

我们可以想象一下对连接池的基本动作，无非就是申请连接，从连接池中获取连接，和业务处理完后，把连接释放回连接池这些动作。

在通常情况下一个连接池在启动时会初始化MIN连接数，这时候通往数据库的一部分管道已经建立起来了，你可以通过这些管道，对数据库进行查询和增删改查，如果一个请求申请管道的时候发现有空闲的管道，那么直接可以拿来用了，如果所有的管道都在忙，但管道的数量没有达到MAX连接数，那么不需要等待，直接申请创建一个新的连接，用完了再把他放回去，当发现没有空闲的管道，并且活跃的管道已经到达MAX连接数了，那么这时候你只能选择暂时等待，等待的时间取决于block-timeout，在这等待期间如果有管道空闲下来，那么恭喜你，你有机会拿到这个连接，如果超出等待时间还没有拿到连接，那么就抛出个拿不到连接的异常，连接池基本的逻辑就是这样了，另外的功能无非就是对连接池使用状态的监控，比如一个连接如果空闲下来了，多久没有使用需要被关闭，比如哪些错误情况下需要重新创建一下连接再放入池子，比如如何定时来验证连接是否有效，等等。

刚才提到了连接池的MIN和MAX连接，需要大家的关注，因为连接池是无法感知数据库的运行情况以及负载的，通过经验值或者计算模型，合理的加以设置，对于应用服务器和数据库来说，都非常的重要，即要能发挥出应用服务器的最大能力，也要能有效利用数据库的连接资源和处理能力，换句话说不想在有能力处理时让请求在队列中等待，也不想让运行的请求超出DB的处理能力。我们具体来看一下，如果连接池MIN设置过小的话，在应用业务量突增或者启动时，就可能短时间内产生连接风暴，这对于数据库是不小的冲击，但是如果MIN值设置过大，就会出现数据库连接过剩的情况，连接一方面超出空闲时间被销毁，而销毁后发现又小于MIN连接数，又开始创建，结果就发生循环，浪费资源浪费电。那如果连接池MAX值设置过大，在极端情况下，当应用发生异常时，会导致连接数被撑到MAX值，有可能导致数据库的连接数被耗尽，或者超出数据库的处理能力，进而导致业务受到影响。并且当连接数被撑到MAX值，在获取连接等待超时的时候，应用的线程池也有可能受到影响，会形成一系列的连锁反应，乃至雪崩。

所以平时有开发同学抱怨连接池的配置不够，让我们加大MAX值，我都会解释下，能不能加连接要看DB是否还有余量，如果DB还有余量，加连接也许是一种临时的解决办法，如果DB已经容量不足，加大MAX会放进更多的请求到DB，只会让性能变得更差。