

文档名称	安卓知识合辑
文档标识	MFOAK-LG
版本	Beta 0.0.2
修改日期	2018.11.17

目录

1. 基础知识.....	1
1.1. Java 基础.....	1
1.2. 数据结构.....	3
1.3. 面向对象思想.....	4
1.4. 设计模式.....	4
1.5. 开发环境.....	4
1.5.1. Eclipse.....	4
1.5.2. AndroidStudio.....	4
1.6. AndroidSDK.....	4
1.7. Activity 和生命周期.....	4
1.8. Service 和生命周期.....	4
1.9. BroadcastReceiver.....	4
1.9.1. 静态注册.....	4
1.9.2. 动态注册.....	4
1.10. ContentProvider.....	4
1.11. ActionBar.....	4
1.12. Fragment.....	4
2. UI.....	4
2.1. Layout.....	5
2.1.1. FragmentLayout.....	5
2.1.2. LinearLayout.....	5
2.1.3. TableLayout 和 RableRow.....	5
2.1.4. GridLayout.....	5
2.1.5. RelativeLayout.....	5
2.1.6. DrawerLayout.....	5
2.1.7. SlidingPaneLayout.....	5
2.2. View.....	5
2.2.1. Widgets.....	5
2.2.2. Containers.....	6
2.2.3. Date&Time.....	7
2.2.4. Expert.....	7
2.2.5. Custom.....	9
2.3. CustomView.....	9
2.3.1. extendsystemView.....	9
2.3.2. combinationView.....	9
2.3.3. extendView.....	9
2.4. anim.....	9
2.4.1. ViewAnimation.....	9
2.4.2. PropertyAnimation.....	10
2.5. Resource.....	10
2.5.1. asset 和 AssetManager.....	10
2.5.2. res.....	10
2.6. OpenGL.....	11

3. 通讯.....	12
3.1. Http.....	12
3.1.1. HttpClient.....	12
3.1.2. HttpConnection.....	12
3.2. Socket.....	12
3.3. Bluetooth.....	12
3.4. NFC.....	12
3.5. Headset.....	12
3.6. USB.....	12
4. 数据持久化.....	12
4.1. Sqlite.....	12
4.1.1. SQLiteOpenHelper.....	12
4.1.2. ContentProvider.....	12
4.2. File.....	12
4.2.1. InternalStorage.....	12
4.2.2. ExternalStorage.....	13
4.3. SharedPreferences.....	13
5. 性能.....	13
5.1. UI 优化.....	13
5.1.1. 布局层次结构.....	13
5.1.2. 抽象布局使用.....	13
5.1.3. HierarchyViewer.....	13
5.1.4. Lint 工具.....	13
5.2. 内存优化.....	13
5.2.1. OOM.....	13
5.2.2. ANR.....	13
5.2.3. 分析.....	13
5.3. 电量优化.....	14
5.4. 流量优化.....	14
6. 调试.....	14
6.1. Logcat.....	14
6.2. adb.....	14
6.3. HierarchyViewer.....	14
6.4. TraceView.....	14
6.5. Heap.....	14
6.6. Lint.....	14
7. 适配.....	14
7.1. minSDK 和 OSVersion.....	15
7.2. layout 和 ScreenSize.....	15
7.3. drawable 和 Screenpx.....	15
8. 测试.....	15
8.1. Monkey.....	15
8.2. MonkeyRunner.....	15
8.3. JUnit.....	15

8.4.	Robotium.....	15
8.5.	Appium.....	15
8.6.	Athrun (TMTS).....	15
8.7.	UIAutomator.....	15
9.	安全.....	15
9.1.	服务器安全.....	15
9.2.	通讯安全.....	15
9.2.1.	与服务器端通讯.....	15
9.2.2.	组件间通讯.....	16
9.3.	数据加密.....	16
9.4.	数据验签.....	16
9.5.	代码混淆.....	16
9.6.	webView/JS 安全调用.....	16
9.7.	MD5、DES、RSA、https、证书、权限.....	16
10.	NDK.....	16
10.1.	JNI.....	16
10.2.	C 语言.....	16
10.3.	C++.....	16
11.	手机功能.....	16
11.1.	电话.....	16
11.1.1.	联系人.....	16
11.1.2.	通话记录.....	16
11.2.	短/彩信.....	16
11.3.	Camera.....	17
11.4.	Audio.....	17
11.5.	SD 卡.....	17
11.6.	感应器.....	17
11.6.1.	加速.....	17
11.6.2.	方向.....	17
11.6.3.	重力.....	17
11.6.4.	光线.....	17
11.6.5.	陀螺仪.....	17
11.6.6.	磁场.....	17
11.6.7.	接近.....	17
11.6.8.	温度.....	17
11.6.9.	压力.....	17
11.6.10.	线性加速度.....	17
11.6.11.	旋转.....	17
12.	第三方扩展.....	18
12.1.	地图.....	18
12.2.	语音识别.....	18
12.3.	支付.....	18
12.4.	统计分析.....	18
12.5.	广告.....	18

