# Android Note--- Waiting for the upload

没有目标的学习，会感觉到 后面没什么成果，在1年前，我也打算学习android开发的，但是目的就是学习，到网上去下载很多学习的视频，然后把开发环境搭建起来，能把 Helloworld运行起来，能打些log，Activity之间也能互相切换了，但是后面也就不了了之了，因为不知道学了要干什么。依葫芦画瓢的做了 几个例子，因为里面的问题都是已经解决的，所以也没能深入的系统学习。

## Window

requestWindowFeature(Window.FEATURE\_NO\_TITLE);

//设置成无标题栏用于全屏显示

**getWindow().setFlags(WindowManager.LayoutParams.FLAG\_FULLSCREEN, WindowManager.LayoutParams.FLAG\_FULLSCREEN);**

 //设置窗体全屏

**getWindow().setFlags(WindowManager.LayoutParams.FLAG\_KEEP\_SCREEN\_ON,**

**WindowManager.LayoutParams.FLAG\_KEEP\_SCREEN\_ON);**

 //设置窗体始终点亮

**getWindow().setFlags(WindowManager.LayoutParams.FLAG\_BLUR\_BEHIND,  
                WindowManager.LayoutParams.FLAG\_BLUR\_BEHIND);**

//设置窗体背景模糊

## AlertDialog

**AlertDialog.Builder builder = new AlertDialog.Builder(**

**MainActivity.this);**

**setTitle设置标题  
setIcon设置对话框图标  
setMessage简单消息框  
setItems设置要在对话框中显示的项目列表  
setView设置自定义的对话框样式  
setSingleChoiceItems设置对话框显示一个单选框  
setMultiChoiceItems设置对话框显示一系列的复选框  
setPositiveButton设置对话框显示一个确定的按钮  
setNeutralButton设置对话框显示一个退出按钮  
setNegativeButton 设置对话框显示一个取消按钮  
create创建一个AlertDialog对话框  
show显示对话框**

**LockScreen**

@Override

**public** **boolean** onKeyDown(**int** keyCode, KeyEvent event) {

**if** (keyCode == KeyEvent.*KEYCODE\_BACK*) {

**return** **true**;

//返回false是不吃掉，后面的监听也能得到这个事件，而返回true是吃掉，后面的监听就得不到这个事件了。这里不需要执行父类的点击事件，所以直接return true

} **else** {

**return** **super**.onKeyDown(keyCode, event);

}

}

//按下back键不能返回主界面

## WindowManager.LayoutParams

<http://hubingforever.blog.163.com/blog/static/171040579201175111031938/>

<http://hubingforever.blog.163.com/blog/static/1710405792011758245990/>

## Intent Action

<http://www.cnblogs.com/playing/archive/2010/09/15/1826918.html>

## 连续点击两次退出程序

1、在Activity中设置一个long型的全局变量 firstTime=0，用来记录第一次按下的时间（单位为毫秒）；  
   2、响应Activity的 onKeyUp事件：  
       
@Override   
    public boolean onKeyUp(int keyCode, KeyEvent event) {   
        if (keyCode == KeyEvent.KEYCODE\_BACK) {   
            long secondTime = System.currentTimeMillis();   
            if (secondTime - firstTime > 800) {//如果两次按键时间间隔大于800毫秒，则不退出   
                Toast.makeText(MainActivity.this, "再按一次退出程序...",   
                        Toast.LENGTH\_SHORT).show();   
                firstTime = secondTime;//更新firstTime   
                return true;   
            } else {   
                System.exit(0);//否则退出程序   
            }   
        }   
        return super.onKeyUp(keyCode, event);   
    }

## 快速连击按钮响应事件

**public** **class** HideClick **extends** Thread {

**public** **static** **volatile** **int** *isChecked* = 0;

@Override

**public** **void** run() {

*isChecked*++;

**try** {

Thread.*sleep*(3000);

} **catch** (InterruptedException e) {

e.printStackTrace();

}

**if** (*isChecked* > 0) {

*isChecked*--;

