

第三章 数据库 SQL 操作语言

复杂工程问题——数据库 SQL 操作实践

一、数据库 SQL 操作工程问题探讨

1、了解 PostgreSQL 数据库的体系结构？创建数据库时要考虑哪些参数？

(1) PostgreSQL 数据库是由：连接管理系统（系统控制器）、编译执行系统、存储管理系统、事务系统、系统表五大部分组成。

- **连接管理系统：**接收外部操作对系统的请求，对操作请求进行预处理和分发，起系统逻辑控制作用。
- **编译执行系统：**由查询编译器、查询执行器组成，完成操作请求在数据库中的分析处理和转化工作，最终实现物理存储介质中数据的操作。
- **存储管理系统：**由索引管理器、内存管理器、外存管理器组成，负责存储和管理物理数据，提供对编译查询系统的支持；
- **事务系统：**是由事务管理器、日志管理器、并发控制、锁管理器组成，日志管理器和事务管理器完成对操作请求处理的事务一致性支持，锁管理器和并发控制提供对并发访问数据的一致性支持；
- **系统表：**是 PostgreSQL 数据库的元信息管理中心，包括数据库对象信息和数据库管理控制信息。系统表管理元数据信息，将 PostgreSQL 数据库的各个模块有机地连接在一起，形成一个高效的数据管理系统。

(2) 创建数据库时的可选参数：

```
CREATE DATABASE name
    [ [ WITH ] [ OWNER [=] user_name ]           指定数据库用户
      [ TEMPLATE [=] template ]                 指定数据库模板
      [ ENCODING [=] encoding ]                 指定数据库使用的字符集编码
      [ LC_COLLATE [=] lc_collate ]             指定数据库的字符排序规则
      [ LC_CTYPE [=] lc_ctype ]                 指定数据库的字符分类规则
      [ TABLESPACE [=] tablespace_name ]       指定数据库使用的表空间
      [ CONNECTION LIMIT [=] conlimit ] ]       指定数据库的并发连接数
```

2、建立一个应用系统的数据库的基本步骤？需要哪些 SQL 语句？

(1) **数据库建立基本步骤：**规划、需求分析、概念模型设计、逻辑设计、物理设计、程序编制及调试、运行及维护。

数据库操作基本步骤：数据库创建、修改、删除；数据库表创建、修改、删除；索引对象创建、修改、删除；数据插入、更新、删除；及各种数据查询。

(2) **数据库创建、修改、删除：**

```
CREATE DATABASE <数据库名>;
ALTER DATABASE <数据库名>;
DROP DATABASE <数据库名>;
```

数据库表创建、修改、删除：

```
CREATE TABLE <表名>;
ALTER TABLE <表名>;
DROP TABLE <表名>;
```

索引对象创建、修改、删除：

```
CREATE INDEX <索引名> ON <表名>< (列名[,...]) >;  
ALTER INDEX <索引名> RENAME TO <索引名>;  
DROP INDEX;
```

数据插入、更新、删除：

```
INSERT INTO <基本表>[<列名表>] VALUES(列值表);  
  
UPDATE <基本表>  
SET <列名1>=<表达式1> [,<列名2>=<表达式2>...]  
[WHERE <条件表达式>];  
  
DELETE FROM <表名>  
[WHERE <条件表达式>];
```

各种数据查询：

```
SELECT [ALL|DISTINCT] <目标列>[,<目标列>...]  
[ INTO <新表> ]  
FROM <表名>[,<表名>...]  
[ WHERE <条件表达式> ]  
[ GROUP BY <列名> [HAVING <条件表达式> ]  
[ ORDER BY <列名> [ ASC | DESC ] ];
```

3、提高数据库 SQL 查询操作性能有哪些技术方法？

- (1) 减少在查询条件中使用函数，避免全表扫描；
- (2) 减少不必要的表连接；
- (3) 使用临时表暂存中间结果；
- (4) 使用视图查询，给视图建立索引进行优化；
- (5) 避免多层嵌套的 SQL 语句，分解查询任务。

4、如何剖析 SQL 查询性能？

一段 SQL 代码写好以后，可以通过查看 SQL 的执行计划，初步预测该 SQL 在运行时的性能好坏，尤其是在发现某个 SQL 语句的效率较差时，我们可以通过查看执行计划，分析出该 SQL 代码的问题所在。

- (1) 打开执行计划查看工具：PL/SQL Developer；
- (2) 查看总 COST，获得资源耗费的总体印象；
- (3) 按照从左至右，从上至下的方法，了解执行计划的执行步骤；
- (4) 分析表的访问方式；
- (5) 分析表的连接方式和连接顺序。

5、如何优化 SQL 查询性能？

- (1) 减少在查询条件中使用函数，避免全表扫描；
- (2) 减少不必要的表连接；
- (3) 使用临时表暂存中间结果；
- (4) 使用视图查询，给视图建立索引进行优化；
- (5) 避免多层嵌套的 SQL 语句，分解查询任务。

6、SQL 聚类索引、非聚类索引如何应用？

动作描述	使用聚集索引	使用非聚集索引
列经常被分组排序	✓	✓
返回某范围内的数据	✓	✗
一个或极少不同值	✗	✗
小数目的不同值	✓	✗
大数目的不同值	✗	✓
频繁更新的列	✗	✓
外键列	✓	✓
主键列	✓	✓
频繁修改索引列	✗	✓

