

به نام خدا



«بررسی فرآیند و روش کار برنامه FileDownloader»

کاری از:

مصطفی قدیمی

مهرداد کرابی

استاد راهنما:

امید جعفری نژاد

بهار ۹۸

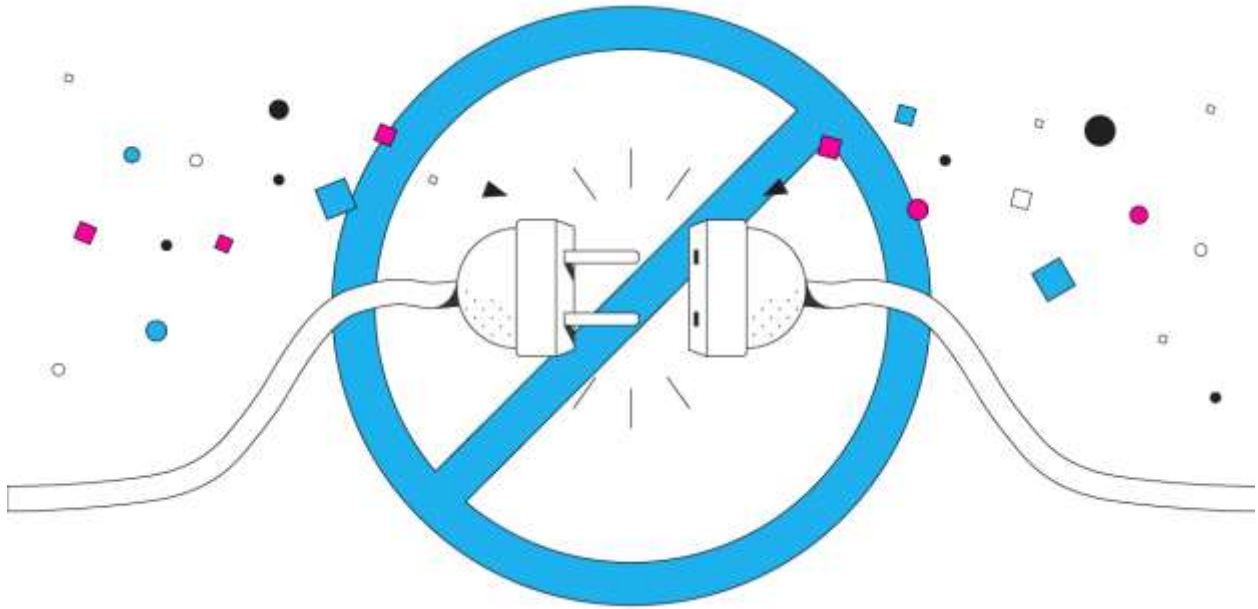
در این قسمت ابتدا به ترتیب به سراغ هر پکیج می‌رویم و سپس به بررسی هریک می‌پردازیم. ساختار کلی پروژه به صورت زیر است:

```
.
├── BaseDownloadTask.java
├── connection
│   ├── DefaultConnectionCountAdapter.java
│   ├── FileDownloadConnection.java
│   ├── FileDownloadUrlConnection.java
│   └── RedirectHandler.java
├── database
│   ├── FileDownloadDatabase.java
│   ├── NoDatabaseImpl.java
│   ├── RemitDatabase.java
│   ├── SqliteDatabaseImpl.java
│   └── SqliteDatabaseOpenHelper.java
├── download
│   ├── ConnectionProfile.java
│   ├── ConnectTask.java
│   ├── CustomComponentHolder.java
│   ├── DownloadLaunchRunnable.java
│   ├── DownloadRunnable.java
│   ├── DownloadStatusCallback.java
│   ├── FetchDataTask.java
│   └── ProcessCallback.java
├── DownloadSpeedMonitor.java
├── DownloadTaskHunter.java
├── DownloadTask.java
├── event
│   ├── DownloadEventPoolImpl.java
│   ├── DownloadEventSampleListener.java
│   ├── DownloadServiceConnectChangedEvent.java
│   └── IDownloadEvent.java
```

