePath*　ファイルパス
入力可能な形式について下記に説明します。

- Javaの標準的なクラス「java.io.File」が扱う絶対パス文字列
  Android 10 (API 29)以降をアプリのターゲットとする場合には、一部のファイルが直接扱えなくなりますのでご注意ください。
  詳細については、「3.5 注意事項 - 対象範囲別ストレージについて」を参照してください。

- Androidに用意されたクラス「android.net.Uri」が扱う下記スキーム名のURI文字列
  ・file://
  ・content://
  本パラメータには"ストレージ アクセス フレームワーク"から取得したURI文字列を指定する必要があります。"ストレージ アクセス フレームワーク"から取得せずに作成されたURIはファイルを開けない可能性がありますのでご注意ください。

登録する外字のファイル名を指定してください。
サポートするファイルの拡張子は.binです。
外字データについては、「ディスプレイ用技術説明書」のディスプレイコマンド「外字の登録」を参照してください。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。
外字が登録済みの状態で本メソッドを実行した場合は、外字は上書きされます。

外字の表示にはエンコード指定がShift_JISのテンプレートを使用してください。
外字を表示する前に**setCodePage**の*codePage*に**CODE_PAGE_KATAKANA**を指定してください。
外字を表示する際は**setTemplateTextData**の*textl*に指定可能な文字コードを指定してください。
指定可能な文字コードはE000h～E05Dhです。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

unregisterUserDefinedCharacter　外字の削除

登録済みの外字を削除します。

構文　public void **unregisterUserDefinedCharacter**() throws **PrinterException**

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

登録されている外字は全て削除されます。
外字を削除しても使用済みのメモリは解放されません。使用済みのメモリは、**defragment**を実行すると再度利用できます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

registerOptionFont　オプションフォントの登録

ディスプレイにオプションフォントを登録します。

構文　public void **registerOptionFont**(int *startCode,*
int *endCode,*
int *width,*
int *height,*
String *filePath*) throws **PrinterException**

パラメータ　*startCode*　登録開始文字コード
有効範囲は20h～FFhのASCII文字コードです。

endCode　　登録終了文字コード
有効範囲は20h～FFhのASCII文字コードです。

width　　文字幅（ピクセル：px）
有効範囲は1～255です。

height　　文字高さ（ピクセル：px）
有効範囲は1～255です。

filePath　　ファイルパス
入力可能な形式について下記に説明します。

- Javaの標準的なクラス「java.io.File」が扱う絶対パス文字列
  Android 10 (API 29)以降をアプリのターゲットとする場合には、一部のファイルが直接扱えなくなりますのでご注意ください。
  詳細については、「3.5 注意事項 – 対象範囲別ストレージについて」を参照してください。

- Androidに用意されたクラス「android.net.Uri」が扱う下記スキーム名のURI文字列
  ・file://
  ・content://
  本パラメータには"ストレージ アクセス フレームワーク"から取得したURI文字列を指定する必要があります。"ストレージ アクセス フレームワーク"から取得せずに作成されたURIはファイルを開けない可能性がありますのでご注意ください。

登録するオプションフォントのファイル名を指定してください。
サポートするファイルの拡張子は.binです。
オプションフォントデータについては、「ディスプレイ用技術説明書」のディスプレイコマンド「オプションフォントの登録」を参照してください。

例外　　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

オプションフォントが登録済みの状態で本メソッドを実行した場合は、登録済みのオプションフォントは削除され、新たにメモリ領域を確保してオプションフォントを登録します。
登録済みのオプションフォントが削除されても使用済みのメモリは解放されません。使用済みのメモリは、**defragment**を実行すると再度利用できます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

unregisterOptionFont　オプションフォントの削除

登録済みのオプションフォントを削除します。

構文　public void **unregisterOptionFont**() throws **PrinterException**

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

登録されているオプションフォントは全て削除されます。
オプションフォントを削除しても使用済みのメモリは解放されません。使用済みのメモリは、**defragment**を実行すると再度利用できます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

controlMacroRegistration　マクロ登録の開始・終了

マクロ登録の開始・終了を指定します。

構文　public void **controlMacroRegistration**(int *macroID,*
MacroRegistrationFunction *control*) throws **PrinterException**

パラメータ　*macroID*　マクロID
有効範囲は-1～127です。
マクロID=120～126はシステムで利用していますので、指定しないでください。

control　マクロ登録処理
利用可能な定数は、
「4.2.1(3)⑫ マクロ登録処理(MacroRegistrationFunction)」を参照してください。

例外　**PrinterException**
本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

説明　マクロ登録処理の手順は下記の通りです。
① マクロ登録処理を開始します。
*macroID*に-1を指定してください。
*control*に**MACRO_REGISTRATION_START**を指定してください。

② メソッドを実行します。
マクロ登録処理対象メソッドの場合、送信データのバッファリングを開始します。
バッファリング中に実行したマクロ登録処理対象メソッドの送信データは、デバイスへ送信せずにマクロデータバッファーにバッファリングします。
バッファリングできる最大送信データのサイズは1024バイトです。
バッファリングされた送信データが最大サイズを超える場合、超えた時点のマクロ登録対象メソッドがエラーとなります。
エラーとなった場合は登録中のデータは破棄され、マクロモードが解除されます。
保持された送信データは、手順③でマクロ登録処理を終了してください。
マクロ登録処理対象以外のメソッドの場合は、送信データはバッファリングされずに即実行されます。

マクロ登録処理対象メソッドを下記に示します。

- **showTemplate**
- **showSlide**
- **selectTemplate**
- **setTemplateImageData**
- **selectTemplateTextObject**
- **setTemplateTextAlignment**
- **setTemplateTextLeftMargin**
- **setTemplateTextLineSpacing**
- **setTemplateTextBold**
- **setTemplateTextUnderline**
- **setTemplateTextSize**
- **setTemplateTextFont**
- **setTemplateTextRegisteredFont**
- **setTemplateTextRightSpacing**
- **setTemplateTextColor**
- **setTemplateTextData**
- **setTemplateBarcodeData**
- **setTemplateQrCodeData**

③ マクロ登録処理を終了します。
*macroID*に登録したいマクロID（0～127）を指定してください。
*control*に**MACRO_REGISTRATION_REGIST**を指定するとバッファリングされた送信データをデバイスへ送信します。バッファリングされた送信データはデバイス送信後も保持されます。
本メソッドを実行するとディスプレイはスタンバイモードになります。
選択中のテンプレートは選択が解除されます。

保持されている送信データは下記のいずれかにより破棄されます。
- **MACRO_REGISTRATION_CLEAR**を指定する。
- **MACRO_REGISTRATION_START**を指定する。
- **disconnect**を実行する。

登録したマクロは**executeMacro**で実行できます。

登録済みのマクロを削除する手順は下記の通りです。

*control*に**MACRO_REGISTRATION_START**を指定し、*macroID*に-1を指定して本メソッドを呼び出してください。

*control*に**MACRO_REGISTRATION_REGIST**を指定し、*macroID*に削除するマクロIDを指定して本メソッドを呼び出してください。

