

北京邮电大学



实验报告： GaussDB 数据库实验

学院： 计算机学院（国家示范性软件学院）

专业： 计算机科学与技术

班级： 2022211305

学号： 2022211683

姓名： 张晨阳

目录

实验一 GaussDB(for openGauss)数据库创建与维护实验	1
一、实验目的.....	1
二、实验平台及环境.....	1
三、实验内容.....	1
四、实验步骤.....	2
五、实验结果及分析.....	2
1 创建数据库	2
2 删除数据库	3
六、实验小结.....	3
实验二 数据库表的创建与维护实验.....	4
一、实验目的.....	4
二、实验环境.....	4
三、实验内容.....	4
四、实验步骤.....	4
五、实验结果及分析.....	5
1 表的创建	5
2 数据导入	7
3 数据修改	8
4 表的删除	9
六、实验总结.....	10
实验三 数据查询实验.....	11
一、实验目的.....	11
二、实验环境.....	11
三、实验内容.....	11
四、实验要求.....	12
五、实验结果及分析.....	13
1 单表查询	13
2 多表查询	17
3 嵌套查询	21
六、实验总结.....	25

实验一 GaussDB(for openGauss)数据库创建与维护实验

一、实验目的

1. 通过对 GAUSSDB(FOR OPENGauss)数据库创建与访问：
 - (1) 了解华为云分布式数据库 GAUSSDB(FOR OPENGauss)的软件环境和创建方法；
 - (2) 掌握并熟悉 GAUSSDB(FOR OPENGauss)数据库软件的使用方法；
 - (3) 掌握并熟悉 GAUSSDB(FOR OPENGauss)数据库软件的构成和相关工具；
 - (4) 通过 GAUSSDB(FOR OPENGauss)数据库软件的使用，深入理解数据库系统的基本概念。
2. 通过创建 GAUSSDB(FOR OPENGauss)数据库及进行相应的维护，了解并掌握 GaussDB(for openGauss)数据库的创建与维护的不同方法和途径，进而通过这一具体的数据库理解实际数据库所包含的各要素。

二、实验平台及环境

- 华为云: GaussDB 2.7.2
- 数据库: PostgreSQL

三、实验内容

1. GAUSSDB(FOR OPENGauss)数据库软件的使用：
 - (1) 登录并运行 GAUSSDB(FOR OPENGauss)数据库；
 - (2) 了解华为云数据库 GAUSSDB(FOR OPENGauss)的机制；
 - (3) 熟悉 GAUSSDB(FOR OPENGauss)数据库的各项功能。
2. 数据库创建与维护：
 - (1) 创建“疫情数据”数据库；

- (2) 对数据库属性和参数进行相应的修改和维护；
- (3) 练习数据库的删除等维护操作。

四、实验步骤

1. 熟悉 GaussDB(for openGauss)数据库的创建过程；
2. 创建一个名为“疫情数据”的数据库；
3. 删除“疫情数据”数据库。

五、实验结果及分析

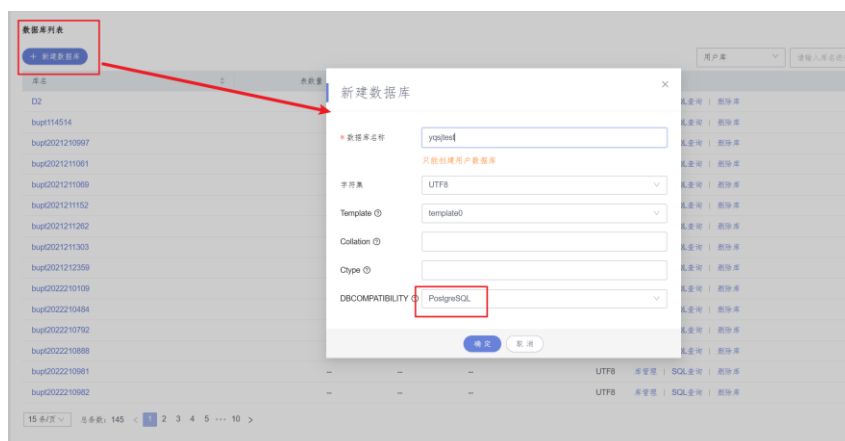
1 创建数据库

登录华为云平台，选择 IAM 用户登录，原华为云账号为 luxqbupt123，使用学校提供的账号登入。进入控制台后点击数据库，进入数据库管理服务 DAS，进入开发工具。

登录第一节创建的数据库实例连接。数据库引擎选择为 GaussDB，数据库实例选择为 gauss-3c93_root_0，角色为 master。



创建数据库疫情数据(yqsjtest):



指定兼容的数据库的类型为 PostgreSQL.

