

北京邮电大学

数据库系统原理



第二次实验

实验五 创建和管理索引和视图

姓 名 郑毓恒
学 院 计算机学院
专 业 计算机科学与技术
班 级 2020211302
学 号 2020211262
任课教师 邓芳

2022 年 11 月

实验目的

1. 通过实验让学生熟悉并了解 GaussDB(for openGauss)数据库的基本机制与操作。
2. 通过索引管理、视图管理等管理的操作，让学生熟悉并了解 DAS 环境下如何使用 GaussDB(for openGauss)。

实验平台及环境

1. 本实验环境为华为云 GaussDB(for openGauss)数据库；
2. 为了满足本实验需要，实验环境采用以下配置：
 - 1) 设备名称：数据库
 - 2) 设备型号：GaussDB(for openGauss) 8 核 | 64 GB
 - 3) 软件版本：GaussDB(for openGauss) 2020 主备版

实验内容

1. 本实验通过索引管理、视图管理等管理的操作，让学生熟悉并了解 DAS 环境下如何使用 GaussDB(for openGauss)；
2. 本实验通过视图管理等管理的操作，让学生熟悉并了解 DAS 环境下如何使用 GaussDB(for openGauss)。

实验步骤

1 创建和管理索引

(1) 基本概念：

索引可以提高数据的访问速度，但同时也增加了插入、更新和删除表的处理时间。所以是否要为表增加索引，索引建立在哪些字段上，是创建索引前必须要考虑的问题。需要分析应用程序的业务处理、数据使用、经常被用作查询条件或者被要求排序的字段来确定是否建立索引。

openGauss 支持 4 种创建索引的方式：唯一索引、多字段索引、部分索引、表达式索引。

(2) 创建索引

(3) 管理索引

a. 查询索引

b. 删除索引

(4) 索引练习：

对美国各州县确诊与死亡数统计表创建唯一索引、多字段索引、部分索引和表达式索引四类索引，尝试比较未建索引与创建索引后，查询效率的不同。

2 创建和管理视图

(1) 基本概念：

当用户对数据库中的一张或者多张表的某些字段的组合感兴趣，而又不想每次键入这些查询时，用户就可以定义一个视图，以便解决这个问题。

视图与基本表不同，不是物理上实际存在的，是一个虚表。数据库中仅存放视图的定义，而不存放视图对应的数据，这些数据仍存放在原来的基本表中。若基本表中的数据发生变化，从视图中查询出的数据也随之改变。从这个意义上讲，视图就像一个窗口，透过它可以看到数据库中用户感兴趣的数据及变化。视图每次被引用的时候都会运行一次。

