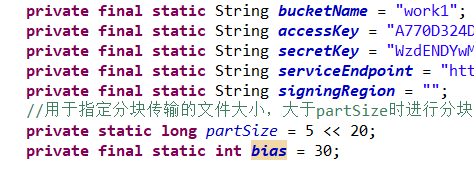
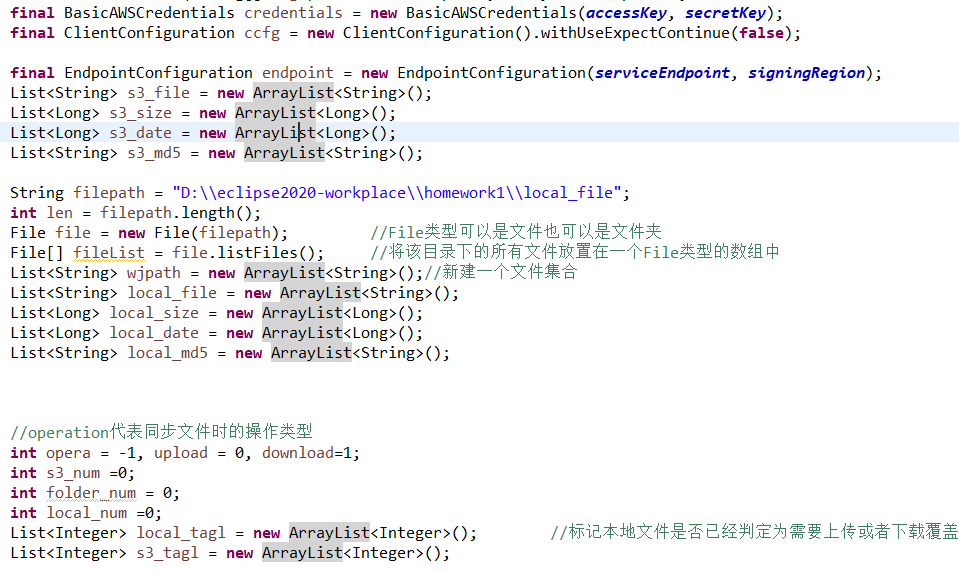
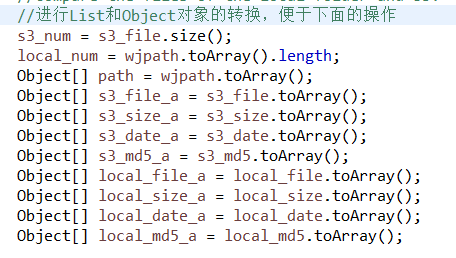
设计说明书

**班别：软件工程3班 姓名：周秀俊**

1. **设计程序的目的：**使用一个指定的本地文件夹与一个S3的bucket，实现简单的文件同步器，在程序启、动后，自动同步本地文件与S3指定bucket中的文件。
2. **定义：各个变量的声明与定义，作用**







