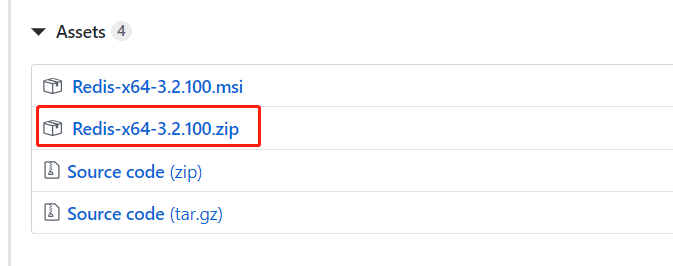
**Redis环境搭建说明**

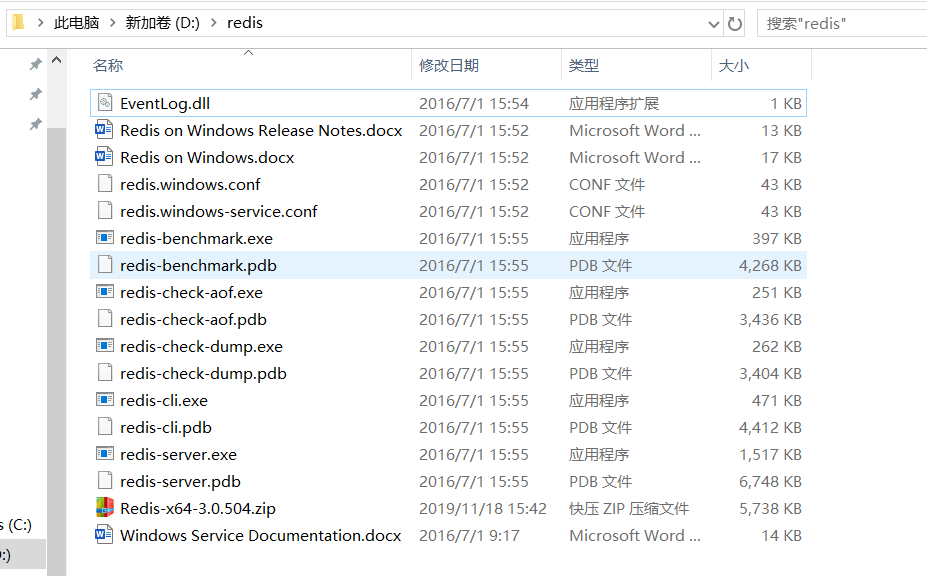
# 环境搭建

## 1 standalone模式

windows 64位系统下载redis路径：<https://github.com/MSOpenTech/redis/releases>，下载zip包

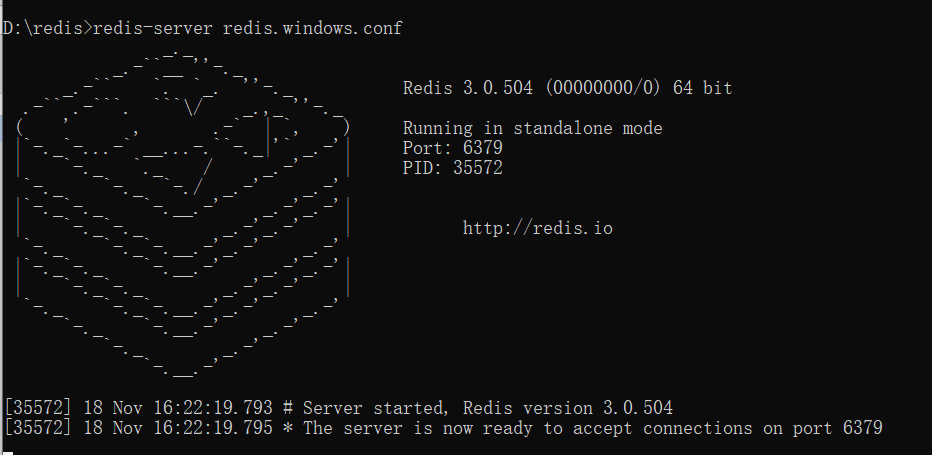


下载完解压文件，我这里解压到了D盘



在此目录下打开cmd’窗口，执行redis-server redis.windows.conf

如图：



Redis服务已启动。

## 2 sentinel模式

### 2.1 原理

**监控(Monitoring)**: 哨兵(sentinel) 会不断地检查你的Master和Slave是否运作正常。

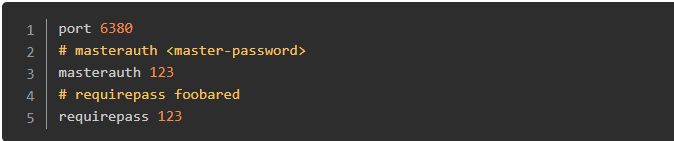
**提醒(Notification)**:当被监控的某个 Redis出现问题时, 哨兵(sentinel) 可以通过 API 向管理员或者其他应用程序发送通知。