(2) 创建视图

(3) 管理视图

a. 查询普通视图

b. 查看普通视图的具体信息

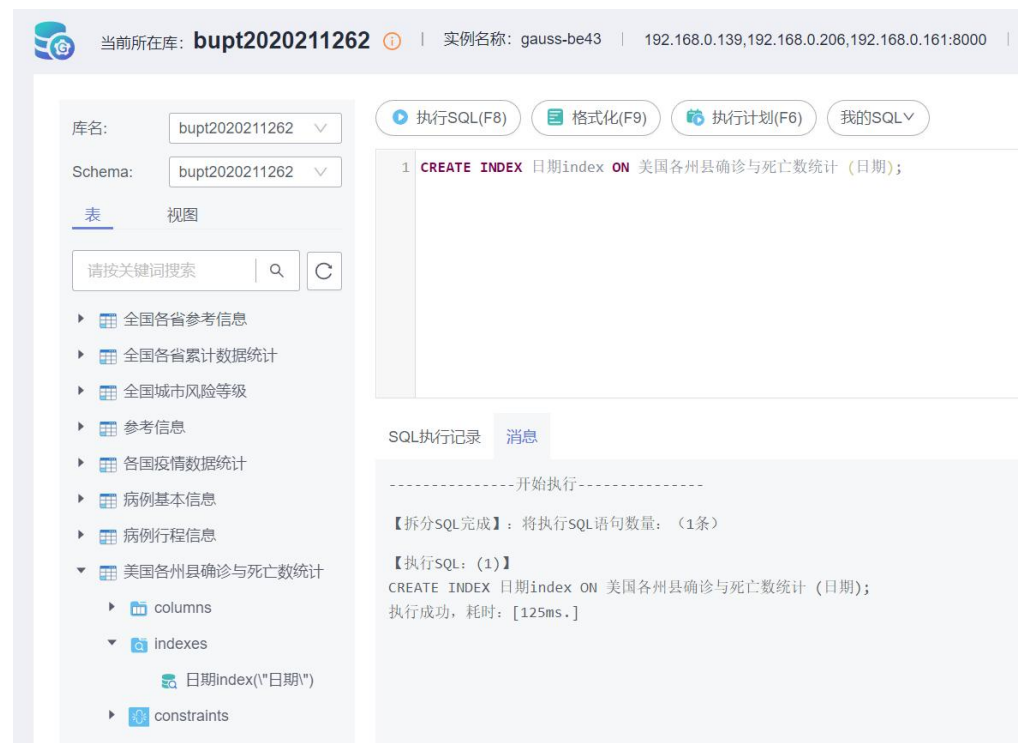
c. 查看视图详情

(4) 创建北京市病例信息的视图，包括行程号，病例号，性别，日期信息（选用病例行程信息表日期）和行程信息。

(5) 通过上述视图查询临床分型为普通型的病例号、行程号、性别和日期信息，按照病例号进行升序显示（截前五条记录）。

实验结果及分析

1. 创建和管理索引



在“美国各州县确诊与死亡统计”，输入以上语句，创建索引。在左侧，表视图的 **indexes** 中，显示刚刚创建的索引“日期 index”。



输入以上语句，删除刚刚创建的索引。在左侧的 **indexes** 一栏中，可见该索引已被删除。

2. 创建索引练习

创建唯一索引

执行SQL(F8) 格式化(F9) 执行计划(F6) 我的SQL▼

1 **SELECT** 日期 **FROM** 美国各州县确诊与死亡数统计 **WHERE** 日期='2020-12-24'

SQL执行记录 消息 结果集1 X

-----开始执行-----

【拆分SQL完成】：将执行SQL语句数量：（1条）

【执行SQL：（1）】

SELECT 日期 FROM 美国各州县确诊与死亡数统计 WHERE 日期='2020-12-24'
执行成功，当前返回：[50]行，耗时：[34ms.]

未创建索引时，查询耗时 34ms。

执行SQL(F8) 格式化(F9) 执行计划(F6) 我的SQL▼

1 **CREATE INDEX** 日期index1 **ON** 美国各州县确诊与死亡数统计 (日期);
2 **SELECT** 日期 **FROM** 美国各州县确诊与死亡数统计 **WHERE** 日期='2020-12-24';

SQL执行记录 消息 结果集1 X

-----开始执行-----

【拆分SQL完成】：将执行SQL语句数量：（2条）

【执行SQL：（1）】

CREATE INDEX 日期index1 ON 美国各州县确诊与死亡数统计 (日期);
执行失败，失败原因：ERROR: relation "日期index1" already exists

【执行SQL：（2）】

SELECT 日期 FROM 美国各州县确诊与死亡数统计 WHERE 日期='2020-12-24';
执行成功，当前返回：[50]行，耗时：[7ms.]

创建索引后，查询耗时 7ms，比未创建时耗时更短。

创建多字段索引

 执行SQL(F8)

 格式化(F9)

 执行计划(F6)

我的SQL▼

1

`SELECT * FROM 美国各州县确诊与死亡数统计 WHERE 日期= '2020-12-24' AND 累计确诊 > 1000;`

SQL执行记录 消息 结果集1 X

-----开始执行-----

【拆分SQL完成】：将执行SQL语句数量：（1条）

【执行SQL：(1)】

SELECT * FROM 美国各州县确诊与死亡数统计 WHERE 日期= '2020-12-24' AND 累计确诊 > 1000;

执行成功，当前返回：[50]行，耗时：[38ms.]

未创建索引时，查询耗时 38ms。

 执行SQL(F8)

 格式化(F9)

 执行计划(F6)

我的SQL▼

1

`CREATE INDEX 累计index ON 美国各州县确诊与死亡数统计 (日期 ,累计确诊);`

2

`SELECT * FROM 美国各州县确诊与死亡数统计 WHERE 日期= '2020-12-24' AND 累计确诊 > 1000;`

SQL执行记录 消息 结果集1 X

-----开始执行-----

【拆分SQL完成】：将执行SQL语句数量：（2条）

【执行SQL：(1)】

CREATE INDEX 累计index ON 美国各州县确诊与死亡数统计 (日期 ,累计确诊);

执行成功，耗时：[205ms.]

