WindowManager是Android中一个重要的服务(Service)。WindowManagerService 是全局且唯一的。它将用户的操作翻译成指令，发送给呈现在界面上的各个Window。Activity会将顶级的控件注册到 WindowManager中，当用户触碰屏幕或键盘的时候，WindowManager就会被通知到，而当控件有一些请求产生，也会经ViewParent送回到WindowManager中。从而完成整个通信流程。

整个Android的窗口机制都是基于WindowManager的，通过它可以添加view到屏幕，也可以从屏幕删除view。它面向的对象一端是屏幕，另一端就是View，通过WindowManager的addView()方法添加View，这样产生出来的View根据WindowManager.LayoutParams属性的不同，效果也不同。比如创建系统顶级窗口，实现悬浮窗口效果等。WindowManager的方法很简单，最常用的有3个：addView()、removeView()、updateViewLayout()。

**public class MainActivity extends AppCompatActivity** {  
 **private Button button**;  
 @Override  
 **protected void** onCreate(**Bundle** savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(**R.layout**.***activity\_main***);  
  
 **button** = (**Button**) findViewById(**R.id**.***button***);  
 **button**.setOnClickListener(**new View.OnClickListener**() {  
 @Override  
 **public void** onClick(**View** v) {  
  **// 1.使用java代码生成一个Button  
 Button** btn = **new** Button(getApplicationContext());  
 **// 2.通过getSystemService()方法得到WindowManager实例** WindowManager wm = (WindowManager) getSystemService(**Context**.***WINDOW\_SERVICE***);  
   
 **// 3.定义WindowManager.LayoutParams对象并为其设置属性  
 WindowManager.LayoutParams** layoutParams = **new** WindowManager.LayoutParams();  
 layoutParams.**type** = 2002;  
 layoutParams.**format** = 1;  
 layoutParams.**flags** = 40;  
 layoutParams.**width** = 40;  
 layoutParams.**height** = 40;  
   
 **// 4.调用WindowManager的addView()方法添加View到屏幕中，View的属性由layoutParams控制** wm.addView(btn, layoutParams);  
 }  
 });  
 }  
}