

KeyboardEmulator Development Instructions

Version 1.0.3

Catalog

1. Keyboardemulator Development Abstract.....	8
1.1 Development Intro	8
1.2 Intended For	8
1.3 IDE	8
1.4 Development Steps.....	8
2. API.....	8
2.1 open()	8
Interface	8
public void open(Context context,BarcodeUtility.ModuleType modul).....	8
Function Description	8
Switch On scanning function.....	8
Para. Description.....	8
context.....	8
Value: context object modul.....	9
Value:	9
BARCODE_1D	9
BARCODE_2D BARCODE_2D_H.....	9
Return.....	9
void	9
2.2 close().....	9
Interface.....	9
public void close(Context context,BarcodeUtility.ModuleType modul)	9
Function Description	9

Switch OFF barcode scanning.....	9
Para. Description.....	9
context.....	9
Value: context object module	9
Value:	9
BARCODE_1D BARCODE_2D BARCODE_2D_H	9
Return.....	9
void	9
2.3 startScan()	9
Interface	9
public void startScan(Context context,BarcodeUtility.ModuleType modul).....	9
Function Description	9
Start Scanning	9
Para. Description.....	9
context.....	9
Value: context object modul.....	9
Value:	10
BARCODE_1D BARCODE_2D BARCODE_2D_H	10
Return.....	10
void	10
2.4 stopScan().....	10
Interface	10
public void stopScan(Context context,BarcodeUtility.ModuleType modul)	10
Function Description	10
Stop Scanning.....	10
Para. Description.....	10
context.....	10
Value: context object modul.....	10

Value:	10
BARCODE_1D BARCODE_2D BARCODE_2D_H	10
Return.....	10
void	10
2.5 setOutputMode().....	10
Interface	10
public void setOutputMode(Context context,int mode).....	10
Function Description	10
OFF scan	10
Para. Description.....	10
context.....	10
Value: context object mode	10
Value:	10
0: on cursor.....	10
1: clipboard.....	11
2: broadcast.....	11
3: HID	11
Return.....	11
void	11
2.6 setScanResultBroadcast()	11
Interface	11
public void setScanResultBroadcast(Context context,java.lang.String broadcastAction,java.lang.String extraName).....	11
Function Description	11
Para. Description	11
Context.....	11
Value: context object broadcastAction.....	11
Value:	11

Action of receiving scanning data.....	11
ExtraName.....	11
Value:	11
The name of the data item returned by intent	11
Return.....	11
void	11
2.7 stopScan().....	11
Interface	11
public void enableEnter(Context context,boolean isEnter).....	11
Function Description	11
Enable Enter	11
Para. Description.....	11
context.....	11
Value: context object isEnter.....	11
Value:	12
true Enable	12
false disable	12
Return.....	12
void	12
2.8 stopScan().....	12
Interface	12
public void enableTAB(Context context,boolean isTab).....	12
Function Description	12
Enable TAB.....	12
Para. Description.....	12
context.....	12
Value: context object isTab.....	12
Value:	12

true Enable	12
false Disable	12
Return.....	12
void	12
2.9 setReleaseScan.....	12
Interface	12
public void setReleaseScan(Context context,boolean enable).....	12
Function Description	12
Release Key Stop Scan?	12
Para. Description.....	12
Context.....	12
Value: context object enable	12
Values:	12
true Stop scan.....	13
false Not stop	13
Return.....	13
void	13
2.10 enablePlaySuccessSound()	13
Interface	13
public void enablePlaySuccessSound(Context context,boolean isSuccessSound) ..	13
Function Description	13
Play success sound?	13
Para. Description.....	13
Context.....	13
Value: context object enable	13
value:	13
true play sound	13
false not to play sound	13

Return.....	13
void	13
2.11 enablePlayFailureSound()	13
Interface.....	13
public void enablePlayFailureSound(Context context,boolean isFailureSound)	13
Function Description	13
Play failure sound?	13
Para. Description.....	13
Context.....	13
Value: context object enable	13
Value:	13
true Play sound.....	13
false Not to play sound.....	13
Return.....	13
void	13
2.12 setPrefix()	14
Interface.....	14
public void setPrefix(Context context,java.lang.String prefix)	14
Function Description	14
prefix.....	14
Para. Description.....	14
Context.....	14
Value: context object enable	14
Value:	14
Prefix.....	14
Return.....	14
void	14
2.13 setSuffix().....	14

