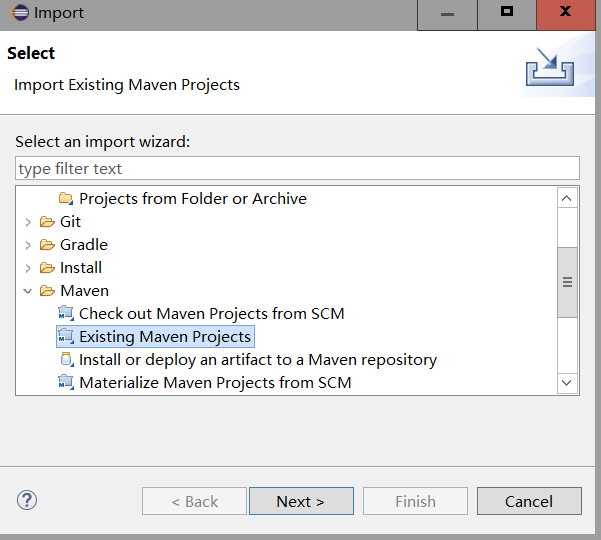
目前，电脑上应该有两个文件，一个是工程文件，另一个是数据。

工程文件文件名是：Storm\_Experiment

数据文件名是：2013-10 - Citi Bike trip data.csv和query.csv。

使用Eclipse这个IDE，使用java 8 作为语言。使用的是windows 10系统。

打开Eclipse，导入Maven项目，具体为：点击Eclipse左上角的File，选择import，选择Maven，选择Existing projects into Workspace。

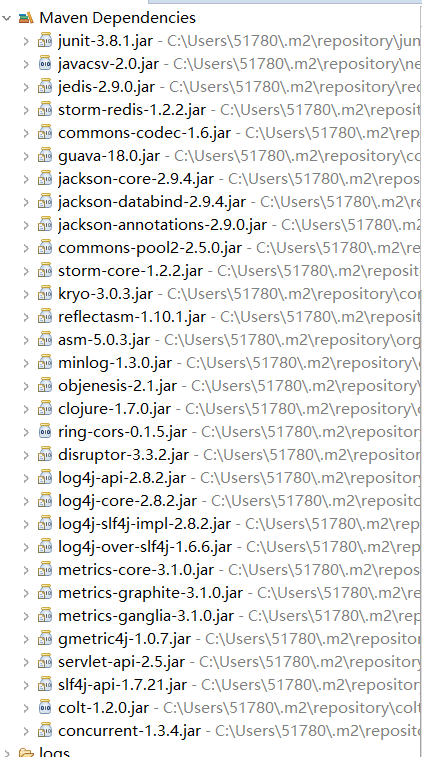


导入工程文件。接下来，等待一段时间，让电脑下载pom.xml中声明了依赖的关系包。

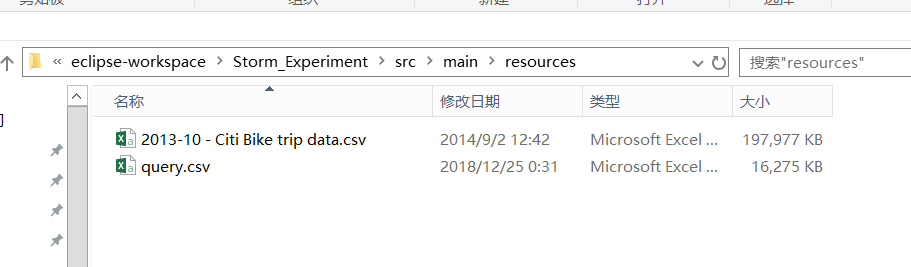
（如果发现没有自动下载，在Eclipse中打开pom.xml，随便删除一个依赖关系后重新添加上去）



