

第五部分 数据库部分

1. 维护数据库的完整性、一致性、你喜欢用触发器还是自写业务逻辑？为什么？

答：尽可能用约束（包括 CHECK、主键、唯一键、外键、非空字段）实现，这种方式的效率最好；其次用触发器，这种方式可以保证无论何种业务系统访问数据库都能维持数据库的完整性、一致性；最后再考虑用自写业务逻辑实现，但这种方式效率最低、编程最复杂，当为下下之策。

2. 什么是事务？什么是锁？

答：事务是指一个工作单元，它包含了一组数据操作命令，并且所有的命令作为一个整体一起向系统提交或撤消请求操作，即这组命令要么都执行，要么都不执行。

锁是在多用户环境中对数据的访问的限制。SqlServer 自动锁定特定记录、字段或文件，防止用户访问，以维护数据安全或防止并发数据操作问题，锁可以保证事务的完整性和并发性。

3. 什么是索引，有什么优点？

答：索引象书的目录类似，索引使数据库程序无需扫描整个表，就可以在其中找到所需的数据，索引包含了一个表中包含值的列表，其中包含了各个值的行所存储的位置，索引可以是单个或一组列，索引提供的表中数据的逻辑位置，合理划分索引能够大大提高数据库性能。

4. 视图是什么？游标是什么？

答：视图是一种虚拟表，虚拟表具有和物理表相同的功能，可以对虚拟表进行增该查操作，视图通常是一个或多个表的行或列的子集，视图的结果更容易理解（修改视图对基表不影响），获取数据更容易（相比多表查询更方便），限制数据检索（比如需要隐藏某些行或列），维护更方便

游标对查询出来的结果集作为一个单元来有效的处理，游标可以定位在结果集的特定行、从结果集的当前位置检索一行或多行、可以对结果集中当前位置进行修改、

5. 什么是存储过程？有什么优点？

答：存储过程是一组予编译的 SQL 语句，它的优点有 1. 允许模块化程序设计，就是说只需要创建一次过程，以后在程序中就可以调用该过程任意次。2. 允许更快执行，如果某操作需要执行大量 SQL 语句或重复执行，存储过程比 SQL 语句执行的要快。3. 减少网络流量，例如一个需要数百行的 SQL 代码的操作有一条执行语句完成，不需要在网络中发送数百行代码。4. 更好的安全机制，对于没有权限执行存储过程的用户，也可授权他们执行存储过程。

6. 触发器的作用？

答：触发器是一中特殊的存储过程，主要是通过事件来触发而被执行的。它可以强化约束，来维护数据的完整性和一致性，可以跟踪数据库内的操作从而不允许未经许可的更新和变化。可以联级运算。如，某表上的触发器上包含对另一个表的数据操作，而该操作又会导致该表触发器被触发。

7. 什么是内存泄漏？

答：一般我们所说的内存泄漏指的是堆内存的泄漏。堆内存是程序从堆中为其分配的，大小任意的，使用完后要显示释放内存。当应用程序用关键字 new 等创建对象时，就从堆中为它分配一块内存，使用完后程序调用 free 或者 delete 释放该内存，否则就说该内存就不能被使用，我们就说该内存被泄漏了。

8. 为管理岗位业务培训信息, 建立 3 个表:

S (S#, SN, SD, SA) S#, SN, SD, SA 分别代表学号、学员姓名、所属单位、学员年龄

C (C#, CN) C#, CN 分别代表课程编号、课程名称

SC (S#, C#, G) S#, C#, G 分别代表学号、所选修的课程编号、学习成绩

1. 使用标准 SQL 嵌套语句查询选修课程名称为' 税收基础' 的学员学号和姓名

--实现代码:

```
Select SN, SD FROM S
Where [S#] IN(
Select [S#] FROM C, SC
Where C. [C#]=SC. [C#]
AND CN=N' 税收基础' )
```

2. 使用标准 SQL 嵌套语句查询选修课程编号为' C2' 的学员姓名和所属单位

--实现代码:

```
Select S. SN, S. SD FROM S, SC
Where S. [S#]=SC. [S#]
AND SC. [C#]=' C2'
```

3. 使用标准 SQL 嵌套语句查询不选修课程编号为' C5' 的学员姓名和所属单位

--实现代码:

```
Select SN, SD FROM S
Where [S#] NOT IN(
Select [S#] FROM SC
Where [C#]=' C5' )
```

