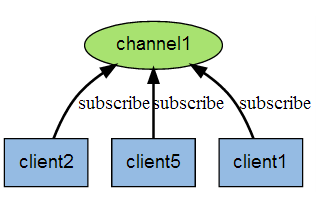
# Redis 发布订阅

## 简介

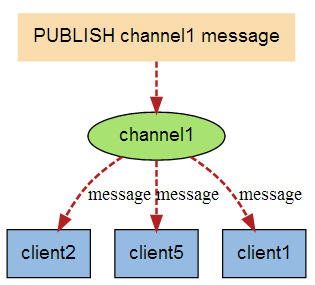
1. Redis 发布订阅 (pub/sub) 是一种消息通信模式：发送者 (pub) 发送消息，订阅者 (sub) 接收消息。
2. Redis 客户端可以订阅任意数量的频道。

## 工作流程

1. 频道 channel1，以及订阅这个频道的三个客户端 —— client2 、 client5 和 client1 之间的关系：

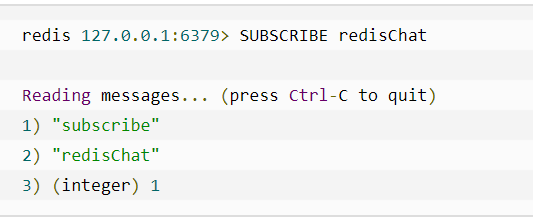


1. 当有新消息通过 PUBLISH 命令发送给频道 channel1 时， 这个消息就会被发送给订阅它的三个客户端：



## 实例

* 1. 创建订阅频道redisChat



* 1. 重新开启个 redis 客户端，然后在同一个频道 redisChat 发布两次消息，订阅者就能接收到消息。



## 命令



# Redis 事务

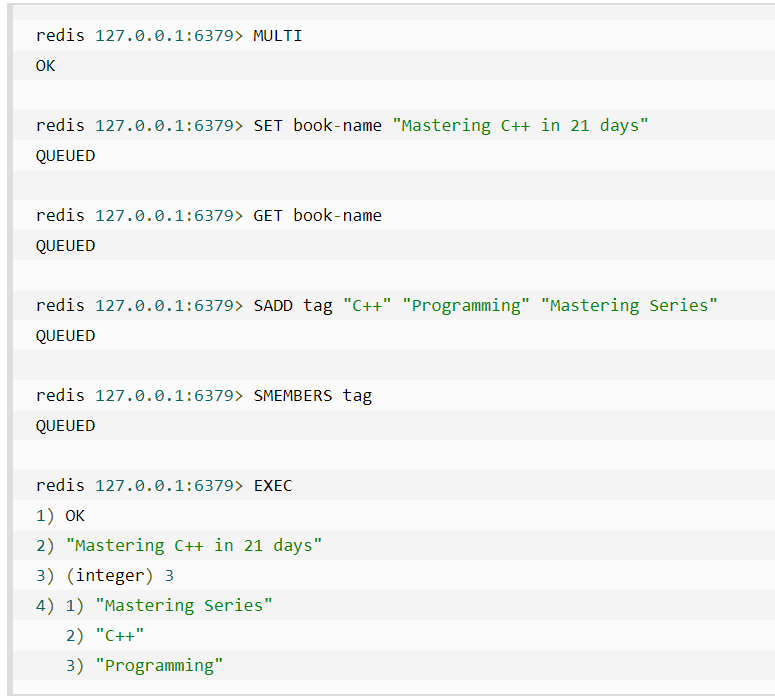
## 简介

1.1 Redis 事务可以一次执行多个命令， 并且带有以下三个重要的保证：

1. 批量操作在发送 EXEC 命令前被放入队列缓存。
2. 收到 EXEC 命令后进入事务执行，**事务中任意命令执行失败，其余的命令依然被执行**。
3. 在事务执行过程，其他客户端提交的命令请求不会插入到事务执行命令序列中。
   1. 一个事务从开始到执行会经历以下三个阶段：
4. 开始事务。
5. 命令入队。
6. 执行事务。

## 实例

1. 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：



1. 事务开始命令是MULTI,结束命令是 EXEC
2. 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。
3. 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。

## 相关命令



# Redis 脚本

## 简介

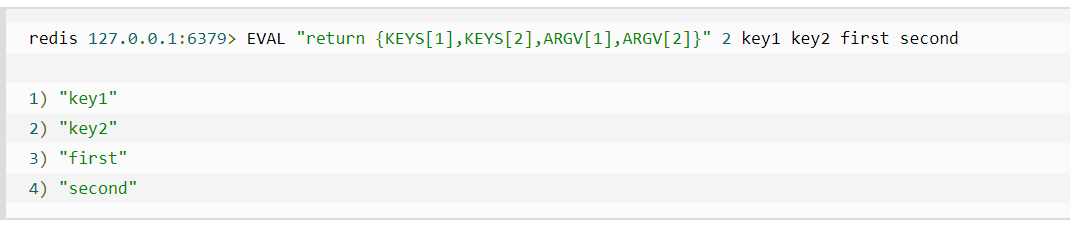
Redis 脚本使用 Lua 解释器来执行脚本。 Redis 2.6 版本通过内嵌支持 Lua 环境。

## 基本语法

1. Eval命令

redis 127.0.0.1:6379> EVAL script numkeys key [key ...] arg [arg ...]

1. 实例



## 相关命令



<https://www.cnblogs.com/huangxincheng/p/6230129.html>

脚本具体看这篇

# Redis 连接

## 简介

Redis 连接命令主要是用于连接 redis 服务。

## 实例



## 相关命令



# Redis 服务器

## 简介

Redis 服务器命令主要是用于管理 redis 服务。

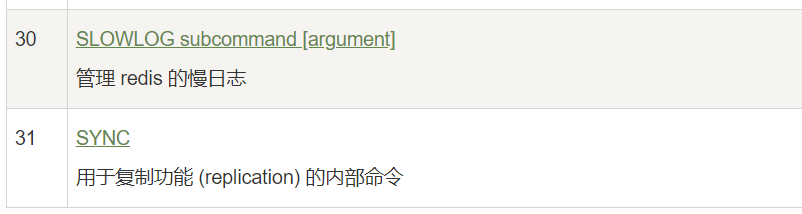
## 服务器命令











## 具体实例

见文件