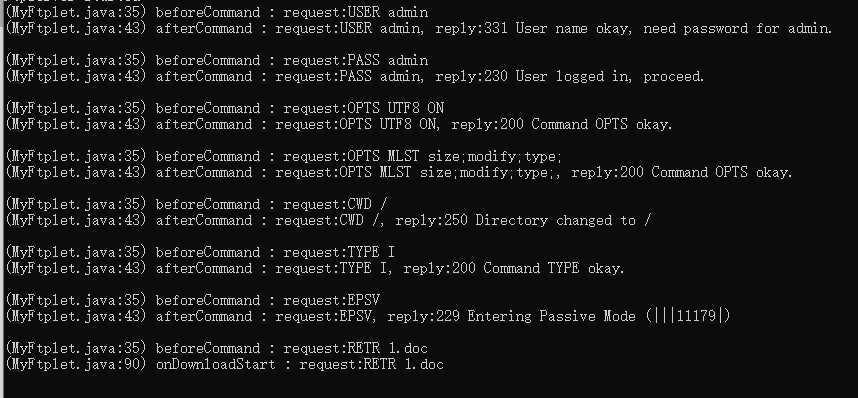
点击下载的时候：



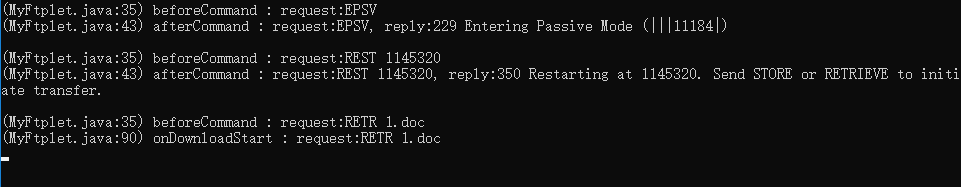
这么多的命令其实大概就是先刷新一下登录、然后刷新一下远程列表，然后设置目录，设置传输类型，设置被动模式，最后就下载命令

然后中断：



捕捉到异常，并输出信息

然后续传：



先设为被动模式，然后发送REST + 文件大小的命令（这里不是文件名，之前说错的，因为忘记了这个命令），session中就会记住了这个偏移，在下次的下载中就能获取这个偏移（session.getFileOffset();）。

测试环境：

win10 64位

Inter(R) Core(TM) i5-6200U CPU @ 2.30GHz 2.40GHz

16GB内存

设置max-threads = 100，max-logins = 80

符合预期：有7个同时下载10个文件，第8个同时下载2个文件，即

7 \* (10 + 1) + (1\*2 +1) = 80

这时总线程数



设置max-threads = 500，max-logins = 480

这个时候由于程序太多，已经无法统计同时下载数量

线程数在505的时候，就已经是峰值，收到421登录限制



测试的时候，大概在第40个以上，访问速度就有点慢了，就是连接到服务器，不能马上就连上，会等一小段时间才收到欢迎。大概43-45个窗口的时候，就已经收到421回复了，主要是没有设置空闲时间，所以有一些超过2分钟然后退出了，导致无法统计，但估算一下，一个窗口正常情况占1个登录10个下载，44\*11=484，的确是符合预期的480，这里是没有超过threads的，如果超过是收不到回复的。