

# Android项目总结

Android项目总结，最近在公司里，写了几个Android的小项目，在学习过程中学习了一些新的东西，也学到了很多，所以打算记录一下。

**1. 涉及到多选和单选的时候，可以用checkbox和radiobutton，要是有必选项目的时候可以用&&符号来判断，确保用户都已经选择了。**

**2. 想检测editext输入的情况的时候，可以用mEditText.addTextChangedListener();**

```
erIdCode.addTextChangedListener(new TextWatcher() {  
    @Override  
    public void beforeTextChanged(CharSequence sequence, int i, int i1,  
int i2) {  
  
    }  
  
    @Override  
    public void onTextChanged(CharSequence sequence, int i, int i1, int  
i2) {  
  
    }  
  
    @Override  
    public void afterTextChanged(Editable editable) {  
  
    }  
}
```

用这个方法，可以监测到用户输入的过程，并且可以同时获取到用户的操作信息。

**3. 获取手机imei码**

```
//需要的权限  
<uses-permission android:name="Android.permission.READ_PHONE_STATE" />  
  
//获取手机唯一标识符的代码  
  
TelephonyManager telephonyManager = (TelephonyManager)  
this.getSystemService(this.getApplicationContext().TELEP  
HONY_SERVICE);  
String imei = telephonyManager.getDeviceId();
```

**4. 将时间戳转换为时间**

```

    /**
     * 将时间戳转换为时间
     */
    public static String stampToDate(String s) {
        String res;
        SimpleDateFormat simpleDateFormat = new SimpleDateFormat("yyyy-MM-dd
HH:mm");
        long lt = new Long(s);
        Date date = new Date(lt);
        res = simpleDateFormat.format(date);
        return res;
    }

```

## 5. 将时间转换为时间戳

```

    /**
     * 将时间转换成时间戳
     */
    public static String dateToStamp(String s) throws ParseException {
        String res;
        SimpleDateFormat simpleDateFormat = new SimpleDateFormat("yyyyMMdd");
        Date date = simpleDateFormat.parse(s);
        long ts = date.getTime();
        res = String.valueOf(ts);
        return res;
    }

```

## 6. 设置edittext的提示文字的颜色和字体大小

```

        harSequence hint = mCode.getHint();
        SpannableString ss = new SpannableString(hint);
        AbsoluteSizeSpan ass = new AbsoluteSizeSpan(17, true);
        mCode.setHintTextColor(0xffeeeeee);
        ss.setSpan(ass, 0, ss.length(), Spanned.SPAN_EXCLUSIVE_EXCLUSIVE);
        mCode.setHint(new SpannedString("短信验证码"));

```

## 7. 让Fragment拥有和Activity一样的onResume事件

```

protected boolean isCreate = false;

@Override
public void onCreate(@Nullable Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    isCreate = true;
}

@Override
public void setUserVisibleHint(boolean isVisibleToUser) {
    super.setUserVisibleHint(isVisibleToUser);
    if (isVisibleToUser && isCreate) {
        initData(true, 1);
        LinLog.lLog("====lin====> onResume");
        //相当于Fragment的onResume
    }
}

```

```
        //在这里处理加载数据等操作
    } else {
        //相当于Fragment的onPause
    }
}
```