**自动故障迁移(Automatic failover)**:当一个Master不能正常工作时，哨兵(sentinel) 会开始一次自动故障迁移操作,它会将失效Master的其中一个Slave升级为新的Master, 并让失效Master的其他Slave改为复制新的Master; 当客户端试图连接失效的Master时,集群也会向客户端返回新Master的地址,使得集群可以使用Master代替失效Master。

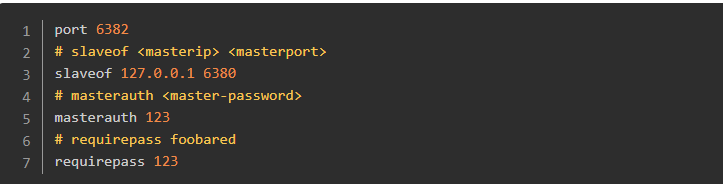
### 2.2 环境搭建

将redis.windows.conf文件复制6份，分别命名为**redis6380.conf，redis6382.conf，redis6384.conf，sentinel26380.conf，sentinel26382.conf，sentinel26384.conf**

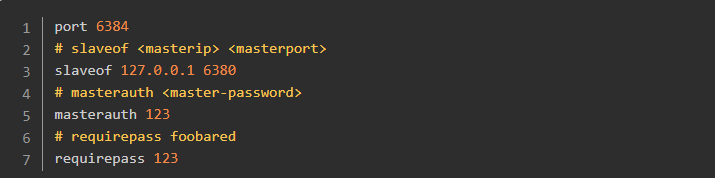
**配置redis6380.conf文件(master)**



**配置redis6382.conf文件(slave)**



**配置redis6384.conf文件(slave)**



**配置sentinel26380.conf文件(*其他两个文件修改端口号就可以*)**



**依次cmd命令启动Redis**

**redis-server.exe redis6380.conf**

**redis-server.exe redis6382.conf**

**redis-server.exe redis6384.conf**

**redis-server.exe sentinel26380.conf --sentinel**

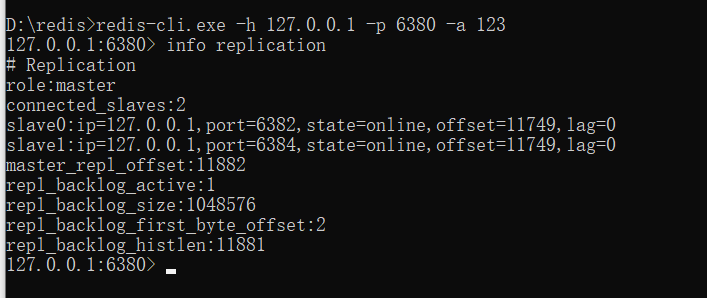
**redis-server.exe sentinel26382.conf --sentinel**

**redis-server.exe sentinel26384.conf --sentinel**

**客户端查看master和哨兵状态**

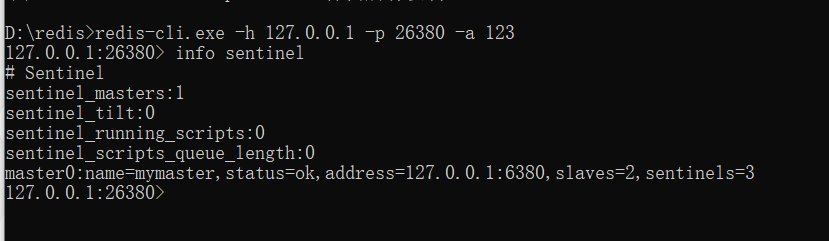
**redis-cli.exe -h 127.0.0.1 -p 6380 -a 123**

**info replication**



**redis-cli.exe -h 127.0.0.1 -p 26380 -a 123**

**info sentinel**



搭建完成。

## 3 cluster模式

### 3.1 集群概述

Redis 集群是一个提供在多个Redis间节点间共享数据的程序集。

Redis集群并不支持处理多个keys的命令,因为这需要在不同的节点间移动数据,从而达不到像Redis那样的性能,在高负载的情况下可能会导致不可预料的错误.

Redis 集群通过分区来提供一定程度的可用性,在实际环境中当某个节点宕机或者不可达的情况下继续处理命令. Redis 集群的优势:

自动分割数据到不同的节点上。

整个集群的部分节点失败或者不可达的情况下能够继续处理命令。

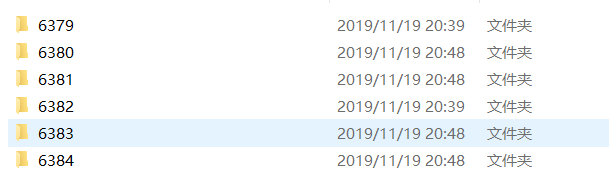
Redis 集群的数据分片 Redis 集群没有使用一致性hash, 而是引入了 哈希槽的概念.

