**一、连接数据库**

1、同一机房的应用服务器连接数据库服务器，必须使用内网ip地址（192.168.7.xxx）；

2、不同机房的应用服务器连接数据库服务器，可以使用外网ip地址连接；

3、使用主从数据库的应用服务器连接数据库服务器，必须使用DBA给出的vip连接；

3、应用服务器连接带有镜像灾备模式的数据库服务器，必须使用DBA给出的镜像模式连接串；

例如：Data Source=192.168.7.200;Failover Partner=192.168.7.219;database=.......

**二、数据帐户使用**

1、每个项目使用的数据库都分三级帐户管理，程序员必须更具应用的不同功能，区分开程序中的读/写账户；

（1）管理员帐户，具有对所管理的数据库dbo权限，并具备创建和执行数据库作业权限；

A、供程序员维护数据和数据结构使用；

B、该账户不能在程序中使用；

（2）可写账户，具有对所管理的数据库update / insert / delete / select权限；

（3）只读帐户，具有对所管理的数据库select权限；

**三、编写sql语句检查规则**

（一）、sql server数据库

1、检查sql语句执行结果所消耗的服务器io及cpu情况；

设置监控参数为打开状态：

set statistics io on

set statistics time on

然后运行sql语句，并检查监控的参数；

2、使用“显示执行计划”检查sql语句对索引的使用；

（二）、mysql数据库

使用explain + sql语句，检查sql语句的执行效率；

**四、程序员对大数据量操作规则：**

1、对数据库中大量数据的操作包括数据导入/导出，数据更新/删除操作，以及更改数据库表结构等有可能影响服务器性能的操作；

2、操作之前，需要和dba一起制定执行和回退方案，并确定执行时间；

3、方案制定完成，需要报相关部门主管审批后方可进行；

5、部门主管需要审批sql语句的逻辑结构是否正确，业务逻辑结构是否正确；并做风险评估；最终返回给相关同事审批结果；

4、操作之前，通知dba监控服务器负载情况和网络情况，通知测试组监控应用程序运行情况；

**五、关于测试数据库的使用：**

1、所有线上数据库都不能提供测试使用；

2、所有线上数据库服务器都不能提供测试数据库；

3、程序员在新产品/新功能开发时，需要提前向dba申请测试数据库；

4、程序员必须在测试数据库上进行相关测试工作，包括应用方面的功能测试、压力测试以及数据结构测试等方面；

**六、程序员新产品使用数据库申请及上线流程**

1、规划数据库的使用方式及数据空间/访问量预测；

2、填写《数据库申请表》，并提交给dba；

3、dba根据程序员对数据库方面的要求和预测，分配数据库资源，以及创建相关备份和灾备方案；

4、项目正式上线前，使用测试服务器进行相关测试和开发工作；

5、开发完成后，程序员在产品正式上线前，通知dba将测试数据库迁移到正式数据库服务器使用；

**七、空查询和无用索引的处理**

空查询和无用索引严重时会影响数据库服务器的响应性能；要求对数据库的访问中，尽量少的有空查询和无用索引；

1、空查询清理阀值，

1. 磁盘io使用率：平均每分钟不超过10万次io访问；
2. Cpu使用率：平均不超过总cpu的5%，即空查询sql总合不超过2000秒；

2、无用索引清理阀值：

未使用索引定义：少于等于1次/天的索引；

**八、提高sql语句效率的一些技巧：**

1. 将SQL语句写在存储过程中，优点就不说了

2. 尽可能少的在存储过程中拼接SQL语句再执行，宁可多写IF判断或拆分存储过程

3. 查询时不要返回不需要的行、列

4. IN适合于外表大而内表小的情况；EXISTS适合于外表小而内表大的情况

5. 尽可能少的在列上使用表达式，会被放弃使用索引

6. 注意in,or,union 在适当的条件下的互换(以前的in or 互换在sql2008下查询优化器已经能够自己识别了，尝试了很多种in or 互换

执行效率都差不多)

7. 如果临时表的数据量较大，需要建立索引，那么应该将创建临时表和建立索引的过程放在单独一个子存储过程中，这样才能保证系统

能够很好的使用到该临时表的索引。

8. 如果使用到了临时表，在存储过程的最后务必将所有的临时表显式删除，先truncate table，然后drop table，这样可以避免系统表

的较长时间锁定。

9. 使用CTE：可以简化嵌套SQL，免除创建临时表的麻烦，配合ROW\_NUMBER()分页可以在数据量较大的时候提高分页速度(非常明显).还

可以让很复杂的语句看起来清晰明了，符合思维习惯，而且支持递归.总之我能想到的都是CTE的优点，除了在数据量小的时候使用CTE分页速

度赶不上TOP MAX外(当然在数据量少的时候慢那几毫秒是可以接受的，万一那天表数据量多了呢)

10. 如果有复杂的字符串处理(如正则匹配)可以考虑使用CLR函数

11. 对较为复杂的sql语句加上注释，说明算法、功能。

12. 尽量少用视图，它的效率低。对视图操作比直接对表操作慢

13. where条件中尽量减少使用常量比较，建议使用变量

14. 大量的排序操作会影响系统性能，所以尽量减少order by和group by排序操作。

对于7,8条还是建议少用临时表，尽量用结果集和表变量来代替它。数据量大的临时表，创建索引也是占用资源和时间的。

对于第3条：在查询Select语句中用Where字句限制返回的行数,避免表扫描，如果返回不必要的数据，浪费了服务器的I/O资源