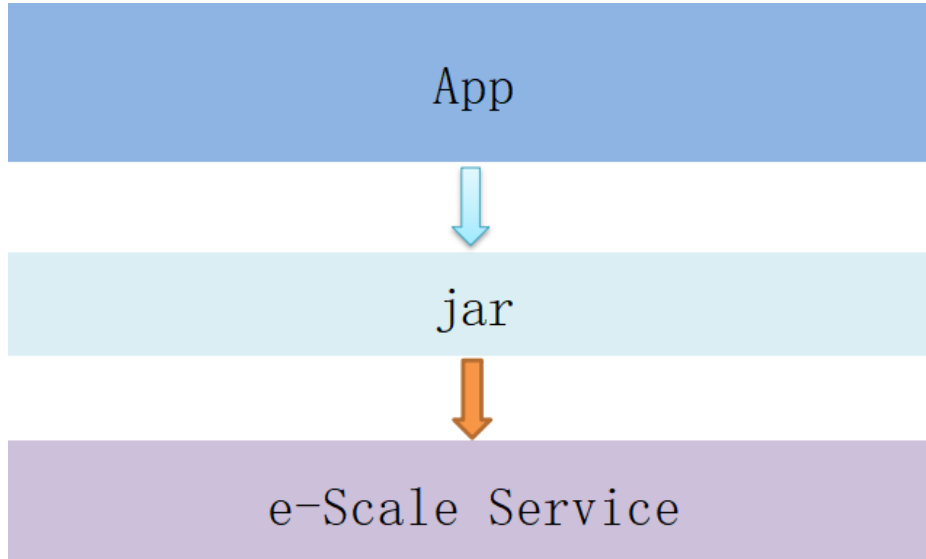


电子秤 2.0 开发者文档

一、框架

采用内置电子秤服务和外置引用 SDK jar 包实现方式；
电子秤功能、兼容等由内部服务适配，开发者只需调用 jar 包标准接口即可；
内置电子秤服务迭代维护可由平台推送独立升级，无需客户额外开发；
示意图：



二、电子秤基础知识

2.1 电子秤参数

型号:	Swan 2 (Scale)
最大量程	6kg≤Max≤15kg
n(等级)	3000 (III)
检定分度	2/5g
工作温度	-10~+40°C

三、计量相关

#1. 零点

每次智能电子秤重新上电的时候，电子秤都会自动记录初始零点作为后续称重的参考。开机重启的零点范围通常是满量程的 10%。当重量超过满量程的 10%的时候，电子秤将无法找到零点位置。例如，对 15kg 的电子秤而言，如果开机的时候秤盘的重量超过 1.5kg 的时候，电子秤将无法找到零点位置。如果重量小于 1.5kg 的时候，电子秤将默认从零点开始称量。

我们推荐您在上电开机的前清空秤盘上所有的物品。

#2. 手动清零/零点设置

在日常称重时，倘若需要回到零点时，可以通过手动清零回到初始零点位置。手动清零的范围是满量程的 2%。对 15kg 的电子秤而言，手动清零的范围是 298g。例如，在您清洁完秤盘后，可能称重显示为-0.004kg。当您点击清零按钮后，称重显示即回归“0.000kg”。

注意：在有皮重的情况下，清零按钮将无效。

#3. 去皮/削皮

在称重过程中，如果需要将商品包装的重量去除的时候，可以通过去皮操作执行。去皮的重量将会影响到称量的范围。例如，如果一台 15kg 的电子秤设置了 5kg 的皮重之后，那么剩余的称重范围将仅剩余 10kg。

称重去皮：先把包装放在秤盘上，点击去皮按钮后，包装的重量将自动记录为皮重；

预置皮重 如果您已知包装重量的时候，可以手动输入皮重值；

对多量程的电子秤而言，最大去皮重量通常是 Max1-e1。例如，6/15kg 的电子秤，最大皮重是-5.998kg。

注意：预置皮重的值必须准确设置。首先，预置皮重的值必须小于最大皮重；其次，预置皮重值必须按照正确的分度值设置。例如，对一台 6/15kg（n=3000）的电子秤，6.005kg 或者是 0.019kg 的预置皮重值都是不对。

#4. 净重

净重是指消费者应该支付商品的重量值。如果商品带包装，需要先将包装作为皮重去掉。

#5. 毛重

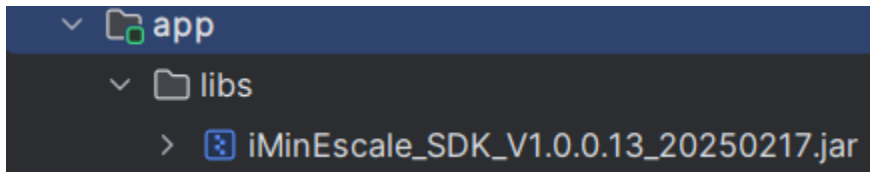
毛重=皮重+净重

四、电子秤服务使用方法

3.1 下载 SDK

3.2 初始化 SDK

导入 jar 包后



implementation files('libs/iMinEscale_SDK_V1.0.0.13_20250217.jar')

