# CpclCommand API Introduction

Package name ：com.smart.command

File name：CpclCommand.java

**void** addInitializePrinter()

**Function：initial label**

**Overload method：void** addInitializePrinter(**int** qty)

**void** addInitializePrinter(**int** height,**int** qty)

**void** addInitializePrinter(**int** offset,**int** height,**int** qty)

**Parameter：**

qty : count

height : height of label

offset : Label Horizontal Offset

**Return:void**

**void** addPrint()

**Function：**print label

**parameter：:null**

**return : null**

**void** addText(**TEXT\_FONT** font,int size， **int** x, **int** y, **String** text)

**Function：**print text on the label

**parameter :**

**font : type**

|  |  |  |  |
| --- | --- | --- | --- |
| **TEXT\_FONT** | **English fonts** | **size** | **Chinese fonts** |
| *FONT\_0*("0") | 12\*12;  20x12;  20x20;  32x20;  36x20;  32x36;  16x16;  55x55 | 0: 12x12  1: 20x12  2: 20x20  3: 32x20  4: 36x20  5: 32x36  26: 16x16  55: 55x55 | 12\*12Simplified ChineseGB18030 |
| *FONT\_1*("1") | 24\*24 | 0: 24x24 | 24\*24Simplified ChineseGB18030 |
| *FONT\_2*("2") | 24\*24 | 0: 24x24 | 24\*24Simplified ChineseGB18030 |
| *FONT\_3*("3") | 20\*20 | 0: 20x20 | 20\*20Simplified ChineseGBK |
| *FONT\_4*("4") | 32\*32;48x48;64x48 | 0: 32x32  3: 48x48  4: 64x48 | 32\*32Simplified ChineseGBK |
| *FONT\_5*("5") | 24\*24 | 0: 24x24 | 24\*24Simplified ChineseGB18030 |
| *FONT\_7*("7") | 24\*24 | 0: 24x24 | 24\*24Simplified ChineseGB18030 |
| *FONT\_8*("8") | 24\*24 | 0: 24x24 | 24\*24Simplified ChineseGB18030 |
| *FONT\_20*("20") | 16\*16 | 0: 16x16 | 16\*16Simplified ChineseGBK |
| *FONT\_24*("24") | 24\*24;48x48 | 0: 24x24  3: 48x48 | 24\*24Simplified ChineseGB18030 |
| *FONT\_28*("28") | 28\*28 | 0: 28x28 | 28\*28Traditional ChineseBIG5 |
| *FONT\_55*("55") | 16\*16;32x32 | 0: 16x16  3: 32x32 | 16\*16Simplified ChineseGB18030 |
| *FONT\_56*("56") | 32\*32 | 0: 32x32 | 32\*32Simplified ChineseGB18030 |
| *FONT\_57*("57") | 12\*12 | 0: 12x12 | 12\*12Simplified  ChineseGB18030 |
| *FONT\_88*("88") | 11\*11;32x32 | 0: 11x11  32: 32x32 | UTF-8 |
| *FONT\_89*("89") | 24\*24 | 0: 24x24 | 24x24Korean  KS5601 |
| *FONT\_90*("90") | 24\*24 | 0: 24x24 | 24\*24Traditional ChineseBIG5 |
| *FONT\_91*("91") | 24\*24 | 0: 24x24 | 28\*24Traditional ChineseBIG5 |

size : Select font size

x : Horizontal starting position

y : Vertical starting position

text : text going to print

**return :** null

**Related methods : void** addText90(**TEXT\_FONT** font,int size, **int** x, **int** y, **String** text)

**void** addText180(**TEXT\_FONT** font,int size, **int** x, **int** y, **String** text)

**void** addText270(**TEXT\_FONT** font,int size, **int** x, **int** y, **String** text)

**Function** :Text rotation angle of 90,180,270 degrees

**void** addTextConcat(**int** x, **int** y,**String []** var)

**Function：**Adding text strings to labels

**Parameter :**

x : Horizontal starting position

y : Vertical starting position

var : Text in series, formatted as font size offset data

Such as String[] var ={“2 0 2 hello”,”4 0 2 hello”}

font : Font name/number

size : Ignore this parameter, any number

offset : Offset value of the text relative to the starting position

data : Text content.

