

# Coreaiot 信标 SDK Android

## 安装

Gradle (推荐)

```
compile 'com.coreaiot.rtls:sensor:1.0.1'
```

Maven

```
<dependency>
  <groupId>com.coreaiot.rtls</groupId>
  <artifactId>sensor</artifactId>
  <version>1.0.1</version>
  <type>pom</type>
</dependency>
```

Ivy

```
<dependency org='com.coreaiot.rtls' name='sensor' rev='1.0.1'>
  <artifact name='sensor' ext='pom' ></artifact>
</dependency>
```

下载地址

<https://github.com/coreaiot/RTLSSensor/releases>

# 使用

kotlin

```
import com.coreaiot.rtls.sensor.Ble

val ble = Ble(context)
ble.start({ ->
    ok()
}, { errorCode ->
    fail(errorCode)
})

// Stop
ble.stop()
```

## Ble 类

### 构造方法

```
class Ble(context: Context)
```

### 属性

```
/**
 * ID
 * MAC 地址的后两个字节
 * 值域: [0, 0xffff]
 */
var id: Int

/**
 * 用户自定义数据
 * 值域: [0, 0x7f]
 */
var customFlag: Int

/**
 * 报警状态
```

```

*/
var alarm: Boolean

/**
 * 电池电量
 * 值域: [0, 10]
 * 实际电量为 值 * 10 * 100%
 * 例: 值为 5 表示电池电量为 50%
 */
var battery: Int

/**
 * 广播频率
 * 默认为 AdvertiseSettings.ADVERTISE_MODE_BALANCED
 * 详情请看:
https://developer.android.com/reference/android/bluetooth/le/AdvertiseSettings
 */
var advertiseMode: Int

/**
 * 广播功率
 * 默认为 AdvertiseSettings.ADVERTISE_TX_POWER_HIGH
 * 详情请看:
https://developer.android.com/reference/android/bluetooth/le/AdvertiseSettings
 */
var txPowerLevel: Int

```

## 方法

```

/**
 * 开始广播
 */
fun start()
fun start(callback: (() -> Unit)?)
fun start(callback: (() -> Unit)?, fail: ((errorCode: Int) -> Unit)?)

/**
 * 停止广播
 */
fun stop()

```