本メソッドを実行するとディスプレイはスタンバイモードになります。選択中のテンプレートは選択が解除されます。

プリンター経由で使用する場合、プリンターにディスプレイが未接続の場合は無視されます。

注意　　工場出荷時に登録されるデータは、品質向上のため予告無く追加・変更される場合があります。

getDisplayResponse　　ディスプレイからの各種応答取得

ディスプレイからの応答データを取得します。

構文　public void **getDisplayResponse**(int *id,* int *param,* Object *buf)* throws **PrinterException**

パラメータ　*id*　ディスプレイ応答種別定数
利用可能な定数は、「4.2.1(2)⑤ ディスプレイ応答種別」を参照してください。

param　コマンドパラメータ
ディスプレイ応答種別定数によりバッファーの型が異なります。
指定する値の説明は、下表を参照してください。

buf　取得した応答データを格納するバッファー
*id*で指定した応答データを*buf*で指定されたオブジェクトに格納します。
ディスプレイ応答種別定数によりバッファーの型が異なります。
バッファーの型は、下表を参照してください。

応答種別定数	
パラメータ	説明
DISPLAY_RESPONSE_REQUEST（実行応答リクエスト）	
param	int型で0～15（00h～0Fh）を指定してください。
buf	長さ1のint型の配列を指定してください。 *buf[0]*には0～15(00h～0Fh)を指定してください。 応答取得成功時には、*buf[0]*に実行応答リクエストの応答コードが64～79(40h～4Fh)で格納されます。
DISPLAY_RESPONSE_USER_AREA（ユーザ領域の残り容量の送信）	
param	int型で0を指定してください。
buf	長さ1のint型の配列を指定してください。 応答取得成功時には、ユーザ領域の残り容量(単位：バイト)が数値で格納されます。

応答種別定数	
パラメータ	説明
DISPLAY_RESPONSE_TEMPLATE_ID_LIST（テンプレートIDの送信）	
param	int型で0を指定してください。
buf	ArrayList<Integer>型の配列を指定してください。 応答取得成功時には、登録済みのテンプレートIDがintの配列で格納されます。
DISPLAY_RESPONSE_IMAGE_ID_LIST（イメージIDの送信）	
param	int型で0を指定してください。
buf	ArrayList<Integer>型の配列を指定してください。 応答取得成功時には、登録済みのイメージIDがintの配列で格納されます。
DISPLAY_RESPONSE_SLIDE_ID_LIST（スライドIDの送信）	
param	int型で0を指定してください。
buf	ArrayList<Integer>型の配列を指定してください。 応答取得成功時には、登録済みのスライドIDがintの配列で格納されます。
DISPLAY_RESPONSE_TEMPLATE_LABEL（テンプレート名の送信）	
param	int型で0～127（00h～7Fh）を指定してください。
buf	ArrayList<String>型の配列を指定してください。 応答取得成功時には、テンプレートの登録時に指定したテンプレート名が文字列で格納されます。
DISPLAY_RESPONSE_IMAGE_LABEL（イメージ名の送信）	
param	int型で0～63（00h～3Fh）を指定してください。
buf	ArrayList<String>型の配列を指定してください。 応答取得成功時には、イメージデータの登録時に指定したイメージ名が文字列で格納されます。
DISPLAY_RESPONSE_SLIDE_LABEL（スライド名の送信）	
param	int型で0～91（00h～5bh）を指定してください。
buf	ArrayList<String>型の配列を指定してください。 応答取得成功時には、スライドデータの登録時に指定したスライド名が文字列で格納されます。

例外　**PrinterException**

本メソッド呼び出し時にエラーが発生した場合は、**PrinterException**をスローします。
エラーについての詳細は、「**4.2.5 PrinterExceptionクラス**」を参照してください。
データ送信に失敗すると、デバイスとの通信を終了し、**PrinterException**をスローすることがあります。デバイスとの通信状態の確認は、**isConnect**を参照してください。

4.2.2 PrinterEventクラス

PrinterEventクラスは、**startDiscoveryPrinter**、または**startDiscoveryDevice**が終了した際に発生する終了イベントを取得するクラスです。

(1) メソッド一覧

PrinterEventクラスで提供されるメソッドを下記に示します。

名前	説明
getEventType	終了イベントの取得

(2) 終了イベント定数

終了イベントの取得で利用する定数を下記に示します。

定数名	説明	値
EVENT_FINISHED_DISCOVERY	**startDiscoveryPrinter**、または **startDiscoveryDevice**の終了	1

(3) メソッド詳細

getEventType 終了イベントの取得

startDiscoveryPrinter、または**startDiscoveryDevice**が終了した際の終了イベントを取得します。

構文　public int **getEventType**()

戻り値　値については、「4.2.2(2) 終了イベント定数」を参照してください。

説明　終了イベントの判定条件を下記に示します。
・**startDiscoveryPrinter**が終了
・**startDiscoveryDevice**が終了
デバイスが発見できなかった場合も**EVENT_FINISHED_DISCOVERY**が返ります。

4.2.3 PrinterListenerインターフェイス

PrinterListenerインターフェイスは、**startDiscoveryPrinter**、または**startDiscoveryDevice**が終了した際の終了イベントを取得するためのインターフェイスです。

(1) メソッド一覧

PrinterListenerインターフェイスのメソッドを下記に示します。

名前	説明
finishEvent	デバイス探索の終了イベント

(2) メソッド詳細

finishEvent デバイス探索の終了イベント

startDiscoveryPrinter、または**startDiscoveryDevice**が終了した場合に呼び出される終了イベントです。

構文 public void **finishEvent**(**PrinterEvent** *event*)

パラメータ *event* 終了イベント
PrinterEventクラスで指定されます。

説明 インターフェイスのため実装はありません。
本メソッドを**startDiscoveryPrinter**、または**startDiscoveryDevice**の終了によって、終了イベントの通知を受け取るユーザアプリケーションに実装してください。**PrinterEvent**クラスの**getEventType**により、終了イベントの種別を判定してください。

4.2.4 PrinterInfoクラス

PrinterInfoクラスは、**startDiscoveryPrinter**、または**startDiscoveryDevice**で発見されたデバイスのデバイス情報を格納するクラスです。

(1) メソッド一覧

ポート名を取得できます。**PrinterInfo**クラスで提供されるメソッドの一覧を下記に示します。

名前	説明
getDevicePath	デバイスパスの取得

(2) メソッド詳細

getDevicePath デバイスパスの取得

startDiscoveryPrinter、または**startDiscoveryDevice**で発見されたデバイスのデバイス情報からUSBデバイスファイルのパスの文字列を取得します。

構文　public String **getDevicePath**()

