

# Label Printer

---

LPAPI.js Interface Description

Android Version

V1.2

**DothanTech**

**3<sup>rd</sup> Dec. 2019**

## Content

Content.....	1
Document Modification Records .....	3
1. Overview.....	4
2. Brief Description of Use.....	4
2.1 Import the LPAPI.jar package in the native project.....	4
2.2 Create the label printing interface in the Activity where the WebView control resides .....	4
2.3 Call the LPAPI.js interface in user code .....	4
3. Connect the printer.....	5
3.1 getAllPrinters Return all the specified models of printers in the pair list as a string .....	5
3.2 openPrinter Open a printer of a specified name or model asynchronously...	5
3.3 openPrinterSync Open a printer of a specified name or model synchronously	6
3.4 getPrinterName Get the name of the connected printer .....	6
3.5 isPrinterOpened Determine whether the printer is connected .....	6
3.6 cancel Cancel print job .....	6
3.7 closePrinter Close the currently connected printer.....	7
3.8 reopenPrinter Reopen the last connected printer asynchronously .....	7
3.9 reopenPrinterSync Reopen the last connected printer synchronously.....	7
4. Method instructions for print jobs and pages .....	8
4.1 startJob Start draw task .....	8
4.2 abortJob Cancel draw task .....	8
4.3 endJob End draw task.....	8
4.4 commitJob Submit data to print .....	8
4.5 startPage Start drawing a page .....	9
4.6 endPage End drawing a page.....	9
5. Set or acquire print content parameters .....	9
5.1 setItemOrientation Set clockwise rotation angle of the subsequent drawings ...	9
5.2 getItemOrientation Get clockwise rotation of the current drawing.....	10
5.3 setItemHorizontalAlignment Set horizontal alignment of subsequent drawings .....	10
5.4 getItemHorizontalAlignment Get horizontal alignment of current drawing .....	10

5.5 setItemVerticalAlignment Set vertical alignment of subsequent drawings.....	10
5.6 getItemVerticalAlignment Get vertical alignment of current drawing .....	11
5.7 setItemPenAlignment Set Line Brush Alignment .....	11
5.8 getItemPenAlignment Get Line Brush Alignment.....	11
5.9 setDrawParam Set values of drawing-related parameters.....	12
6. Description of draw text strings.....	12
6.1 Uniform fonts used for text drawing .....	12
6.2 drawText Draw text strings.....	13
7. Description of draw Barcode .....	13
7.1 draw1Dbarcode Draw Barcode.....	14
7.2 draw2DQRCode Draw QR Code.....	15
7.3 draw2DPdf417 Draw PDF417 QR Code.....	15
8. Description of draw vector graphics .....	16
8.1 drawRectangle Draw a rectangular box with the specified line width.....	16
8.2 fillRectangle Draw a fill rectangular box with the specified line width.....	16
8.3 drawRoundRectangle Draw a rounded rectangular box with the specified line width.....	17
8.4 fillRoundRectangle Draw a fill rounded rectangular box with the specified line width.....	17
8.5 drawEllipse Draw an ellipse with the specified line width .....	18
8.6 fillEllipse Draw a fill ellipse with the specified line width .....	18
8.7 drawCircle Draw a circle centered on the specified position.....	19
8.8 fillCircle Draw fill circle.....	19
8.9 drawLine Draw line (straight/slash) .....	20
8.10 drawDashLine Draw dotted line.....	20
9. Description of draw image .....	21
9.1 drawImage Draw an image from specified file .....	21
9.2 drawImageWithActualSize Draw the actual size of the given image .....	22
9.3 drawImageWithThreshold Draw image with grayscale thresholds.....	22

**Document Modification Records**

SN	Version	Descriptions	Modifier	Date
1	V0.1	Document Creation	Hu Dianxing	2017-09-09
2	V0.2	Documentation Improvement	Yang Lingmei	2017-10-10
3	V1.0	Document Format Improvement	Yang Lingmei	2017-11-11
4	V1.1	Add setDrawParam	Yang Lingmei	2018-03-20
5	V1.2	Modify Document Format	Yang Lingmei	2019-12-03

## 1. Overview

LPAPI.js interface method description document, is based on the native java docking package through the JavaScript secondary encapsulation, to facilitate the user in the secondary development of the interface to call, shorten the development cycle, speed up the development process.

