

第一天

1. `Android`是基于(**linux**) 的移动端开源操作系统
2. `Android`系统是有(**谷歌**)公司推出的
3. `Android11`对应的 API 编号是(**30**)
4. `APP`除了在手机上运行,还能在电的`模拟器(简称 AVD)`运行
5. `Android Studio`创建模拟器的管理工具名为(**Device Manager**)
6. 第一部`Android`手机由(**HTC**)制造
7. `Android Studio`由(**idea**)演变而来
8. `Android Studio`只能在(**64 位操作**)系统上运行
9. 运行`APP`是指运行某个模块,而非运行某个项目
10. `APP`不能在电脑上直接运行,要在模拟器上运行
11. 智能手机的两大操作系统是 (**Android 和 IOS**)
12. 下列那些设备可以运行`Android`操作系统(**智能电视, 平板电脑, 智能手机, 车载大屏**)
13. `Android`提供的 App 专用开发工具包名为(**SDK**)
14. `Android`APP 开发的主要编程语言是(**JAVA**)
15. 打印调试级别的日志方法名为(**Log.d**)
16. 列出导入 APP 工程的几种方式
 - a) 第一种,直接导入一个工程
 - b) 第二种,导入一个模块

第二天

1. 除了开启开发者选项之外,还需打开手机上的(**USB**)调试开关,然后才能在手机上调试 App。
2. App 开发的两大技术路线包括(**原生开发**)和(**混合开发**)。
3. App 工程的编译配置文件名为(**build.gradle**)。
4. Android Studio 使用(**gradle**)工具完成 App 工程的构建操作。
5. 在 Java 代码中调用(**startActivity**)方法能够跳到新的 App 页面。
6. Android Studio 默认支持到 Java 8。
7. Kotlin 语言也能用于 App 开发。
8. App 属于(**客户端**)程序。
9. 一个 App 项目可以包含多个 App 模块。
10. App 工程的图片资源放在(**drawable**)目录下。
11. 通过 (**USB 接口**) 可以连接手机和电脑
12. 如果手机无法安装调试 App,可能是哪个原因造成的(a:**处于锁屏状态**, b:**选择了充电模式**)
13. App 可以直接连接的数据库是(**SQLite**)
14. App 界面布局采用的文件格式是(**XML**)
15. (**id**) 属性表示 TextView 标签的控件编号

第三天

1. (`res/values`)目录下面存放字符串定义的资源文件名
2. (`dp`)指的是与设备无关的显示单位
3. Android 的色值 alpha 透明度和(`RGB`)三色联合定义
4. 线性布局利用属性 `layout_weight` 设置下级控件的尺寸权重时,要将下级控件的宽度设置为(`0dp`)
5. 按钮控件被按住超过(`500 毫秒`)之后,会触发长按事件
6. Android 控件类都有(`View`)派生而来
7. 线性布局 `LinearLayout` 默认下级控件在(`水平方向排列`)
8. 在相对布局内部,如果不设定下级视图的参照物,那么下级视图默认显示在布局 (`左上角`)
9. 滚动视图 `scrollView` 默认下级布局在(`垂直`)方向排列
10. 按钮控件上的英文默认显示(`大写字母`)
11. Java 代码中, `setTextSize` 方法默认的字号单位是(`sp`)
12. 网格布局 `GridLayout` 指定网格行数的属性名称是(`rowCount`)
13. 图像视图财务缩放类型(`FIT_XY`)的时候,图像可能会被拉伸变形
14. 图像按钮 `ImageButton` 由(`ImageView`)派生而来
15. 在按钮控件中,把图片放在文本右边属性名称是(`drawableRight`)

