# 文档前言

## 基本描述

俗话说的好：“好记性，不如烂笔头”。说明的一个道理就是纪录的重要性，哪怕是一个记忆天才，纪录在其他地方，可以随时看到也是非常重要的。何况我是如此一个平凡的人啊！谦虚学习，踏踏实实，不能好高骛远，天道酬勤才是王道。

本文档旨在纪录开发过程中遇到的一些重要点、难点，也能帮助我纪录我的成长过程。

少波，好好纪录，这是很重要的一个事情，不要觉得我写了一些知道的东西，没什么价值，磨刀不误砍柴工。

## 内容概括

本文包括如下内容：iOS开发、设计模式、数据结构、java服务器开发、前段开发等，若有新技术的添加会及时的添加到文档中。

IOS开发部分

# 网络部分

## 实现离线断点下载功能（2016年4月19日星期二）

### 基本描述：

实现大文件的离线重新下载，在程序运行过程可以暂停、继续以及在程序app下一次启动后，可以继续下载，上次下载的文件进行保存，下次从上次下载完成的部分继续下载。

基本的实际案例：百度音乐下载、喜马拉雅音乐下载、百度云文件的下载。

其实本人认为，网络部分断点续传算是难度和复杂度都比较高的部分。

### 解决方案

#### 解决方案理论

用户发送下载文件请求（如下载音乐、大文件等），当用户首次发送下载文件，将请求头部设置为请求的文件二进制位置为0，在下载过程中不断将下载的二进制文件存储在本地，并且记录下载量，一旦用户暂停下载，app记录下最后下载完成的数量以及下载请求的位置，从多少字节开始下载。服务器端则根据客户端的请求，从head区域的文件请求位置，开始传输二进制数据，发送给客户端，在完成总的文件下载后，客户端取消任何关于本文件的下载信息。

当前，一般的网络处理，都会用到AFNetworking 网络请求库，但是此框架对实现离线形式（重启app，从上次的下载点继续下载）的断点续传支持不好。因此我此处使用到的是apple系统api原生的网络请求方式。

#### 解决实践

1. 用到NSURLSession ->session，调用含有代理设置的初始化方法:

+ (NSURLSession \*)sessionWithConfiguration:(NSURLSessionConfiguration \*)configuration delegate:(nullable id <NSURLSessionDelegate>)delegate delegateQueue:(nullable NSOperationQueue \*)queue;

2. 使用NSMutableRequest->request，设置头部文件为：

//设置表头信息

NSString \*rang = [NSString stringWithFormat:”@bytes=%zd-”,currrentSize];

[request setValue:rang forHTTPHeanderField:”@Range”];

3.调用创建task方法

//创建task

NSURLSessionDataTask \*task = [session dataTaskWithRequest:request];

下图是参考别人写的，我觉的很好，直接贴上来，如果觉的图片不清楚，可以通过放大文档的形式查看，如下图：