LPAPI.js interface provides the caller with easy-to-use methods for drawing labels. The interface provides methods for drawing text, barcodes, QR Codes, images, and graphics, as well as the ability to rotate the drawing object and the page

## 2. Brief Description of Use

The basic call flow of LPAPI interface is simply as follows:

### 2.1 Import the LPAPI.jar package in the native project

### 2.2 Create the label printing interface in the Activity where the WebView control resides

```
private LPAPI mPrinter;  
  
private WebView mWeb;  
  
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
  
    mWeb = (WebView) findViewById(R.id.webviewer);  
  
    ...  
  
    // Initialize label print interface before WebView loads html file.  
    mPrinter = LPAPI.Factory.createInstance(mWeb);  
  
    // Load html file  
    mWeb.loadUrl("file:///android_asset/index.html");  
}
```

### 2.3 Call the LPAPI.js interface in user code

Import the LPAPI.js file where it needs to be called, and call the LPAPI.js method where it needs to be printed, as shown below.

```
// Get LPAPI Object  
  
var api = new LPAPI ();  
  
// Connect the first printer in the pair  
  
api.openPrinter("");
```

```
// Start draw task, pass parameters (page width, page height)
```

```
api.startJob(40, 30, 0);
```

```
// Starts a page draw, draw text strings
```

```
api.drawText(content, 4, 5, 40, 30, 4, 0);
```

```
// End draw task to submit it for printing
```

```
api.commitJob();
```

### 3. Connect the printer

#### 3.1 getAllPrinters Return all the specified models of printers in the pair list as a string

Function	Return all the specified models of printers in the pair list as a string.	
Method	<b>LPAPI.prototype.getAllPrinters = function (printerName);</b>	
Parameter	printerName	Printer model filter, printer model number (e.g., DT20S) or printer name (DT20S-71000001); Multiple printer models or printer names can be specified, separated by a comma ",", e.g., "DT20S, DP20S"; If not specified, all paired printer names are returned.
Return Value	<a href="#">Printer Name List String multiple printer names separated by a comma ",".</a>	
Instructions	The list of printer names needs to be separated by the user according to comma ",".	
Note		

#### 3.2 openPrinter Open a printer of a specified name or model asynchronously

Function	Open a printer of a specified name or model asynchronously.	
Method	<b>LPAPI.prototype.openPrinter = function (printerName);</b>	
Parameter	printerName	The parameters can be data of the following types: 1. The name displayed on the printer screen: used to open the printer with the specified name; 2. Printer model (name prefix): to open the first printer of the corresponding model in the pairing list; null or "", to open the first printer in the pairing list;
Return Value	Successful or not	
Instructions	Connect the printer.	
Note	This method, or a method in the way, must be called before starting a print job.	

**3.3 openPrinterSync** Open a printer of a specified name or model synchronously

Function	Open a printer of a specified name or model synchronously	
Method	LPAPI.prototype.openPrinterSync = function (printerName);	
Parameter	printerName	The parameters can be data of the following types: 3. The name displayed on the printer screen: used to open the printer with the specified name; 4. Printer model (name prefix): to open the first printer of the corresponding model in the pairing list; 5. null or "", to open the first printer in the pairing list;
Return Value	Successful or not	
Instructions		
Note	Make sure the printer is turned on and the pairing operation is performed.	

