## 常用代码

### 隐藏标题栏

**getSupportActionBar**().**hide**();

### 去掉状态栏

**getWindow**().**setFlags**(WindowManager.LayoutParams.*FLAG\_FULLSCREEN*,  
 WindowManager.LayoutParams.*FLAG\_FULLSCREEN*);  
**setContentView**(R.layout.*activity\_main*);

### 延时跳转

mHandler = new Handler();  
runnable = new Runnable() {  
 **@Override** public void run() {  
 Intent intent = new Intent(MainActivity.this, LoginViewImpl.class);  
 **startActivity**(intent);  
 MainActivity.this.**finish**();  
 }  
};

//通过handler的postDelayed()方法实现  
mHandler.**postDelayed**(runnable,*DELAYED\_TIME*);

### 得到Spinner下拉框的值

A.

*// 类别Spinner*sp\_type = (Spinner) **findViewById**(R.id.*sp\_type*);  
**queryDeviceList**();  
ArrayAdapter deviceTypeAdapter = new ArrayAdapter(this, android.R.layout.*simple\_dropdown\_item\_1line*, mDeviceTypeList);  
sp\_type.**setAdapter**(deviceTypeAdapter);  
sp\_type.**setOnItemSelectedListener**(new AdapterView.OnItemSelectedListener() {  
 **@Override** public void onItemSelected(AdapterView<?> **parent**, View **view**, int **position**, long **id**) {  
 mDeviceTypeList.**get**(**position**).**toString**(); *// 得到下拉框选项的name* }  
  
 **@Override** public void onNothingSelected(AdapterView<?> **parent**) {  
  
 }  
});

B.

**sp\_type**.**getSelectedItem**().**toString**();

A跟B的区别主要是A每次下拉选择都能获取，但B只能获取一次。

## 工具类

### Toast工具类

public class ToastUtil {  
 private static Toast *toast*;  
 public static void showShortToast(Context **context**, String **content**) {  
 if (*toast* == null) {  
 *toast* = Toast.**makeText**(**context**, **content**, Toast.*LENGTH\_SHORT*);  
 } else {  
 *toast*.**setText**(**content**);  
 }  
 *toast*.**show**();  
 }  
 public static void showLongToast(Context **context**, String **content**) {  
 if (*toast* == null) {  
 *toast* = Toast.**makeText**(**context**, **content**, Toast.*LENGTH\_LONG*);  
 } else {  
 *toast*.**setText**(**content**);  
 }  
 *toast*.**show**();  
 }  
}

### Activity工具类

public class ActivityUtil {  
 private static List<Activity> *mActivityList* = new ArrayList<Activity>();  
 public static void addActivity(Activity **activity**){  
 *mActivityList*.**add**(**activity**);  
 }  
 public static void removeActivity(Activity **activity**){  
 *mActivityList*.**remove**(**activity**);  
 }  
 public static void finishAll(){  
 for (Activity activity:*mActivityList*){  
 if (!activity.**isFinishing**()){  
 activity.**finish**();  
 }  
 }  
 }  
}