

## 期中测试题答案-这些问题，你都答对了吗？

你好，我是蒋德钧。今天，我来公布一下主观题的答案。

### 第一题

Redis在接收多个网络客户端发送的请求操作时，如果有一个客户端和Redis的网络连接断开了，Redis会一直等待该客户端恢复连接吗？为什么？

答案：

Redis不会等待客户端恢复连接。

原因是，Redis的网络连接是由操作系统进行处理的，操作系统内核负责监听网络连接套接字上的连接请求或数据请求，而Redis采用了IO多路复用机制epoll，不会阻塞在某一个特定的套接字上。epoll机制监测到套接字上有请求到达时，就会触发相应的事件，并把事件放到一个队列中，Redis就会对这个事件队列中的事件进行处理。这样一来，Redis只用查看和处理事件队列，就可以了。当客户端网络连接断开或恢复时，操作系统会进行处理，并且在客户端能再次发送请求时，把接收到的请求以事件形式通知Redis。

### 第二题

Redis的主从集群可以提升数据可靠性，主节点在和从节点进行数据同步时，会使用两个缓冲区：复制缓冲区和复制积压缓冲区。这两个缓冲区的作用各是什么？会对Redis主从同步产生什么影响吗？

答案：

首先来说一下复制缓冲区。

**作用：**主节点开始和一个从节点进行全量同步时，会为从节点创建一个输出缓冲区，这个缓冲区就是复制缓冲区。当主节点向从节点发送RDB文件时，如果又接收到了写命令操作，就会把它们暂存在复制缓冲区中。等RDB文件传输完成，并且在从节点加载完成后，主节点再把复制缓冲区中的写命令发给从节点，进行同步。

**对主从同步的影响：**如果主库传输RDB文件以及从库加载RDB文件耗时长，同时主库接收的写命令操作较多，就会导致复制缓冲区被写满而溢出。一旦溢出，主库就会关闭和从库的网络连接，重新开始全量同步。所以，我们可以通过调整client-output-buffer-limit slave这个配置项，来增加复制缓冲区的大小，以免复制缓冲区溢出。

再看看复制积压缓冲区。

**作用：**主节点和从节点进行常规同步时，会把写命令也暂存在复制积压缓冲区中。如果从节点和主节点间发生了网络断连，等从节点再次连接后，可以从复制积压缓冲区中同步尚未复制的命令操作。

**对主从同步的影响：**如果从节点和主节点间的网络断连时间过长，复制积压缓冲区可能被新写入的命令覆盖。此时，从节点就没有办法和主节点进行增量复制了，而是只能进行全量复制。针对这个问题，应对的方法是调大复制积压缓冲区的大小（可以参考第6讲中对repl\_backlog\_size的设置）。

### 第三题

假设在业务场景中，我们有20GB的短视频属性信息（包括短视频ID、短视频基本信息，例如短视频作者、创建时间等）要持久化保存，并且线上负载以读为主，需要能快速查询到这些短视频信息。

现在，针对这个需求，我们想使用Redis来解决，请你来设计一个解决方案。我来提几个问题，你可以思考一下。

首先，你会用Redis的什么数据类型来保存数据？如果我们只用单个实例来运行的话，你会采用什么样的持久化方案来保证数据的可靠性？

另外，如果不使用单实例运行，我们有两个备选方案：一个是用两台32GB内存的云主机来运行主从两个Redis实例；另一个是用10台8GB的云主机来运行Redis Cluster，每两台云主机分别运行一个Redis实例主库和从库，分别保存4GB数据，你会用哪种方案呢？请聊一聊你的想法。

**答案：**

Redis的Hash类型属于典型的集合类型，可以保存key-value形式的数据。而且，当Hash类型中保存较多数据时，它的底层是由哈希表实现的，哈希表的存取复杂度是 $O(1)$ ，所以可以实现快速访问。在这道题中，短视频属性信息属于典型key-value形式，所以，我们可以使用Hash类型保存短视频信息。具体来说，就是将一个短视频ID作为Hash集合的key，将短视频的其他属性信息作为Hash集合内部的键值对，例如“作者”：“实际姓名”，“创建时间”：“实际时间”。这样既满足了保存数据的需求，也可以利用Hash快速查询的特点，快速查询相应的信息。

Redis的AOF日志会记录客户端发送给实例的每一次写操作命令，在Redis实例恢复时，可以通过重新运行AOF文件中的命令，实现恢复数据的目的。在这道题的业务场景中，负载以读为主，因此，写命令不会太多，AOF日志文件的体量不会太大，即使实例故障了，也可以快速完成恢复。所以，当使用单实例运行时，我们可以使用AOF日志来做持久化方案。

关于使用多实例的运行方案：两种方案各有优势，我们来分析一下。

### 方案一

优势：可以节省云主机数量和成本。虽然主从节点进行第一次全量同步时，RDB文件较大，耗时会长些，但是因为写请求少，所以复制缓冲区的压力不大。

不足：如果网络环境不好，需要频繁地进行全量同步的话，这种方案的优势就小了，每次全量同步时的RDB生成和传输压力都很大。

### 方案二

优势：每个实例只用保存4GB数据，和从库同步时的压力较小。而且，这种方案的可扩展性更好，如果有新增数据，可以更好地应对。

不足：需要较多的云主机，运维和资源成本较高。

好了，这节课就到这里。假期很快就要结束了，希望你抓住最后的几天时间，好好地巩固一下所学的内容。我们下节课见。

**精选留言：**

-----

哈哈，国庆终于把进度补上来了

拼课微信：699250