1. 搭建Android开发环境
   1. 环境组成介绍

搭建android应用开发环境需要搭建以下工具：

1. JDK（Jave Development Kit）
2. Ecilpse
3. Android SDK（Software Development Kit）
4. ADT （Android Development Tools）
   1. 安装JDK
5. Jdk官方下载地址：http:// [www.oracle.com/technetwork/java/javase/downloads/index.html](http://www.oracle.com/technetwork/java/javase/downloads/index.html)
6. Jdk安装（jdk集成了jre）

Jdk是java开发环境；jre是java运行环境

（安装过程中，需要选择安装路径）

1. 检测jdk是否成功安装：在cmd下输入 java -version
   1. 集成环境下载

集成环境下载地址：

windows（32位） : http://pan.baidu.com/s/1ixc9S 密码: vsot

windows（64位） : http://pan.baidu.com/s/1gukk2 密码: mltn

mac系统       : http://pan.baidu.com/s/1kTKas8v 密码: n2p7

* 1. 环境变量介绍

桌面—右键“属性”—高级系统设置—环境变量

环境变量的配置

1. JAVA\_HOME：C:\Program Files\Java\jdk1.8.0\_45
2. CLASSPATH：.;%JAVA\_HOME%\lib\tools.jar;%JAVA\_HOME%\lib\dt.jar;%JAVA\_HOME%\bin;
3. Path:在开始追加%JAVA\_HOME%\bin

JAVA\_HOME : 指明 JDK 的安装路径（注意：要配置成你的JDK安装目录）

CLASSPATH : 指明 java 虚拟机要装载类的路径

PATH : 指明系统可以在任何路径下识别java 命令

* 1. 安装ADT

ADT安装方式：

1. 在eclipse的插件安装界面中通过在线安装，在线安装地址：<http://dl-ssl.google.com/android/eclipse/>
2. 通过下面地址下载ADT压缩包，本地安装：<http://developer.android.com/sdk/installing/installing-adt.html>
3. Android项目结构介绍

2-1 新建android项目（File—New—Android Application Project）

Application Name: 应用名

Project Name： 项目名

Pacakge Name：包名

2-2 项目结构介绍



1. 在界面中显示以及输入文本信息

3-1 控件概述

TextView：显示文本框控件

EditView：输入文本框控件

3-2 控件属性

TextView控件的常用属性：

android:id----控件的id

android:layout\_width----控件的宽度

android:layout\_height---控件的高度  
  
 android:text----文本内容

android:textSize----文本大小

android:textColor----文本颜色

android:background----控件背景

EditText控件的常用属性：

android:id----控件的id

android:layout\_width----控件的宽度

android:layout\_height---控件的高度

android:text----文本内容

android:textSize----文本大小

android:textColor---文本颜色

android:background----控件的背景

android:hint ----输入提示文本

android:inputType ----输入文本类型

wrap\_content 、match\_parent、fill\_parent的区别：

wrap\_content: 包裹实际文本内容（自适应内容）

match\_parent: 当前控件铺面父类容器：2.3api之后添加的属性值

fill\_parent: 当前控件铺面父类容器：2.3之前的属性值

3-3 TextView实现跑马灯效果

1. 跑马灯：

第一步：android:focusable="true"

android:focusableInTouchMode="true"

android:singleLine="true"

android:ellipsize="marquee"

第二步: 自定义一个继承TextView的类

1. android:ellipsize属性

android:ellipsize=”start”—–省略号显示在开头 "...pedia"

android:ellipsize=”end”——省略号显示在结尾 "encyc..."

android:ellipsize=”middle”—-省略号显示在中间 "en...dia"

android:ellipsize=”marquee”–以横向滚动方式显示(需获得当前焦点时)

1. 在界面中显示图片

4-1 ImageView的概述

ImageView：显示图片的控件

属性：

1. android:src="@drawable/ic\_lancher"—内容
2. android:background="@drawable/ic\_launcher"—背景图
3. android:background="#00ff00"—RGB颜色

4-2 使用ImageView显示图片

ImageView 中的src属性和background属性区别与联系

1. src: 尺寸不随控件大小而改变,不填充；单独想要一个图标时使用
2. background: 尺寸始终和控件大小相同（还可以实现#f0f0f0，此时要设置高度）。需要给图标设置背景时使用

这两个属性可以一起使用

4-3 不同分辨率下图片的显示

drawable-ldpi低分辨率

drawable-mdpi中等分辨率

drawable-hdpi 高分辨率

drawable-xhdpi比hdpi高一点的分辨率

drawable-xxhdpi最高的分辨率

ImageView控件引用的图片在drawable文件夹中（文件夹有多个，因对应不同的手机分辨率，但名称与格式必须一样）：程序会自动根据手机分辨率选择显示相应分辨率drawable文件夹下的图片。