**各个类或者方法的定义：**

**Main.java ：**主类，在其中定义各个使用的变量以及调用main包中的其他类的方法。

**void getfile(String dir,List<String> wjpath)方法：**位于Main.Java中，也可以调用封装于getlist类的方法，作用相同，都是递归地得到本地文件列表。

**uploadtos3.java ：**是一个单次上传类，不进行分块，用于将本地文件一次性上传。

**loadfroms3.java ：**是一个单次下载类，不进行分块，用于将文件一次性下载。

**deletes3.java ：**删除类，调用该类可以对s3中的指定文件进行删除操作。

**multipart\_up.java ：**用于将文件分块上传。

**multipart\_down.java ：**用于将文件分块下载。

**watchFile.java ：**监视类，使用watchService服务对指定的文件夹进行监视，得到文件或文件夹的创建，修改，删除信息，实时上传新文件，或对S3桶中的对应文件进行修改或删除。

1. **程序**

**3.1、程序设计思路**

在登录上s3服务器的账号后，使用ListObjectsV2Request得到bucket中的文件列表，通过调用函数和元数据得到文件的四个属性，文件名，大小，最后修改时间，md5值。再通过调用File对象得到相应的本地文件列表和属性。接着将属性通过md5，大小，修改时间进行比较，得到两个关于上传和下载的文件列表。通过调用aws-java-sdk中的接口对s3的bucket和本地文件夹进行文件的上传和下载，当遇到大文件时，使用分块传输。这是同步过程。同步结束后，使用watchService服务进行对本地文件夹的监听，当有创建，修改，删除操作时就会触发相应的事件，此时再调用上传或者下载的方法进行同步即可。

**3.2、程序功能**

（1）当启动程序时，程序自动登录获取用户给定的bucket中的文件目录，以及一个给定的本地文件夹中的全部文件以及子文件夹和其中的文件，实现文件同步。

（2）在（1）同步结束后，若未结束该程序的运行，则接下来程序会调用一个监视类，通过watchService服务对本地指定的文件夹进行文件的变化监视，变化包括，新文件的创建，文件的内容修改，文件名的修改（会被当作一个旧文件的删除以及新文件的创建的多操作过程），文件的删除，空文件夹的创建以及删除，大文件的分块上传以及分块下载。这些操作都会触发一个事件，程序会在文件保存完成后将相应的文件上传到bucket或将其中对应的文件删除。

（3）对于超过限定长度的文件，程序将会自动启动分块传输，包括分块上传和分块下载。

**3.3、初始同步时的文件冲突解决策略：**

在获取两方的文件列表时，还会获取文件的文件名，文件大小，文件最后修改时间，文件的md5校验码。

一种情况是，先判断得到两个文件名相等的文件a和b。在双方都存在md5校验码时先进行md5的比较，若相等则无需任何操作，若不等，则若不等则需要进行文件的同步；根据文件的最后修改时间，将新文件同步到另一方覆盖同名的旧文件。

若有一方不存在md5，则根据文件的大小和最后修改时间来判定是否进行同步。当a和b大小不一致时，直接将新文件覆盖旧文件进行同步。当文件大小相等，且文件a和b的最后修改时间相差在30秒内时不进行同步（偏差时间可以根据实际情况自行设定）。否则，当时间相差大于设定的容许偏差时将新的一个文件进行同步覆盖旧文件。

另一种是，当某一方存在另一方不存在的文件时，将新文件进行同步，若是空文件夹，则不作处理。

**3.4、输入输出项**

不需要输入，但程序会有一些说明输出，可以从中得到文件的一些基础信息，或者文件的传输是否完成，传输的文件的分块传输信息。

**3.5、限制和尚未解决的问题**

（1）分块传输时，传输中断、程序重启可以继续原来的进度。

（2）不支持对有文件存在的文件夹进行直接重命名。

（3）不支持对有文件的文件夹直接进行删除，需要手动将目录下的所有文件删除后才可以对文件夹进行重命名或者删除操作。

（4）在从外部复制大文件进入指定的本地文件夹后会触发多次同步上传，但小文件不受影响。下载也不受影响。

（5）在指定文件夹的子文件夹中创建文件夹并在其中添加文件后会触发异常，尚未解决。因此只能在指定文件夹中创建一级文件夹，无法创建二级文件夹并同步其中的文件。但如果在程序运行前创建好二级和二级以下的文件夹，也是支持在其中创建新文件并进行同步的，该bug尚未解决。

