Redis的使用

# Redis是什么？

Redis是分布式的非关系型数据库，是NOSQL的一种。以key-value方式展示。Redis的存储方式是采用内存和磁盘存储， 这点和memcache不同，而memcache是存储在内存，要是一旦宕机，数据就会丢失。而Redis在一般情况下则不会，因为它支持持久化。Redis自动会把内容从内存中同步到磁盘，即使宕机或者服务重启，数据还可以从磁盘中还原过来。Redis是支持主从同步，在大流量，高并发的场景下使用。Redis3.0以上还支持集群。

# Redis具体的使用场景？

Redis的数据因为存储在内存，所以不适合海量数据的存储，是和来处理一些对于并发较高的一些场景的使用，并且Redis在使用时，都要设置一个时效性，如果长期存储，就会浪费不必要的内存浪费。

Redis有5种基本数据类型，字符串(string), 哈希( hash), 队列（list）,集合（set）,有序集合（zset）。下面分别来对这5种字符串具体的使用场景来进行介绍。

## 2.1对于变更少，读取频繁的数据

如果对于一些频繁读取，但变更又不多的热点数据，可以考虑存储在Redis中，比如我们项目中的一些配置，像微信支付配置，accessToken，全局设置等存储。这些热点数据存储在Redis中的好处就是，不用频繁的去请求数据库，可以直接从redis数据库中去读取出来。可以减少磁盘IO的开销。这个可以采取redis的Hash类型来处理。

## 2.2流控以及事务提交的并发锁

我们在做限时购，秒杀的时候，为了有效的防止库存超卖的现象，在提交订单事务的时候，可以采用Redis分布式的并发锁来处理,主要流程是这样子，一波用户进来，就加上并发锁，等用户处理完业务流程的时候，再将并发锁放开。当并发锁在没有放开的时候，别的用户是进不来的。并发锁是怎么来设置呢，就是用户在提交事务时在redis中设置一个值，是string类型的，并且设置时效性，流程处理完或者时间过期，就将这个值删除掉。

## 2.3抢优惠券

我们平常抢优惠券的时候，是从优惠券日志表中去读取一条出来，然后把抢优惠券的使用状态，用户id更新到优惠券使用记录表中去，这个时候，其实会存在一个性能问题，就是操作数据库太频繁。这个时候，可以考虑先从后台把优惠券数据存在队列中，然后再在前台把队列中的优惠券数据去拉取一条出来，这样子用户就可以抢到优惠券了。同时在计划下异步将优惠券数据状态更新掉。这个可以采用list的数据类型来处理

## 2.4计数

计数功能比较有用，其实我们最近做的广告的PV,UV统计就可以采用计数的方式来统计，就是在点击广告，链接跳转的时候记下数，UV也可以采用类似于这样的方式来做。先记录到redis中去，并且定时同步到数据库中去。

## 2.5异步任务操作

对于一些耗时比较长的事务，比如在订单提交的同时，需要发短信，推送微信消息模板等第三方消息模板的时候，可以采用将需要处理的类名，方法名组成一个json串，存入到队列中。后端的计划任务再从计划任务中去获取出来，根据类名，方法名来进行处理。

## 2.6 SESSION的存储方式

如果我们有多台Web服务器的时候，采用常规的session存储的时候，就会出现一开始登陆了，再刷新下页面的时候，登录失效了，这是由于，session是存在服务器下的文件中，如果在多台服务器下面，就会出现session信息不同步。这个时候可以采用一台Redis服务器，把session信息存在redis下。这样子多台web服务器在使用session时可以规避这个问题。

# Redis的缺点

是数据库容量受到物理内存的限制,不能用作海量数据的高性能读写,因此Redis适合的场景主要局限在较小数据量的高性能操作和运算上。

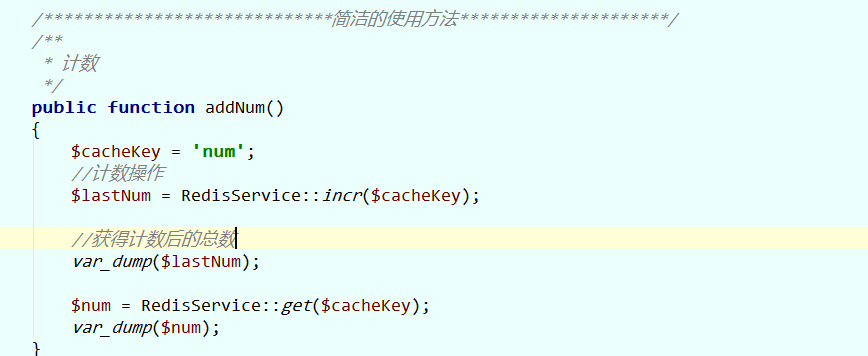
# Redis结合PHP使用的例子

我在wap\_greencampus项目下，有专门的Redis的一个demo，是放在Base/TestAction类。

我从五个类型分别来说明：

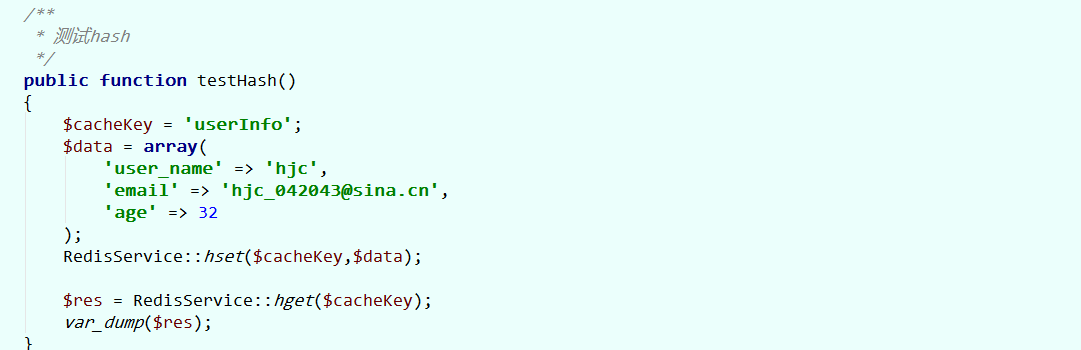
1. 字符串类型

这个是用来计数使用，在统计PV,UV的时候可以用这个方式来采用



1. Hash类型

在对于一些配置可以采用hash来写



1. 集合类型



1. 队列类型

这是队列的设置，像优惠券等可以设置到队列中去

