

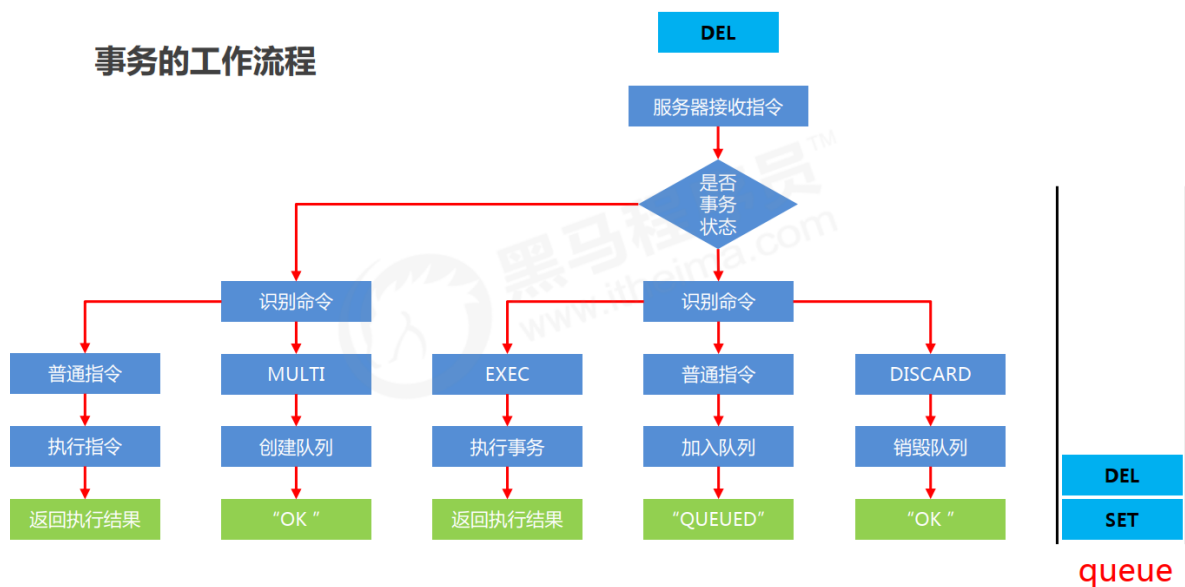
Redis事务

一个队列中，一次性，顺序性，排他性的执行一系列命令

事务的基本操作

- 开启事务
 - multi: 设定事务的开启位置，此指令执行后，后续的所有指令均加入到事务中
- 执行事务
 - exec: 设定事务的结束位置，同时执行事务，与multi成对出现，成对使用
- 取消事务
 - discard: 终止当前事务的定义，发生在multi之后，exec之前。
- 注意：加入事务的命令暂时进入到任务队列中，并没有立即执行，只有执行exec命令才开始执行

事务的工作流程



事务的注意事项

- 定义事务的过程中，命令格式输入错误怎么办？

整体事务中的所有命令均不会执行，包括那些语法正确的命令

- 定义事务中，命令执行出现错误怎么办？

命令格式正确，但是无法正确执行，例如对list进行incr操作。redis会运行能够正确执行的命令，而运行错误的命令不会被执行

基于特定条件的事务执行---锁

- 对key添加监视锁，在执行exec前，如果可以发生了变化，终止当前事务

watch key1 key2。。。。

unwatch：取消对所有key的监视

基于特定条件的事务执行---分布式锁

使用setnx设置一个公共锁

setnx lock-key value

利用setnx命令的返回值特征，有值则返回设置失败，无值则返回设置成功

- 对于返回设置成功的，拥有控制权，进行下一步的具体业务操作
- 对于返回设置失败的，不具有控制权，排队或等待

操作完毕通过del操作释放锁

基于特定条件的事务执行---分布式锁改良

使用expire为锁key添加时间限定，到时不释放，放弃锁，防止造成死锁