戻り値　デバイスパス

4.2.5 PrinterExceptionクラス

(1) メソッド一覧

PrinterExceptionクラスで提供されるメソッドの一覧を下記に示します。

名前	説明
PrinterException	コンストラクタ
getErrorCode	エラーコードの取得

(2) 定数一覧

① エラーコード

エラーコードの取得で利用される定数を下記に示します。

定数名	説明	値
ERROR_ACCESS_DENIED	ハンドルの取得に失敗しました。*1	-1
	利用できないポートが指定されました。	
	サポートされていないメソッドが指定されました。	
ERROR_SHARING_VIOLATION	すでにオープン済みのポートが指定されました。	-11
ERROR_PORT_NOT_OPENED	ポートがオープンされていません。	-12
ERROR_DEVICE_NOT_CONNECTED	Androidデバイスとプリンターの接続に問題があります。	-21
ERROR_OFFLINE	切断状態またはプリンターがオフラインです。	-22
ERROR_EXTERNAL_DEVICE_NOT_CONNECTED	ディスプレイが接続されていません。	-23
ERROR_DEVICE_INITIALIZE_FAILED	プリンターの設定変更に失敗しました。プリンターへのデータ送信が送信タイムアウト時間内に完了してないか、プリンターからのデータ受信が受信タイムアウト時間内に完了していません。	-31
ERROR_DATA_SIZE_ZERO	データサイズが0バイトのデータが指定されました。	-101
ERROR_OVER_MAX_DATA_SIZE	データサイズが最大値を超過しています。	-102
ERROR_DATA_SIZE_INVALID	既定のデータサイズではありません。	-103
ERROR_ENCODE_FAILED	テキストデータのエンコードでエラーが発生しました。*1	-111
ERROR_TIMEOUT	送信タイムアウトが発生しました。	-201
	受信タイムアウトが発生しました。	
ERROR_FILE_NOT_FOUND	指定されたファイルが存在しません。	-301
ERROR_FILE_USED	指定されたファイルは別プロセスで使用中です。	-302
ERROR_FILE_INVALID	指定されたファイルは無効です。	-303
ERROR_LOW_MEMORY	イメージデータファイルの読み込みでメモリ不足が発生しました。	-311

定数名	説明	値
ERROR_OVER_MAX_IMAGE	イメージデータの幅と高さのいずれか、または両方が印字可能な最大ドット数を超過しています。	-312
ERROR_LOGO_NOT_DEFINED	ロゴが登録されていません。	-313
ERROR_LOW_USER_AREA	ユーザ領域の残り容量が不足しています。	-401
ERROR_LOW_EXTERNAL_RAM	RAMの残り容量が不足しています。	-402
ERROR_NOT_REGISTERD	テンプレートが登録されていません。 イメージデータが登録されていません。 スライドデータが登録されていません。 オプションフォントが登録されていません。 外字が登録されていません。	-403
ERROR_NOT_UNREGISTERD	テンプレートが削除されていません。 イメージデータが削除されていません。 スライドデータが削除されていません。 オプションフォントが削除されていません。 外字が削除されていません。	-404
ERROR_INVALID_NO	指定されたロゴIDの値は無効です。	-501
ERROR_INVALID_DATA	指定されたデータは無効です。	-503
ERROR_INVALID_PARAM	指定されたパラメータは無効です。	-9999

*1: 異常処理の可能性があります。

(3) メソッド詳細

PrinterException　コンストラクタ

com.seikoinstruments.sdk.thermalprinter.PrinterExceptionクラスのコンストラクタです。

構文　public **PrinterException**(int *code*, String *message*)

getErrorCode　エラーコードの取得

スローされた例外のエラーコードを取得します。

構文　public int **getErrorCode**()

戻り値　エラーコードの内容は、「4.2.5(2) 定数一覧」を参照してください。

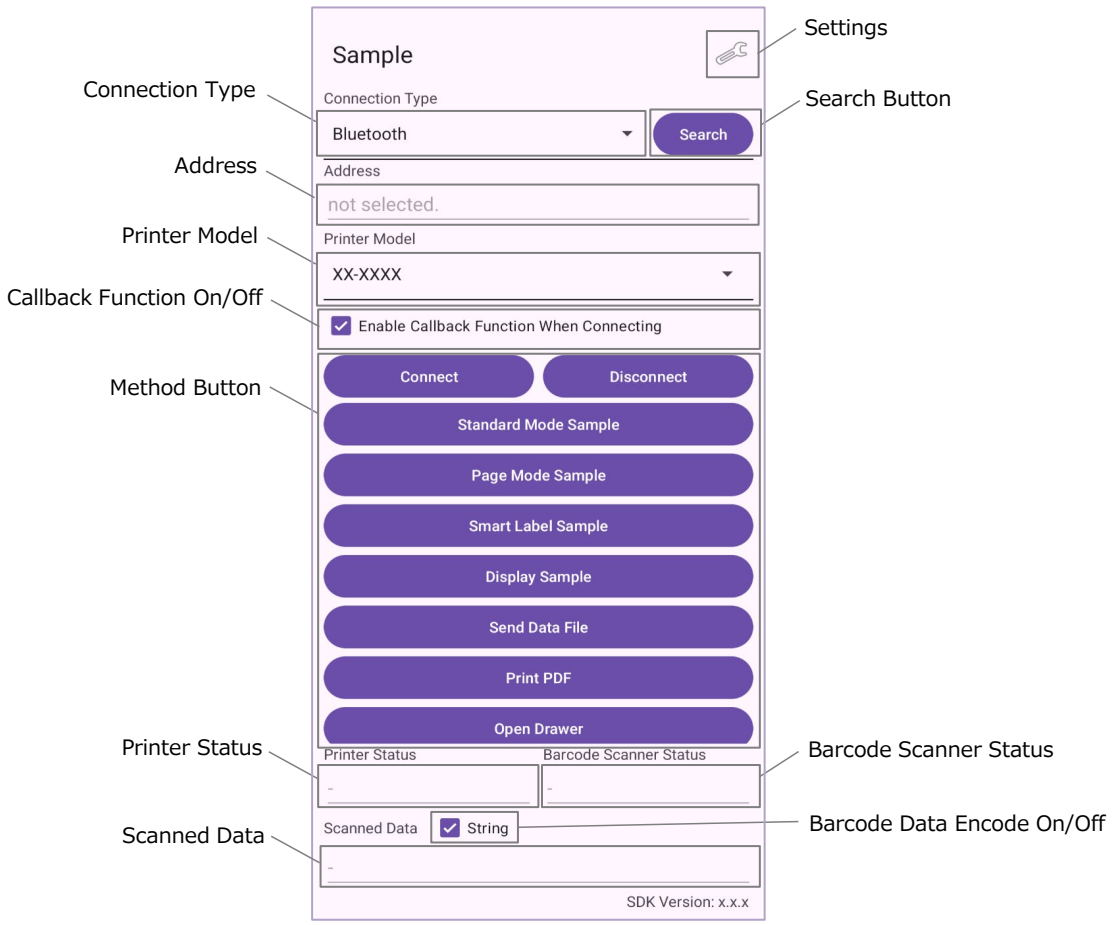
5章　サンプルプログラム

本章では、SIIプリントクラスライブラリで提供するサンプルプログラムについて説明します。

5.1　画面構成

SIIプリントクラスライブラリには、Android Studioのプロジェクト形式のサンプルプログラムが含まれています。
サンプルプログラムの画面について説明します。