Interface	14
public void setSuffix(Context context,java.lang.String suffix)	14
Function Description	14
Suffix	14
Para. Description	14
Context	14
Value: context object suffix	14
Value: suffix	14
Return.....	14
void	14
3. Operation Description	14
3.1 Import jar	14
3.2 Use operation	15
3.2.1 Registering Broadcast:	15
3.2.2 Receive Data:	15
3.2.3 Parameters Setup:	15

1. Keyboardemulator Development Abstract

1.1 Development Intro

This document mainly describes how to control the whole scanning process through the code control device with its own Keyboardemulator. Keyboardemulator is the code scanning software of the device. In the actual development, the keyboard assistant is used to obtain the scanning data and set other scanning options.

1.2 Intended For

Software developers

Software tester

1.3 IDE

Android Studio

1.4 Development Steps

-->Get the tool class barcodeutility to control the keyboardemulator through the SDK and instantiate it.

-->Set the relevant parameters of keyboardemulator scanning through the method of tool class.

-->The scan value is obtained by registering the broadcast.

2. API

This document mainly describes the common related methods of barcodeutility class, which is applicable to the control operation of keyboardemulator. For more details, please refer to the API document.

2.1 open()

Interface	public void open(Context context,BarcodeUtility.ModuleType modul)
Function Description	Switch On scanning function
Para. Description	context

	Value: context object modul Value: BARCODE_1D BARCODE_2D BARCODE_2D_H
Return	void

2.2 close()

Interface	public void close(Context context,BarcodeUtility.ModuleType modul)
Function Description	Switch OFF barcode scanning
Para. Description	context Value: context object module Value: BARCODE_1D BARCODE_2D BARCODE_2D_H
Return	void

2.3 startScan()

Interface	public void startScan(Context context,BarcodeUtility.ModuleType modul)
Function Description	Start Scanning
Para. Description	context Value: context object modul

	Value: BARCODE_1D BARCODE_2D BARCODE_2D_H
Return	void

2.4 stopScan()

Interface	public void stopScan(Context context,BarcodeUtility.ModuleType modul)
Function Description	Stop Scanning
Para. Description	context Value: context object modul Value: BARCODE_1D BARCODE_2D BARCODE_2D_H
Return	void

2.5 setOutputMode()

Interface	public void setOutputMode(Context context,int mode)
Function Description	OFF scan
Para. Description	context Value: context object mode Value: 0: on cursor

	1: clipboard 2: broadcast 3: HID
Return	void

2.6 setScanResultBroadcast()

Interface	public void setScanResultBroadcast(Context context,java.lang.String broadcastAction,java.lang.String extraName)
Function Description	Set the broadcast to receive scan data
Para. Description	Context Value: context object broadcastAction Value: Action of receiving scanning data ExtraName Value: The name of the data item returned by intent
Return	void

2.7 stopScan()

Interface	public void enableEnter(Context context,boolean isEnter)
Function Description	Enable Enter
Para. Description	context Value: context object

	isEnter Value: true Enable false disable
Return	void

2.8 stopScan()

Interface	public void enableTAB(Context context,boolean isTab)
Function Description	Enable TAB
Para. Description	context Value: context object isTab Value: true Enable false Disable
Return	void

2.9 setReleaseScan

Interface	public void setReleaseScan(Context context,boolean enable)
Function Description	Release Key Stop Scan?
Para. Description	Context Value: context object enable Values:

	true Stop scan false Not stop
Return	void

2.10 enablePlaySuccessSound()

Interface	public void enablePlaySuccessSound(Context context,boolean isSuccessSound)
Function Description	Play success sound?
Para. Description	Context Value: context object enable value: true play sound false not to play sound
Return	void

2.11 enablePlayFailureSound()

Interface	public void enablePlayFailureSound(Context context,boolean isFailureSound)
Function Description	Play failure sound?
Para. Description	Context Value: context object enable Value: true Play sound false Not to play sound
Return	void

2.12 setPrefix()