如果下载成功，便可以在Maven Depencies中看到与下图一致的所有包。



接下来，将下载的数据（两个.csv文件放到src/main/resources文件夹下）



接下来，配置redis数据库。

（如果电脑上过着SS等代理软件，请关闭，否则会出现连接失败的玄学问题）

到网站：

<https://github.com/MSOpenTech/redis/releases>

去下载redis，我下载的是.zip包。



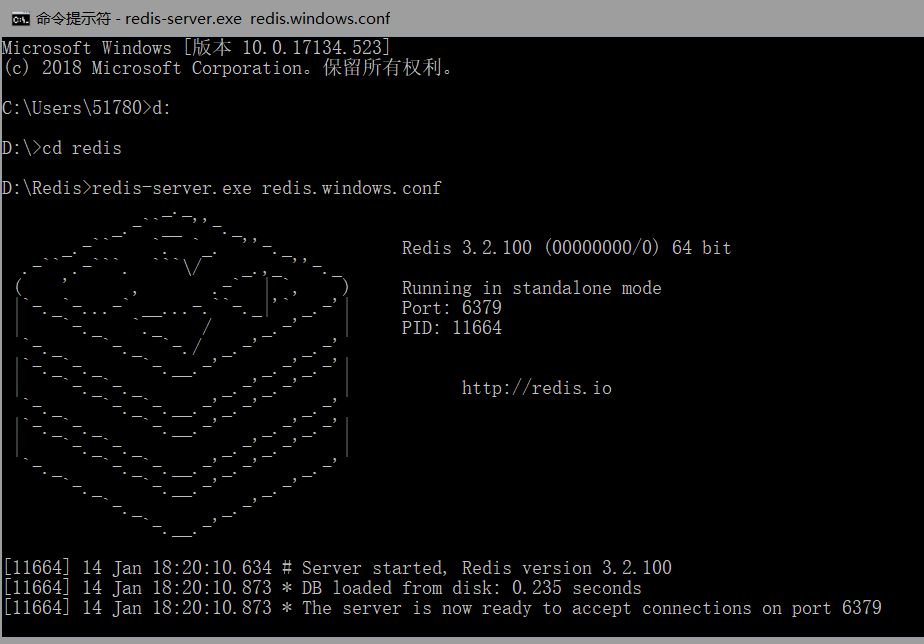
在D盘解压（所有的配置我都放在D盘，这个可以自行修改，但是后面的文件路径必须一样修改）。

打开windows的cmd，切换到D盘，用cd命令进入redis中，

输入:

redis-server.exe redis.windows.conf

可以看见如下界面：



不要关闭此cmd！

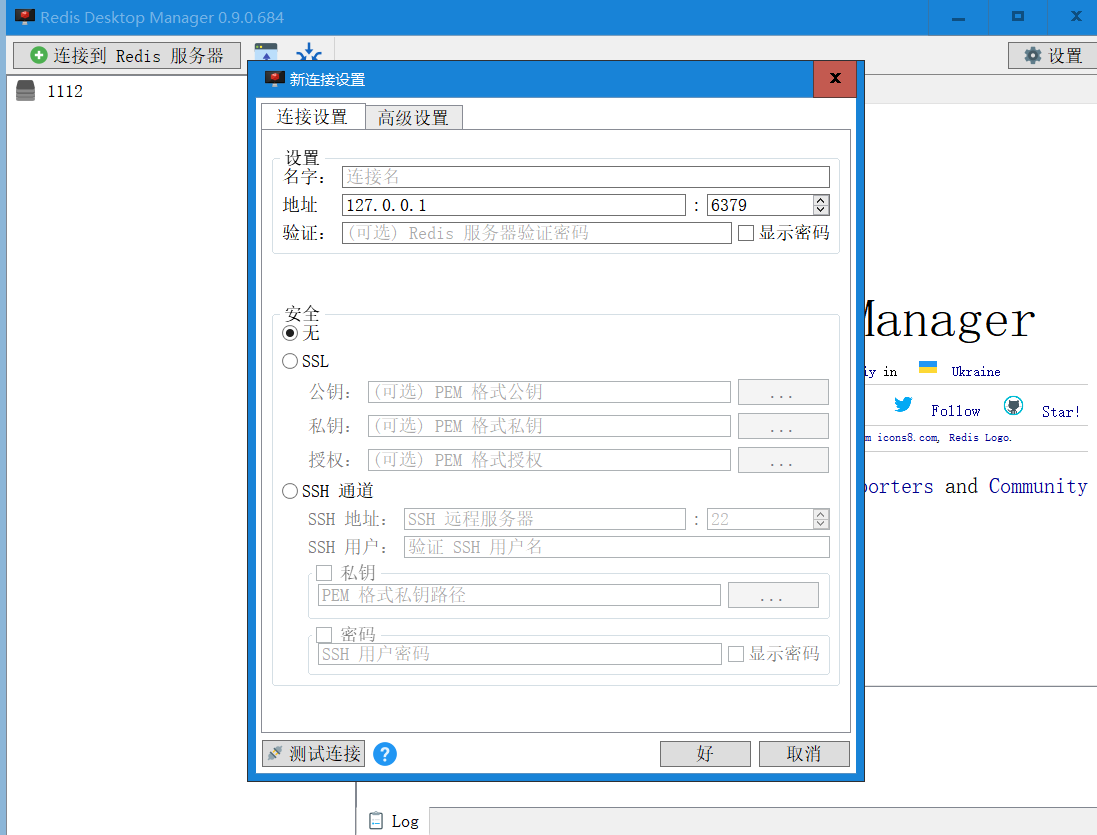
接下来，下载redis-desktop-manager（redis的可视化工具）