5.1.1　メイン画面

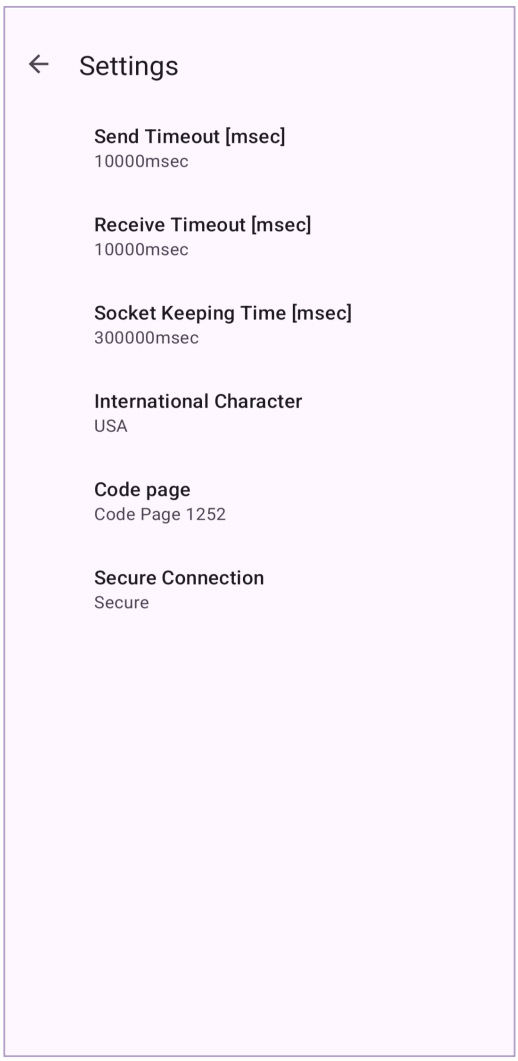


項目	説明
Settings	[Settings]ボタンをタップすると、機能設定画面が開きます。メイン画面に戻る場合は、画面左上の ← ボタンをタップしてください。
Connection Type	プリンターへの接続種別を選択します。
Search Button	[Connection Type]で指定した種別のプリンターの探索を開始します。 プリンター探索画面に遷移します。探索されたプリンターは一覧に表示されます。探索済みのプリンターをタップするとプリンターが選択され、メイン画面に戻ります。
Address	選択したプリンターの情報を表示します。
Printer Model	プリンターモデルを指定します。
Callback Function On/Off	プリンターとの接続時にコールバック機能を有効にするか選択します。 On：接続時にコールバックを開始します。 Off：コールバックは応答しません。
Method Button*1*2	**connect**及び**disconnect**を行うメソッドボタンの他、各メソッドを組み合わせたサンプルの印字や周辺デバイスの動作確認ができます。
Printer Status	プリンターステータスを表示します。 [Callback Function On/Off]がOnの場合は、最新のプリンターステータスが表示されます。
Barcode Scanner Status	バーコードスキャナーの接続状態を表示します。 DSP-A01はバーコードスキャナーをサポートしていません。
Barcode Data Encode On/Off	バーコードスキャナーで読み取ったバーコードデータの変換を選択します。 DSP-A01はバーコードスキャナーをサポートしていません。
Scanned Data	バーコードスキャナーで読み取ったバーコードデータを表示します。 DSP-A01はバーコードスキャナーをサポートしていません。

*1: サポートする機能は機種により異なります。サポートしている機能以外は動作しません。
*2: プリンター経由で使用する場合のみ有効

5.1.2 [Settings]画面

[Settings]では各種設定機能を表示します。



5.2 注意事項

サンプルプログラムは予告なしに変更される場合があります。

サンプルプログラムに対しての動作保証及びサポートは行っておりません。

付録 A　文字セット

A.1 コードページ表(文字コードテーブル)

国際文字セットでCOUNTRY_USA設定時のコードページを下記に示します。
設定する国際文字セットにより、特定の文字コードの表示結果が異なります。
特定の文字コードについては、「A.2 国際文字セット」を参照してください。

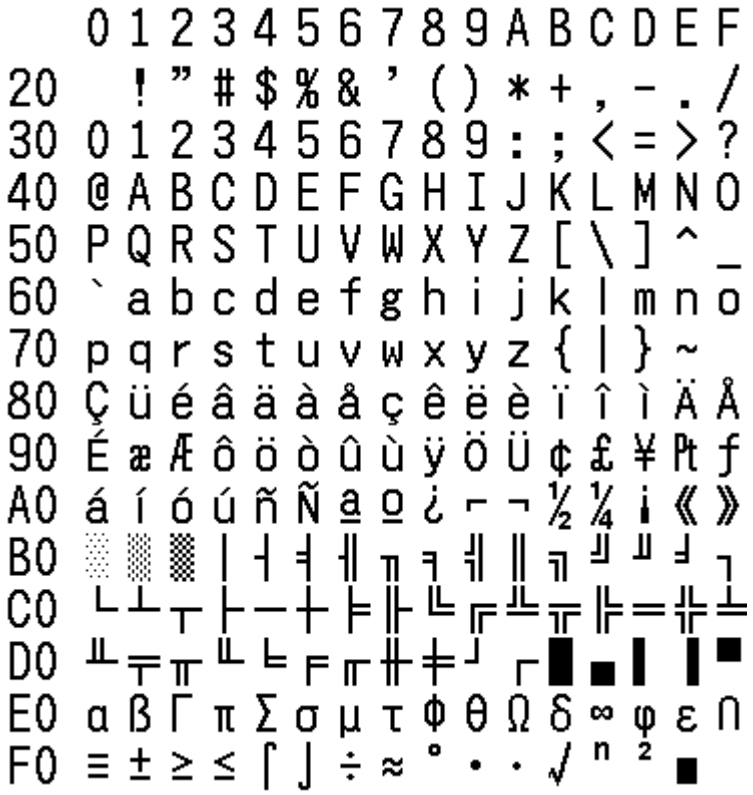


図 A-1　**CODE_PAGE_437** (USA, Standard Europe)