}

**super**.run();

}

}

//调用

@Override

**public** **void** onClick(View v) {

**new** HideClick().start();

**if** (HideClick.*isChecked* >= 5) {

Intent intent = **new** Intent(

android.provider.Settings.*ACTION\_SETTINGS*);

startActivity(intent);

}

}

## 防止按钮连续点击

public class Utils {  
    private static long lastClickTime;  
    public static boolean isFastDoubleClick() {  
        long time = System.currentTimeMillis();  
        long timeD = time - lastClickTime;  
        if ( 0 < timeD && timeD < 500) {     
            return true;     
        }     
        lastClickTime = time;     
        return false;     
    }  
}  
  
调用  
public void onClick(View v) {  
    if (Utils.isFastDoubleClick()) {  
        return;  
    }  
}

## 锁屏

<http://www.2cto.com/kf/201404/296561.html>

## Android程序框架

## 长按的两种实现

### ContextMenu

>>ContextMenu和OptionsMenu相比主要有以下区别：

1，ContextMenu必须通过Activity的[registerForContextMenu(View)](http://developer.android.com/reference/android/app/Activity.html" \l "registerForContextMenu%28android.view.View%29)来进行注册，而OptionsMenu不用。

2，ContextMenu不支持icon,而OptionsMenu支持。

3，ContextMenu可以有头，可以通过setHeaderIcon，setHeaderTitle，setHeaderView来设置头，否则就没有头。

4，弹出的方式不一样。

5，Options Menu的拥有者是Activity，而上下文菜单的拥有者是Activity中的View。每个Activity有且只有一个Options Menu，它为整个Activity服务。而一个Activity往往有多个View，并不是每个View都有上下文菜单，这就需要我们调用[registerForContextMenu](http://developer.android.com/reference/android/app/Activity.html#registerForContextMenu%28android.view.View%29)([View](http://developer.android.com/reference/android/view/View.html) view)来指定。

注意事项：ContextMenu的任何项在被选中后，整个ContextMenu会自动被系统关闭掉，即使多选项也是这样。这样的话，多选项的行为就有些怪异。查看Android的源码后，发现这是个Android的bug,应该是google没考虑多选项的情况。

1.在activity的onCreate(...)方法中为一个view注册上下文菜单

2.在onCreateContextMenuInfo(...)中生成上下文菜单。

3.在onContextItemSelected(...)中响应上下文菜单项。

eg:

@Override

**protected** **void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_main*);

**this**.registerForContextMenu(findViewById(R.id.*button*));

}

@Override

**public** **void** onCreateContextMenu(ContextMenu menu, View v,

ContextMenuInfo menuInfo) {

// menu.setHeaderTitle("title");

menu.add(0, 1, Menu.*NONE*, "发送");

menu.add(0, 2, Menu.*NONE*, "标记为重要");

}

@Override

**public** **boolean** onContextItemSelected(MenuItem item) {

**switch** (item.getItemId()) {

**case** 1:

Toast.*makeText*(**this**, "!!!!!", Toast.*LENGTH\_SHORT*).show();

**break**;

**case** 2:

**break**;

**default**:

**return** **super**.onContextItemSelected(item);

}

**return** **true**;

}

### setOnLongClickListener

findViewById(R.id.*button*).setOnLongClickListener(

**new** OnLongClickListener() {

@Override

**public** **boolean** onLongClick(View arg0) {

Toast.*makeText*(MainActivity.**this**, "~~~~~", 1).show();

**return** **true**;

}

});

## S

<http://www.oschina.net/code/snippet_1014520_27327>

<http://blog.csdn.net/yueqinglkong/article/details/44781103>

<http://zhidao.baidu.com/link?url=WV254JoXIUsnIo8YtiVA9c6Y7bWJT0o0bSTVNv-5IIMCqYxk9bg0JnJh13WGvdzorQ1mk18Wz09Q6P5p5BFhLE_ISnt86QCrMq7MNOqwHbW>

<http://www.xuebuyuan.com/2028919.html>

<http://blog.csdn.net/e421083458/article/details/17192513>

<http://www.tuicool.com/articles/ERvAfyn>

<http://blog.csdn.net/xyz_lmn/article/details/8132420>

<http://download.csdn.net/download/u014578633/8685433>