（可以在网站：

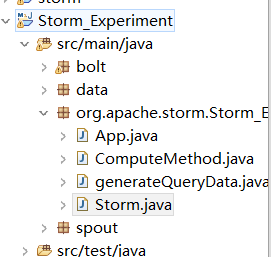
<https://sourceforge.net/projects/redis-desktop-manager.mirror/files/latest/download>

下载，官网是收费的）

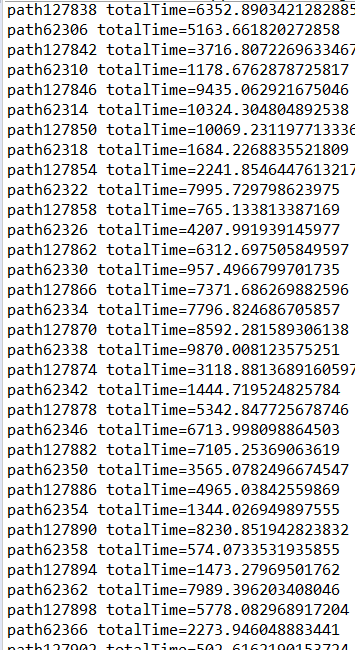
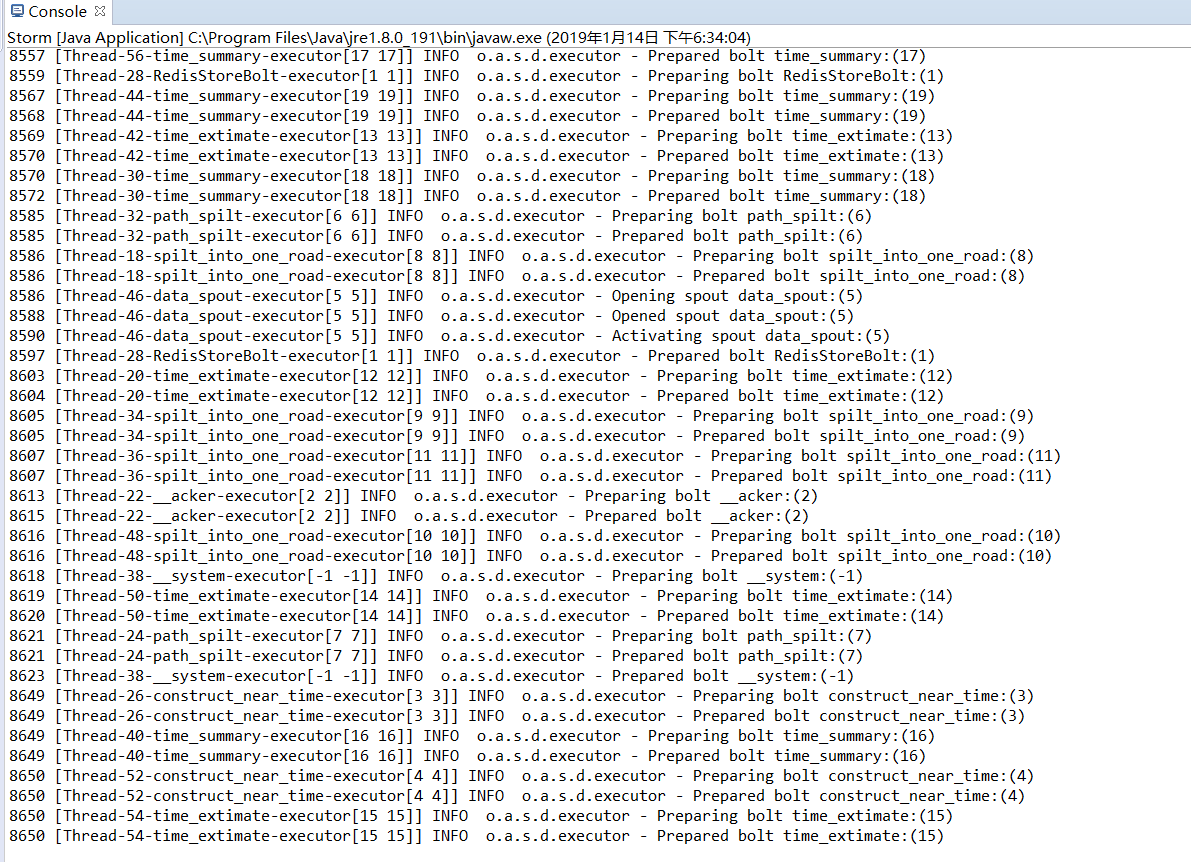
下载后进行安装，然后可以运行，出现如下界面，点击连接到Redis服务器，弹出一下窗口，自己随便输入一个连接名即可连接上Redis。