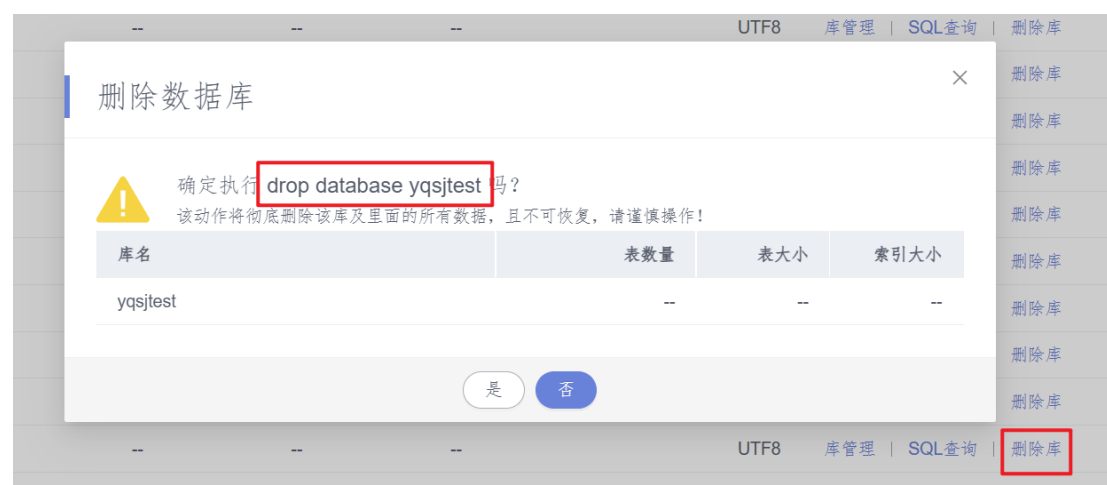
对应的 PostgreSQL 语句为:

```
CREATE DATABASE yqsjtest
WITH
    ENCODING = 'UTF8'
    TEMPLATE = template0;
```

ENCODING = 'UTF8': 指定数据库的字符集为 UTF8;

TEMPLATE = template0: 确保创建的数据库是一个干净的数据库, 而不继承系统数据库模板的任何数据或设置。

2 删除数据库



对应的 PostgreSQL 语句为:

```
CREATE DATABASE yqsjtest
```

六、实验小结

通过本次实验, 我掌握了 GAUSSDB(FOR OPENGAUSS)数据库的基础操作, 了解了其软件环境和创建方法。通过创建数据库及进行相应的维护, 了解并掌握其数据库的创建与维护的不同方法和途径, 为以后的实验打下了基础。

实验二 数据库表的创建与维护实验

一、实验目的

1. 通过进行数据库表的建立操作，熟悉并掌握 GAUSSDB(FOR OPENGauss)数据库表的建立方法，理解关系型数据库表的结构，巩固 PostgreSQL 中关于数据库表的建立语句；
2. 通过进行数据库表数据的增加、删除和插入等维护操作，熟悉并掌握 GAUSSDB(FOR OPENGauss)数据库数据的操作方法，巩固 PostgreSQL 中关于数据维护的语句。

二、实验环境

- 华为云: GaussDB 2.7.2
- 数据库: PostgreSQL

三、实验内容

建立相应的表并熟悉基本操作，例如建表、对表进行增、删、改、查。

四、实验步骤

1. 熟悉课程实验背景知识；
2. 使用 GAUSSDB(FOR OPENGauss)数据库软件创建相应的表；
3. 将提供的数据导入各表，掌握 GAUSSDB(FOR OPENGauss)数据库数据导入的方法；

注意：

- 1) 表中空列的处理；
 - 2) 表结构与数据类型的匹配。
4. 修改“病例基本信息”表数据，增加名为“备注”的列,数据类型 varchar()型；
 5. 修改“病例基本信息”表数据，将“备注”列的数据类型改为 int；