**3.4 getPrinterName** Get the name of the connected printer

Function	Get the name of the connected printer.	
Method	LPAPI.prototype.getPrinterName = function ();	
Parameter	Null	
Return Value	<a href="#">Return the name of the corresponding printer if it is connected, otherwise return null.</a>	
Instructions		
Note		

**3.5 isPrinterOpened** Determine whether the printer is connected

Function	Determine whether the printer is connected.	
Method	LPAPI.prototype.isPrinterOpened = function ();	
Parameter	Null	
Return Value	<a href="#">true: indicates that the printer is connected and ready to print;</a> <a href="#">false: indicates that the printer is not connected and you need to connect the printer before printing.</a> <a href="#">The printer needs to be connected before printing.</a>	
Instructions		
Note		

**3.6 cancel** Cancel print job

Function	Cancel print job	
Method	LPAPI.prototype.cancel = function ();	
Parameter	Null	
Return Value	<a href="#">Null</a>	

<b>Instructions</b>	Cancel a print job that has been committed (by executing the commitJob function), but has not yet been started;
<b>Note</b>	

### 3.7 closePrinter Close the currently connected printer

<b>Function</b>	Close the currently connected printer.
<b>Method</b>	LPAPI.prototype.closePrinter = function ();
<b>Parameter</b>	Null
<b>Return Value</b>	<a href="#">Null</a>
<b>Instructions</b>	If multiple users need to share the same printer, this function can be used to disconnect the printer after printing is complete, otherwise other users will not be able to connect to the printer;
<b>Note</b>	Try to avoid calling this function before printing is complete, or it may cause the print job to terminate;

### 3.8 reopenPrinter Reopen the last connected printer asynchronously

<b>Function</b>	Reopen the last connected printer asynchronously.
<b>Method</b>	LPAPI.prototype.reopenPrinter = function ();
<b>Parameter</b>	Null
<b>Return Value</b>	<a href="#">Successful or not</a>
<b>Instructions</b>	
<b>Note</b>	Return success only means that the operation was submitted, it does not mean that the connection was successful, the result of the connection will be notified through the callback function. The specific connection result will be notified through the callback function.

### 3.9 reopenPrinterSync Reopen the last connected printer synchronously

<b>Function</b>	Reopen the last connected printer synchronously.
<b>Method</b>	LPAPI.prototype.reopenPrinterSync = function ();
<b>Parameter</b>	Null
<b>Return Value</b>	<a href="#">Successful or not</a>
<b>Instructions</b>	
<b>Note</b>	



## 4. Method instructions for print jobs and pages

### 4.1 startJob Start draw task

Function	Start draw task.	
Method	LPAPI.prototype.startJob = function (width, height, orientation);	
Parameter	width	Label width (based on the draw view, label rotation is not taken into account. In mm).
	height	Label height (based on the draw view, label rotation is not taken into account. In mm).
	orientation	Clockwise rotation angle of the label page of the draw task, 0/90/180/270, default value 0.
Return Value	<a href="#">Successful or not</a>	
Instructions	This method is called when starting a draw task.	
Note	When starting a draw task, all previous tasks/data will be discarded.	

### 4.2 abortJob Cancel draw task

Function	Cancel draw task.	
Method	LPAPI.prototype.abortJob = function ();	
Parameter	Null	
Return Value	<a href="#">Successful or not</a>	
Note	The current task/data will be completely discarded, but the parameter settings will be retained.	

### 4.3 endJob End draw task

Function	End draw task.	
Method	LPAPI.prototype.endJob = function ();	
Parameter	Null	
Return Value	Null	

### 4.4 commitJob Submit data to print

Function	Submit data to print	
Method	LPAPI.prototype.commitJob = function ();	
Parameter	Null	
Return Value	<a href="#">Successful or not</a>	

**4.5 startPage Start drawing a page**

<b>Function</b>	Start drawing a page.	
<b>Method</b>	LPAPI.prototype.startPage = function ();	
<b>Parameter</b>	Null	
<b>Return Value</b>	<a href="#">Successful or not</a>	

**4.6 endPage End drawing a page**

<b>Function</b>	End drawing a page.	
<b>Method</b>	LPAPI.prototype.endPage = function ();	
<b>Parameter</b>	Null	
<b>Return Value</b>	Null	

**5. Set or acquire print content parameters**

[ItemAlignment](#) Styles of drawing barcodes

Value	Description
0	Horizontal left/vertical top
1	Horizontal Centering / Vertical Centering
2	Horizontal right/vertical bottom
3	Sub-elements are aligned in the same way as objects and are currently used for horizontal alignment of barcode text.