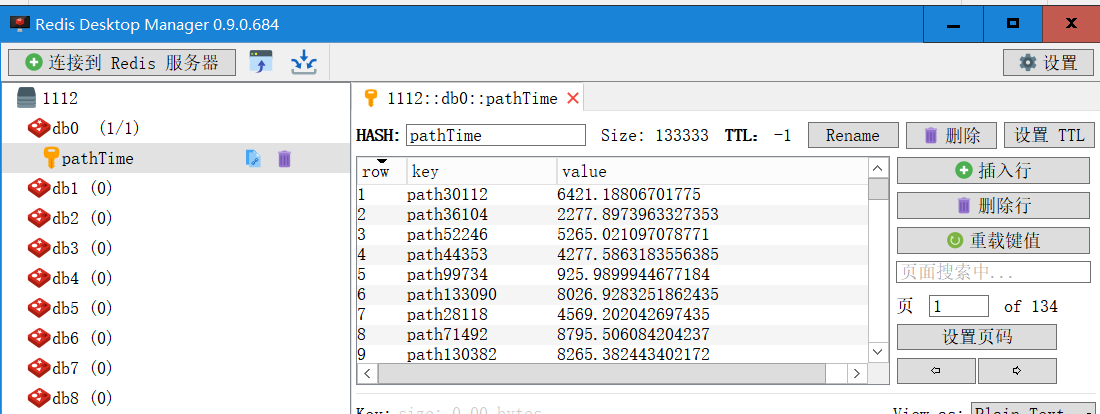
接下来，在Eclipse的项目中，切换到：org.apache.storm.Storm\_Experiment的Storm.java中，然后点击运行，即可以开始跑程序了（这时候使用的是Storm提供的模拟集群运行模式）。



正确运行的输出结果如下：



在redis-desktop-maneger中，可以看到结果如下：



（key值顺序并不一定一样，因为并行程序执行的顺序是无法估计的）

那么程序运行成功。

伪集群提交任务：

这一部分对于程序的执行没什么帮助，只是说明了伪集群的搭建过程，如果只是关心程序的执行结果，那么这一部分是可以跳过的，之前的结果已经是storm模拟集群的执行结果了。