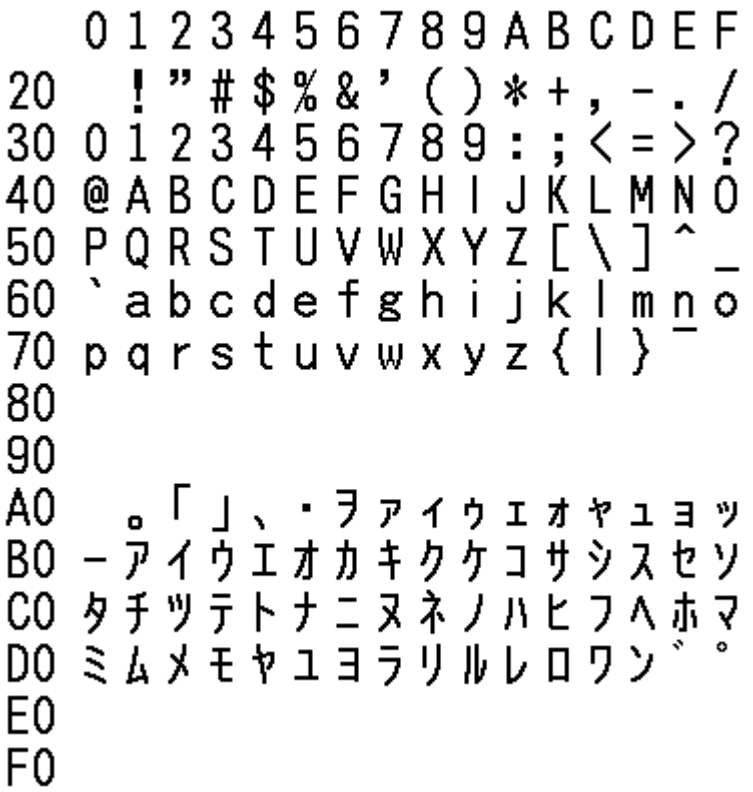


図 A-2 **CODE_PAGE_KATAKANA**

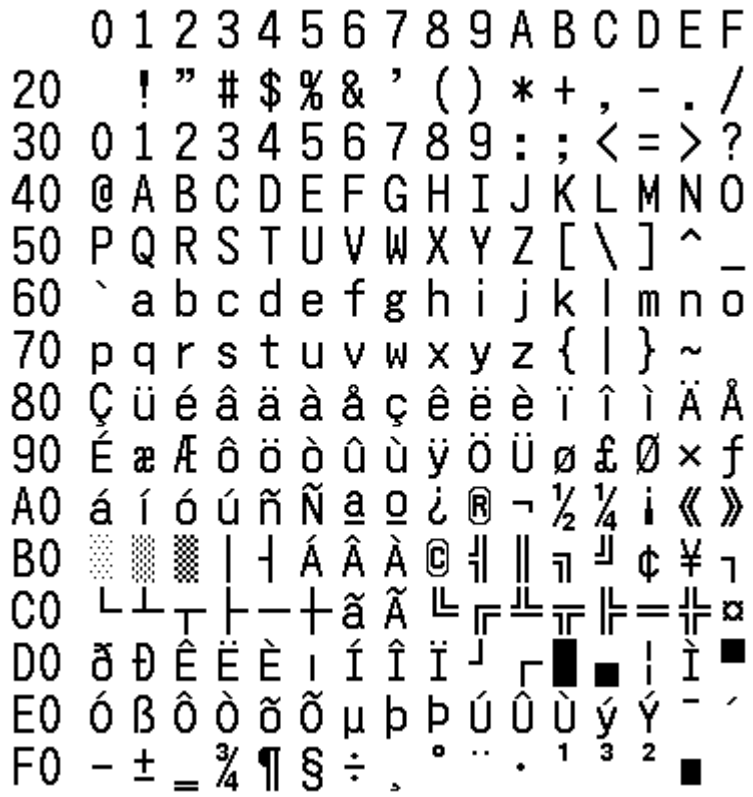


図 A-3 **CODE_PAGE_850** (Multilingual)

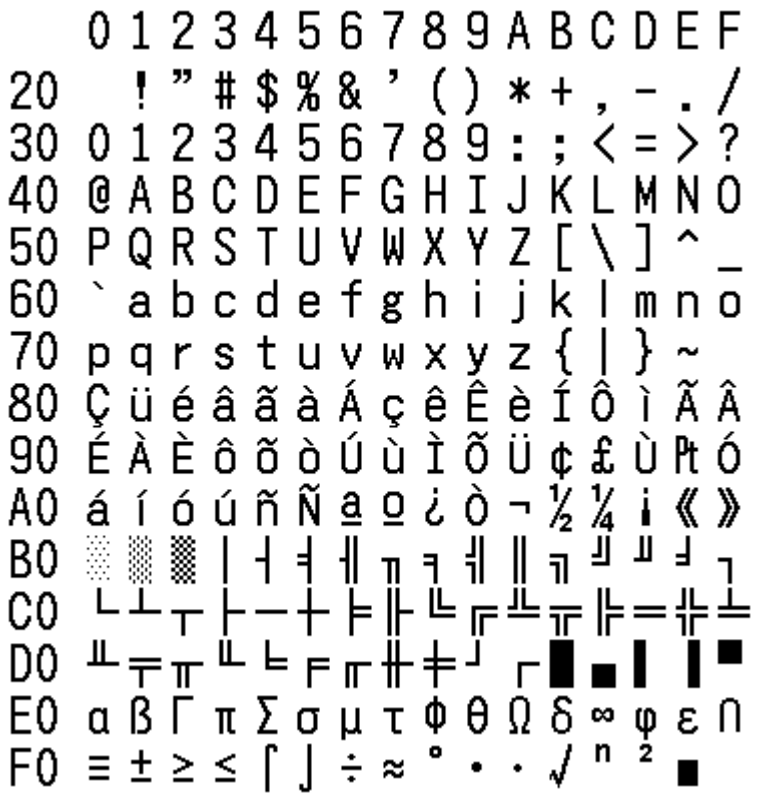


図 A-4 **CODE_PAGE_860** (Portuguese)

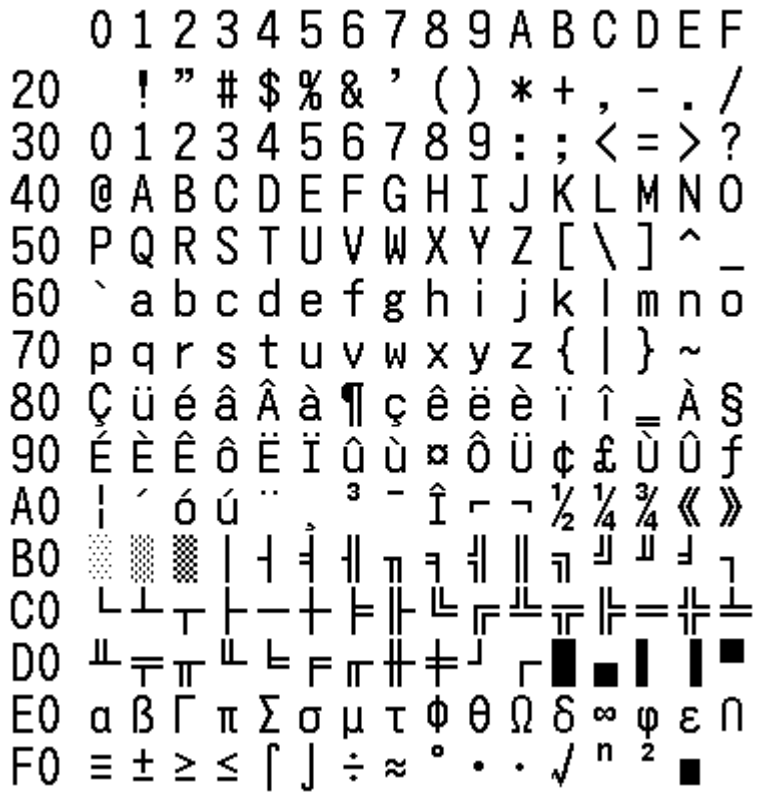


図 A-5 **CODE_PAGE_863** (Canadian-French)