13. 其他.....	18
13.1. Intent.....	18
13.2. AndroidManifest.....	18
13.3. AIDL.....	18
13.4. WiFi.....	18
13.5. 国际化.....	18
13.6. PopuWindow.....	18

1. 基础知识

Android 基础知识, 主要包括 Java 基础、面向对象思想、数据结构、Android 开发环境、Android SDK 和 Android 四大组件等内容。以下分类进行展开, 从网上寻找资料进行知识编辑。

1.1. Java 基础

JVM、JRE 和 JDK 的区别:

(1) JVM(Java Virtual Machine): java 虚拟机, 用于保证 java 的跨平台的特性。java 语言是跨平台的, jvm 不是跨平台的。(2) JRE(Java Runtime Environment): java 的运行环境, 包括 jvm+java 的核心类库。(3) JDK(Java Development Kit): java 的开发工具, 包括 jre+开发工具。

环境变量 path 和 classpath 的作用是什么?

(1) path 是配置 Windows 可执行文件的搜索路径, 即扩展名为 .exe 的程序文件所在的目录, 用于指定 DOS 窗口命令的路径。(2) Classpath 是配置 class 文件所在的目录, 用于指定类搜索路径, JVM 就是通过它来寻找该类的 class 类文件的。

变量有什么用? 为什么要定义变量? 什么时候用?

变量的作用: 用来存储数据。为什么要定义变量: 用来不断的存放同一类型的常量, 并可以重复使用。在执行方法时进行调用。

&和&&的区别?

(1) &&会出现短路, 如果可以通过第一个表达式判断出整个表达式的结果, 则不继续后面表达式的运算; 只能操作 boolean 类型数据; (2) &不会出现短路, 将整个表达式都运算。既可以操作 boolean 数据还可以操作数。

标示符命名规则:

(1) 由数字(0-9), 大小写英文字母, 以及_和\$组成。(2) 不能以数字开头。(3) 不能使用关键字来自定义命名。

数据类型:

(1) 基本数据类型(4类8种): 整数类型: byte、short、int、long; 浮点数类型: float、double; 字符类型: char; 布尔类型: boolean(true false); (2) 引用数据类型: 类、接口、数组

类型转换

(1) 精度从高到低 double float long int short(char) byte。(2) 自动类型转换, 将一个低精度转换到高精度。(3) 强制类型转换, 将一个高精度转换到低精度(精度会下降)。

java 语言的三种技术架构

(1) J2EE: 企业版, 是为开发企业环境下的应用程序提供的一套解决方案。该技术体系中包含的技术如 Servlet、Jsp 等, 主要针对于 Web 应用程序开发。(2) J2SE: 标准版, 是为开发普通桌面和商务应用程序提供的解决方案。该技术体系是其他两者的基础, 可以完成一些桌面应用程序的开发。比如 Java 版的扫雷。(3) J2ME: 小型版, 是为开发电子消费产品和嵌入式设备提供的解决方案。该技术体系主要应用于小型电子消费类产品, 如手机中的应用程序等。

java 的跨平台性:

通过 Java 语言编写的应用程序在不同的系统平台上都可以运行。只要在需要运行 java 应用程序的操作系统上, 先安装一个 Java 虚拟机(JVM Java Virtual Machine)即可。由 JVM 来负责 Java 程序在该系统中的运行。

有符号数据的表示法(次重点)

原码, 反码(原码取反), 补码(反码+1)。

函数

(1) 定义: 函数就是定义在类中的具有特定功能的一段独立小程序。

(2) 特点: 定义函数可以将功能代码进行封装; 便于对该功能进行复用; 函数只有被调用才会被执行; 函数的出现提高了代码的复用性; 对于函数没有具体返回值的情况, 返回值类型用关键字 void 表示, 那么该函数中的 return 语句如果在最后一行可以省略不写。(3) 函数的应用两个明确: 明确要定义的功能最后的结果是什么? 明确在定义该功能的过程中, 是否需要未知内容参与运算。