6. 修改“病例基本信息”表数据，删除“备注”列；
7. 删除“病例基本信息”数据表。

五、实验结果及分析

1 表的创建

在 SQL 查询界面，依次输入下面的 SQL 语句并执行，即可完成建表。

```
CREATE TABLE 全国城市风险等级表
(
    省 VARCHAR(255) NULL,
    市 VARCHAR(255) NULL,
    区 VARCHAR(255) NULL,
    地址详情 VARCHAR(255) NULL,
    风险等级 VARCHAR(255) NULL
)

CREATE TABLE 病例基本信息表 (
    病例号 INT PRIMARY KEY,
    省 VARCHAR(255) NULL,
    市 VARCHAR(255) NULL,
    区 VARCHAR(255) NULL,
    日期 DATE NULL,
    性别 VARCHAR(10) NULL,
    年龄 INT NULL,
    患者信息 VARCHAR(255) NULL,
    其它信息 VARCHAR(255) NULL,
    信息来源 VARCHAR(255) NULL
)

CREATE TABLE 病例行程信息表 (
    行程号 INT PRIMARY KEY,
    病例号 INT NULL,
    日期信息 VARCHAR(255) NULL,
    行程信息 VARCHAR(255) NULL
)

CREATE TABLE 参考信息表 (
    组合码 VARCHAR(255) PRIMARY KEY,
    国家 VARCHAR(255) NOT NULL,
    省州 VARCHAR(255) NULL,
```

```

    市县 VARCHAR(255) NULL,
    纬度 DECIMAL(10, 6) NULL,
    经度 DECIMAL(10, 6) NULL,
    人口数 BIGINT NULL
)

CREATE TABLE 各国疫情数据统计表 (
    日期 DATE NULL,
    国家 VARCHAR(255) NULL,
    省州 VARCHAR(255) NULL,
    累计确诊 INT NULL,
    累计治愈 INT NULL,
    累计死亡 INT NULL
)

CREATE TABLE 美国各州县确诊与死亡统计表 (
    日期 DATE NULL,
    州 VARCHAR(255) NULL,
    县 VARCHAR(255) NULL,
    累计确诊 INT NULL,
    累计死亡 INT NULL
)

CREATE TABLE 全国各省参考信息表 (
    中文名称 VARCHAR(255) NULL,
    英文名称 VARCHAR(255) NULL,
    组合码 VARCHAR(255) NULL,
    人口数 BIGINT NULL,
    省会城市 VARCHAR(255) NULL,
    纬度 DECIMAL(10, 6) NULL,
    经度 DECIMAL(10, 6) NULL
)

CREATE TABLE 全国各省累计数据统计表 (
    日期 DATE NULL,
    省 VARCHAR(255) NULL,
    累计确诊 INT NULL,
    累计治愈 INT NULL,
    累计死亡 INT NULL
)

```


2 数据导入

在“导入”界面，点击“新建任务”，导入属性配置如图所示：

新建任务

导入类型

sqlCSV

文件来源

上传文件从OBS中选择

附件存放位置 ?

obs0630bupt

没有OBS桶? 创建OBS桶

选择附件

点击或将文件拖动到此处后上传文件 (.csv)

最大不能超过1GB，且只能上传一个附件

数据库

bupt2022211683

表

"buptceshi"."全国各省参考信息表"

数据位置

第1行为属性第1行为数据

字符集

UTF8GBK自动检测

选项

☐ 忽略报错,即SQL执行失败时跳过

☒ 导入完成后删除上传的文件

☒ 清空表,执行导入前先对相应的表执行Truncate操作

备注

创建导入任务

取消

将八张表全部导入后结果如下：

表名	行数 (预估值)	表大小 (预估值)	索引大小 (预估值)	字符集	操作
全国各省参考信息表	0 (预估值)	16KB (预估值)	0B (预估值)	UTF8	SQL查询 打开表 查看表详情
全国各省累计数据统计表	2040 (预估值)	152KB (预估值)	0B (预估值)	UTF8	SQL查询 打开表 查看表详情
全国城市风险等级表	68 (预估值)	16KB (预估值)	0B (预估值)	UTF8	SQL查询 打开表 查看表详情
参考信息表	4140 (预估值)	432KB (预估值)	240KB (预估值)	UTF8	SQL查询 打开表 查看表详情
各国疫情数据统计表	14400 (预估值)	880KB (预估值)	0B (预估值)	UTF8	SQL查询 打开表 查看表详情
病例基本信息表	1187 (预估值)	256KB (预估值)	48KB (预估值)	UTF8	SQL查询 打开表 查看表详情
病例行程信息表	6852 (预估值)	880KB (预估值)	168KB (预估值)	UTF8	SQL查询 打开表 查看表详情
美国各州县确诊与死亡统计表	200400 (预估值)	11.77MB (预估值)	0B (预估值)	UTF8	SQL查询 打开表 查看表详情

3 数据修改

1 修改“病例基本信息”表数据，增加名为“备注”的列,数据类型 varchar()型
执行 SQL 语句为:

```
ALTER TABLE 病例基本信息表 ADD COLUMN 备注 VARCHAR(255)
```

执行效果如下:

The screenshot shows a database management interface. On the left, a tree view displays the database structure, including a table named '病例基本信息表' (Case Basic Information Table) with columns like '病例号' (Case Number), '省' (Province), '市' (City), '区' (District), '日期' (Date), '性别' (Gender), '年龄' (Age), '患者信息' (Patient Information), '其它信息' (Other Information), '信息来源' (Information Source), and '备注' (Remarks). The '备注' column is highlighted. In the center, the SQL statement 'ALTER TABLE 病例基本信息表 ADD COLUMN 备注 VARCHAR(255)' is entered. On the right, the 'SQL 执行记录' (SQL Execution Log) table shows the execution history:

执行时间	SQL 语句	消耗时间	执行结果
2024-11-07 15:27:33	ALTER TABLE 病例基本信息表 ADD COLUMN 备注 VARCHAR(255)	11 ms	执行成功
2024-11-07 15:27:32	set time zone 'Etc/GMT-8'	7 ms	执行成功
2024-11-07 15:26:53	ALTER TABLE 病例基本信息表 ADD COLUMN 备注 VARCHAR(255)	30 ms	执行失败
2024-11-07 15:26:52	set time zone 'Etc/GMT-8'	7 ms	执行成功
2024-10-24 11:04:20	CREATE TABLE 全国各省累计数据统计表 (日期 DATE NULL, 省 VARCHAR(25) NULL, 累计确诊 INT NULL, 累计治愈 INT NULL, 累计死亡 INT NULL)	8 ms	执行成功

2 修改“病例基本信息”表数据，将“备注”列的数据类型改为 int

执行 SQL 语句为:

```
ALTER TABLE 病例基本信息表 ALTER COLUMN 备注 TYPE INT
```

执行效果如下:

The screenshot shows the same database management interface. The '备注' column in the '病例基本信息表' is highlighted. The SQL statement 'ALTER TABLE 病例基本信息表 ALTER COLUMN 备注 TYPE INT' is entered. The 'SQL 执行记录' table shows the execution history:

执行时间	SQL 语句	消耗时间	执行结果
2024-11-07 15:29:15	ALTER TABLE 病例基本信息表 ALTER COLUMN 备注 TYPE INT	34 ms	执行成功
2024-11-07 15:29:14	set time zone 'Etc/GMT-8'	8 ms	执行成功
2024-11-07 15:27:33	ALTER TABLE 病例基本信息表 ADD COLUMN 备注 VARCHAR(255)	11 ms	执行成功
2024-11-07 15:27:32	set time zone 'Etc/GMT-8'	7 ms	执行成功
2024-11-07 15:26:53	ALTER TABLE 病例基本信息表 ADD COLUMN 备注 VARCHAR(255)	30 ms	执行失败
2024-11-07 15:26:52	set time zone 'Etc/GMT-8'	7 ms	执行成功

3 修改“病例基本信息”表数据，删除“备注”列

执行 SQL 语句为:

```
ALTER TABLE 病例基本信息表 DROP COLUMN 备注
```

执行效果如下:

库名: bupt2022211683
Schema: buptoeshi
表 视图
请按关键词搜索

- 全国各省参考信息表
- 全国各省累计数据统计表
- 全国城市风险等级表
- 参考信息表
- 各国疫情数据统计表
- 病例基本信息表
 - 病例号 (int4(32,0))
 - 省 (varchar(255))
 - 市 (varchar(255))
 - 区 (varchar(255))
 - 日期 (date)
 - 性别 (varchar(10))
 - 年龄 (int4(32,0))
 - 患者信息 (varchar(255))
 - 其它信息 (varchar(255))
 - 信息来源 (varchar(255))

1 ALTER TABLE 病例基本信息表 DROP COLUMN 备注

SQL 执行记录 消息

执行时间	SQL 语句	消耗时间	执行结果
2024-11-07 15:30:59	ALTER TABLE 病例基本信息表 DROP COLUMN 备注	6 ms	执行成功
2024-11-07 15:30:57	set time zone 'Etc/GMT-8'	8 ms	执行成功
2024-11-07 15:29:15	ALTER TABLE 病例基本信息表 ALTER COLUMN 备注 TYPE INT	34 ms	执行成功
2024-11-07 15:29:14	set time zone 'Etc/GMT-8'	8 ms	执行成功
2024-11-07 15:27:33	ALTER TABLE 病例基本信息表 ADD COLUMN 备注 VARCHAR(255)	11 ms	执行成功
2024-11-07 15:27:32	set time zone 'Etc/GMT-8'	7 ms	执行成功
2024-11-07 15:26:53	ALTER TABLE 病例基本信息表 ADD COLUMN 备注 VARCHAR(255)	30 ms	执行失败
2024-11-07 15:26:52	set time zone 'Etc/GMT-8'	7 ms	执行成功
2024-10-24 11:04:20	CREATE TABLE 全国各省累计数据统计表 (日期 DATE NULL, 省 VARCHAR(255) NULL, 累计确诊 INT NULL, 累计治愈 INT NULL, 累计死亡 INT NULL)	8 ms	执行成功

4 表的删除

执行 SQL 语句为:

```
DROP TABLE 病例基本信息表
```

执行效果如下:

库名: bupt2022211683
Schema: buptoeshi
表 视图
请按关键词搜索

- 全国各省参考信息表
- 全国各省累计数据统计表
- 全国城市风险等级表
- 参考信息表
- 各国疫情数据统计表
- 病例行信息表
- 美国各州确诊与死亡统计表

1 DROP TABLE 病例基本信息表

SQL 执行记录 消息

执行时间	SQL 语句	消耗时间	执行结果
2024-11-07 15:32:15	DROP TABLE 病例基本信息表	13 ms	执行成功
2024-11-07 15:32:14	set time zone 'Etc/GMT-8'	8 ms	执行成功
2024-11-07 15:29:15	ALTER TABLE 病例基本信息表 ALTER COLUMN 备注 TYPE INT	34 ms	执行成功

9

六、实验总结

本次实验中第一个遇到的问题是空列的处理，对于 csv 文件中的空列，如果不处理直接导入数据时，文件格式与表的格式就会不一致导致导入异常，经过尝试，手动删除该空列即可。

除此之外，另一个问题是 varchar 的设置长度，为了避免由于长度限制导致数据丢失，经查阅资料，varchar 的最长长度为 255，故 varchar 属性都设置为 255。

在本实验中，我熟悉了 GAUSSDB(FOR OPENGAUSS)数据库表的有关操作，以及数据库表数据的增加、删除和插入等维护操作的 SQL 语句使用。

建议：更新一下实验的 PPT（似乎仍然是之前版本的），本次实验的账号和密码已经更换，如果完全参考 PPT 会导致错误。

实验三 数据查询实验

一、实验目的

通过对实验二建立的数据库关系表的各种查询的操作，加深对 SQL 语言和 PostgreSQL 查询语言的了解，掌握相关查询语句的语法及使用方法。

二、实验环境

- 华为云: GaussDB 2.7.2
- 数据库: PostgreSQL

三、实验内容

1.单表查询

- 1-1 查询国内确诊病例基本信息的所有信息来源。
- 1-2 给出河南省、西藏自治区、台湾省的英文名称和人口数。
- 1-3 查询 2021 年 1 月 20 日各省现有确诊病例数据，按现有确诊病例数降序排列输出。
- 1-4 顺义区中风险地区的数量。
- 1-5 计算截至 2021 年 1 月 20 日全国累计确诊病例数。
- 1-6 查询 1005 号病例确诊后，其所在市新增的所有确诊病例。
- 1-7 在“病例基本信息表”中查询石家庄市在 2021 年 1 月 11 日当天以及之前的所有 60 岁以上的患者信息。
- 1-8 统计截止到 2020 年 12 月 30 日美国累计确诊病例数最多的 10 个州。
- 1-9 统计截至 2021 年 1 月 20 号中国发病率最高的人群（人群按照年龄划分，儿童<18，18<=青壮年<60，老年>=60）。

2.多表查询：

- 2-1 借助病例行程信息粗略查询曾去过“源升品质生活坊”的所有患者的基本信息。

2-2 根据病例行程信息表和病例基本信息表，查询行程信息中存在“家庭聚餐”的病例被确诊的日期。

2-3 对比中美两国累计确诊病例数，输出格式为(日期，中国累计确诊，美国累计确诊)。

2-4 计算截止到 2021 年 1 月 20 日，美国有些县的累计确诊是同一个州的其他县的 2 倍或以上，列出这些县，以及他们所在的州和他们的累计确诊。

2-5 计算世界上人口数排名前 10 位的国家地区。

2-6 列出美国人口超千万的大州中，截至 2021 年 1 月 20 日新冠肺炎疫情死亡率超过 2%的州。

2-7 截至 2021 年 1 月 20 日，河北省哪些区出现了新冠确诊病例但不属于中高风险地区。

2-8 在病例行程信息表的基础上根据病例基本信息表，查询河北省病例的全部信息。

3.嵌套查询：

3-1 查询披露的确诊患者信息中年龄最大的患者，输出其基本信息。(未注明年龄的患者不进行比较)。

3-2 查询 2020 年 12 月份新增确诊患者最多的城市。

3-3 结合“全国各省参考信息表”和“病例基本信息表”给出没有新增确诊病例或未披露病例信息的省份。

3-4 2021 年 1 月 20 日全国中高风险地区所在省中，哪些省在 1 月 20 日没有新增确诊信息披露。

3-5 根据病例基本信息表查询一月份国内新增患者病例最多的城市。

3-6 查询除中美两国以外的其余国家中，进入 2021 年以来单日新增确诊病例始终不低于一万例的国家。

四、实验要求

1.要求学生独立完成以上内容；

2.按照实验步骤完成实验后，撰写报告内容。

五、实验结果及分析

1 单表查询

1-1 查询国内确诊病例基本信息的所有信息来源

执行 SQL 语句为：

```
SELECT DISTINCT 信息来源
FROM 病例基本信息表
```

执行结果如下：

1 SELECT DISTINCT 信息来源

2 FROM 病例基本信息表

SQL执行记录 消息 结果集1 ×

以下是SELECT DISTINCT 信息来源 FROM 病例基本信息表的执行结果集 ① 该表不可编辑。

	信息来源
1	新京报
2	大众网
3	中国新闻网
4	广西新闻网
5	辽宁日报
6	黑龙江卫健委网站
7	河北日报
8	北京青年报
9	上海本地宝
10	松原发布
11	光明网