[PenAlignment](#) Styles of drawing fonts

Value	Description
0	The line drawn is centered at the specified position
1	The drawn line is inside the specified position

**5.1 setItemOrientation Set clockwise rotation angle of the subsequent drawings**

<b>Function</b>	Set clockwise rotation angle of the subsequent drawings.	
<b>Method</b>	LPAPI.prototype.setItemOrientation = function (orientation);	
<b>Parameter</b>	orientation	Clockwise rotation angle of the subsequently drawn content (0/90/180/270), default value 0.
<b>Return Value</b>	Null	

**5.2 getItemOrientation** Get clockwise rotation of the current drawing

Function	Get clockwise rotation of the current drawing.	
Method	LPAPI.prototype.getItemOrientation = function ();	
Parameter	Null	
Return Value	Clockwise rotation of the currently drawn content (0/90/180/270)	

**5.3 setItemHorizontalAlignment** Set horizontal alignment of subsequent drawings

Function	Set horizontal alignment of subsequent drawings.	
Method	LPAPI.prototype.setItemHorizontalAlignment = function (alignment);	
Parameter	alignment	Alignment of subsequent drawings ( <a href="#">ItemAlignment</a> ), default value <a href="#">ItemAlignment.LEFT</a> .
Return Value	Null	
Instructions	The horizontal direction is based on the horizontal drawing direction of the subsequent drawn content, same as the clockwise rotation angle of the drawn content.	

**5.4 getItemHorizontalAlignment** Get horizontal alignment of current drawing

Function	Get horizontal alignment of current drawing.	
Method	LPAPI.prototype.getItemHorizontalAlignment = function ();	
Parameter	Null	
Return Value	Horizontal alignment of current drawing ( <a href="#">ItemAlignment</a> )	
Instructions	The horizontal direction is based on the horizontal drawing direction of the subsequent drawn content, same as the clockwise rotation angle of the drawn content.	

**5.5 setItemVerticalAlignment** Set vertical alignment of subsequent drawings

Function	Set vertical alignment of subsequent drawings.	
Method	LPAPI.prototype.setItemVerticalAlignment = function (int alignment);	
Parameter	alignment	Alignment of subsequent drawings ( <a href="#">ItemAlignment</a> ), default value <a href="#">ItemAlignment.TOP</a> .
Return Value	Null	
Instructions	The vertical direction is based on the vertical drawing direction of the subsequent drawn content, same as the clockwise rotation angle of the drawn content.	

**5.6 getItemVerticalAlignment** Get vertical alignment of current drawing

Function	Get vertical alignment of current drawing.
Method	LPAPI.prototype.getItemVerticalAlignment = function ();
Parameter	Null
Return Value	Vertical alignment of current drawing ( <a href="#">ItemAlignment</a> )
Instructions	The vertical direction is based on the vertical drawing direction of the subsequent drawn content, same as the clockwise rotation angle of the drawn content.

**5.7 setItemPenAlignment** Set Line Brush Alignment

Function	Set Line Brush Alignment.
Method	LPAPI.prototype.setItemPenAlignment = function (penAlignment);
Parameter	<div>penAlignment</div> <div>Line brush alignment (<a href="#">PenAlignment</a>):  <a href="#">PenAlignment.CENTER</a>: The drawn line is centered at the specified position;  <a href="#">PenAlignment.INSET</a>: The line drawn is inside the specified position.</div>
Return Value	Null

**5.8 getItemPenAlignment** Get Line Brush Alignment

Function	Get Line Brush Alignment.
Method	LPAPI.prototype.getItemPenAlignment = function ();
Parameter	Null
Return Value	<div>Line brush alignment (<a href="#">PenAlignment</a>):</div> <div><a href="#">PenAlignment.CENTER</a>: The drawn line is centered at the specified position;</div> <div><a href="#">PenAlignment.INSET</a>: The line drawn is inside the specified position.</div>

