# NoSQL数据库简介

## 技术发展

技术分类：

* 解决功能性问题：Java,jsp,Tomcat,HTML,Linux,JDBC
* 解决扩展性问题：Spring,SpringMVC,Mybatis
* 解决性能问题：NoSQL,Java线程，Hadoop,Nginx,MQ,ElasticSearch

## NoSQL概述以及适用场景

NoSQL = Not Only SQL,意思是不仅仅是SQL,泛指非关系型数据库。NoSQL不依赖业务逻辑方式存储，而是以简单的key-value模式存储，因此大大增加了数据库的拓展能力。

* 不支持SQL标准
* 不支持ACID原则
* 远超于SQL性能

NoSQL适用场景

* 对数据高并发读写
* 海量数据的读写
* 对数据高可扩展性

NoSQL不适应场景

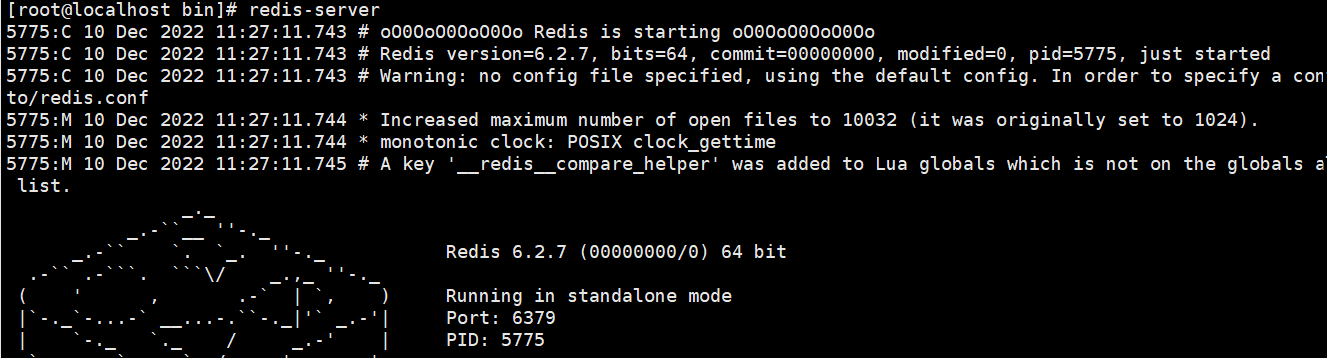
* 需要事务支持
* 基于sql的结构化存储查询，处理复杂关系

常见NoSQL：Redis,MongoDB

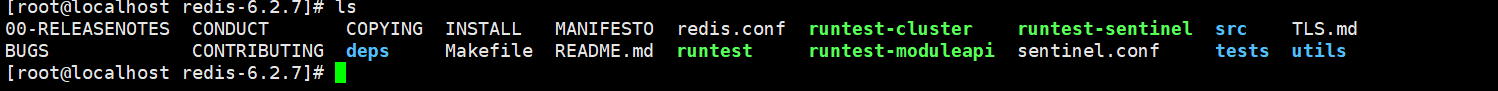
## Redis概述

* Redis是一个开源的key-value存储系统
* 数据类型都支持push/pop,add/remove及取交集并集等更丰富的操作。
* 不考虑对windows的支持，基于Linux系统
* Linux具体安装步骤百科，安装成功后默认安装目录/usr/local/bin，该目录下启动redis服务
  + redis-server：Redis服务器启动命令
  + redis-cli：客户端，操作入口

Redis前台启动（不推荐）演示：可以看到端口号是6379，进程ID5775，前台启动后窗口不能关闭（ctrl+c关闭），否则redis就关闭服务



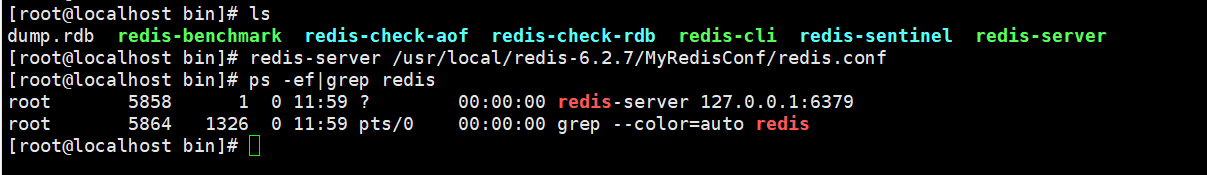
Redis后台启动（推荐）：Redis解压缩的目录中有一个文件redis.conf



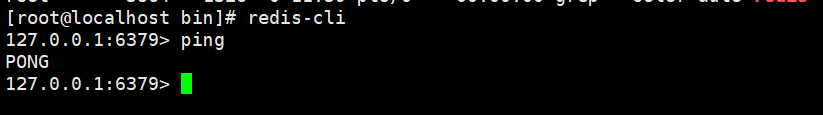
（这里备份了redis.conf文件在MyRedisConf目录中，该目录创建在redis的解压缩目录中，同时将该配置文件中的daemonize no改为yes）

进入/usr/local/bin目录，

后台启动命令（使用配置文件）：redis-server /usr/local/redis-6.2.7/MyRedisConf/redis.conf



连接redis客户端：redis-cli，pong表示当前正常连接状态



关闭redis：

* redis-cli shutdown
* 进入客户端，输入命令shutdown
* kill -9 进程id

Resi是单线程加多路IO复用技术

### Redis基础命令

默认端口号：6379

默认右16个数据库，类似于数组，下表从0开始，初始默认使用0号库

库命令：

* select 下标：切换对应下标的数据库
* dbsize：查看当前数据库的key数量
* flushdb：清空当前数据库
* flushall：清空全部库

key命令：

* keys \*：查看当前库所有key
* exists key：判断该key是否存在
* type key：查看key是什么类型
* del key：删除指定的key数据
* unlink key：根据value选择非阻塞删除（仅将keys从keyspace元数据中删除，真正的删除会在后续异步操作）
* expire key xxx：设置key的过期时间，xxx是整数，代表了多少秒
* ttl key：查看key的过期时间，-1表示永不过期，-2表示已经过期

set命令：

* set key value：添加一个键值对，如果设置键值对中key已存在，则value值覆盖
* get key：获取key的值
* append key value：将value值追加到key对应值的后面
* strlen key：获取值的长度
* setnx key value：只有key不存在时，设置key的值
* incr key：将key中存储的数字值加一，原子性操作
* decr key：将key存储的数字值减一，原子性操作

### Redis基本数据类型

#### 字符串String

string是redis最基本的类型，且是二进制安全的，意味着redis的string可以包含任何数据，如图片或者序列化对象，在Redis中字符串value最多可以是512M