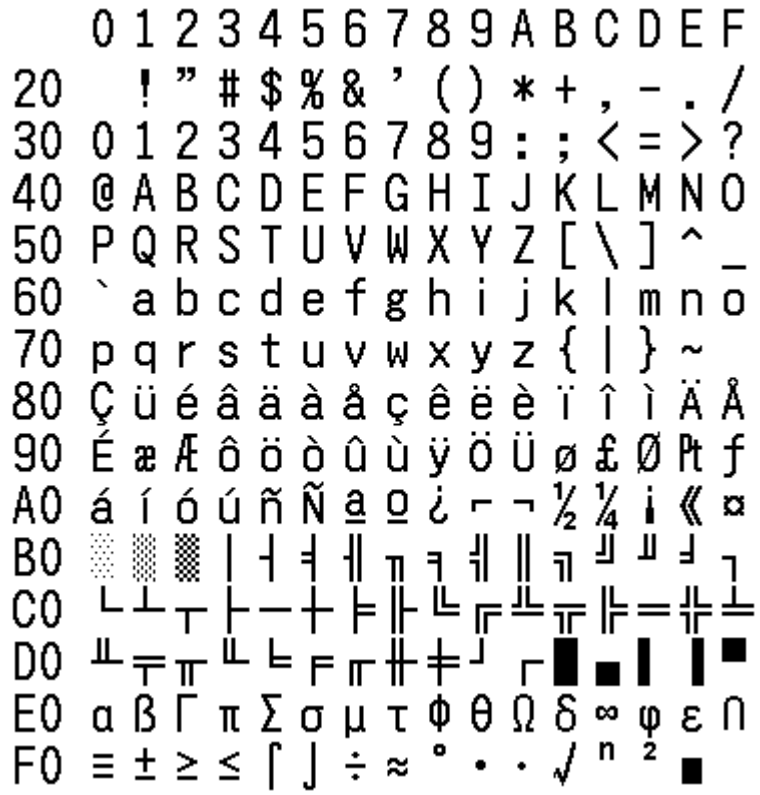


図 A-6 **CODE_PAGE_865** (Nordic)

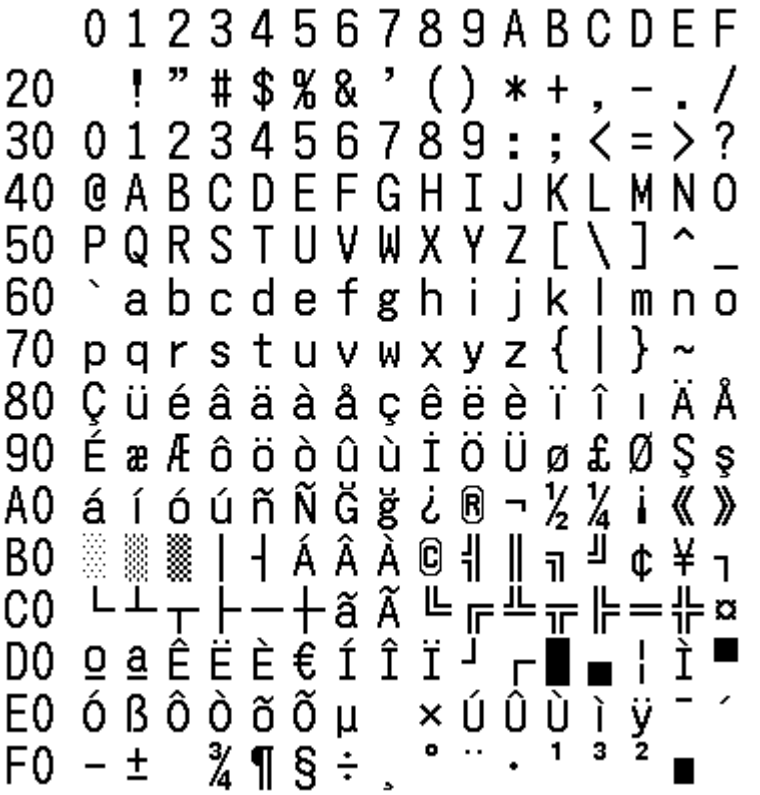


図 A-7 **CODE_PAGE_857** (Turkish)

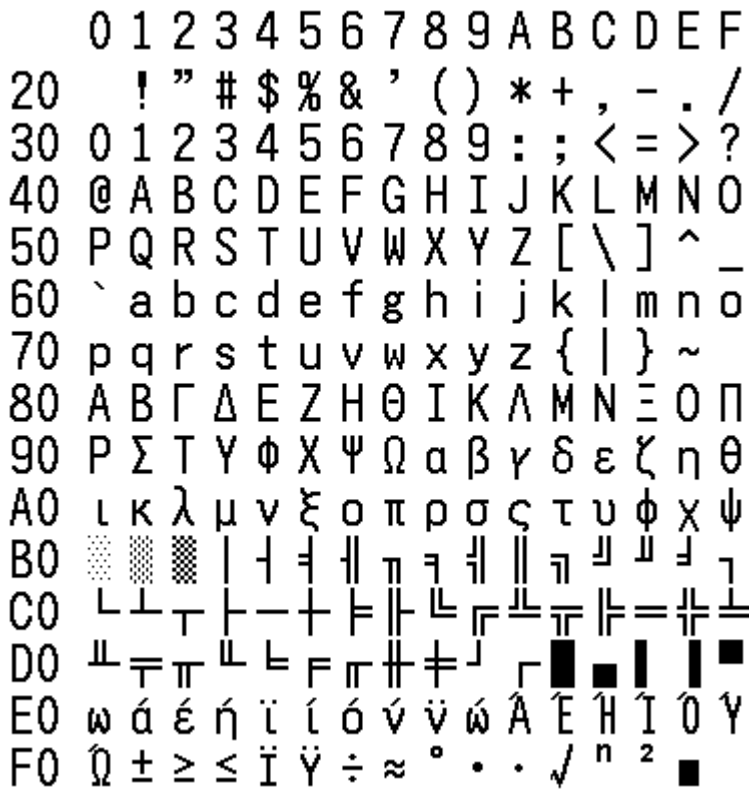


図 A-8 **CODE_PAGE_737** (Greek)

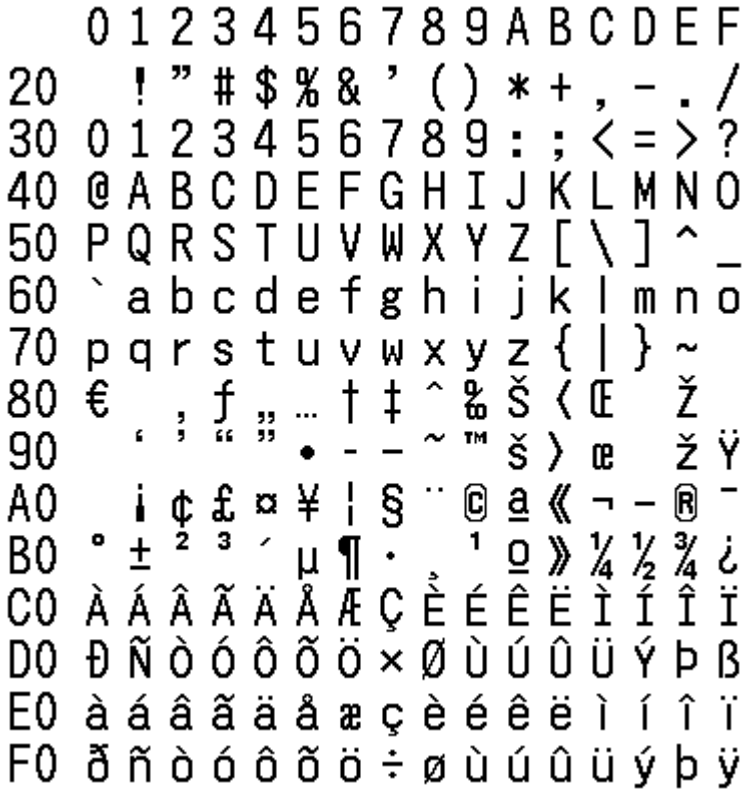


図 A-9 **CODE_PAGE_1252** (Latin)