重载

(1) 概念: 在同一个类中, 允许存在一个以上的同名函数, 只要它们的参数个数或者参数类型不同即可。(2) 特点: 与返回值类型无关, 只看参数列表(参数类型以及参数个数)。(3) 好处: 方便于阅读, 优化了程序设计。

数组

(1) 概念: 同一种数据类型的集合。(2) 好处: 可以自动给数组中的元素从 0 开始编号, 方便操作这些元素。

内存结构

(1) 栈内存: 用于存储局部变量, 当数据使用完, 所占空间会自动释放。(2) 堆内存: 数组和对象, 通过 new 建立的实例都存放在堆内存中。(3) 方法区: 静态成员、构造函数、常量池、线程池。(4) 本地方法区: window 系统占用。(5) 寄存器: 所有进程都使用寄存器。Java 虚拟机使用下列寄存器管理系统堆栈: 程序记数寄存器, 跟踪程序执行的准确位置; 堆栈指针寄存器, 指示操作栈项; 框架寄存器, 指向当前执行的环境; 变量寄存器, 指向当前执行环境中第一个本地变量。Java 开发组决定 Java 只使用四个寄存器, 这是因为如果使用的寄存器数多于处理器端口数, 那么处理器的效率将严重地降低。Java 虚拟机中的堆栈用于存放变量, Java 程序向 Java 虚拟机传递字节码, Java 虚拟机为每个方法创建堆栈框架, 每个框架维护三种信息: 局部变量, 由变量寄存器指向的 32 位变量数组; 执行环境, 由框架寄存器指向和执行的方法; 操作堆栈, 执行先进先出规则(FIFO), 它是 32 位宽度的,

它为操作码维护必要的参数，该堆栈的顶部是由堆栈指针寄存器索引的。

1.2. 数据结构

https://blog.csdn.net/qq_29631809/article/details/72599708

- 1.3. 面向对象思想
- 1.4. 设计模式
- 1.5. 开发环境
 - 1.5.1. Eclipse
 - 1.5.2. AndroidStudio
- 1.6. AndroidSDK
- 1.7. Activity 和生命周期
- 1.8. Service 和生命周期
- 1.9. BroadcastReceiver
 - 1.9.1. 静态注册
 - 1.9.2. 动态注册
- 1.10. ContentProvider
- 1.11. ActionBar
- 1.12. Fragment
- 2. UI

2.1. Layout

2.1.1. FragmentLayout

2.1.2. LinearLayout

2.1.3. TableLayout 和 TableRow

2.1.4. GridLayout

2.1.5. RelativeLayout

2.1.6. DrawerLayout

2.1.7. SlidingPaneLayout

2.2. View

2.2.1. Widgets

2.2.1.1. TextView

2.2.1.2. Button

2.2.1.3. RadioButton

2.2.1.4. CheckBox

2.2.1.5. Switch

2.2.1.6. ToggleButton

2.2.1.7. ImageButton

2.2.1.8. ImageView

2.2.1.9. ProgressBar

2.2.1.10. SeekBar

2.2.1.11. RatingBar

2.2.1.12. Spinner

2.2.1.13. WebView

2.2.2. Containers

2.2.2.1. RadioGroup

2.2.2.2. ListView

2.2.2.3. GridView

2.2.2.4. ExpandableListView

2.2.2.5. ScrollView

2.2.2.6. TabHost

2.2.2.7. SlidingDrawer

2.2.2.8. Gallery

2.2.2.9. VideoView

2.2.2.10. DialerFilter

2.2.2.11. RecyclerView

2.2.2.12. CardView

2.2.3. Date&Time

2.2.3.1. TextClock

2.2.3.2. AnalogClock

2.2.3.3. Chronometer

2.2.3.4. DatePicker

2.2.3.5. TimePicker

2.2.3.6. CalendarView

2.2.4. Expert

2.2.4.1. Space

2.2.4.2. CheckedTextView

- 2.2.4.3. QuickContactBadge
- 2.2.4.4. ExtractEditText
- 2.2.4.5. AutoCompleteTextView
- 2.2.4.6. MultiAutoCompleteText
- 2.2.4.7. NumberPicker
- 2.2.4.8. ZoomButton
- 2.2.4.9. ZoomController
- 2.2.4.10. GestureOverlayView
- 2.2.4.11. SurfaceView
- 2.2.4.12. TextttureView
- 2.2.4.13. StackView
- 2.2.4.14. ViewStub
- 2.2.4.15. ViewAnimator
- 2.2.4.16. ViewFlipper
- 2.2.4.17. ViewSwitcher