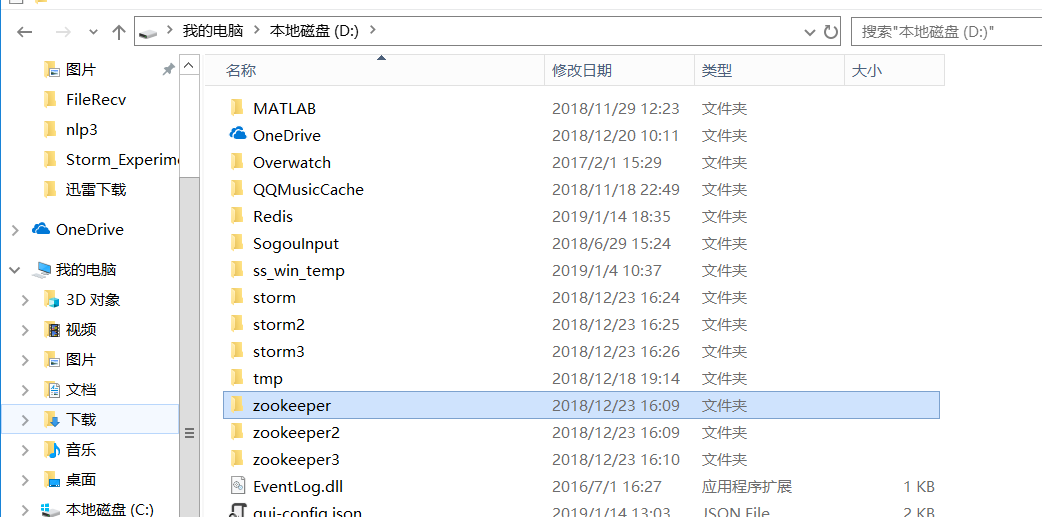
这一部分只是证明了：已经成功搭建了伪集群，对程序的运行结果说明没有任何帮助。

Storm提供了模拟集群的运行，之前执行的就是这一过程。

接下来做的是搭建伪集群，但是和结果没有影响，少了这一过程，程序如之前所示，也运行成功了。

具体过程：

在<https://zookeeper.apache.org/releases.html#download>下载zookeeper压缩包，解压后，拷贝3份，放在D盘，分别命名为zookeeper，zookeeper2，zookeeper3。