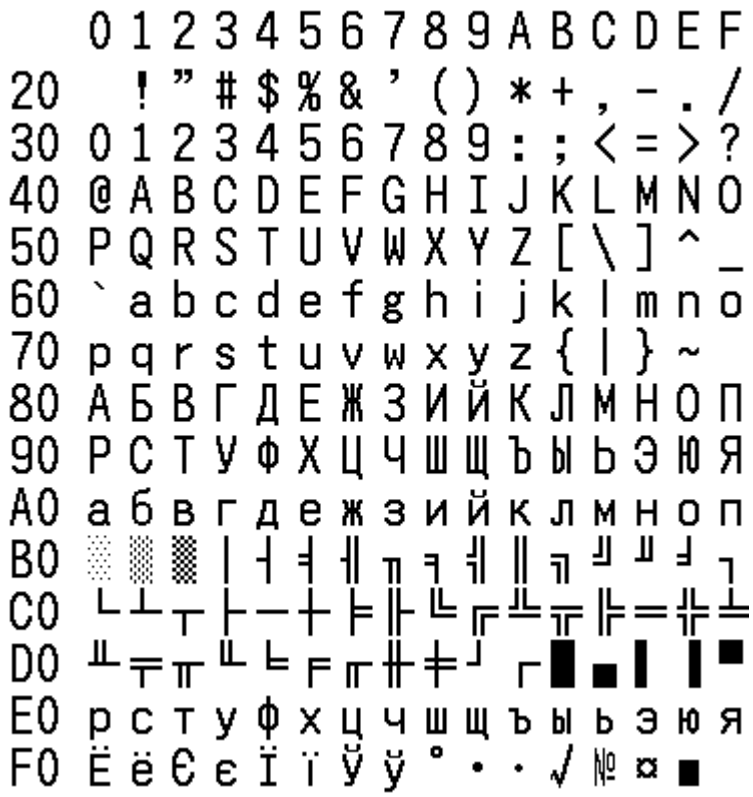


図 A-10 **CODE_PAGE_866** (Russian)

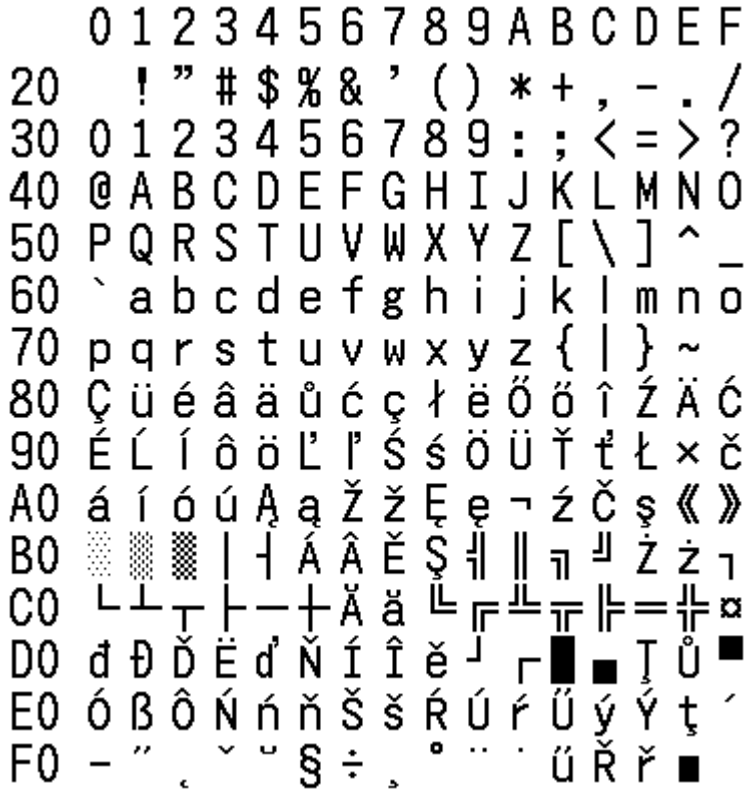


図 A-11 **CODE_PAGE_852** (Eastern Europe)

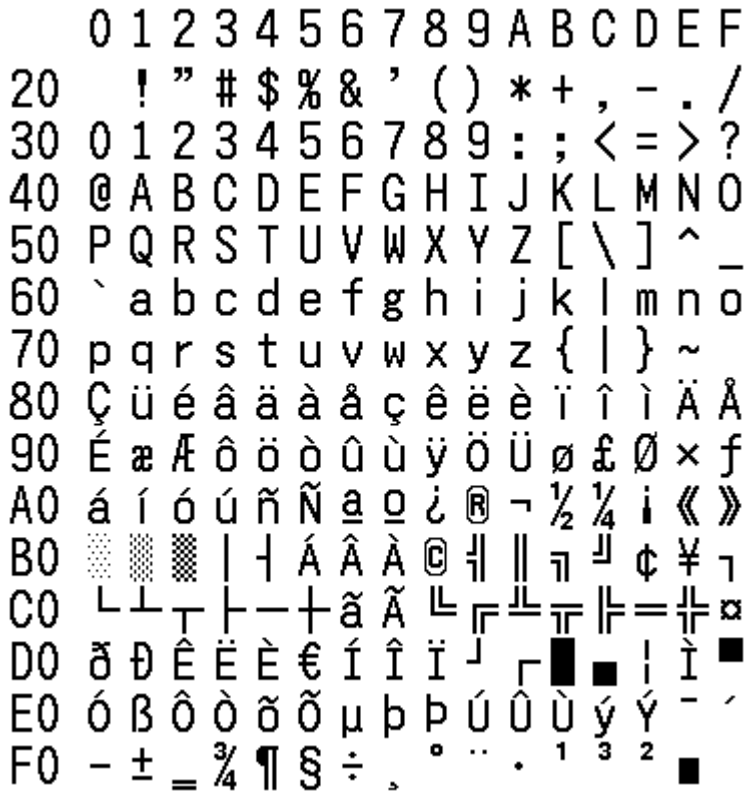


図 A-12 **CODE_PAGE_858** (Euro)