1. 认识按钮Button以及ImageButton

5-1 Button概述

Button和ImageButton特征

1. 共有的特征：

都可以作为一个按钮产生点击事件

都有background属性

1. 不同点：

Button 有text 属性 , Imagrbutton 没有text属性

Button 没有src属性，Iimagebutton有src属性

1. 产生明显的点击效果：

产生一个click效果

5-2 使用按钮Button

注意1：所有控件的文字最好不要直接在xml中写，而是先把字符串按照一定格式写在res\values\strings.xml里面，然后在xml中通过字符串的id引用使用。

注意2：res文件下的所有资源都会对应在gen目录R.java下生成一个资源ID。

5-3 适应按钮ImageButton

1. 监听按钮的点击事件

6-1 onClick事件

1. Button 和 ImageButton都拥有一个 onclick 事件，通过自身的 .setOnClickListener(OnclickListener) 方法添加点击事件
2. 所有控件都有onclick事件
3. 通过点击事件的监听可以实现点击相应按钮之后要发生什么动作
4. 监听事件实现的几种写法: 匿名内部类的实现、独立类的实现、实现接口的方式来实现。

6-2 匿名内部类Button的监听事件

1. 初始化控件：loginButton = (Button) findViewById(R.id.button)返回的是View的对象。

注意：res下的资源都会在gen目录下的R文件中自动生成一个唯一的id，通过R调用id来查找res下的资源

1. 设置监听器：

longinButton.setonClickListener(new OnClickListeenrr(){

publick void onClick(View arg0){

//在当前onClick方法中的监听点击Button的动作

System.out.println("我的Button被点击了");

}

})

6-3 外部类监听点击事件

1. Toast.makeTect(this," string ", Toast.LENGTH\_LONG).show();

ToToast:是一个类，主要管理消息的提示。

makeText()，是Toast的一个方法，用来显示信息，分别有三个参数。

第一个参数：this，是上下文参数，指当前页面显示

第二个参数：“string ”是你想要显示的内容。

第三个参数：Toast.LENGTH\_LONG，提示消息显示的时间（ToastLENGTH\_SHORT）。

show()，表示显示这个Toast消息提醒。

1. v.setAlpha(float) 设置透明度，0f--1f;

6-4 接口方式监听按钮点击事件

接口方式实现监听事件:

1. 实现一个接口：implements OnClickListener--在类里面会生成一个方法
2. 在onclick里面打印一个数据log.i("tag","第三种方式实现")
3. 将监听器绑定到button上---用imagebutton举例
4. 初始化private ImageButton imgbt；
5. 将imgBt=(ImageButton)findViewById(R.id.imageButton1);
6. 设置一个监听器：imgBt.setOnClickListener(this);//this就是访问当前接口的对象。

三种监听方式的对比：

1. 匿名内部类：直接在OnClickListener中new一个匿名内部类
2. 独立类实现：把OnClickListener单独地在外边实现一个--new一个对象，这个对象是可以被当前方法访问的
3. 通过实现一个接口的方式实现监听方式：在类上实现一个接口implements OnClickListener
4. 实现动态自动匹配输入的内容

7-1 AutoCompleteTextView

1. 功能：动态匹配输入的内容
2. 独特属性：android：completetionThreshold="2"//设置输入多少字符时自动匹配

7-2 AutoCompleteTextView实现自动匹配输入的内容

1. 初始化AutoCompleteTextView控件
2. 需要一个适配器：ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,android.R.layout.simple\_list\_item\_1,objects)
3. 初始化数据源（适配器中的objects），用数据源去匹配文本框输入的内容
4. 将adapter与当前AutoCompleteTextView绑定：AutoCompleteTextView对象.setAdapter(adapter)
5. 要设置控件属性android:completethreshold控制输入几个字符时开始进行匹配

7-3 MultiAutoCompleteTextView

1. 功能：可支持多个值（在多次输入的情况下），分别用分隔符分开，并且在每个值选中的时候再出输入值时会自动去匹配，可用在发短信发邮件时选择联系人这种类型当中
2. 独特属性：android:completionThreshold = "2" 设置输入多少字符时自动匹配
3. 设置分隔符：mtxt.setTokenizer(new MultiAutoCompleteTextView.CommaTokenizer());

7-4 MultiAutoCompleteTextView实现自动匹配输入的内容

1. 初始化控件，记住findViewById(R.id.\*\*)方法
2. 需要一个适配器
3. 初始化数据源-->此数据源去匹配文本框内容
4. 将Adapter与MultiAutoCompleteTextView绑定，调用setAdapter()方法
5. 设置输入几个字符后才进行自动提示
6. 设置分隔符
7. 使用多状态按钮ToggleButton

8-1 ToggleButton概述

1. 什么是ToggleButton

ToggleButton：有选中和未选中两种状态，分别显示不同的文本。

1. ToggleButton属性：

android:checkeced="true"

android:textOff="关"

android:textOn="开"