（6）双方文件名的匹配检索使用双重循环的算法进行暴力匹配，未使用优化方法，在文件较多时可能有些影响，但是相比文件同步传输时所消耗的时间，影响就较为弱了。

1. **程序逻辑**

程序入口

错误操作

获取bucket和本地文件夹的文件列表

网络异常

有md5

程序退出

新事件产生

监听中

进行上传或下载

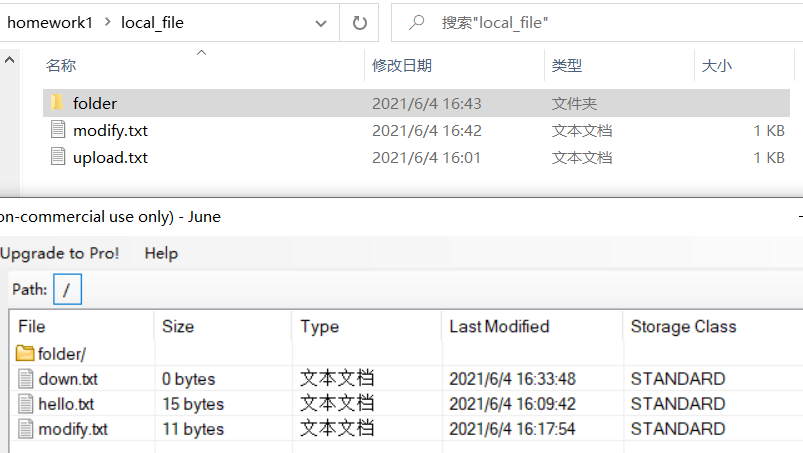
无md5

使用大小和修改时间判断