第四天

1. 打开一个新页面,新页面的生命周期周期方法依次为 `onCreate->onStart->onResume`
2. 关闭现有的页面,现有页面的生命周期方法依次为 `onPause->onStop->onDestroy`
3. `Intent` 意图对象的(`action`)方法用于指定意图的动作行为
4. 上一个页面要在(`onActivityResult`)方法中处理下一个页面返回的数据
5. 在 `AndroidManifest.xml` 的 `activity` 节点添加(`meta-data`)标签,表示给该活动设置元数据信息
6. 活动也处于(`活跃`)状态时,允许用户在界面上输入文字
7. 设置了启动标志 `intent.FLAG_ACTIVITY_SINGLE_TOP` 之后,当栈顶为待跳转的活动实例之时,会(`重用栈顶的实例`)
8. (`显式 intent`)直接指定来源活动与目标活动,他属于(`精确匹配`)
9. 调用(`startActivityForResult`)方法也能获得下个页面返回的意图数据
10. 在桌面(`长按应用图标`),会弹出该应用的快捷方式菜单(如有配置的话)
11. 在当前页面调用(`finish`)方法会回到上一个页面
12. 从 A 页面跳到 B 页面,再从 B 页面返回 A 页面,此时 A 页面会先执行(`onRestart`)方法
13. 栈是一种(`先进后出`)的数据结构
14. `Bundle` 内部用于存放消息的数据结构是(`Map`)
15. (`name, resource, value`)是 `meta-data` 标签拥有的属性

第五天

1. 图形描述文件的扩展名是(`xml`)
2. 形状图形 `shape` 的下级节点(`size`)描述了形状图形的宽高尺寸
3. 由复合按钮 `compoundButton` 派生而来的控件包括(`CheckBox`、`RadioButton`)和 `Switch`
4. `EditText` 的属性(`maxLength`)可指定文本允许输入的最大长度
5. 输入法软键盘有系统服务(`INPUT_METHOD_SERVICE`)管理
6. (形状图形)可以描述(圆角)的定义
7. 单选组 `RadioGroup` 默认内部控件在(垂直)方向排列
8. 首次点击编辑框,就会触发它的(`focus`)事件
9. 提醒对话框 `alertDialog` 支持同时设置(3个)按钮
10. 时间对话框会显示当前的(时、分)
11. 状态列表图形的(`state_pressed`)属性用于描述是否按下的图形列表
12. 在一组按钮中只选择其中一个按钮,应当选用(`RadioButton`)控件
13. 若想让编辑框 `EditText` 输入数字密码,则要将 `inputType` 属性设置为(`numberPassword`)
14. 若想在编辑框的文本改变之后补充处理,应当在(`afterTextChanged`)方法中增加代码
15. 日期选择对话框上能够看到那些时间单位(年、月、日、星期)

第六天

1. `SharedPreferences` 采用的存储结构是(key/value)的键值对方式。
2. Android 可以直接操作的数据库名为 (`SQLite`)。
3. (`SQLiteDatabase`) 是 Android 提供的 `SQLite` 数据库管理器。
4. 数据库记录的修改动作由 (`update`) 命令完成。
5. 为了确保在 App 运行期间只有唯一的 `Application` 实例, 可以采取 (单例模式)实现。
6. 共享参数只能保存(字符串、`int`、`long` 等)类型的数据。
7. `SQLite`(不可以)直接读写布尔类型(备注:要转为 `int` 类型)的数据。
8. 从 Android 7.0 开始, 系统(默认禁止)App 访问公共存储空间。(备注:需要给权限)
9. App 在私有空间上读写文件(无须任何授权)。
10. App 终止时(不会)调用 `Application` 的 `onTerminate` 方法。
11. (全局变量)不是持久化的存储方式。
12. (共享参数、数据库、文件)持久化的存储方式
13. DDL 语言包含哪些数据库操作 (创建表、删除表、修改表结构)。
14. 调用 (`query`、`rawQuery`) 方法会返回结果集的 `Cursor` 对象。
15. 位图工厂 `BitmapFactory` 的(`decodeStream`、`decodeFile`、`decodeResource`)方法支持获取图像数据。
16. 已知某个图片文件的存储卡路径, 可以调用(`setImageURL`)方法将它显示到图像视图上。