, 连接 imin 电子秤服务, 当服务连接后即可使用 SDK 中的各个功能

```
ScaleManager scaleManager = ScaleManager.getInstance(context);
```

```
//iMin Scale service
```

```
scaleManager.connectService(newScaleManager.ScaleServiceConnection() {
```

```
@Override
```

```
public void onServiceConnected() {
```

```
    //Service binding
```

```
}
```

```
@Override
```

```
public void onServiceDisconnect(){
```

```
    //Service unbundling
```

```
}
```

```
}
```

[向 信息警告提示

右] **Android 11** 为了加强了隐私保护策略, 引入了很多变更和限制, 其中软件包可见性变更, 将会导致第三方应用无法成功初始化 SDK

因此需要在 AndroidManifest.xml 中添加如下:

```
<manifest package="开发者应用包名">
```

```
<queries>
```

```
<package android:name="com.imin.peripherservice" />
```

```
</queries>
```

```
...
```

```
</manifest>
```

特别需要注意的是, **Android11** 的该变更只会影响到升级 targetSdkVersion=30 的应用, 未升级的应用暂不受影响

3.3 获取称重数据

通过接口回调的方式快速获取电子秤数据的方法

```
scaleManager.getData(new ScaleResult(){
```

```
@Override
```

```
public void getResult(int net,int tare, boolean isStable) throws RemoteException {
```

```
    //Return weighing result
```

```
}
```

```
@Override
```

```
public void getStatus(boolean isLightWeight, boolean overload, boolean clearZeroErr, boolean calibrationErr) throws
```

```
RemoteException {
```

```
    //Return to weighing state
```

```
}
```

```
@Override
```

```

public void getPrice(int net, int tare, int unit, String unitPrice, String totalPrice, boolean isStable, boolean isLightWeight) {
    //Return valuation result
}

@Override
public void error(int errorCode) {
    //crc exception callback
}
};

```

称重结果参数说明

参数	类型	含义	说明
net	int	获取称量净重	单位：克
tare	int	获取称量皮重	单位：克
isStable	boolean	秤稳定状态	true:稳定 false: 浮动

称重状态参数说明

参数	类型	含义	说明
isLightWeight	boolean	秤是否过轻	true:过轻 false:正常
overload	boolean	秤是否过载	true:过载 false:正常
clearZeroErr	boolean	秤是否有清零错误	true:错误 false:正常
calibrationErr	boolean	秤是否有标定错误	true:错误 false:正常

称重价格参数说明

参数	类型	含义	说明
net	int	获取称量净重	单位：克
tare	int	获取称量皮重	单位：克
unit	int	重量单位	0: g (默认) 1 : 100g 2: 500g 3: kg
unitPrice	String	当前计价设置的单价	默认为 0, 无单位 (由业务方自定义)
totalPrice	String	电子秤计算后的总价	当前计价根据重量和设置单价计算的总价, 默认 0
isStable	boolean	秤稳定状态	true:稳定 false: 浮动
isLightWeight	boolean	秤是否过轻	true:过轻 false:正常

CRC 结果

errorcode: 0 表示称重数据正常 非 0 表示数据异常

[向右]对于称重结果在业务方软件中的展示推荐做法是:

醒目显示净重、皮重结果并有提示当前稳定状态、零点状态和净重状态的标志

零点状态：皮重+净重 = 0

净重状态：皮重 ≠ 0

当称重异常时可通过回调结果提示当前欠载、过载和通信异常状态

欠载状态：需要判断当前重量是否小于 -20e, e 为最小分度值可通过其他 API 获取

3.4 其它接口名称及说明

功能	含义	说明
String getServiceVersion()	获取电子秤服务版本号	返回当前电子秤服务的版本号字符串
String getFirmwareVersion()	获取固件版本号	接返回当前固件的版本号，如10034
void zero()	清零	清零操作只可清除 300g 以内的结果偏差
void tare()	去皮/清皮	秤上有重量时为去皮，没有时为清皮
void digitalTare(int i)	数字去皮	直接给电子秤下发去皮的重量
void cancelGetData()	取消获取数据	退出应用时调用，与 getData 成对使用
int[] readAcceleData()	读取加速度数据	[0][1][2]分别为 X, Y, Z方向数据
int readSealState()	获取铅封状态	0: 正常 1: 铅封被破坏
int getCalStatus()	读取标定按钮开关状态	0: 未按下 1: 按下
int[][] getCalInfo()	读取电子秤参数信息	返回值为一个多个量程的二维数组 例如量程为 6/15kg e=2/5g 多量程电子秤将返回[[6, 2],[15,5]]
void restart()	重启电子秤	电子秤重启会重新读取零点请谨慎调用此方法防止秤重读数不准确
void setUnitPrice(String unitPrice)	设置单价	由电子秤服务计算价格时设置，将影响返回的计价结果 支持小数点两位的计算
String getUnitPrice()	获取当前已经设置的单价	由电子秤服务计算价格时使用的单价设置
void setUnit(int unit)	设置价格计算时的重量单位	0: 按 g 计重 1: 按 100g 计重 2: 按 500g 计重 3: 按 kg 计重
int getUnit()	获取当前价格计算的重量单位	由电子秤服务计算价格时使用的价格重量单位，会影响计算总价的结果
List<String> getCityAccelerations()	获取城市重力加速表	例如一条数据：安徽,97947
int setGravityAcceleration(int index)	设置城市重力加速	index 对应城市重力加速列表脚标；返回值 0 为设置成功，其它设置失败