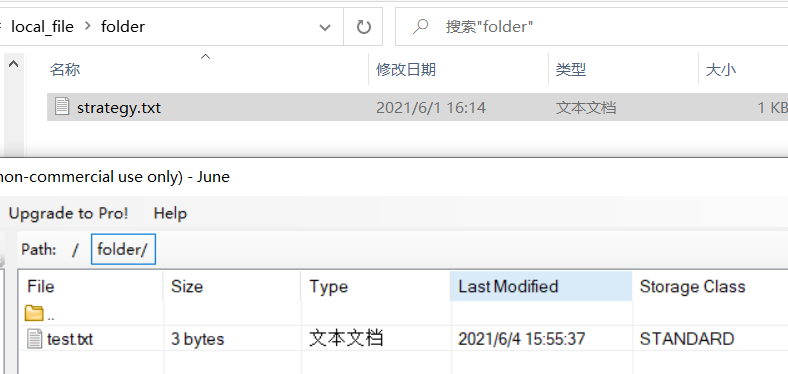
进行md5的比较，判定上传还是下载

1. **部分程序运行成功结果图**

运行前截图：

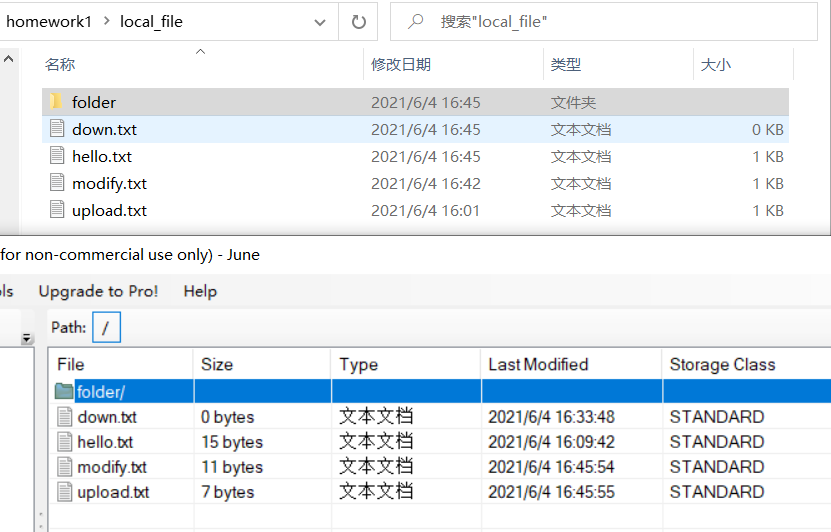


S3的folder文件夹和本地文件夹：

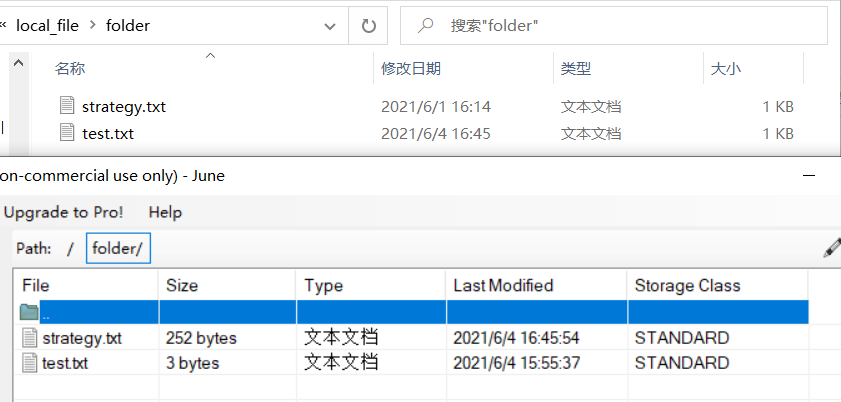


运行后截图：

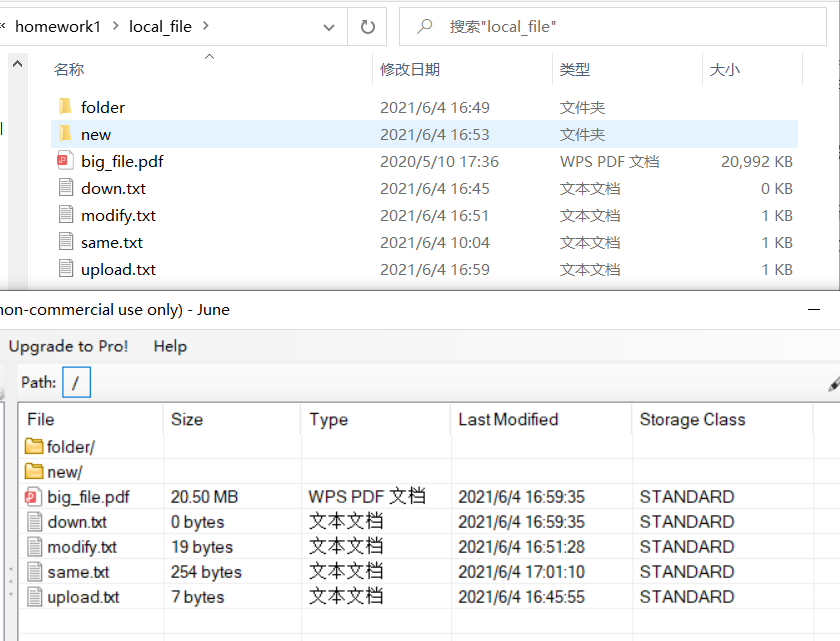