第七天

1. 在 AndroidManifest.xml 里面声明内容提供器的标签名称是(contentprovider)
2. 在互动代码中调用 getContentResolver 方法,得到的是(getContentResolver)实例
3. Manifest.permission.READ_CONTACTS 表示(读取联系人)权限
4. MediaStore.Images.Media.DATA 保存了媒体库中图片文件的(路径)
5. ContentProvider 属于中间接口,本身并(不直接保存数据)
6. 内容解析器 ContentResolver 是客户端 App 操作(服务端数据的工具)
7. 只要调用 ContentResolver 的(2)次 insert 方法,就能向通讯录写入一条联系人数据
8. 内容观察器 ContentObserver(能够实时获取)新增的数据
9. 短信和彩信都只能发送文本内容(还有图片)
10. 内容组件由 3 个部分组成(contentProvider、contentObserver、contentResolver)
11. App 读取短信需要申请(READ_SMS)权限
12. Content://mms 是(彩信)的内容路径
13. FileProvider 的 getUriForFile 方法返回的数据是(uri)类型
14. 安卓 App 安装包的文件扩展名是(APK)

第八天

1. Spinner 是种多选 (1) 的下拉框控件。
2. 若想在页面中部弹出 Spinner 的列表对话框, 要把 spinnerMode 属性设置为(dialog)。
3. 在 XML 文件中, 如果 ListView 后面还有其他平级的控件, 就要将 ListView 的高度设为 (0dp), 同时权重设为 1 , 确保列表视图扩展到剩余的页面区域。
4. 翻页视图 ViewPager 设置当前页面的方法是(setCurrentItem)。
5. Fragment 有两种注册方式, 分别是(静态注册)和(动态注册)。
6. 简单适配器只能展示纯文本列表(还有图片)。
7. 列表视图只支持列表项的点击事件, (也可以)支持列表项的长按事件。
8. 网格视图只能指定(列数)。
9. 引入翻页标签栏 PagerTabStrip, 它能够在翻页视图上方显示 (页面标题)。
10. 采取动态注册方式的时候, 碎片需要配合翻页视图才能正常使用(这句话描述不恰当)。
11. 下拉框可使用(数组适配器、简单适配器、基本适配器)
12. 从 BaseAdapter 派生的数据适配器, 要在 (getView) 方法中补充各控件的处理逻辑。
13. 在列表视图当中, 若想不让列表中的控件抢占列表项的焦点, 应当将内部视图的焦点抢占方式设置为(blockDescendants)
14. 在网格视图当中, 若想让每行的剩余空间均匀分配给该行的每个网格, 应当将拉伸模式设置为(spacingWidthUniform)
15. 若想让翻页视图在滚动结束后触发某种动作,应当重写翻页适配器的(onPageSelected)方法。

第九天

1. 活动只能一对一通信,而广播可以(**一对多**)通信
2. 通过静态方式注册广播,就要在 AndroidManifest.xml 中添加名为(**receiver**)的接收器标签
3. (**PendingIntent**)代表延迟的意图,它指向的组件不会马上激活
4. 手机的屏幕方向默认是(**竖屏**)
5. 开启(**画中画模式**)模式之后,可将 APP 界面缩小为屏幕上的一个方块
6. 标准广播是(**无序的**),有可能后面注册的接收器反而比前面注册的接收器(**先收到广播**)
7. 通过 setPriority 方法设置优先级,优先级(**越大**)的接收器,越先收到有序广播
8. 普通应用能够通过(**动态注册**)方式来监听系统广播
9. 闹钟管理器 AlarmManager 的 setRepeatIn 方法保证能够按时发送广播(**API19 之后不行**)
10. 旋转手机使得屏幕(**由竖屏变为横屏**),App 默认会重新加载整个页面(先销毁原页面再创建新页面)
11. 在接受器内部调用(**abortBroadcast**)方法,就会中断有序广播
12. Android.permission.VIBRATE 表达的是(**震动器**)权限
13. 网路类型(**TYPE_WIMAX**)表示手机的数据连接(含 2G/3G/4G/5G)
14. 网路状态(**CONNECTED**)表示已经连接
15. (**orientation、screenLayout、screenSize、keyBoard**)属于 configChanges 属性配置的显示变更豁免情况