当前第 1 页 上一页 下一页 50 条/页 跳转到 1 GO 获取总行数

1-2 给出河南省、西藏自治区、台湾省的英文名称和人口数

执行 SQL 语句为：

```
SELECT 英文名称,人口数
FROM 全国各省参考信息表
WHERE 中文名称 IN ('河南省','西藏自治区','台湾省')
```

执行结果如下：

	英文名称	人口数
1	Henan	96050000
2	Taiwan	23816775
3	Tibet	3440000

1-3 查询 2021 年 1 月 20 日各省现有确诊病例数据，按现有确诊病例数降序排列输出

执行 SQL 语句为：

```
SELECT 省, 累计确诊-累计死亡-累计治愈 AS 现有确诊
FROM 全国各省累计数据统计表
WHERE 日期 = '2021-01-20'
ORDER BY 现有确诊 DESC
```

执行结果如下：

	省	现有确诊
1	河北省	829
2	香港特别行政区	766
3	黑龙江省	259
4	吉林省	162
5	上海市	95
6	北京市	44
7	广东省	39
8	陕西省	26
9	辽宁省	25
10	天津市	24

当前第 1 页 上一页 下一页 50 条/页 跳转到 1 GO 获取总行数

1-4 顺义区中风险地区的数量

执行 SQL 语句为：

```
SELECT COUNT(*)
FROM 全国城市风险等级表
WHERE 区 = '顺义区' AND 风险等级 = '中风险地区'
```

执行结果如下：

	count
1	4

1-5 计算截至 2021 年 1 月 20 日全国累计确诊病例数

执行 SQL 语句为：

```
SELECT SUM (累计确诊) AS 全国累计确诊数
FROM 全国各省累计数据统计表
WHERE 日期 = '2021-1-20'
```

执行结果如下：

	全国累计确诊数
1	98545

1-6 查询 1005 号病例确诊后，其所在市新增的所有确诊病例

执行 SQL 语句为：

```
SELECT *
FROM 病例基本信息表
WHERE 日期 >= (SELECT 日期
                FROM 病例基本信息表
                WHERE 病例号 = '1005')
AND
市 = (SELECT 市
      FROM 病例基本信息表
      WHERE 病例号 = '1005')
```

执行结果如下：

	病例号	省	市	区	日期	性别	年龄
1	982	黑龙江省	黑河市	爱辉区	2021-01-07	女	54
2	1004	黑龙江省	黑河市	爱辉区	2021-01-06	女	41
3	1005	黑龙江省	黑河市	爱辉区	2021-01-06	女	18

1-7 在“病例基本信息表”中查询石家庄市在 2021 年 1 月 11 日当天以及之前的所有 60 岁以上的患者信息

执行 SQL 语句为：

```
SELECT *
FROM 病例基本信息表
WHERE 市 = '石家庄市' AND 日期 <= '2021-1-11' AND 年龄 > '60'
```

执行结果如下：

	病例号	省	市	区	日期	性别	年龄
1	925	河北省	石家庄市	井陉县	2021-01-08	女	65
2	741	河北省	石家庄市	藁城区	2021-01-11	女	78
3	743	河北省	石家庄市	藁城区	2021-01-11	女	75
4	744	河北省	石家庄市	藁城区	2021-01-11	女	68
5	746	河北省	石家庄市	藁城区	2021-01-11	女	77
6	747	河北省	石家庄市	藁城区	2021-01-11	男	70

当前第 1 页 上一页 下一页 50 条/页 跳转到 1 GO 获取总行数 刷新 查看详情

1-8 统计截止到 2020 年 12 月 30 日美国累计确诊病例数最多的 10 个州

执行 SQL 语句为：

```
SELECT 州
FROM 美国各州县确诊与死亡统计表
WHERE 日期 = '2020-12-30'
GROUP BY 州
```

```
ORDER BY SUM(累计确诊) DESC
LIMIT 10
```

执行结果如下：

	州
1	California
2	Texas
3	Florida
4	New York
5	Illinois
6	Ohio
7	Georgia
8	Pennsylvania
9	Tennessee
10	North Carolina