7、如何合理定义 SQL 索引？

- (1) 在经常进行连接，但是没有指定为外键的列上建立索引，而不经常连接的字段则由优化器自动生成索引。
- (2) 在频繁进行排序或分组（即进行 GROUP BY 或 ORDER BY 操作）的列上建立索引。
- (3) 在条件表达式中经常用到的不同值较多的列上建立检索，在不同值少的列上不要建立索引。比如在雇员表的“性别”列上只有“男”与“女”两个不同值，因此就无必要建立索引。如果建立索引不但不会提高查询效率，反而会严重降低更新速度。
- (4) 如果待排序的列有多个，可以在这些列上建立复合索引（Compound Index）。
- (5) 使用系统工具。如 Informix 数据库有一个 tbcheck 工具，可以在可疑的索引上进行检查。在一些数据库服务器上，索引可能失效或者因为频繁操作而使得读取效率降低，如果一个使用索引的查询不明不白地慢下来，可以试着用 tbcheck 工具检查索引的完整性，必要时进行修复。另外，当数据库表更新大量数据后，删除并重建索引可以提高查询速度。

8、如何合理使用 SQL 内外连接查询数据？

- (1) 内连接：两张表进行匹配的时候，如果表中任意一条数据在另一张表中都找不到对应的数据，那么结果表不会有这一条数据。也就是说必须是两张表中的任意两条数据相互匹配才能被存入到结果表中，类似于“取交集”。适用于一旦某条数据为空便没有意义的场景，这时给它设置成 null 也就毫无意义了。表中的数据也因此显得简练很多。
- (2) 外连接：与内连接相反，如果某张表中的数据在另一张表中找不到对应的条目并不影响它依然出现在查询的结果中，这对于两张表都是满足的，两边都有出现 null 的可能，类似于“取并集”。

9、如何切分 SQL 查询？

- (1) 使用 SELECT-LIKE 语句与通配符：

```
SELECT * FROM <表名>
WHERE <列名> LIKE '[通配符表达式]'
```

(2) 通配符说明:

通配符	描述
%	替代一个或多个字符
_	仅替代一个字符
[charlist]	字符列中的任何单一字符
[^charlist]或者 [!charlist]	不在字符列中的任何单一字符

10、如何分解关联查询?

例如:

```
SELECT *  
FROM tag JOIN tag_post ON tag_post.tag_id = tag.id  
JOIN post ON tag_post_id = post.id  
WHERE tag.tag = 'mysql';
```

可分解为以下三步查询:

```
SELECT * FROM tag WHERE tag = 'mysql';  
SELECT * FROM tag_post WHERE tag_id = 1234;  
SELECT * FROM post WHERE post.id in (124,324,553,2324);
```

二、数据库工程实践案例问题探讨

针对一个工程项目管理系统数据库操作访问, 探讨该数据库 SQL 操作访问实践。

1、在工程项目管理系统开发中, 如何执行 SQL 语句创建 ProjectDB 数据库?

```
CREATE DATABASE "ProjectDB";
```

2、在工程项目管理系统开发中, 如何在 SQL 语句定义数据库表之间参照约束?

```
CONSTRAINT CoursePlanID_FK FOREIGN KEY(CoursePlanID)  
REFERENCES Plan(CoursePlanID)  
ON UPDATE CASCADE  
ON DELETE CASCADE,
```

3、在工程项目管理系统开发中, 如何解决数据库表之间的级联数据操作?

级联修改: ON UPDATA CASCADE

级联删除: ON DELETE CASCADE

4、在工程项目管理系统开发中, 可在哪些表中创建索引支持查询性能?

表内有明显顺序特征的列时, 我们可以对该表创建索引, 来支持查询性能, 加快表中的数据查询。

5、在工程项目管理系统开发中, 如何操作这些数据?

(1) 插入、修改、删除:

```
INSERT INTO <基本表>[<列名表>] VALUES(列值表);
```

```
UPDATE <基本表>
SET <列名1>=<表达式1> [,<列名2>=<表达式2>...]
[WHERE <条件表达式>];

DELETE FROM <表名>
[WHERE <条件表达式>];
```

(2) 各种数据查询:

```
SELECT [ALL|DISTINCT] <目标列>[,<目标列>...]
[ INTO <新表> ]
FROM <表名>[,<表名>...]
[ WHERE <条件表达式> ]
[ GROUP BY <列名> [HAVING <条件表达式> ]
[ ORDER BY <列名> [ ASC | DESC ] ];
```

6、在工程项目管理系统开发中, 如何使用 JOIN ..ON 关联多表实现数据查询?

```
SELECT <目标列>[,<目标列>...]
FROM <表名 1> JOIN <表名 2> ON <连接条件>;
```

7、在工程项目管理系统开发中, 如何使用 SQL 视图查询项目成本数据?

```
CREATE VIEW <视图名> AS
SELECT ...
FROM ...
WHERE ...
```

之后可直接将视图看作一个新表对其数据进行操作。

8、在工程项目管理系统开发中, 如何对系统中的表格进行权限设置?

(1) 授权语句:

```
GRANT <权限列表> ON <数据库对象>
TO <用户或角色> [ WITH GRANT OPTION ];
```

(2) 权限回收语句:

```
REVOKE <权限列表> ON <数据库对象> FROM <用户或角色>
```

(3) 拒绝授权语句:

```
DENY <权限列表> ON <数据库对象> TO <用户或角色>
```