https://blog.csdn.net/huaxun66/article/details/52434528

<https://blog.csdn.net/superloveboy/article/details/49626223>

https://blog.csdn.net/icoollike/article/details/49475427

Android开发本质上就是手机和互联网中的web服务器之间进行通信，就必然需要从服务端获取数据，而反复通过网络获取数据是比较耗时的，特别是访问比较多的时候，会极大影响了性能；以及在断网情况下Android获取到的数据就为空。Android中可通过二级缓存来减少频繁的网络操作，减少流量、提升性能，缓存页面的json数据。

所谓二级缓存实际上并不复杂，当Android端需要获得数据时比如获取网络中的图片，我们首先从内存中查找（按键查找），内存中没有的再从磁盘文件或sqlite中去查找，若磁盘中也没有才通过网络获取；当获得来自网络的数据，就以key-value对的方式先缓存到内存（一级缓存），同时缓存到文件或sqlite中（二级缓存）。注意：内存缓存会造成堆内存泄露，所有一级缓存通常要严格控制缓存的大小，一般控制在系统内存的1/4。

　　理解了二级缓存大家可能会有个问题网络中的数据是变化的，数据一旦放入缓存中，再取该数据就是从缓存中获得，这样岂不是不能体现数据的变化？我们在缓存数据时会设置有效时间，比如说30分钟，若超过这个时间数据就失效并释放空间，然后重新请求网络中的数据。那么30分钟内咋办？就下拉刷新啦， 实际上这不是问题。

在Android中当加载大量图片时首先需要考虑的一个问题是如何避免OOM。为了保证内存的使用始终维持在一个合理的范围，通常会把移出屏幕的图片进行回收处理，此时垃圾回收器会认为你不再持有这些图片的引用，从而对这些图片进行GC。然而当某些图片被回收之后用户又将它重新滑入屏幕时，这时又会去重新加载一遍刚刚加载过的图片。这样频繁地处理图片的加载和回收不利于操作的流畅性，而内存和硬盘的Cache就会帮助解决这个问题，实现快速加载已加载过的图片。   
在缓存上，主要有两种级别的Cache：LruCache和DiskLruCache。 前者是基于内存的，后者是基于硬盘的。