**return : null**

**Refer to :** CPCL programming command :CONCAT

**void** addSetmag(**int** w, **int** h)

**Function :** Enlarge resident fonts by a specified magnification

**Parameter:**

w : Width magnification, from 1 to 16

h : High magnification, from 1 to 16

**return : null**

**Refer to :** CPCL programming command :SETMAG

**void** addSetbold(**BOLD** value)

**Function：**Bold the resident font

**Parameter :**

value : Bolded or not

enum BOLD{

*ON*("1")

*OFF*("0")

}

**return : null**

**Refer to** : CPCL programming command :SETBOLD

**void** addBarcode(**COMMAND** command,**CPCLBARCODETYPE** type,

**int** height, **int** x, **int** y, **String** text)

**Function :** Print barcodes in portrait and landscape orientation at the specified width and height.

**Overload methods :**

**① void** addBarcode(**COMMAND** command,**CPCLBARCODETYPE** type,**int** width,**BARCODERATIO** ratio,**int** height,**int** x,**int** y,**String** text)

**② void** addBarcode(**COMMAND** command,**CPCLBARCODETYPE** type, **int** height, **int** x, **int** y,**int** number,**int** offset,**String** text)

**③ void** addBarcode(**COMMAND** command,**CPCLBARCODETYPE** type,**int** width,**BARCODERATIO** ratio,**int** height,**int** x,**int** y,**int** number,**int** offset,**String** text)

**Parameter :** Methods ① and ② do not have 'number' and 'offset', add barcode comments, the latter two add barcode comments automatically.

**Parameter :**

command : Printing in landscape or portrait orientation.

enum COMMAND{

BARCODE("BARCODE") VBARCODE("VBARCODE")

}

type : barcode type

enum CPCLBARCODETYPE{

CODE128("128")

UPC\_A("UPCA")

UPC\_E("UPCE") EAN\_13("EAN13") EAN\_8("EAN8")

CODE39("39")

CODE93("93")

CODABAR("CODABAR")

}

width : Width of narrow barcode strip

ratio : The ratio of the width of the barcode

enum BARCODERATIO{

Wide and narrow proportions

Point0("0"), 1.5:1

Point1("1") 2.0:1

Point2("2") 2.5:1

Point3("3") 3.0:1

Point4("4") 3.5:1

Point20("20") 2.0:1

Point21("21") 2.1:1

Point22("22") 2.2:1

Point23("23") 2.3:1

Point24("24") 2.4:1

Point25("25") 2.5:1

Point26("26") 2.6:1

Point27("27") 2.7:1

Point28("28") 2.8:1

Point29("29") 2.9:1

Point30("30") 3.0:1

}

height : the height of barcode

x : Horizontal starting position

y : Vertical starting position

number : Font number to be used when annotating barcodes

offset : offset of text from barcode

text : Barcode content

**return :** null

**Refer to :** CPCL programming command :BARCODE

**void** addBarcodeText(**int** font,**int** offset)

**Function :** Adding barcode comments

**Parameter :**

font : Font number to be used when annotating barcodes

offset : Unit offset of text from barcode

**return : null**

**Related methods : void** addBarcodeTextOff()

Function ; Disable comments

**Refer to :** CPCL programming command :BARCODE-TEXT

**void** addBQrcode(**QRCODE\_LEVEL** level,**MODE** mode,**int** x,**int** y,**int** n,**int** u,**String** text)

**Function :** Add QRcode

**Overload : void** addBQrcode(**QRCODE\_LEVEL** level,**MODE** mode,**int** x,**int** y,**String** text)

**Function :** Select default u and n to print barcodes

**Parameter :**

level : QR code error correction level.

mode : Input mode.

x : Horizontal starting position

y : Vertical starting position

n : QR Code Specification number,1 or 2，default: 2

u : 模块的宽度/高度 1-32，default: 6

text : barcode content

**return : null**

**Related methods : void** addVBQrcode(**QRCODE\_LEVEL** level,**MODE** mode,**int** x,**int** y,**int** n,**int** u,**String** text)