- | |— IDownloadEventPool.java
- | |— IDownloadListener.java
- |— exception
- | |— FileDownloadGiveUpRetryException.java
- | |— FileDownloadHttpException.java
- | |— FileDownloadNetworkPolicyException.java
- | |— FileDownloadOutOfSpaceException.java
- | |— FileDownloadSecurityException.java
- | |— PathConflictException.java
- |— FileDownloadConnectListener.java
- |— FileDownloader.java
- |— FileDownloadEventPool.java
- |— FileDownloadLargeFileListener.java
- |— FileDownloadLineAsync.java
- |— FileDownloadLine.java
- |— FileDownloadListener.java
- |— FileDownloadList.java
- |— FileDownloadMessageStation.java
- |— FileDownloadMessenger.java
- |— FileDownloadMonitor.java
- |— FileDownloadQueueSet.java
- |— FileDownloadSampleListener.java
- |— FileDownloadServiceProxy.java
- |— FileDownloadServiceSharedTransmit.java
- |— FileDownloadServiceUIGuard.java
- |— FileDownloadTaskLauncher.java
- |— IDownloadSpeed.java
- |— IFileDownloadMessenger.java
- |— IFileDownloadServiceProxy.java
- |— ILostServiceConnectedHandler.java
- |— IQueuesHandler.java
- |— ITaskHunter.java

- |— IThreadPoolMonitor.java
- |— LostServiceConnectedHandler.java
- |— message
 - | |— BlockCompleteMessage.java
 - | |— IFlowDirectly.java
 - | |— IMessageSnapshot.java
 - | |— LargeMessageSnapshot.java
 - | |— MessageSnapshotFlow.java
 - | |— MessageSnapshot.java
 - | |— MessageSnapshotTaker.java
 - | |— MessageSnapshotThreadPool.java
 - | |— SmallMessageSnapshot.java
- |— MessageSnapshotGate.java
- |— model
 - | |— ConnectionModel.java
 - | |— FileDownloadHeader.java
 - | |— FileDownloadModel.java
 - | |— FileDownloadStatus.java
 - | |— FileDownloadTaskAtom.java
- |— notification
 - | |— BaseNotificationItem.java
 - | |— FileDownloadNotificationHelper.java
 - | |— FileDownloadNotificationListener.java
- |— PauseAllMarker.java
- |— QueuesHandler.java
- |— services
 - | |— BaseFileServiceUIGuard.java
 - | |— DefaultIdGenerator.java
 - | |— DownloadMgrInitialParams.java
 - | |— FDServiceSeparateHandler.java
 - | |— FDServiceSharedHandler.java
 - | |— FileDownloadBroadcastHandler.java

- | |— FileDownloadManager.java
- | |— FileDownloadService.java
- | |— FileDownloadThreadPool.java
- | |— ForegroundServiceConfig.java
- | |— IFileDownloadServiceHandler.java
- |— stream
- | |— FileDownloadOutputStream.java
- | |— FileDownloadRandomAccessFile.java
- |— util
- | |— DownloadServiceNotConnectedHelper.java
- | |— ExtraKeys.java
- | |— FileDownloadExecutors.java
- | |— FileDownloadHelper.java
- | |— FileDownloadLog.java
- | |— FileDownloadProperties.java
- | |— FileDownloadSerialQueue.java
- | |— FileDownloadUtils.java



پکیج Connection:

DefaultConnectionCountAdapter: این کلاس تعداد کانکشن‌ها را بر اساس حجم فایل دانلودی مشخص می‌کند.

FileDownloadConnection: این رابط، بخش کلیدی را از مقداره‌ی اولیه کردن تا پایان کانکشن را بر عهده دارد که در خلال آن به تنظیم کردن هدر، دریافت ورودی به صورت استریم، ادامه دادن دانلود و ... می‌پردازد.

FileDownloadURLConnection: این کلاس علاوه‌بر پیاده‌سازی اینترفیس FileDownloadConnection، برای ریکوئست زدن به یک آدرس اینترنتی مشخص، یک کانکشن جدید باز کرده و سپس به اندازه مدت زمان معلومی منتظر می‌ماند و در صورتی که از آن مدت زمانی بگذرد، کانکشن، تایم‌اوت می‌شود.

RedirectHandler: این کلاس به نوعی کمک‌کننده‌ی کلاس قبل است. به طوری که status ریسپانس آدرس اینترنتی را نگاه می‌کند در صورتی که ریدایرکت باشد (کد ۳۰۷ یا ۳۰۸) تا ماکسیمم ۱۰ بار اتصال مجدد را مدیریت می‌کند.



پکیج Database:

`FileDownloadDatabase`: این اینترفیس حالت‌های فایل‌های دانلودی را در دیتابیس ذخیره می‌کند که دارای توابعی است که در کلاس‌های بعدی باید پیاده‌سازی شود.