【执行SQL：(2)】

SELECT * FROM 美国各州县确诊与死亡数统计 WHERE 日期= '2020-12-24' AND 累计确诊 > 1000;

执行成功，当前返回：[50]行，耗时：[8ms.]

创建索引后，查询耗时 8ms，比未创建时耗时更短。

创建部分索引

[执行SQL\(F8\)](#) [格式化\(F9\)](#) [执行计划\(F6\)](#) [我的SQL](#)

1 **SELECT** * **FROM** 美国各州县确诊与死亡数统计 **WHERE** 日期='2020-12-24';

SQL执行记录 消息 结果集1 X

-----开始执行-----

【拆分SQL完成】：将执行SQL语句数量：（1条）

【执行SQL：（1）】

SELECT * FROM 美国各州县确诊与死亡数统计 WHERE 日期='2020-12-24';
执行成功，当前返回：[50]行，耗时：[32ms.]

未创建索引时，查询耗时 32ms。

[执行SQL\(F8\)](#) [格式化\(F9\)](#) [执行计划\(F6\)](#) [我的SQL](#)

1 **CREATE INDEX** 日期index **ON** 美国各州县确诊与死亡数统计 (日期) **WHERE** 日期 = '2020-12-24';
2 **SELECT** * **FROM** 美国各州县确诊与死亡数统计 **WHERE** 日期='2020-12-24';

SQL执行记录 消息 结果集1 X

-----开始执行-----

【拆分SQL完成】：将执行SQL语句数量：（2条）

【执行SQL：（1）】

CREATE INDEX 日期index ON 美国各州县确诊与死亡数统计 (日期) WHERE 日期 = '2020-12-24';
执行成功，耗时：[54ms.]

【执行SQL：（2）】

SELECT * FROM 美国各州县确诊与死亡数统计 WHERE 日期='2020-12-24';
执行成功，当前返回：[50]行，耗时：[11ms.]

创建索引后，查询耗时 11ms，比未创建时耗时更短。

创建表达式索引

[执行SQL\(F8\)](#) [格式化\(F9\)](#) [执行计划\(F6\)](#) [我的SQL√](#)

1 **SELECT** * **FROM** 美国各州县确诊与死亡数统计 **WHERE** trunc(累计确诊) > 1000;

SQL执行记录 消息 结果集1 X

-----开始执行-----
【拆分SQL完成】：将执行SQL语句数量：（1条）
【执行SQL：(1)】
SELECT * FROM 美国各州县确诊与死亡数统计 WHERE trunc(累计确诊) > 1000;
执行成功，当前返回：[50]行，耗时：[168ms.]

未创建索引时，查询耗时 168ms。

[执行SQL\(F8\)](#) [格式化\(F9\)](#) [执行计划\(F6\)](#) [我的SQL√](#)

1 **CREATE INDEX** 累计确诊_index **ON** 美国各州县确诊与死亡数统计 (trunc(累计确诊));
2 **SELECT** * **FROM** 美国各州县确诊与死亡数统计 **WHERE** trunc(累计确诊) > 1000;

SQL执行记录 消息 结果集1 X

-----开始执行-----
【拆分SQL完成】：将执行SQL语句数量：（2条）
【执行SQL：(1)】
CREATE INDEX 累计确诊_index ON 美国各州县确诊与死亡数统计 (trunc(累计确诊));
执行成功，耗时：[255ms.]
【执行SQL：(2)】
SELECT * FROM 美国各州县确诊与死亡数统计 WHERE trunc(累计确诊) > 1000;
执行成功，当前返回：[50]行，耗时：[122ms.]

创建索引后，查询耗时 122ms，比未创建时耗时更短。

3. 创建和管理视图

执行SQL(F8)

格式化(F9)

执行计划(F6)

我的SQL

1

CREATE VIEW bj_yq AS

2

SELECT 行程号, x.病例号, 性别, x.日期信息, 行程信息

3

FROM 病例行程信息 AS x LEFT JOIN 病例基本信息 AS y

4

ON x.病例号 = y.病例号 WHERE y.省 = '北京市'

SQL执行记录

消息

-----开始执行-----

【拆分SQL完成】：将执行SQL语句数量：（1条）

【执行SQL：（1）】

CREATE VIEW bj_yq AS

SELECT 行程号, x.病例号, 性别, x.日期信息, 行程信息

FROM 病例行程信息 AS x LEFT JOIN 病例基本信息 AS y

ON x.病例号 = y.病例号 WHERE y.省 = '北京市'

执行成功, 耗时: [23ms.]