<1.进入zookeeper/conf,修改zoo.cfg为：

tickTime=2000

initLimit=10

syncLimit=5

dataDir=D://zookeeper//data

clientPort=2181

server.1=localhost:2887:3887

server.2=localhost:2888:3888

server.3=localhost:2889:3889

<2.进入zookeeper2/conf,修改zoo.cfg为：

# The number of milliseconds of each tick

tickTime=2000

# The number of ticks that the initial

# synchronization phase can take

initLimit=10

# The number of ticks that can pass between

# sending a request and getting an acknowledgement

syncLimit=5

# the directory where the snapshot is stored.

# do not use /tmp for storage, /tmp here is just

# example sakes.

dataDir=D://zookeeper2//data

# the port at which the clients will connect

clientPort=2182

server.1=localhost:2887:3887

server.2=localhost:2888:3888

server.3=localhost:2889:3889

# the maximum number of client connections.

# increase this if you need to handle more clients

#maxClientCnxns=60

#

# Be sure to read the maintenance section of the

# administrator guide before turning on autopurge.

#

# http://zookeeper.apache.org/doc/current/zookeeperAdmin.html#sc\_maintenance

#

# The number of snapshots to retain in dataDir

#autopurge.snapRetainCount=3

# Purge task interval in hours

# Set to "0" to disable auto purge feature

#autopurge.purgeInterval=1

<3..进入zookeeper3/conf,修改zoo.cfg为：

# The number of milliseconds of each tick

tickTime=2000

# The number of ticks that the initial

# synchronization phase can take

initLimit=10

# The number of ticks that can pass between

# sending a request and getting an acknowledgement

syncLimit=5

# the directory where the snapshot is stored.

# do not use /tmp for storage, /tmp here is just

# example sakes.

dataDir=D://zookeeper3//data

# the port at which the clients will connect

clientPort=2183

server.1=localhost:2887:3887

server.2=localhost:2888:3888

server.3=localhost:2889:3889

# the maximum number of client connections.

# increase this if you need to handle more clients

#maxClientCnxns=60

#

# Be sure to read the maintenance section of the

# administrator guide before turning on autopurge.

#

# http://zookeeper.apache.org/doc/current/zookeeperAdmin.html#sc\_maintenance

#

# The number of snapshots to retain in dataDir

#autopurge.snapRetainCount=3

# Purge task interval in hours

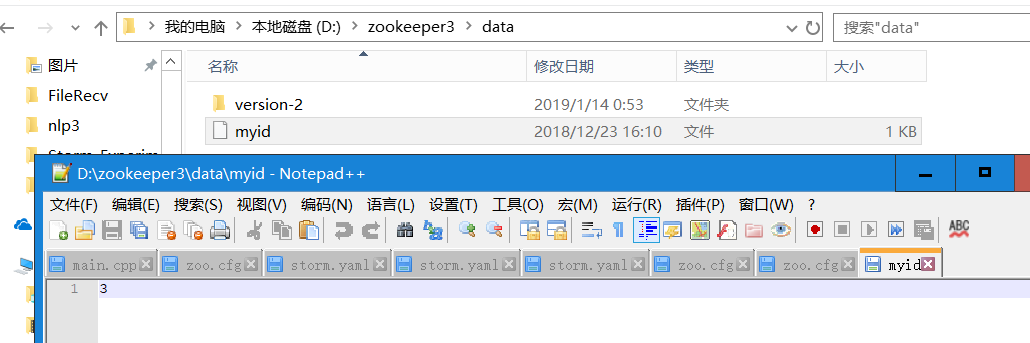
# Set to "0" to disable auto purge feature

#autopurge.purgeInterval=1

接下来进入：zookeeper/data文件夹，创建文件myid，里面填上1

Zookeeper2/data文件夹，创建文件myid，里面填上2

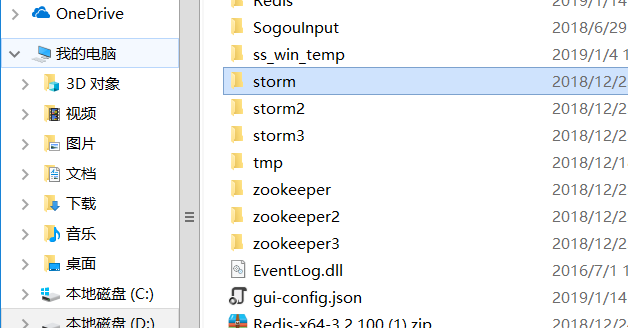
Zookeeper3/data文件夹，创建文件myid，里面填上3



接下来，到如下网站下载storm压缩包。

<http://storm.apache.org/downloads.html>

解压后一样复制三份，放在D盘，命名为storm，storm2，storm3



<1.修改storm/conf的storm.yaml内容如下：

########### These MUST be filled in for a storm configuration

storm.zookeeper.servers:

- "localhost"

storm.zookeeper.port: 2181

nimbus.host: "localhost"

supervisor.slots.ports:

- 6700

- 6701

- 6702

- 6703

storm.local.dir: "D:\\storm\\localdata"

storm.messaging.transport: "backtype.storm.messaging.netty.Context"

storm.messaging.netty.server\_worker\_threads: 1

storm.messaging.netty.client\_worker\_threads: 1

storm.messaging.netty.buffer\_size: 5242880

storm.messaging.netty.max\_retries: 100

storm.messaging.netty.max\_wait\_ms: 1000

storm.messaging.netty.min\_wait\_ms: 100

<2.修改storm2/conf的storm.yaml内容如下：

########### These MUST be filled in for a storm configuration

storm.zookeeper.servers:

- "localhost"

storm.zookeeper.port: 2182

nimbus.host: "localhost"

supervisor.slots.ports:

- 7700

- 7701

- 7702

- 7703

storm.local.dir: "D:\\storm2\\localdata"

storm.messaging.transport: "backtype.storm.messaging.netty.Context"

storm.messaging.netty.server\_worker\_threads: 1

storm.messaging.netty.client\_worker\_threads: 1

storm.messaging.netty.buffer\_size: 5242880

storm.messaging.netty.max\_retries: 100

storm.messaging.netty.max\_wait\_ms: 1000

storm.messaging.netty.min\_wait\_ms: 100

<3.修改storm3/conf的storm.yaml内容如下：

########### These MUST be filled in for a storm configuration

storm.zookeeper.servers:

- "localhost"

storm.zookeeper.port: 2183

nimbus.host: "localhost"

supervisor.slots.ports:

- 8700

- 8701

- 8702

- 8703

storm.local.dir: "D:\\storm3\\localdata"

storm.messaging.transport: "backtype.storm.messaging.netty.Context"

storm.messaging.netty.server\_worker\_threads: 1

storm.messaging.netty.client\_worker\_threads: 1

storm.messaging.netty.buffer\_size: 5242880

storm.messaging.netty.max\_retries: 100

storm.messaging.netty.max\_wait\_ms: 1000

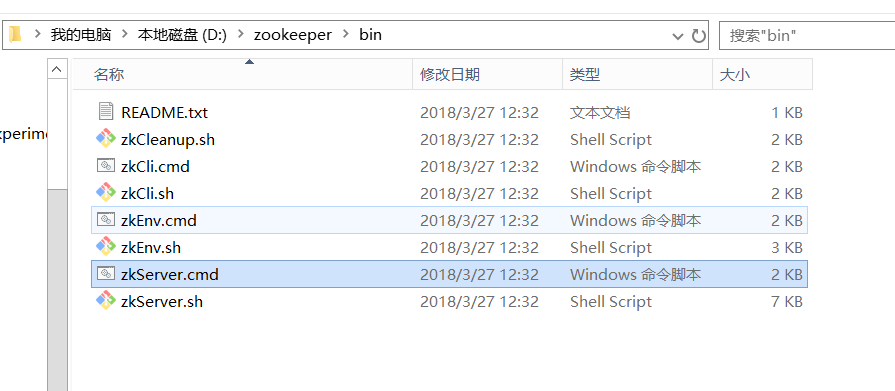
storm.messaging.netty.min\_wait\_ms: 100