### 3.2 环境搭建

#### 3.2.1 构建集群节点目录

集群正常运作至少需要三个主节点，不过在刚开始试用集群功能时， 强烈建议使用六个节点： 其中三个为主节点， 而其余三个则是各个主节点的从节点。主节点崩溃，从节点的Redis就会提升为主节点，代替原来的主节点工作，崩溃的主Redis回复工作后，会成为从节点

将下载的redis的压缩包解压到6个新文件夹，分别以节点端口命名



在每个文件夹中新建文件，输入以下命令

title redis-6379;

redis-server.exe redis.windows.conf

然后保存为start.bat 下次启动时直接执行该脚本即可

#### 3.2.2 修改配置

打开每个节点文件夹的redis.windows.conf文件，修改配置

port 6379 *//修改为与当前文件夹名字一样的端口号*

cluster-enabled yes *//开启集群模式*

cluster-config-file nodes-6379.conf *//保存节点配置,自动创建,自动更新(建议命名时加上端口号)*

cluster-node-timeout 15000 *//集群超时时间，节点超过这个时间没反应就断定是宕机*

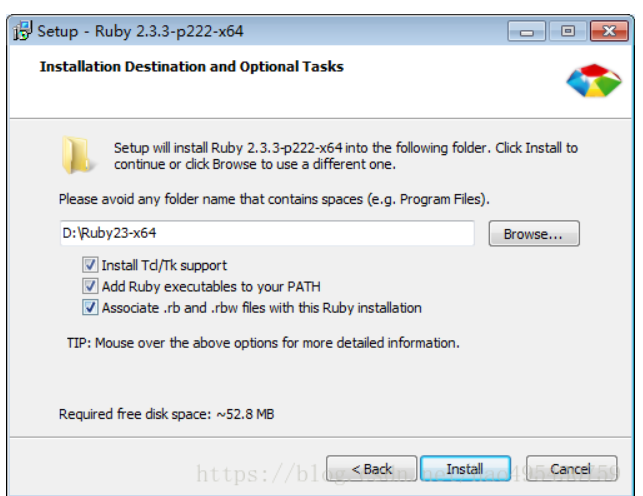
appendonly yes *//指定是否在每次更新操作后进行日志记录*

其他文件节点 6380~6384也修改相应的节点配置信息和建立启动脚本

#### 3.2.3 下载Ruby并安装

下载地址：<http://railsinstaller.org/en>

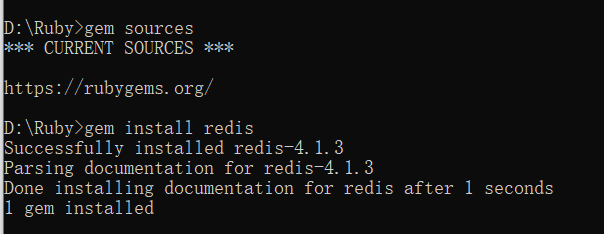
下载完成后安装，一步步点next知道安装完成(安装时勾选3个选项)



然后对ruby进行配置：

命令：gem sources

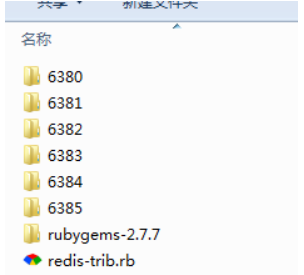
gem install redis



#### 3.2.4 构建集群脚本redis-trib.rb

可以打开 <https://raw.githubusercontent.com/antirez/redis/unstable/src/redis-trib.rb> 然后复制里面的内容到本地并保存为redis-trib.rb；

与redis集群节点保存在同一个文件夹下



#### 3.2.5 创建集群

依次启动所有集群节点start.bat

然后cmd进入redis集群节点目录后，执行： （–replicas 1 表示为集群中的每个主节点创建一个从节点）

命令：redis-trib.rb create --replicas 1 127.0.0.1:6379 127.0.0.1:6380 127.0.0.1:6381 127.0.0.1:6382 127.0.0.1:6383 127.0.0.1:6384

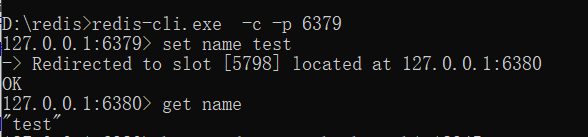
中途会询问是否打印更多详细信息，输入yes即可，然后redis-trib 就会将这份配置应用到集群当中,让各个节点开始互相通讯

集群构建完成

#### 3.2.6 测试集群

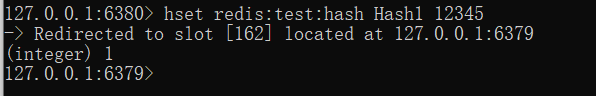
进入任意一个集群节点，cmd执行  redis-cli.exe  -c -p 6379

写入任意一个value，查询



写一个hash:

hset redis:test:hash Hash1 12345



可以看到集群会用CRC16算法来取模得到所属的slot，然后将这个key分到哈希槽区间的节点上CRC16(key) % 16384

所以，可以看到我们set的key计算之后被分配到了slot-162 上， 而slot-162处在节点6379上，因此redis自动redirect到了6379节点上。