PendingIntent

1.1 知识解析

PendingIntent 是封装了 Intent 的“延迟 Intent”。

PendingIntent 和 Intent 不同的地方在于，把对应的 Intent 交给别的程序，别的程序根据这个 Intent 在后面的别的时间做你安排做的事情。也就是说，一个 Intent 是会被马上用到的，而 PendingIntent 可能在未来的某个时间点被用到。

通常，这个对象会通过其自身的静态方法 getActivity(Context, int, Intent, int)、getBroadcast(Context, int, Intent, int)、getService(Context, int, Intent, int) 被创建。

PendingIntent 可以独立存在，因此即使它所在的应用被终止（killed），它也能被其他接受了这个对象的应用所使用。

如果后续应用又试图产生同样一个 PendingIntent 对象，则它会直接返回前面仍然可用的 PendingIntent 对象。

可以在创建这个 PendingIntent 的应用中使用 PendingIntent 上的 cancel()方法来删除 PendingIntent 对象。

有 4 个方法来获得 PendingIntent 对象：

- public static PendingIntent getBroadcast(Context context, int requestCode, Intent intent, int flags)
- public static PendingIntent getActivity(Context context, int requestCode, Intent intent, int flags)
- public static PendingIntent getService(Context context, int requestCode, Intent intent, int flags)
- public static PendingIntent getActivities(Context context, int

requestCode, Intent[] intents, int flags)

其中：

- requestCode 为用于标识发送 PendingIntent 的发送者，可以通过它来标识不同的 PendingIntent 发送者
- intent：需要发送的信息封装在这里
- Flags 的值有 4 个选择，分别是 PendingIntent 上的 4 个常量：
  - FLAG_CANCEL_CURRENT ：如果 PendingIntent 已经存在，则在创建新的 PendingIntent 之前会先取消当前的 PendingIntent
  - FLAG_NO_CREATE ：如果 PendingIntent 不存在，则返回 null 而不创建它
  - FLAG_UPDATE_CURRENT ：只替换原来的 PendingIntent 中 Intent 的 Extra data 值
  - FLAG_ONE_SHOT：这个 PendingIntent 只能被使用一次

PendingIntent 常用到的三个地方：AlarmManager 设置闹钟、AppWidget 桌面小组件以及 Notification 通知栏通知上。

1.3 实战操作

```
AlarmManager alarm = (AlarmManager) getSystemService(Context.ALARM_SERVICE);
                    Intent intent = new Intent(MainActivity.this, MyBroadcastReceiver.class);
                    PendingIntent pi = PendingIntent.getBroadcast(getApplicationContext(),
112233, intent, PendingIntent.FLAG_ONE_SHOT);
                    alarm.set(AlarmManager.RTC_WAKEUP,
System.currentTimeMillis()+delay*1000, pi);
```

1.4 职业素质

pendingIntent 是一种特殊的 Intent。主要的区别在于 Intent 的执行立刻的，而 pendingIntent 的执行不是立刻的。pendingIntent 执行的操作实质上是参数传进来的

Intent 的操作，但是使用 pendingIntent 的目的在于它所包含的 Intent 的操作的执行是需要满足某些条件的。

主要的使用的地方和例子：通知 Notificatio 的发送，短消息 SmsManager 的发送和警报器 AlarmManager 的执行等等。