通过ToggleButton操纵其他控件，来显示（切换）不同的状态。

8-2 使用ToggleButton按钮实现开关效果

使用ToggleButton按钮实现开关效果----来切换其他控件的状态：

1. 放一个ToggleButton；textOn：当点击后显示的文字 textoff同理
2. 放一个ImageView
3. 将2个控件初始化
4. 给当前的tb设置监听器：tb.setOnCheckedChangeListener(this);
5. 判断条件：（A？B：C）
6. 当点击这个tb 的时候，更换img的背景

onCheckedChanged(CompoundButton buttonView,boolean isChecked)方法：

//buttonView：代表被点击控件的本身

//ischecked--代表被点击的控件的状态

1. 使用CheckBox

9-1 checkBox概述

CheckBox:复选框

1. 两种状态：选中状态（true）&未选中状态（false）
2. 属性：

android:id="@+id/checkbox"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:checked="false"

android:text="男"

自定义CheckBox的样式应该设置下列style属性

9-2 使用checkBox

CheckBox的checked属性默认是false

1. CheckBox不管是选中还是取消，动作都可被监听到，打印isCheck时需要加上一个字符串
2. 使用getText（）方法获取CheckBox文本内容后，还需toString()把其转换为String对象
3. 使用RadioGroup与RadioButton

10-1 RadioGroup和RadioButton的概述

1. RadioGroup：是一个集合，可以提供多选一机制
2. 属性：android:orientation:vertical/ horizontal

决定当前RadioGroup和RadioButton以什么形式排列

RadioButton选中后不可取消，当选中其他按钮后会自动取消。

10-2 使用RadioGroup和RadioButton

1. 用onCheckedChanged方法来监听，监听参数是checkedId，用以判断哪个RadioButton被选中。导入的包有两个，要找对应于RadioGroup的那个
2. 单独的RadioButton并不拿来使用，更多的是依靠RadioGroup来实现多选一的功能
3. 设置监听的过程与其它控件相同，都是:取得对象->添加监听器->注册监听器->修改监听方法

RadioGroup中的RadioButton状态改变既可以通过RadioButton来监听也可以通过RadioGroup来监听:

1. 通过RadioButton：

定义一个RadioButton对象，然后设置其监听器。

private RadioButton rb;

rb.setOnCheckedChangeListener(this);

1. 通过RadioGroup：

定义一个RadioGroup对象，然后设置其监听器。

1. 五布局之线性布局LinearLayout

11-1 理解LinearLayout

1. LinearLayout：把控件以纵向或横向的方式排列
2. LinearLayout常用属性
3. android:orientation = “vertical” —— 该属性决定它子类控件的排布方式（vertical：垂直；horizontal：水平）
4. android:gravity = “center” —— 该属性决定它子类控件的xy的位置

center\_vertical：垂直（Y轴）居中

center\_horizontal：水平（X轴）居中

center：垂直水平都居中

right：子类控件位于当前布局的右

left：子类控件位于当前布局的左

buttom：子类控件位于当前布局的下面

11-2 使用LinearLayout

1. 子类控件在LinearLayout中常用到的属性
2. android:gravity：决定其子类控件在该控件里的x、y的位置

（作为LinearLayout的属性使用）

1. android:layout\_gravity：该控件本身在父容器的x、y的位置

（作为LinearLayout的子控件的属性来使用）

1. android:layout-weight="1" 本身控件所在父容器的比例

注意：

如果Android:layout\_height:" wrap\_content" 则是按照android:layout\_weight的正比例进行；

如果Android:layout\_height:"match\_parent" 则是按照android:layout\_weight的反比例进行。

1. 五布局之相对布局RelativeLayout

RelativeLayout：其所包含的子控件将以控件之间的相对位置或者子类控件相对父类容器的位置的方式排列。

1. 五布局之帧布局FrameLayout

FrameLayout：在这个布局中，所有的子元素都不能被指定放置的位置，它们统统放在这块区域的左上角，并且后面的子元素直接覆盖在前面的子元素之上，将前面的子元素部分和全部遮挡。

android:layout\_gravity 此属性在相对布局中是无效的。

1. 五布局之绝对布局AbsoluteLayout
2. 五布局之表格布局TableLayout
3. 认识Activity

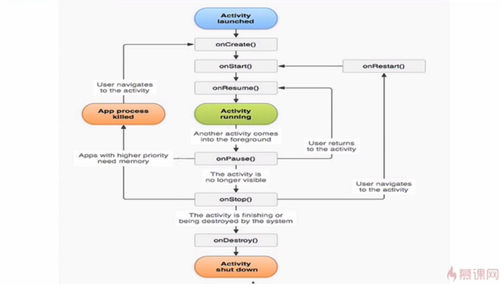
Activity:是一个应用程序组件，提供用户与程序交互的界面

Android的四大组件:

——Activity

——Service

——BroadcastReceiver

——Content Provider

1. 使用Intent实现页面跳转
2. App签名打包

签名的意义：

1. 为了保证每个应该程序开发者的合法
2. 防止部分人通过使用相同的Package Name来混淆替换已经安装的程序，从而出现一些恶意窜改
3. 保证我们每次发布的版本的一致性（如自动更新不会因为版本不一致而无法安装）
4. 使用SDK开发文档
5. 综合练习——制作简易计算器