4. 使用标准 SQL 嵌套语句查询选修全部课程的学员姓名和所属单位

--实现代码

```
Select SN, SD FROM S
Where [S#] IN(
Select [S#] FROM SC
RIGHT JOIN
C ON SC. [C#]=C. [C#] GROUP BY [S#]
HAVING COUNT(*)=COUNT([S#]))
```

5. 查询选修了课程的学员人数

--实现代码:

```
Select 学员人数=COUNT(DISTINCT [S#]) FROM SC
```

6. 查询选修课程超过 5 门的学员学号和所属单位

--实现代码:

```
Select SN, SD FROM
Where [S#] IN(
Select [S#] FROM SC
```

```
GROUP BY [S#]
HAVING COUNT(DISTINCT [C#])>5
```

9. 问题描述: 已知关系模式:

S (SNO, SNAME) 学生关系。SNO 为学号, SNAME 为姓名

C (CNO, CNAME, CTEACHER) 课程关系。CNO 为课程号, CNAME 为课程名, CTEACHER 为任课教师

SC(SNO, CNO, SCGRADE) 选课关系。SCGRADE 为成绩

1. 找出没有选修过“李明”老师讲授课程的所有学生姓名

--实现代码:

```
Select SNAME FROM S
Where NOT EXISTS(
Select * FROM SC,C
Where SC.CNO=C.CNO
AND CNAME='李明'
AND SC.SNO=S.SNO)
```

2. 列出有二门以上(含两门)不及格课程的学生姓名及其平均成绩

--实现代码:

```
Select S.SNO, S.SNAME, AVG_SCGRADE=AVG(SC.SCGRADE)
FROM S,SC,(
Select SNO
FROM SC
Where SCGRADE<60
GROUP BY SNO
Having COUNT(DISTINCT CNO)>=2
)A Where S.SNO=A.SNO AND SC.SNO=A.SNO
GROUP BY S.SNO,S.SNAME
```

3. 列出既学过“1”号课程, 又学过“2”号课程的所有学生姓名

--实现代码:

```
Select S.SNO, S.SNAME
FROM S,(
Select SC.SNO
FROM SC,C
Where SC.CNO=C.CNO
AND C.CNAME IN('1','2')
GROUP BY SNO
Having COUNT(DISTINCT CNO)=2
)SC Where S.SNO=SC.SNO
```

4. 列出“1”号课成绩比“2”号同学该门课成绩高的所有学生的学号

--实现代码:

```
Select S.SNO, S.SNAME
```

```
FROM S, (
  Select SC1.SNO
  FROM SC SC1, C C1, SC SC2, C C2
  Where SC1.CNO=C1.CNO AND C1.NAME='1'
  AND SC2.CNO=C2.CNO AND C2.NAME='2'
  AND SC1.SCGRADE>SC2.SCGRADE
)SC Where S.SNO=SC.SNO
```

5. 列出“1”号课成绩比“2”号课成绩高的所有学生的学号及其“1”号课和“2”号课的成绩

--实现代码:

```
Select S.SNO, S.SNAME, SC.[1号课成绩], SC.[2号课成绩]
FROM S, (
  Select SC1.SNO, [1号课成绩]=SC1.SCGRADE, [2号课成绩]=SC2.SCGRADE
  FROM SC SC1, C C1, SC SC2, C C2
  Where SC1.CNO=C1.CNO AND C1.NAME='1'
  AND SC2.CNO=C2.CNO AND C2.NAME='2'
  AND SC1.SCGRADE>SC2.SCGRADE
)SC Where S.SNO=SC.SNO
```

10. 您能向我简要叙述一下 SQL Server 2000 中使用的一些数据库对象吗？

您希望听到的答案包括这样一些对象：表格、视图、用户定义的函数，以及存储过程；如果他们还能够提到像触发器这样的对象就更好了。如果应聘者不能回答这个基本的问题，那么这不是一个好兆头。

11. 什么是索引？SQL Server 2000 里有什么类型的索引？

简单地说，索引是一个数据结构，用来快速访问数据库表格或者视图里的数据。在 SQL Server 里，它们有两种形式：聚集索引和非聚集索引。聚集索引在索引的叶级保存数据。这意味着不论聚集索引里有表格的哪个（或哪些）字段，这些字段都会按顺序被保存在表格。由于存在这种排序，所以每个表格只会有一个聚集索引。非聚集索引在索引的叶级有一个行标识符。这个行标识符是一个指向磁盘上数据的指针。它允许每个表格有多个非聚集索引。