**void** addVBQrcode(**QRCODE\_LEVEL** level,**MODE** mode,**int** x,**int** y,**String** text)

**Function :** Print the QR code in portrait orientation

**Refer to :** CPCL programming command :QR Code

**void** addBox(**int** x, **int** y, **int** xend, **int** yend, **int** thickness)

**Function :** Prints a rectangle of the specified line width

**Parameter :**

x : X-coordinate in the top left corner

y : Y-coordinate in the top left corner

xend : X-coordinate in the lower right corner

yend : Y-coordinate in the lower right corner

thickness : Width of line

**return :** null

**Refer to :** CPCL programming command:BOX

**void** addLine(**int** x, **int** y, **int** xend, **int** yend, **int** width)

**Function : Print lines of any length, width and angular orientation**

**Parameter :**

x : X-coordinate of the starting point

y : Y-coordinate of the starting point

xend : X-coordinate in the lower right corner

yend : Y-coordinate in the lower right corner

width : Width of line

**return :** null

**Refer to :** CPCL programming command:LINE

**void** addInverseLine(**int** x, **int** y, **int** xend, **int** yend, **int** width)

**Function :** To draw the inverse area, add the content first and then the inverse area.

**Parameter :**

x : X-coordinate of the starting point

y : Y-coordinate of the starting point

xend : X-coordinate of the termination point

yend : Y-coordinate of the termination point

width : Height of counter-coloured area

**return : null**

**Refer to :** CPCL programming command:INVERSE-LINE

**void** addEGraphics(**int** x,**int** y,**int** nWidth,**Bitmap** bitmap)

**Function :** 打印bitmap image

**Parameter :**

x : Horizontal starting position

y : Vertical starting position

nWidth : Image width(dots)

bitmap : Image source file or path

**return : null**

**Related methods : void** addCGraphics(**int** x,**int** y,**int** nWidth,**Bitmap** bitmap)

**Function :** Print the bitmap image with the same parameters as above

**Refer to :** CPCL programming command:GRAPHICS

**void** addJustification(**ALIGNMENT** align)

**Function :** Control the alignment

**Overload methods :void** addJustification(**ALIGNMENT** align,**int** end) **Function :** Control the alignment of the fields according to their width.

**Parameter :** align : Alignment method

enum ALIGNMENT{

CENTER("CENTER")

LEFT("LEFT")

RIGHT("RIGHT");

}

end : End point of alignment

**return : null**

**Refer to :** CPCL programming command:JUSTIFICATION

**void** addPagewidth(**int** width)

**Function :** Set the width of the page

**Parameter :** width : width of the page

**return : null**

**Refer to :** CPCL programming command:PAGE-WIDTH

**void** addSpeed(**CPCLSPEED** level)

**Function :** Setting print speed

**Parameter :** level : Speed grade

enum CPCLSPEED{

SPEED0("0")

SPEED1("1")

SPEED2("2")

SPEED3("3")

SPEED4("4")

SPEED5("5")

}

**return : null**

**Refer to :** CPCL programming command:SPEED

**void** addBeep(**int** beep\_length)

**Function :** Make the buzzer sound for a given length of time

**Parameter :**

beep\_length : Beep duration in 1/8 second increments

**return : null**

**Refer to :** CPCL programming command:BEEP

**void** addQueryPrinterStatus ()

**Function :** Query the current status of the printer

**Parameter :null**

**return : null**

**void** addGapSense()

**Function :** Set check label

**Parameter :** null

**return :** null

**Refer to :** CPCL programming command:GAP-SENSE

**void** addBarSense()

**Function :** Set detection right black mark

**Parameter :** null

**return :** null

**Refer to :** CPCL programming command:BAR-SENSE

**void** addBarSenseLeft()

**Function :** Set detection left black mark

**Parameter :** null

**return :** null

**Refer to :** CPCL programming command:BAR-SENSE LEFT