### 5.9 setDrawParam Set values of drawing-related parameters

Function	Set values of drawing-related parameters.	
Method	<b>LPAPI.prototype.setDrawParam = function (name, value);</b>	
Parameter	name	<p>Parameter name defined in a static string variable of the <b>FONT_NAME</b>: font name file stored in the project assets directory, the value is String type, such as: HeiTi.ttf. Note that font file name, maybe not the font name. If parameter value does not specify a suffix, the .ttf suffix will be added automatically.</p> <p><b>QRCODE_VERSION</b>: QR Code encoding version number (1 to 40), the value is Integer type. If no version number is specified, it means that the encoding version will be used automatically according to the encoding content.</p> <p><b>ERROR_CORRECTION</b>: The error correction level of QRCode (0~3), the value is Integer type, default is 0. The error correction level 0 is recommended for label printing, so that the black dots can be printed as large as possible, and the QR code can be scanned and recognized more easily.</p> <p><b>CHARACTER_SET</b>: The string encoding type of QRCode, the value is String type. Default is UTF-8.</p> <p><b>MARGIN</b>: The white space of QRCode code, value is Integer, defined in the QRCode code specification is 0/2/4. The default is 0, i.e. no white space.</p>
Instructions	value	Parameter values, refer to different parameter names for exact meanings.
Return Value	Null	

## 6. Description of draw text strings

[FontStyle](#) Fonts styles

Value	Description
0	Normal
1	Bold
2	Italic
3	Bold Italic
4	Underline
8	Strikethrough

### 6.1 Uniform fonts used for text drawing

The default fonts of different brands and models of cellphones in Android system are very different, so it is necessary to standardize the fonts. To standardize the text fonts, you can use the following two methods:

- For the case of using one font for the whole label, you can directly put the font file in the assets directory of the Android APP project, and change the name to FONT.ttf. The interface function for drawing text will use this font to draw the text. If the font file is not found, the text drawing function will use the default font;
- For the case of using multiple fonts in the whole label, you can put multiple font files in assets directory of the Android APP project, and then before drawing the text, call [setDrawParam\(IAtBitmap.DrawParamName.FONT\\_NAME, "xxx"\)](#) to specify a font file.

## 6.2 drawText Draw text strings

Function	Draw text strings.	
Method	<b>LPAPI.prototype.drawText = function (text, x, y, width, height, fontHeight, fontStyle);</b>	
Parameter	text	The text string that needs to be drawn.
	x	Horizontal position of the upper left corner of the drawn text box (in mm).
	y	Vertical position of the upper left corner of the drawn text box (in mm).
	width	Horizontal width of the drawn text box in mm. If width is 0, the drawn text will be aligned left-center-right based on the current alignment with x as the reference point according to the display width of the drawn text. The default value is 0.
	height	Vertical height of the drawn text box in mm. If height is 0, the drawn text will be aligned top, center and bottom based on the current alignment with y as the reference point according to the display width of the drawn text. The default value is 0.
	fontHeight	Font size (in mm).
	fontStyle	Font styles, Normal, Bold, italic, Bold Italic, Underline, Strikethrough ( <a href="#">FontStyle</a> ). Default is Normal.
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>• If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw the text string.</li> <li>• Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> <li>• FontHeight is in millimeters. The formula for converting font size to millimeters is: Font size = millimeter * 72 / 25.4 For example, a size 9 character is 3.175 millimeters and a size 12 character is 4.233 millimeters.</li> </ul>	

## 7. Description of draw Barcode

[BarcodeType](#) Barcode Type

Value	Description
20	UPC_A
21	UPC_E
22	EAN13
23	EAN8
24	CODE39
25	ITF25
26	CODEBAR
27	CODE93
28	CODE128
29	ISBN
30	ECODE39
60	AUTO