初始的文件同步

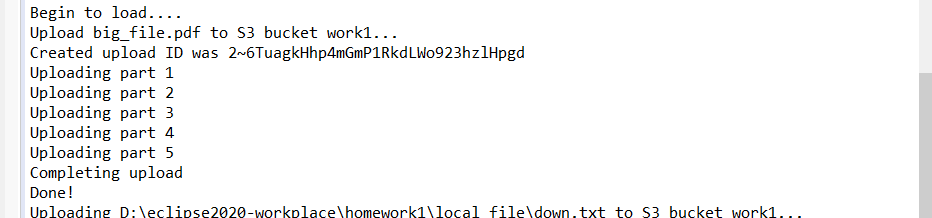


文件夹的同步



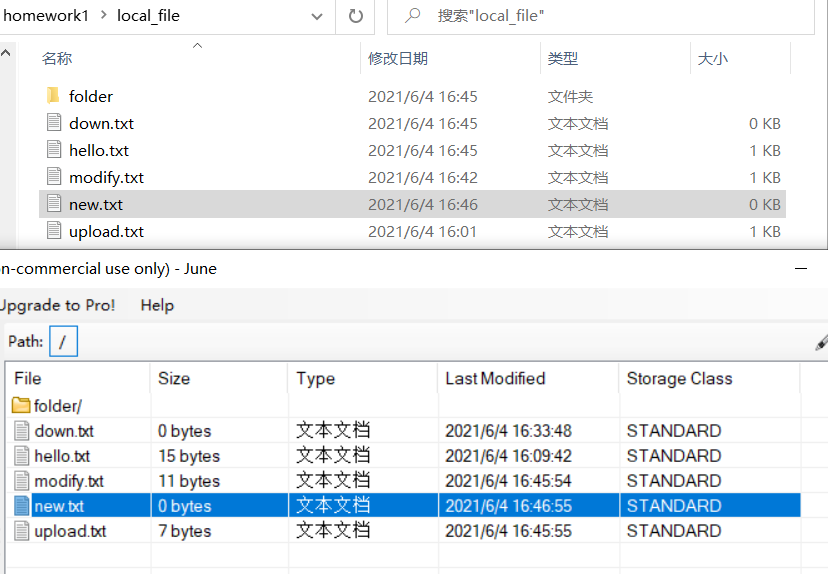
分块上传：



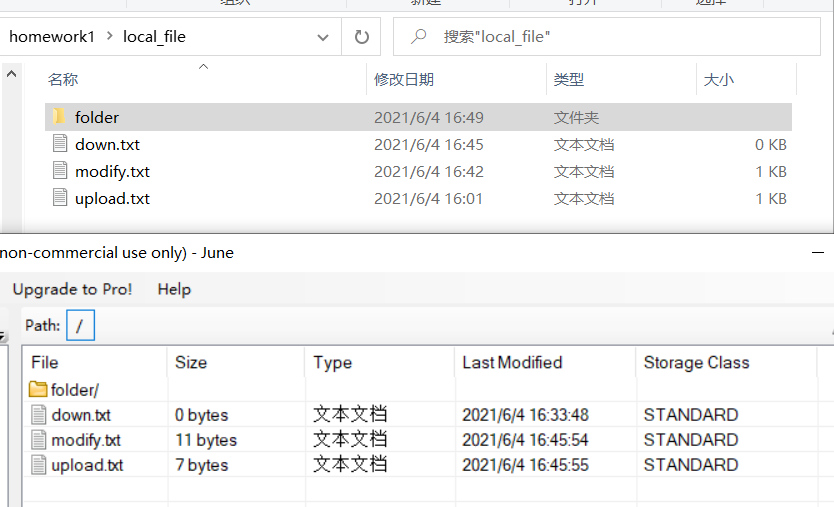


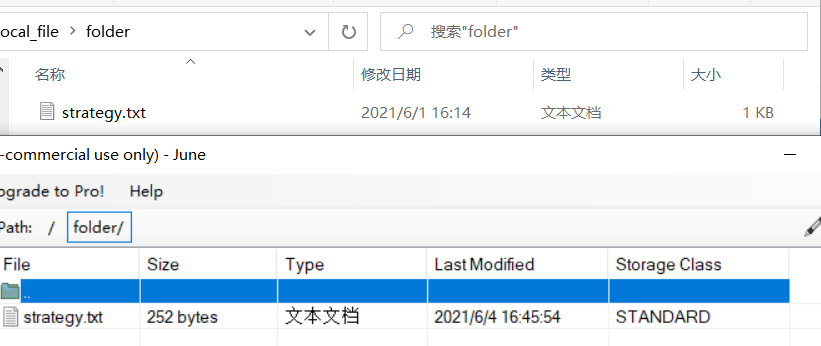
新文件的创建、修改和删除：

创建new.txt:

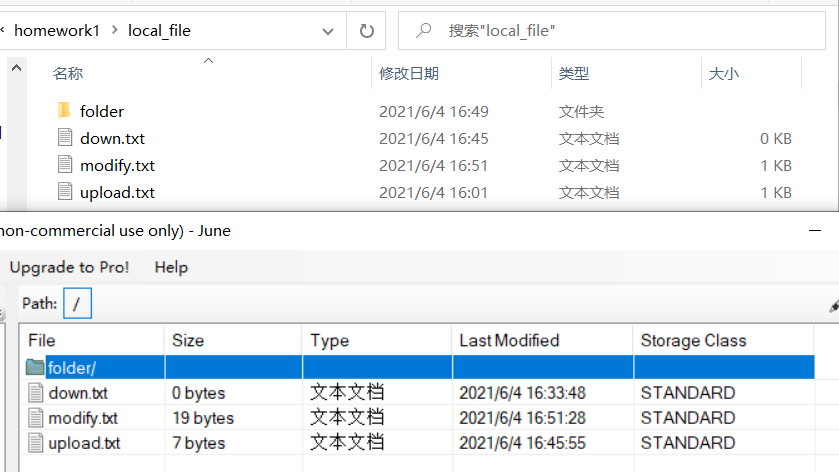


删除new.txt，hello.txt，folder文件夹中的test.txt:

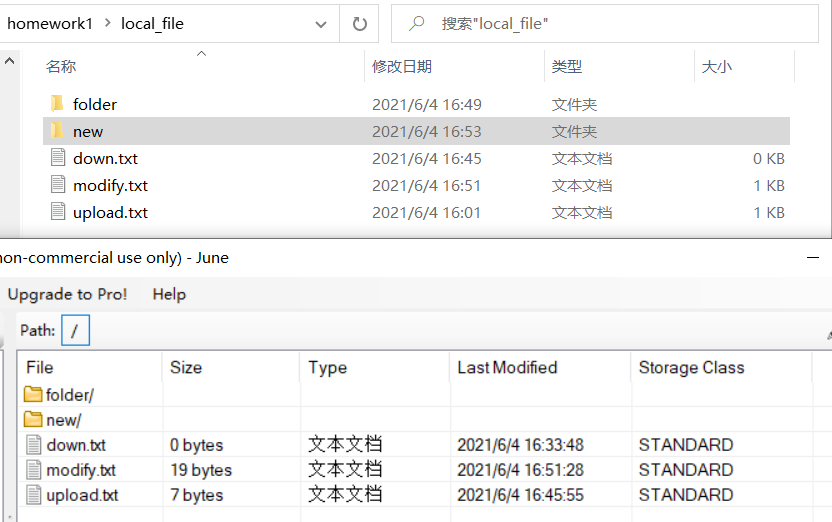


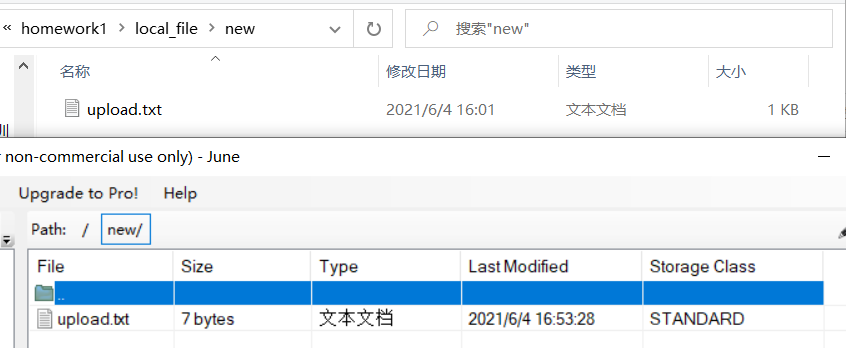


Modify.txt的修改：



新文件夹new的创建并创建一个文件在其中才能同步该文件夹和它的文件：





输出：

