Rxjava2.x流式编程

文档：

外文网站：<http://reactivex.io/documentation/operators.html>

系列教程：<http://www.apkbus.com/myspaceblog-898535.html>

RxLifecycle：<https://github.com/trello/RxLifecycle>

UI教程

<http://hencoder.com/>

Kotlin语言

MVVM框架(databinding)

Databinding教程：

<http://blog.csdn.net/qiang_xi/article/details/73849556>

APP构架指南：

<http://blog.csdn.net/guiying712/article/details/78474177>

AAC官方MVVM组件：

<https://www.jianshu.com/p/f418de23e3ee>

<https://www.jianshu.com/p/29e5e8c75450>

开发者网站

<https://developer.android.com/topic/libraries/architecture/index.html>

系列教程

<https://www.jianshu.com/p/42eb71ec4a19>

MVP框架

Retrofit2.x 网络访问

CompositeDisposable：订阅管理类

Observable、Single、Flowable、Maybe或者其他Completable

数据库框架Room

Glide图片处理

源码：<https://github.com/bumptech/glide>

教程1：<http://www.jcodecraeer.com/a/anzhuokaifa/2017/0928/8567.html>

教程2：<https://juejin.im/entry/5924f28eda2f60005d7725b2>

教程3：[http://blog.csdn.net/u013005791/article/details/74532091#51-requestoptions](http://blog.csdn.net/u013005791/article/details/74532091" \l "51-requestoptions)

dagger2注入

官网：<https://google.github.io/dagger/>

系列文章：<http://blog.csdn.net/mq2553299/article/details/73136396>

注意点：

1. 使用inject完成注入，只有注入之后才能够使用注解生成实例
2. 当module含有参数时，需要在注入的时候将该参数模块初始化注入，即xxxModule(a:Any……)，当没有参数时不需要初始化模块
3. 当要使用其他module中提供的实例时，需要在Component中使用dependencies关键字将被依赖的Component的类加入，并且需要在被依赖的component中提供实例 方法
4. 当一个Component没有域修饰的时候(@Singleton, 使用@Scope自定义的域)可以被任意Component依赖。但是当他被域修饰时，那么依赖这个Component的Component也必须使用域修饰，并且这两个域不能是相同的域

Github源码： <https://github.com/google/dagger>

//查看github的issue可以查看某些问题

系列教程：<http://blog.csdn.net/io_field/article/details/70947365>

系列教程：<https://www.jianshu.com/p/9fee10d4e59f>

<https://www.jianshu.com/p/2cd491f0da01>

热修复框架

推送框架

二维码扫描

Zxing

社会化分享

Mob 友盟

第三方登录

Mob 友盟

数据库Room

文章：<http://blog.csdn.net/LABLENET/article/details/76614991>

官方译文：<http://www.jcodecraeer.com/a/anzhuokaifa/androidkaifa/2017/0525/7971.html>

Google最新sample

<https://github.com/googlesamples>

其他：

isAssignableFrom、instanceof

数据库

Websocket

Github地址：<https://github.com/TooTallNate/Java-WebSocket>

北京时间api：

<http://api.k780.com:88/?app=life.time&appkey=19292&sign=4684f7ddf14f1f63e1e312be09cb080e&format=json>

豆瓣topapi：

<https://api.douban.com/v2/movie/top250>