### 7.1 draw1Dbarcode Draw Barcode

Function	Draw Barcode.	
Method	LPAPI.prototype.draw1Dbarcode = function (text, type, x, y, width, height, textHeight);	
Parameter	text	Content of the barcode that needs to be drawn.
	type	Barcode type ( <a href="#">BarcodeType</a> ), default value 60.
	x	Horizontal position of the upper left corner of barcode (in mm).
	y	Vertical position of the upper left corner of barcode (in mm).
	width	The overall display width of barcode.
	height	Display height of barcode (including human readable text).
	textHeight	Height of human-readable text (in mm), 3 mm recommended.
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw the barcode string.</li> <li>Horizontal and vertical position, width, and height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	

## 7.2 draw2DQRCode Draw QR Code

Function	Draw QR Code.	
Method	<b>LPAPI.prototype.draw2DQRCode = function (text, x, y, width);</b>	
Parameter	text	Content of the QR Code that needs to be drawn.
	x	Horizontal position of the upper left corner of QR Code (in mm).
	y	Vertical position of the upper left corner of QR Code (in mm).
	width	Horizontal width of the drawn QR Code (in mm).
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>• If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw the QR Code string.</li> <li>• Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	

## 7.3 draw2DPdf417 Draw PDF417 QR Code

Function	Draw PDF417 QR Code.	
Method	<b>LPAPI.prototype.draw2DPdf417 = function (text, x, y, width, height);</b>	
Parameter	text	Content of the PDF417 QR Code that needs to be drawn.
	x	Horizontal position of the upper left corner of PDF417 QR Code (in mm).
	y	Vertical position of the upper left corner of PDF417 QR Code (in mm).
	width	Horizontal width of the drawn PDF417 QR Code (in mm).
	height	Vertical height of the drawn PDF417 QR Code (in mm).
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>• If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw PDF417 QR Code string.</li> <li>• Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	



## 8. Description of draw vector graphics

### 8.1 drawRectangle Draw a rectangular box with the specified line width

Function	Draw a rectangular box with the specified line width.	
Method	LPAPI.prototype.drawRectangle = function (x, y, width, height, lineWidth);	
Parameter	x	Horizontal position of upper left corner of the drawn rectangular box (in mm).
	y	Vertical position of upper left corner of the drawn rectangular box (in mm).
	width	Horizontal width of the drawn rectangular box (in mm).
	height	Vertical height of the drawn rectangular box (in mm).
	lineWidth	Line width of rectangular box. Line width of rectangular frame is extended toward inside of the rectangular frame (in mm).
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw rectangular box string.</li> <li>Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	

### 8.2 fillRectangle Draw a fill rectangular box with the specified line width

Function	Draw a fill rectangular box with the specified line width.	
Method	LPAPI.prototype.fillRectangle = function (x, y, width, height);	
Parameter	x	Horizontal position of upper left corner of drawn fill rectangular box (in mm).
	y	Vertical position of upper left corner of drawn fill rectangular box (in mm).
	width	Horizontal width of the drawn fill rectangular box (in mm).
	height	Vertical height of the drawn fill rectangular box (in mm).
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw fill rectangular box string.</li> <li>Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	

**8.3 drawRoundRectangle Draw a rounded rectangular box with the specified line width**

Function	Draw a rounded rectangular box with the specified line width.	
Method	<b>LPAPI.prototype.drawRoundRectangle = function (x, y, width, height, cornerWidth, cornerHeight, lineWidth);</b>	
Parameter	x	Horizontal position of upper left corner of drawn rounded rectangular box (in mm).
	y	Vertical position of upper left corner of drawn rounded rectangular box (in mm).
	width	Horizontal width of the drawn rounded rectangular box (in mm).
	height	Vertical height of the drawn rounded rectangular box (in mm).
	cornerWidth	Rounded corner width (in mm).
	cornerHeight	Rounded corner height (in mm). Default value same as cornerWidth.
	lineWidth	Line width of the rounded rectangle box. Line width of the rounded rectangular frame is extended to the inside of the rounded rectangular frame (in mm).
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>• If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw rounded rectangular box string.</li> <li>• Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	