输入以上命令创建普通视图 bj_yq。

当前所在库: **bupt2020211262** 切换库 | 192.168.0.139,192.168.0.206,192.168.0.161:8000 | 字符集: UTF8

SQL窗口

Schema列表

对象列表

元数据采集

打开视图: bj_yq X

对象列表数据来自DAS后台定时采集任务, 最近一次采集时间: 2022-12-01 14:57:37 立即采集 清空采集数据

表

Schema: **bupt2020211262**

视图

存储过程

视图名称	操作
bj_yq	打开视图 查看视图详情



在视图列表，可见刚刚创建的视图。点击查看视图详情，可以得知视图的内容。

SQL执行记录 消息 结果集1 X

SQL提示 全屏模式

1 SELECT * FROM bj_yq;

以下是SELECT * FROM bj_yq的执行结果集

复制行 复制列 列设置

	行程号	病例号	性别	日期信息	行程信息
1	41	1007	女	12月28日	作为确诊病例的密切接触者进行集中隔离医学观察
2	42	1007	女	1月4日	核酸检测结果为阳性，由120负压救护车转至地坛医院，综合流行病学史、临床症状、实验室检测和影像学
3	121	1020	男	12月26日	作为确诊病例的密切接触者进行集中隔离医学观察
4	122	1020	男	1月3日	核酸检测结果为阳性，由120负压救护车转至地坛医院，综合流行病学史、临床症状、实验室检测和影像学
5	123	1021	男	12月26日	作为确诊病例的密切接触者进行集中隔离医学观察
6	124	1021	男	1月3日	核酸检测结果为阳性，由120负压救护车转至地坛医院，综合流行病学史、临床症状、实验室检测和影像学

输入“SELECT * FROM bj_yq;”，可查询视图内容。

```

SELECT *
FROM bj_yq
WHERE 行程信息 LIKE '%临床分型为普通型%'
ORDER BY 病例号
LIMIT 5

```

通过以上命令，在上述视图查询临床分型为普通型的病例号、行程号、性别和日期信息，按照病例号进行升序显示（截前五条记录）。

执行SQL(F8) 格式化(F9) 执行计划(F6) 我的SQL

SQL提示 全屏模式

```

1 SELECT *
2 FROM bj_yq
3 WHERE 行程信息 LIKE '%临床分型为普通型%'
4 ORDER BY 病例号
5 LIMIT 5

```

SQL执行记录 消息 结果集1 X 覆盖模式

以下是SELECT * FROM bj_yq WHERE 行程信息 LIKE '%临床分型为普通型%' ORDER BY 病例号 LI... 该表不可编辑。

	行程号	病例号	性别	日期信息	行程信息
1	1708	25	男	1月19日	详细行病史、临床表现、实验室检测和影像学检查等结果，当日诊断为确诊病例，临床分型为普通型
2	1825	27	男	1月19日	实验室检测和影像学检查等结果，当日诊断为确诊病例，临床分型为普通型，已转至定点医院治疗
3	1899	28	女	1月19日	详细行病史、临床表现、实验室检测和影像学检查等结果，当日诊断为确诊病例，临床分型为普通型
4	1959	29	女	1月19日	临床表现、实验室检测和影像学检查等结果，当日诊断为确诊病例，临床分型为普通型
5	2096	31	男	1月19日	详细行病史、临床表现、实验室检测和影像学检查等结果，当日诊断为确诊病例，临床分型为普通型

查询结果如上，病例号为升序显示，且临床分型均为普通型。结果正确。

实验小结

通过本次实验，学会了创建四种索引的方法，并通过比较未创建索引和创建索引后的查询效率，发现虽然创建索引的命令耗时长，但有了索引之后的查询耗时低了很多，适合给经常需要查询的内容创建索引来提升查询效率。

其次，在本次实验，也学会了如何创建并使用视图。通过视图，可以为需要的数据创建一张视图，之后频繁查询时不需要多次调用查询语句。