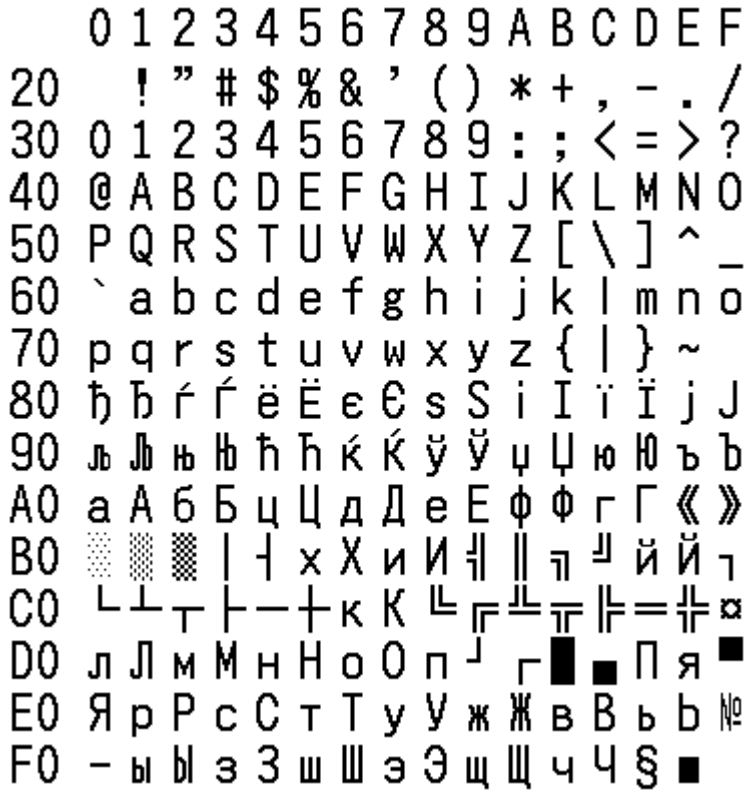


図 A-13 **CODE_PAGE_855** (Cyrillic)

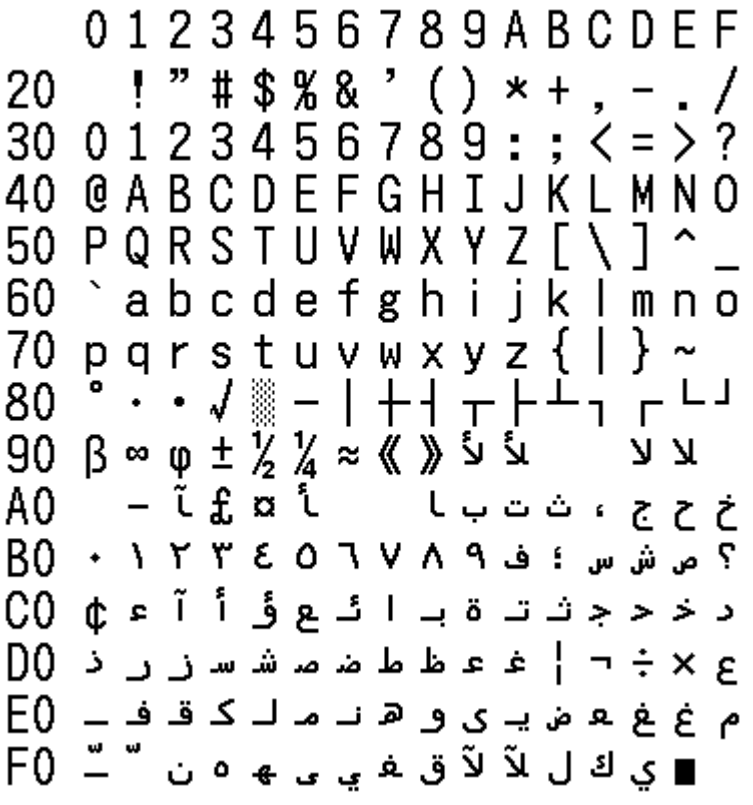


図 A-14 **CODE_PAGE_864** (Arabic)



図 A-15 **CODE_PAGE_1250** (Central European)



図 A-16 **CODE_PAGE_1251** (Cyrillic)

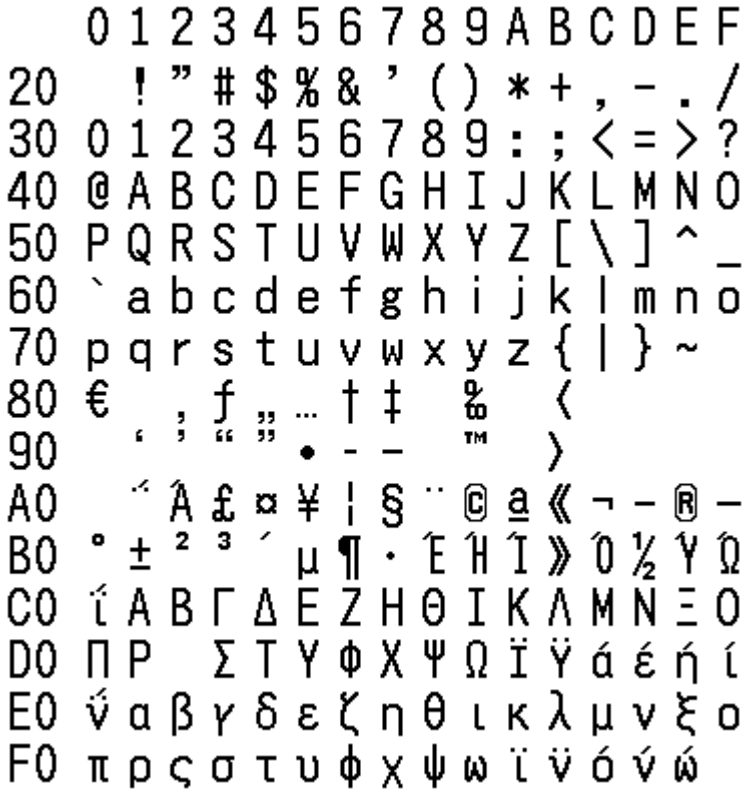


図 A-17 **CODE_PAGE_1253** (Greek)



図 A-18 **CODE_PAGE_1254** (Turkish)

A.2 国際文字セット

設定する国際文字セットにより、特定の文字コードの表示結果が異なります。
特定の文字コード、及びその表示結果を下記に示します。

	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
COUNTRY_USA	#	$	@	[	\	]	^	`	{	|	}	~
COUNTRY_FRANCE	#	$	à	°	ç	§	^	`	é	ù	è	¨
COUNTRY_GERMANY	#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
COUNTRY_ENGLAND	£	$	@	[	\	]	^	`	{	|	}	~
COUNTRY_DENMARK_1	#	$	@	Æ	Ø	Å	^	`	æ	ø	å	~
COUNTRY_SWEDEN	#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
COUNTRY_ITALY	#	$	@	°	\	é	^	ù	à	ò	è	ì
COUNTRY_SPAIN	₧	$	@	¡	Ñ	¿	^	`	¨	ñ	}	~
COUNTRY_JAPAN	#	$	@	[	¥	]	^	`	{	|	}	~
COUNTRY_NORWAY	#	¤	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
COUNTRY_DENMARK_2	#	$	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
COUNTRY_SPAIN_2	#	$	á	¡	Ñ	¿	é	`	í	ñ	ó	ú
COUNTRY_LATIN_AMERICA	#	$	á	¡	Ñ	¿	é	ü	í	ñ	ó	ú
COUNTRY_ARABIA	#	$	@	[	\	]	^	`	{	|	}	~

図 A-19 国際文字セット