**8.4 fillRoundRectangle Draw a fill rounded rectangular box with the specified line width**

Function	Draw a fill rounded rectangular box with the specified line width	
Method	<b>LPAPI.prototype.drawRoundRectangle = function (x, y, width, height, cornerWidth, cornerHeight);</b>	
Parameter	x	Horizontal position of upper left corner of drawn fill rounded rectangular box (in mm).
	y	Vertical position of upper left corner of drawn fill rounded rectangular box (in mm).
	width	Horizontal width of the drawn rounded fill rectangular box (in mm).
	height	Vertical height of the drawn rounded fill rectangular box (in mm).
	cornerWidth	Rounded corner width (in mm).
	cornerHeight	Rounded corner height (in mm). Default value same as cornerWidth.
Return Value	Successful or not	

Instructions	<ul style="list-style-type: none"> <li>If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw fill rounded rectangular box string.</li> <li>Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>
--------------	---

### 8.5 drawEllipse Draw an ellipse with the specified line width

Function	Draw an ellipse with the specified line width.	
Method	LPAPI.prototype.drawEllipse = function (x, y, width, height, lineWidth);	
Parameter	x	Horizontal position of upper left corner of drawn ellipse (in mm).
	y	Vertical position of upper left corner of drawn ellipse (in mm).
	width	Horizontal width of the drawn ellipse (in mm).
	height	Vertical height of the drawn ellipse (in mm). Default value same as width, that is drawing circle.
	lineWidth	Line width of ellipse. Line width of the ellipse frame is extended to the inside of the ellipse (in mm).
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw ellipse string.</li> <li>Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> <li>Width equals height is to print circle.</li> </ul>	

### 8.6 fillEllipse Draw a fill ellipse with the specified line width

Function	Draw a fill ellipse with the specified line width.	
Method	LPAPI.prototype.drawEllipse = function (x, y, width, height);	
Parameter	x	Horizontal position of upper left corner of drawn fill ellipse (in mm).
	y	Vertical position of upper left corner of drawn fill ellipse (in mm).
	width	Horizontal width of the drawn fill ellipse (in mm).
	height	Vertical height of the drawn fill ellipse (in mm). Default value same as width, that is drawing circle.

	lineWidth	Line width of fill ellipse. Line width of the ellipse frame is extended to the inside of the ellipse (in mm).
<b>Return Value</b>	Successful or not	
<b>Instructions</b>	<ul style="list-style-type: none"> <li>● If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw fill ellipse string.</li> <li>● Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> <li>● Width equals height is to print circle.</li> </ul>	

### 8.7 drawCircle Draw a circle centered on the specified position

<b>Function</b>	Draw a circle centered on the specified position.	
<b>Method</b>	LPAPI.prototype.drawEllipse = function (x, y, radius, lineWidth);	
<b>Parameter</b>	x	Center of the x-axis of the drawn circle (in mm).
	y	Center of the y-axis of the drawn circle (in mm).
	radius	Radius of the circle (in mm).
	lineWidth	Line width of circle. Line width of the circle frame is extended to the inside of the circle (in mm).
<b>Return Value</b>	Successful or not	
<b>Instructions</b>	<p>□ If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw ellipse/circle string.</p> <ul style="list-style-type: none"> <li>● Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> <li>● Width equals height is to print circle.</li> </ul>	

### 8.8 fillCircle Draw fill circle

<b>Function</b>	Draw a fill circle centered on the specified position.	
<b>Method</b>	LPAPI.prototype.fillCircle = function (x, y, radius);	
<b>Parameter</b>	x	Center of the x-axis of the drawn fill circle (in mm).
	y	Center of the y-axis of the drawn fill circle (in mm).
	radius	Radius of the fill circle (in mm).
<b>Return Value</b>	Successful or not	

<b>Instructions</b>	<ul style="list-style-type: none"> <li>□ If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw ellipse/circle string.</li> <li>● Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> <li>● Width equals height is to print circle.</li> </ul>
---------------------	--

