## 实例

以下是一个事务的例子， 它先以 **MULTI** 开始一个事务， 然后将多个命令入队到事务中， 最后由 **EXEC** 命令触发事务， 一并执行事务中的所有命令：

redis 127.0.0.1:6379> MULTI

OK

redis 127.0.0.1:6379> SET book-name "Mastering C++ in 21 days"

QUEUED

redis 127.0.0.1:6379> GET book-name

QUEUED

redis 127.0.0.1:6379> SADD tag "C++" "Programming" "Mastering Series"

QUEUED

redis 127.0.0.1:6379> SMEMBERS tag

QUEUED

redis 127.0.0.1:6379> EXEC

1. OK
2. "Mastering C++ in 21 days"
3. (integer) 3

4) 1) "Mastering Series"

2) "C++"

3) "Programming"

单个 Redis 命令的执行是原子性的，但 Redis 没有在事务上增加任何维持原子性的机制，所以 Redis 事务的执行并不是原子性的。

事务可以理解为一个打包的批量执行脚本，但批量指令并非原子化的操作，中间某条指令的失败不会导致前面已做指令的回滚，也不会造成后续的指令不做。