`NoDatabaseImpl`: این کلاس در صورتی که قبلاً دیتابیزی ساخته نشده باشد، شروع به ساختن دیتابیس جدید با استفاده از پیاده‌سازی اینترفیس بالا می‌کند.

`SqliteDatabaseImpl`: این کلاس پس از این که کلاس `NoDatabaseImpl` صدا زده می‌شود، و دیتابیزی از نوع اس‌کیوال‌لایت را با پیاده‌سازی اینترفیس `FileDownloadDatabase` راه‌اندازی اولیه می‌کند.

`RemitDatabase`: این کلاس به صورت یک هندلر بین دو کلاس بالا عمل می‌کند. به صورتی که اگر اطلاعاتی روی حافظه کش باشد و نیاز باشد که در دیتابیس ذخیره شود این عمل را انجام می‌دهد.

`SqliteDatabaseOpenHelper`: این کلاس اینترفیس `SQLiteOpenHelper` را پیاده‌سازی می‌کند که تمام مراحل ایجاد جدول‌های جدید، اضافه کردن ستون و نابودسازی جدول را انجام می‌دهد.



پکیج Download:

ConnectionProfile: این کلاس یک کانکشن می‌گیرد و برای آن یک پروفایل می‌سازد به شرطی که در آن هدرهای آن را نظیر `startoffset`، `currentoffset` و `endoffset` را تنظیم کنیم.

ConnectTask: این کلاس وظیفه تطابق هدرها را برای کانکشن داندلودی با هدری که در قسمت قبل ساخته شده است دارد. همچنین علاوه بر آن قابلیت آپدیت هدرها و اضافه کردن هدرهای مورد نیاز کاربر و ... را دارد.

CustomComponentHolder: این کلاس اطلاعات کلی هر فایل داندلودی شامل تعداد کانکشن‌ها و تولیدکننده آیدی و ... را نگه‌داری می‌کند تا اطلاعات به صورت منسجم در یک جا نگه‌داری شود.

ProcessCallback: این اینترفیس تاریخچه‌ای از حالت‌های مختلف فایل داندلودی را در خود نگه می‌دارد که شامل `onerror` و ... است.

DownloadLaunchRunnable: این کلاس که اینترفیس بالا را پیاده‌سازی می‌کند یک کانکشن داندلودی را با توجه به تاریخچه داندلود اجرا می‌کند تا داندلود با موفقیت آغاز یا ادامه داده شود.



پکیج Event:

`DownloadEventPoolImpl`: این کلاس که اینترفیس `IDownloadEventPool` را پیاده‌سازی می‌کند، استخری از تردها با اندازه ۱۰ نگهداری می‌کند که به صورت سنکرون نگهداری می‌شوند تا خلالی در اجرای برنامه به وجود نیاید.

`DownloadEventSampleListener`: این کلاس ساختار یک لیسنر را تعریف و ساده‌سازی می‌کند.

`DownloadServiceConnectChangedEvent`: این کلاس به عنوان یک درایور برای کانکشن سرویس دانلود فایل عمل می‌کند.

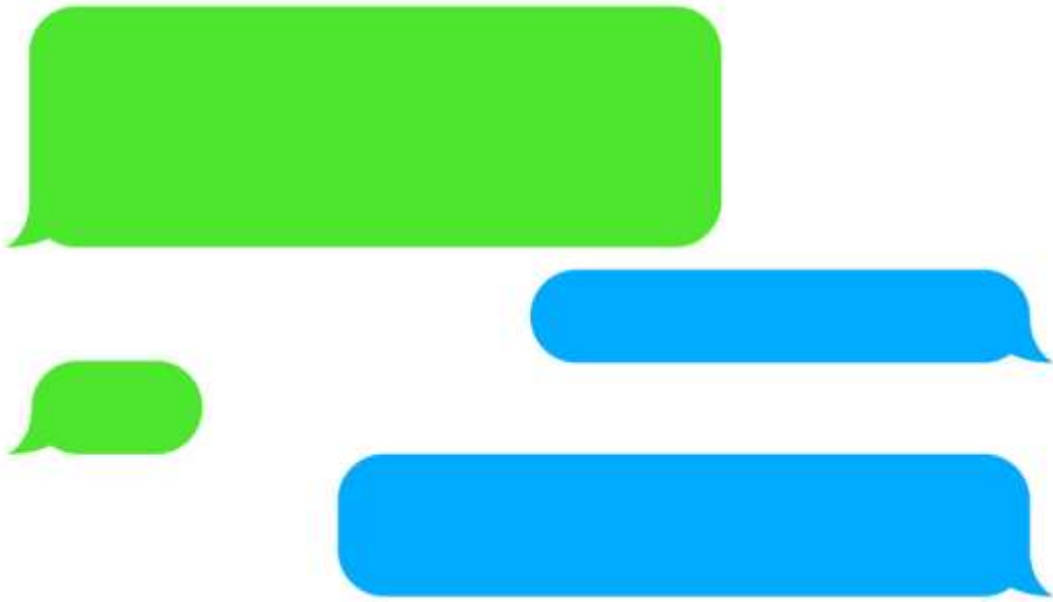
`IDownloadEvent`: این کلاس یک ایوینت اتمیک را پیاده‌سازی می‌کند.