### 8.9 drawLine Draw line (straight/slash)

Function	Draw line (straight/slash).	
Method	LPAPI.prototype.drawLine = function (x1, y1, x2, y2, lineWidth);	
<b>Parameter</b>	x1	Horizontal position of the start point of the line (in mm).
	y1	Vertical position of the start point of the line (in mm).
	x2	Horizontal position of the end point of the line (in mm).
	y2	Vertical position of the end point of the line (in mm).
	lineWidth	Line width (in mm). Line width is extended toward the bottom of the line.
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>● If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw straight/slash line string.</li> <li>● Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	

### 8.10 drawDashLine Draw dotted line

Function	Draw dotted line.	
Method	LPAPI.prototype.drawDashLine = function (x1, y1, x2, y2, lineWidth, dashLen); LPAPI.prototype.drawDashLine2 = function (x1, y1, x2, y2, lineWidth, dashLen1, dashLen2); LPAPI.prototype.drawDashLine4 = function (x1, y1, x2, y2, lineWidth, dashLen1, dashLen2, dashLen3, dashLen4);	
<b>Parameter</b>	x1	Horizontal position of the start point of the line (in mm).
	y1	Vertical position of the start point of the line (in mm).
	x2	Horizontal position of the end point of the line (in mm).
	y2	Vertical position of the end point of the line (in mm).
	lineWidth	Line width (in mm). Line width is extended toward the bottom of the line.

	dashLen1	Length of the first segment of the dotted line (in mm).
	dashLen2	Length of the second segment of the dotted line (in mm).
	dashLen3	Length of the third segment of the dotted line (in mm).
	dashLen4	Length of the fourth segment of the dotted line (in mm).
	dashLen	Array of dotted line segment lengths (in mm).
	dashCount	The number of elements in the dotted line segment length array.
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>● If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw dotted line string.</li> <li>● Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	

## 9. Description of draw image

### 9.1 drawImage Draw an image from specified file

Function	Draw an image from specified file	
Method	LPAPI.prototype.drawImage = function (imageFile, x, y, width, height);	
Parameter	imageFile	Image path.
	x	The position of the print object in the horizontal direction (in mm).
	y	The position of the print object in the vertical direction (in mm).
	width	Width of the print object (in mm). If width is 0, then use width of the loaded bitmap. The default value is 0.
	height	Height of the print object (in mm). If height is 0, then use height of the loaded bitmap. The default value is 0.
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>● If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw image string.</li> <li>● Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	

**9.2 drawImageWithActualSize Draw the actual size of the given image**

Function	Draw the actual size of the given image.	
Method	<b>LPAPI.prototype.drawImageWithActualSize = function (imageFile, x, y);</b>	
Parameter	imageFile	Image path.
	x	The position of the print object in the horizontal direction (in mm).
	y	The position of the print object in the vertical direction (in mm).
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>● If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw image string.</li> <li>● Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	

**9.3 drawImageWithThreshold Draw image with grayscale thresholds**

Function	Draw image with grayscale thresholds.	
Method	<b>LPAPI.prototype.drawImageWithThreshold = function (imageFile, x, y, width, height, threshold);</b>	
Parameter	imageFile	Image path.
	x	The position of the print object in the horizontal direction (in mm).
	y	The position of the print object in the vertical direction (in mm).
	width	Width of the print object (in mm). If width is 0, then use width of the loaded bitmap. The default value is 0.
	height	Height of the print object (in mm). If height is 0, then use height of the loaded bitmap. The default value is 0.
	threshold	The grayscale threshold for drawing bitmap. 256 means drawing grayscale picture, 257 means drawing primary color picture, 0~255 means drawing black and white picture. 255 means to draw black and white picture, the points with original color > grayscale threshold will be considered as white, and the points with original color <= grayscale threshold will be considered as black.
Return Value	Successful or not	
Instructions	<ul style="list-style-type: none"> <li>● If this method is called directly without calling <a href="#">startPage</a>, then this method will automatically call <a href="#">startPage</a> to start drawing a page. Then draw image string.</li> <li>● Horizontal and vertical position, horizontal width, and vertical height of the drawn upper-left corner are based on the horizontal and vertical orientation of the currently drawn page.</li> </ul>	