2.2.4.18. ImageSwitcher

2.2.4.19. TextSwitcher

2.2.4.20. AdapterViewFlipper

2.2.4.21. ViewPager

2.2.5. Custom

2.2.5.1. include

2.2.5.2. fragment

2.2.5.3. requestFocus

2.3. CustomView

2.3.1. extendsystemView

2.3.2. combinationView

2.3.3. extendView

2.4. anim

2.4.1. ViewAnimation

2.4.1.1. TweenAnimation

2.4.1.1.1. alpha

2.4.1.1.2. scale

2.4.1.1.3. translate

2.4.1.1.4. rotate

2.4.1.1.5. Interpolators

2.4.1.2. FrameAnimation 和 animation-list

2.4.2. PropertyAnimation

2.4.2.1. ValueAnimator

2.4.2.2. ObjectAnimator

2.4.2.3. AnimatorSet

2.5. Resource

2.5.1. asset 和 AssetManager

2.5.2. res

2.5.2.1. anim

2.5.2.2. animator

2.5.2.3. color

2.5.2.4. drawable

2.5.2.5. interpolator

2.5.2.6. layout

2.5.2.7. menu

2.5.2.8. raw

2.5.2.9. values

2.5.2.9.1. arrays

2.5.2.9.2. attrs

2.5.2.9.3. bools

2.5.2.9.4. colors

2.5.2.9.5. string

2.5.2.9.6. Styles

2.5.2.10. xml

2.6. OpenGL

3. 通讯

3.1. Http

3.1.1. HttpClient

3.1.2. HttpURLConnection

3.2. Socket

3.3. Bluetooth

3.4. NFC

3.5. Headset

3.6. USB

4. 数据持久化

4.1. Sqlite

4.1.1. SqliteOpenHelper

4.1.2. ContentProvider

4.2. File

4.2.1. InternalStorage

4.2.2. ExternalStorage

4.3. SharedPreferences

5. 性能

5.1. UI 优化

5.1.1. 布局层次结构

5.1.2. 抽象布局使用

5.1.3. HierarchyViewer

5.1.4. Lint 工具

5.2. 内存优化

5.2.1. OOM

5.2.2. ANR

5.2.3. 分析

5.2.3.1. Heap

5.2.3.2. adb shell

5.2.3.2.1. dumpsysmeminfo

5.2.3.2.2. showmap

5.2.3.3. TraceView

5.2.3.4. Dalvik 日志

5.2.3.5. logcat

5.2.3.6. MAT

5.3. 电量优化

5.4. 流量优化

6. 调试

6.1. Logcat

6.2. adb

6.3. HierarchyViewer

6.4. TraceView

6.5. Heap

6.6. Lint

7. 适配

- 7.1. minSDK 和 OSVersion
- 7.2. layout 和 ScreenSize
- 7.3. drawable 和 Screenpx

8. 测试

- 8.1. Monkey
- 8.2. MonkeyRunner
- 8.3. JUnit
- 8.4. Robotium
- 8.5. Appium
- 8.6. Athrun (TMTS)
- 8.7. UIAutomator

9. 安全

- 9.1. 服务器安全
- 9.2. 通讯安全
 - 9.2.1. 与服务器端通讯

9.2.2. 组件间通讯

9.3. 数据加密

9.4. 数据验签

9.5. 代码混淆

9.6. webView/JS 安全调用

9.7. MD5、DES、RSA、https、证书、权限

10. NDK

10.1. JNI

10.2. C 语言

10.3. C++

11. 手机功能

11.1. 电话

11.1.1. 联系人

11.1.2. 通话记录

11.2. 短/彩信

11.3. Camera

11.4. Audio

11.5. SD 卡

11.6. 感应器

11.6.1. 加速

11.6.2. 方向

11.6.3. 重力

11.6.4. 光线

11.6.5. 陀螺仪

11.6.6. 磁场

11.6.7. 接近

11.6.8. 温度

11.6.9. 压力

11.6.10. 线性加速度

11.6.11. 旋转

12. 第三方扩展

12. 1. 地图

12. 2. 语音识别

12. 3. 支付

12. 4. 统计分析

12. 5. 广告

13. 其他

13. 1. Intent

13. 2. AndroidManifest

13. 3. AIDL

13. 4. WiFi

13. 5. 国际化

13. 6. PopuWindow