پکیج Exception:

- | — exception
 - | | — FileDownloadGiveUpRetryException.java
 - | | — FileDownloadHttpException.java
 - | | — FileDownloadNetworkPolicyException.java
 - | | — FileDownloadOutOfSpaceException.java
 - | | — FileDownloadSecurityException.java
 - | | — PathConflictException.java

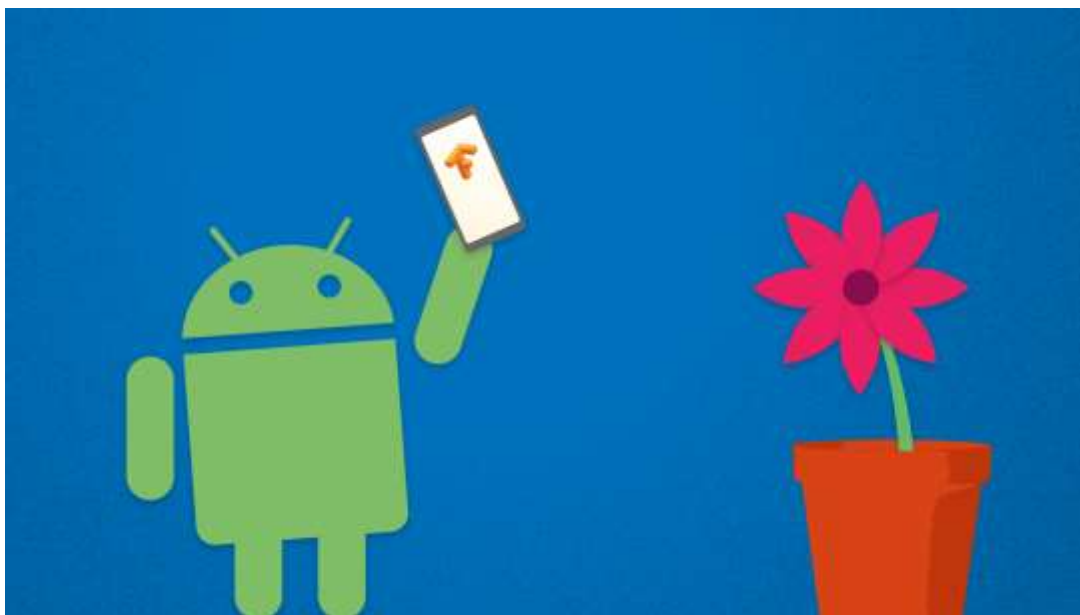
این دسته از کلاس‌ها برای مدیریت مشکلاتی است که ممکن است به دلایل مختلفی نظیر مشکل‌های اینترنتی، خطای آدرس، مشکلات امنیتی و ... به وجود بیاید و خطای مناسب هر کدام را چاپ می‌کند تا کاربر متوجه مشکل به وجود آمده بشود.