12. NULL 是什么意思？

NULL 这个值表示 UNKNOWN（未知）：它不表示“”（空字符串）。假设您的 SQL Server 数据库里有 ANSI_NULLS，当然在默认情况下会有，对 NULL 这个值的任何比较都会生产一个 NULL 值。您不能把任何值与一个 UNKNOWN 值进行比较，并在逻辑上希望获得一个答案。您必须使用 IS NULL 操作符。

13. 什么是主键？什么是外键？

主键是表格里的（一个或多个）字段，只用来定义表格里的行；主键里的值总是唯一的。外键是一个用来建立两个表格之间关系的约束。这种关系一般都涉及一个表格里的主键字段与另外一个表格（尽管可能是同一个表格）里的一系列相连的字段。那么这些相连的字段就是外键。

14. 什么是触发器？SQL Server 2000 有什么不同类型的触发器？

触发器是一种专用类型的存储过程，它被捆绑到 SQL Server 2000 的表格或者视图上。在 SQL Server 2000 里，有 INSTEAD-OF 和 AFTER 两种触发器。INSTEAD-OF 触发器是替代数据操控语言（Data Manipulation Language, DML）语句对表格执行语句的存储过程。例如，如果我有一个用于 TableA 的 INSTEAD-OF-UPDATE 触发器，同时对这个表格执行一个更新语句，那么 INSTEAD-OF-UPDATE 触发器里的代码会执行，而不是我执行的更新语句则不会执行操作。

AFTER 触发器要在 DML 语句在数据库里使用之后才执行。这些类型的触发器对于监视发生在数据库表格里的数据变化十分好用。

15. 您如何确保一个带有名为 Field1 字段的 TableB 表格里只具有 Field1 字段里的那些值, 而这些值同时在名为 TableA 的表格的 Field1 字段里?

这个与关系相关的问题有两个可能的答案。第一个答案 (而且是您希望听到的答案) 是使用外键限制。外键限制用来维护引用的完整性。它被用来确保表格里的字段只保存有已经在不同的 (或者相同的) 表格里的另一个字段里定义了的值。这个字段就是候选键 (通常是另外一个表格的主键)。

另外一种答案是触发器。触发器可以被用来保证以另外一种方式实现与限制相同的作用, 但是它非常难设置与维护, 而且性能一般都很糟糕。由于这个原因, 微软建议开发人员使用外键限制而不是触发器来维护引用的完整性。

16. 对一个投入使用的在线事务处理表格有过多索引需要有什么样的性能考虑?

您正在寻找进行与数据操控有关的应聘人员。对一个表格的索引越多, 数据库引擎用来更新、插入或者删除数据所需要的时间就越多, 因为在数据操控发生的时候索引也必须要维护。

您可以用什么来确保表格里的字段只接受特定范围里的值?

这个问题可以用多种方式来回答, 但是只有一个答案是“好”答案。您希望听到的回答是 Check 限制, 它在数据库表格里被定义, 用来限制输入该列的值。

触发器也可以被用来限制数据库表格里的字段能够接受的值, 但是这种办法要求触发器在表格里被定义, 这可能会在某些情况下影响到性能。因此, 微软建议使用 Check 限制而不是其他的方式来限制域的完整性。

17. 返回参数和 OUTPUT 参数之间的区别是什么?

返回参数总是由存储过程返回, 它用来表示存储过程是成功还是失败。返回参数总是 INT 数据类型。

OUTPUT 参数明确要求由开发人员来指定, 它可以返回其他类型的数据, 例如字符型和数值型的值。(可以用作输出参数的数据类型是有一些限制的。) 您可以在一个存储过程里使用多个 OUTPUT 参数, 而您只能使用一个返回参数。

18. 什么是相关子查询? 如何使用这些查询?

相关子查询是一种包含子查询的特殊类型的查询。查询里包含的子查询会真正请求外部查询的值, 从而形成一个类似于循环的状况。您可以在这篇文章里看到关于这些特殊类型的查询是如何工作的更加具体的描述。