最后：

打开zookeeper/bin，双击zkServer.cmd，不要关闭

打开zookeeper2/bin，双击zkServer.cmd，不要关闭

打开zookeeper3/bin，双击zkServer.cmd，不要关闭



打开使用windows的cmd，切换到storm/bin，输入storm.py nimbus

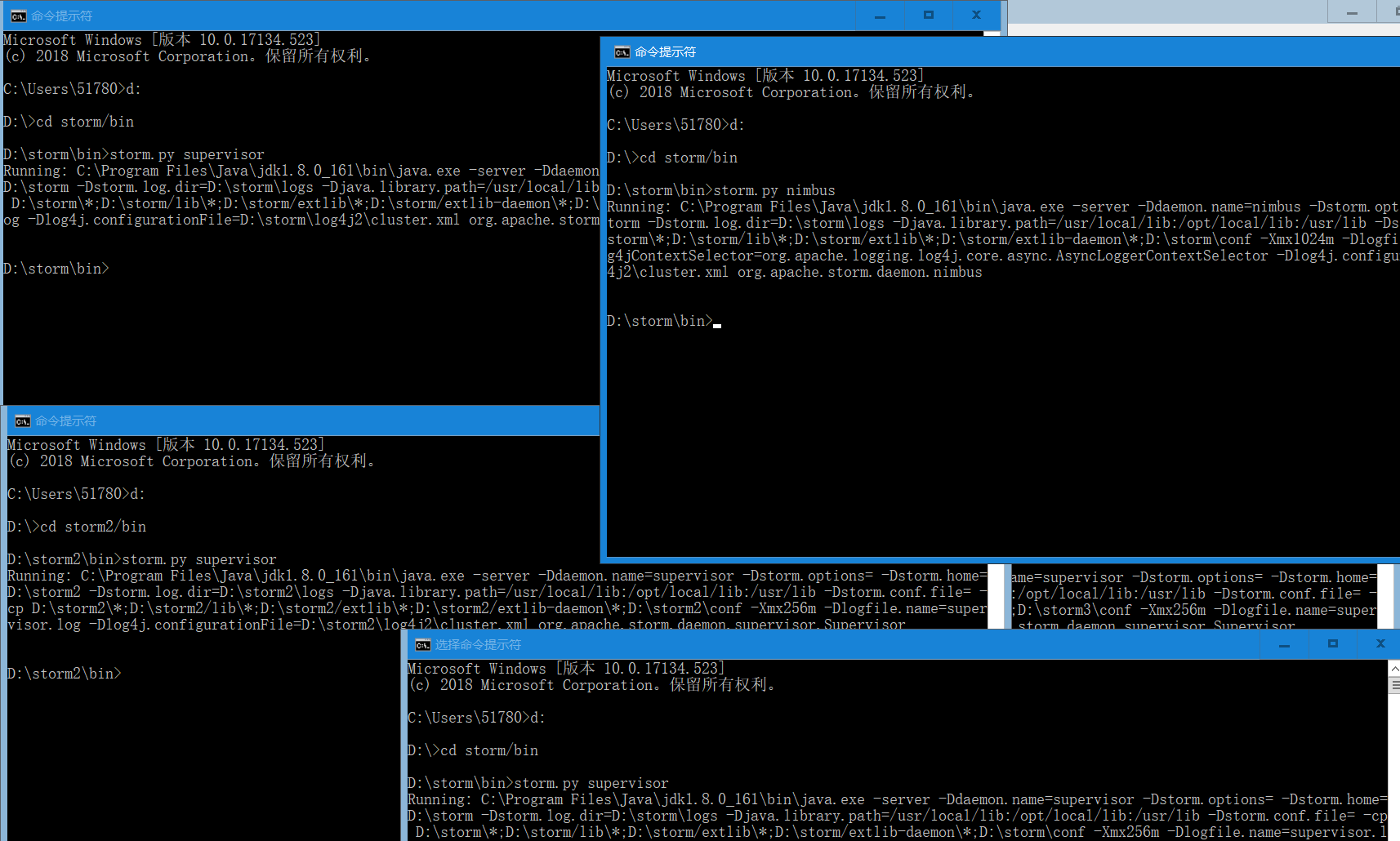
打开使用windows的cmd，切换到storm/bin，输入storm.py supervisor

打开使用windows的cmd，切换到storm/bin，输入storm.py ui

打开使用windows的cmd，切换到storm2/bin，输入storm.py supervisor

打开使用windows的cmd，切换到storm3/bin，输入storm.py supervisor

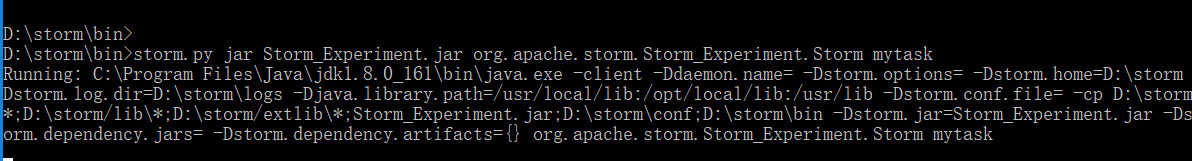
如下图：所有打开的cmd都不要关闭。



接下来，将Storm\_Experiment/target目录下的Storm\_Experiment.jar拷贝到storm/bin目录下。

最后，打开使用windows的cmd，切换到storm/bin，输入：

storm.py jar Storm\_Experiment.jar org.apache.storm.Storm\_Experiment.Storm mytask



一段时间以后：

打开浏览器，输入localhost:8080，可以看到提交的任务mytask，并且Nimbus和Supervisor都有机器，那么伪集群搭建完成。