پکیج Message:

- | — message
 - | | — BlockCompleteMessage.java
 - | | — IFlowDirectly.java
 - | | — IMessageSnapshot.java
 - | | — LargeMessageSnapshot.java
 - | | — MessageSnapshotFlow.java
 - | | — MessageSnapshot.java
 - | | — MessageSnapshotTaker.java
 - | | — MessageSnapshotThreadPool.java
 - | | — SmallMessageSnapshot.java

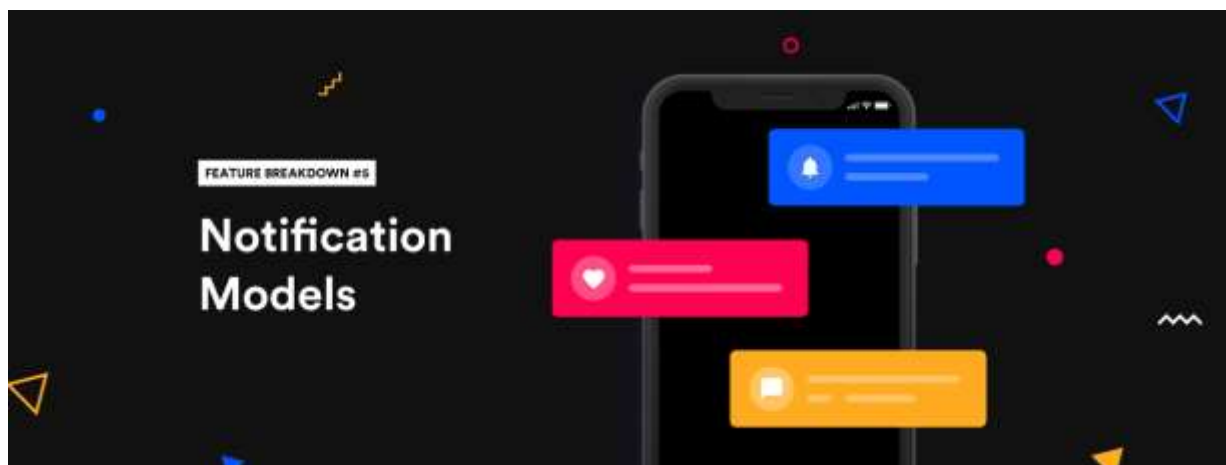
این دسته از کلاس‌ها برای ارسال پیام به پردازنده‌های مختلف که در این جا فایل‌های دانلودی هستند، به وجود آمده است. در آن هر تسک با یک آی‌دی شناسایی می‌شوند و باعث می‌شود که در مورد صدا زده شدن آن‌ها به صورت FIFO اطمینان حاصل شود.



پکیج Model:

- |— model
 - | |— ConnectionModel.java
 - | |— FileDownloadHeader.java
 - | |— FileDownloadModel.java
 - | |— FileDownloadStatus.java
 - | |— FileDownloadTaskAtom.java

این دسته از کلاس‌ها برای مدل کردن موارد مختلفی نظیر کانکشن‌ها، هدرهای فایل‌های دانلودی و ... به وجود آمده‌اند. برای مثال در قسمت کانکشن، یک مدل برای هر کانکشن رکورد روی چندین کانکشن می‌سازد.



پکیج Notification:

- | — notification
- | | — BaseNotificationItem.java
- | | — FileDownloadNotificationHelper.java
- | | — FileDownloadNotificationListener.java

این دسته از کلاس‌ها برای ارسال نوتیفیکیشن‌های اساسی و دانلود به کاربر است. مثلاً کلاس BaseNotificationItem یک نوتیفیکیشن را در صورتی که تسک دانلودی دارای یک آی‌دی مشابه باشند، ارسال می‌کند.



پکیج Stream:

- |— stream
- | |— FileDownloadOutputStream.java
- | |— FileDownloadRandomAccessFile.java

این پکیج شامل دو کلاس است که از آن‌ها برای نوشتن جریان خروجی به روی فایل برای دانلود استفاده می‌شود.



پکیج Util:

└─ util

- └─ DownloadServiceNotConnectedHelper.java
- └─ ExtraKeys.java
- └─ FileDownloadExecutors.java
- └─ FileDownloadHelper.java
- └─ FileDownloadLog.java
- └─ FileDownloadProperties.java
- └─ FileDownloadSerialQueue.java
- └─ FileDownloadUtils.java

این قسمت شامل کلاس‌های نه چندان به هم مرتبطی است که هر کدام امکاناتی را در اختیار ما قرار می‌دهد. برای مثال `FileDownloadSerialQueue` یک صف سریال می‌دهد که باعث می‌شود به صورت پویا، تسک‌ها افزایش بیابند و همچنین تسک‌هایی که درون صف قرار دارند، یک به یک شروع به دانلود شدن کنند.

یا مثلاً کلاس `FileDownloadLog` به ما کمک می‌کند تا همه‌ی لاگ‌های درون فایل دایرکتوری را مدیریت کنیم.