Interface	public void setPrefix(Context context,java.lang.String prefix)
Function Description	prefix
Para. Description	Context Value: context object enable Value: Prefix
Return	void

2.13 setSuffix()

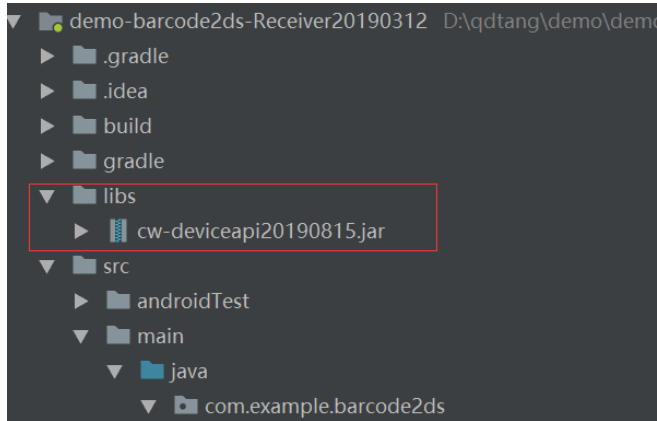
Interface	public void setSuffix(Context context,java.lang.String suffix)
Function Description	Suffix
Para. Description	Context Value: context object suffix Value: suffix
Return	void

3. Operation Description

3.1 Import jar.

A\ Build Android Studio development environment

B\ Add CW- deviceapi.jar Copy to LIBS directory and import



Note: Latest API version will be upgraded, please download from our website according to model.

3.2 Use operation

3.2.1 Registering Broadcast:

```
barcodeDataReceiver=new BarcodeDataReceiver();
IntentFilter intentFilter = new IntentFilter();
intentFilter.addAction("com.scanner.broadcast");
context.registerReceiver(barcodeDataReceiver, intentFilter);
```

3.2.2 Receive Data:

```
class BarcodeDataReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        String barCode = intent.getStringExtra( name: "data");
        String type = intent.getStringExtra( name: "CODE_TYPE");
        if (barCode != null && !barCode.equals("")) {
            // success tvData.setText(barCode);
        } else {
            barCode="Scan fail";
            //fail tvData.setText("Scan fail");
        }
        if(iBarcodeResult!=null)
            iBarcodeResult.getBarcode(barCode + "\n" + type);
        SoundManage.PlaySound(context, SoundManage.SoundType.SUCCESS);
    }
}
```

3.2.3 Parameters Setup:

Instantiation tool class:

```
public Barcode2D() {  
    barcodeUtility= BarcodeUtility.getInstance();  
}
```

Set scan parameters:

```
barcodeUtility.setOutputMode(context, mode: 2); //设置广播接收数据  
barcodeUtility.setScanResultBroadcast(context, broadcastAction: "com.scanner.broadcast", extraName: "data"); //设置接收数据的广播  
barcodeUtility.open(context, BarcodeUtility.ModuleType.BARCODE_2D); //打开2D  
barcodeUtility.setReleaseScan(context, enable: false); //设置松开扫描按键，不停止扫描  
barcodeUtility.setScanFailureBroadcast(context, enable: true); //扫描失败也发送广播  
barcodeUtility.enableContinuousScan(context, isContinuous: false); //关闭键盘助手连续扫描  
barcodeUtility.enablePlayFailureSound(context, isFailureSound: false); //关闭键盘助手 扫描失败的声音  
barcodeUtility.enablePlaySuccessSound(context, isSuccessSound: false); //关闭键盘助手 扫描成功的声音  
barcodeUtility.enableEnter(context, isEnter: false); //关闭回车
```


Start / stop scanning

```
//开始扫码
public void startScan(Context context) {
    if(barcodeUtility!=null) {
        Log.i(TAG, msg: "ScanBarcode");
        barcodeUtility.startScan(context, BarcodeUtility.ModuleType.BARCODE_2D);
    }
}
//停止扫描
public void stopScan(Context context){
    if(barcodeUtility!=null) {
        Log.i(TAG, msg: "stopScan");
        barcodeUtility.stopScan(context, BarcodeUtility.ModuleType.BARCODE_2D);
    }
}
```