@Override

public boolean equals(Object o) {

if (o != null && o.getClass() == this.getClass()) {

MonthInfo data = (MonthInfo) o;

if (data.year == this.year && data.month == this.month) {

return true;

}

}

return false;

}

HashCode()方法被用来获取给定对象的唯一整数。这个整数被用来确定对象被存储在HashTable类似的结构中的位置。默认的，Object类的hashCode()方法返回这个对象存储的内存地址的编号。

1. Padding指view的内容相对于view的边界的距离。Margin指一个view相对于其他view的距离。
2. 在使用include标签时，如果指定了除android:id=”xxx”和layout=”xxx”之外的属性时，必须加上android:layout\_width和android:layout\_height，否则其他属性不会生效。
3. android:layout\_marginBottom=”20dp”

android:layout\_marginRight=”50dp”

1. 1) ~~drawable-fr-rCA/~~ 与设备配置冲突

2) ~~drawable/~~

~~drawable-port-ldpi/~~

~~drawable-port-notouch-12key/~~ 按描述符优先级en匹配

3) ~~drawable-en/~~

~~drawable-en-notouch-12key/~~ 按描述符优先级port匹配

所以drawable-en-port/匹配。

1. onSaveInstanceState在activity被系统杀掉之前调用。如，activity a启动了activity b，b完全遮挡了a，那么a可能在系统资源紧张时被回收，在系统杀掉a之前会调用onSaveInstanceState。
2. Start service将会调用service的onCreate -> onStartCommand -> onDestroy, bind service会调用servcie的onCreate -> onBind -> onUnbind -> onDestroy。

在onStartCommand返回START\_STICKY，则service进程被杀掉后，系统会重启该service。

1. 1)实现数据共享

2)屏蔽实现细节

1. 不是。onReceive()方法将在与registerReceiver(BroadcastReceiver, IntentFilter, String, Handler)时传入的handler关联的线程里执行。默认调用registerReceiver(BroadcastReceiver, IntentFilter)方法不传handler时将在主线程执行。
2. Thread, HandlerThread, AsyncTask, Loader, ThreadPoolExecutor
3. 1)输入事件(touch, key等) 5s内没有得到处理

2)BroadcastReceiver的onReceive方法执行超过10s

1. 当启动一个launchMode为singleTask的activity时，activity会根据taskAffinity去寻找相应的task来驻留自己，当发现没有这样的task时，系统就会创建这样的task，并将activity放进task里，最后将该task置于前台；如果发现有这样的task，则查询该task中是否已有该activity的实例，如果没有，则将activity置于task栈顶，最后将该task置于前台；如果有这样的实例，则查询是否位于task栈顶，如果在栈顶，则调用该activity实例的onNewIntent方法，并将task置于前台；如果不在栈顶，则销毁该activity实例之上的activity，并调用该activity的onNewIntent方法，并将task置于前台。
2. Handler
3. LayoutView1.onInterceptTouchEvent:ACTION\_DOWN

LayoutView2.onInterceptTouchEvent:ACTION\_DOWN

LayoutView2.onTouchEvent:ACTION\_DOWN

LayoutView1.onInterceptTouchEvent:ACTION\_MOVE

LayoutView2.onTouchEvent:ACTION\_MOVE

...

LayoutView1.onInterceptTouchEvent:ACTION\_UP

LayoutView2.onTouchEvent:ACTION\_UP

onInterceptTouchEvent, onTouchEvent小结：

1)所有都默认

LayoutView1:

onInterceptTouchEvent

LayoutView2:

onInterceptTouchEvent

MyView:

onTouchEvent

LayoutView2:

onTouchEvent

LayoutView1:

onTouchEvent

ACTION\_DOWN

任何View在onTouchEvent的ACTION\_DOWN事件时返回false，则不会收到后续move, up事件。返回true则会直接收到后续的move, up事件。

2)当某个ViewGroup的onInterceptTouchEvent返回true，则会发生截获，事件被传到当前控件的onTouchEvent。

LayoutView1:

onInterceptTouchEvent

LayoutView2:

onInterceptTouchEvent

MyView:

onTouchEvent

LayoutView2:

onTouchEvent

LayoutView1:

onTouchEvent

ACTION\_DOWN

3)当某个ViewGroup的onInterceptTouchEvent, onTouchEvent返回true，则后续move, up事件直接传给该控件的onTouchEvent。

LayoutView1:

onInterceptTouchEvent

LayoutView2:

onInterceptTouchEvent

MyView:

onTouchEvent

LayoutView2:

onTouchEvent

LayoutView1:

onTouchEvent

ACTION\_DOWN, MOVE, UP

DOWN

DOWN

后续move,up

4)当parent view截获move、up事件时，会强制给原先接受ACTION\_DOWN事件的子view发送ACTION\_CANCEL事件。如果parent view的onTouchEvent未处理move、up事件，则对应事件会继续往parent的传递。

1. 当需要将一个对象从一个进程传递到另一个进程时，则需要实现Parcelable接口。
2. 使你的类实现Parcelable接口，并实现相应的方法
3. 添加静态字段CREATOR，CREATOR为一个实现了Parcelable.CREATOR接口的类对象
4. 添加.aidl文件`