1-9 统计截至 2021 年 1 月 20 号中国发病率最高的人群（人群按照年龄划分，儿童<18，18<=青壮年<60，老年>=60）

执行 SQL 语句为：

```
SELECT
  CASE
    WHEN 年龄 < '18' THEN '儿童'
    WHEN 年龄 < '60' AND 年龄 >= '18' THEN '青壮年'
    ELSE '老年'
  END AS 人群,
  COUNT(*) AS 病例数
FROM 病例基本信息表
WHERE 日期 <= '2021-1-20'
GROUP BY 人群
ORDER BY 病例数 DESC
LIMIT 1;
```

执行结果如下：

	人群	病例数
1	青壮年	696

2 多表查询

2-1 借助病例行程信息粗略查询曾去过“源升品质生活坊”的所有患者的基本信息

执行 SQL 语句为：

```
SELECT 病例基本信息表.*  
FROM 病例基本信息表  
JOIN 病例行程信息表 ON 病例基本信息表.病例号 = 病例行程信息表.病例号  
WHERE 行程信息 LIKE '%源升品质生活坊%';
```

执行结果如下：

	病例号	省	市	区	日期	性别	年龄
1	157	吉林省	通化市	东昌区	2021-01-18	女	73
2	167	吉林省	通化市	东昌区	2021-01-18	女	76
3	171	吉林省	通化市	东昌区	2021-01-18	男	78
4	568	吉林省	通化市	东昌区	2021-01-13	男	45

2-2 根据病例行程信息表和病例基本信息表，查询行程信息中存在“家庭聚餐”的病例被确诊的日期

执行 SQL 语句为：

```
SELECT 病例基本信息表.日期  
FROM 病例基本信息表  
JOIN 病例行程信息表 ON 病例基本信息表.病例号 = 病例行程信息表.病例号  
WHERE 病例行程信息表.行程信息 LIKE '%家庭聚餐%';
```

执行结果如下：

	日期
1	2021-01-05
2	2021-01-04
3	2021-01-13
4	2021-01-13

2-3 对比中美两国累计确诊病例数，输出格式为(日期，中国累计确诊，美国累计确诊)

执行 SQL 语句为：

```
SELECT a.日期, a.中国累计确诊, b.美国累计确诊  
FROM (
```

```

SELECT 日期, SUM(累计确诊) AS 中国累计确诊
FROM 全国各省累计数据统计表
GROUP BY 日期
) AS a
INNER JOIN (
    SELECT 日期, SUM(累计确诊) AS 美国累计确诊
    FROM 美国各州县确诊与死亡统计表
    GROUP BY 日期
) AS b ON a.日期 = b.日期
ORDER BY a.日期;

```

执行结果如下：

	日期	中国累计确诊	美国累计确诊
1	2020-11-22	92117	12347905
2	2020-11-23	92212	12521898
3	2020-11-24	92297	12697001
4	2020-11-25	92403	12879677
5	2020-11-26	92489	12991818
6	2020-11-27	92587	13199752
7	2020-11-28	92682	13355018

当前第 1 页 上一页 下一页 50 条/页 跳转到 1 GO 获取总行数 刷新

2-4 计算截止到 2021 年 1 月 20 日, 美国有些县的累计确诊是同一个州的其他县的 2 倍或以上, 列出这些县, 以及他们所在的州和他们的累计确诊

执行 SQL 语句为:

```

SELECT A.县, A.州, A.累计确诊
FROM 美国各州县确诊与死亡统计表 AS A
WHERE A.日期 = '2021-01-20'
    AND A.累计确诊 >= 2 * (
        SELECT MAX(B.累计确诊)
        FROM 美国各州县确诊与死亡统计表 AS B
        WHERE B.州 = A.州
            AND B.县 != A.县
            AND B.日期 = '2021-01-20'
    );

```

执行结果如下：

	县	州	累计确诊
1	Hennepin	Minnesota	93303
2	Jefferson	Alabama	63040
3	Maricopa	Arizona	428624
4	Douglas	Nebraska	59230
5	Miami-Dade	Florida	350252
6	District of Columbia	District of Columbia	34403
7	Los Angeles	California	1038738

2-5 计算世界上人口数排名前 10 位的国家地区

执行 SQL 语句为:

```
SELECT 国家, 人口数
FROM 参考信息表
WHERE 人口数 IS NOT NULL AND 组合码 = 国家
UNION ALL
SELECT 'China' AS 国家, SUM(人口数) AS 人口数
FROM 全国各省参考信息表
WHERE 组合码 LIKE '%, China'
ORDER BY 人口数 DESC
LIMIT 10;
```

执行结果如下:

	国家	人口数
1	China	1428493185
2	India	1380004385
3	US	329466283
4	Indonesia	273523621
5	Pakistan	228892331
6	Brazil	212559409
7	Nigeria	206139587
8	Bangladesh	164689383
9	Russia	145934460
10	Mexico	127792286

2-6 列出美国人口超千万的大州中, 截至 2021 年 1 月 20 日新冠肺炎疫情死亡率超过 2%的州

执行 SQL 语句为:

```
SELECT s.州, (s.州累计死亡 / s.州累计确诊) * 100 AS 死亡率
FROM (
    SELECT 州, SUM(累计确诊) AS 州累计确诊, SUM(累计死亡) AS 州累计死亡
    FROM 美国各州县确诊与死亡统计表
    WHERE 日期 = '2021-01-20'
    GROUP BY 州
) AS s
JOIN 参考信息表 AS r ON s.州 = r.省州
WHERE r.国家 = 'US'
    AND r.人口数 > 10000000
    AND 市县 IS NULL
    AND s.州累计确诊 > 0
    AND (s.州累计死亡 / s.州累计确诊) > 0.02
ORDER BY 死亡率 DESC;
```

执行结果如下:

	州	死亡率
1	New York	3.249464178695136
2	Pennsylvania	2.515414228831464

2-7 截至 2021 年 1 月 20 日，河北省哪些区出现了新冠确诊病例但不属于中高风险地区

小型数据集时可执行 SQL 语句为：

```
SELECT 区
FROM 病例基本信息表
WHERE 日期 <= '2021-01-20' AND 省 = '河北省' AND 区 IS NOT NULL
EXCEPT
SELECT DISTINCT 区
FROM 全国城市风险等级表
WHERE 省 = '河北省';
```

大型数据集时可执行：

```
SELECT t1.区
FROM 病例基本信息表 AS t1
WHERE t1.省 = '河北省'
  AND t1.日期 <= '2021-01-20'
  AND t1.区 IS NOT NULL
  AND NOT EXISTS (
    SELECT *
    FROM 全国城市风险等级表 AS t2
    WHERE t2.区 = t1.区
      AND t2.省 = t1.省
  );
```

执行结果如下：

	区
1	栾城
2	井陘县
3	裕华市
4	正定

2-8 在病例行程信息表的基础上根据病例基本信息表，查询河北省病例的全部信息

执行 SQL 语句为：

```
SELECT b.*
FROM 病例基本信息表 AS b
JOIN 病例行程信息表 AS t ON b.病例号 = t.病例号
```

```
WHERE b.省 = '河北省';
```

执行结果如下：

病例号	省	市	区	日期	性别	年龄
1	河北省	石家庄市	鹿城区	2021-01-20	女	54
2	河北省	石家庄市	鹿城区	2021-01-20	女	54

3 嵌套查询

3-1 查询披露的确诊患者信息中年龄最大的患者，输出其基本信息。(未注明年龄的患者不进行比较)

执行 SQL 语句为：

```
SELECT 患者信息
FROM 病例基本信息表
WHERE 年龄 = (
    SELECT MAX(年龄)
    FROM 病例基本信息表
    WHERE 年龄 IS NOT NULL
);
```

执行结果如下：

	患者信息
1	张某某，女，95岁

3-2 查询 2020 年 12 月份新增确诊患者最多的城市

执行 SQL 语句为：

```
SELECT 市, 新增确诊数
FROM (
    SELECT 市, COUNT(*) AS 新增确诊数
    FROM 病例基本信息表
    WHERE 日期 >= '2020-12-01' AND 日期 < '2021-01-01'
    GROUP BY 市
) AS 月统计
ORDER BY 新增确诊数 DESC
LIMIT 1;
```

执行结果如下：

	市	新增确诊数
1	大连市	38

3-3 结合“全国各省参考信息表”和“病例基本信息表”给出没有新增确诊病例或未披露病例信息的省份

执行 SQL 语句为：

```
SELECT SUBSTRING(中文名称, 1, 2) AS 省份
FROM 全国各省参考信息表
WHERE SUBSTRING(中文名称, 1, 2) NOT IN (
    SELECT DISTINCT SUBSTRING(省, 1, 2)
    FROM 病例基本信息表
    WHERE 省 IS NOT NULL
)
```

执行结果如下：

	省份
1	安徽
2	重庆
3	福建
4	甘肃
5	广东
6	贵州
7	海南
8	河南
9	湖北

3-4 2021 年 1 月 20 日全国中高风险地区所在省中，哪些省在 1 月 20 日没有新增确诊信息披露

执行 SQL 语句为：

```
SELECT 省
FROM 全国城市风险等级表
EXCEPT
SELECT DISTINCT 省
FROM 病例基本信息表
WHERE 日期 = '2021-1-20'
```

执行结果如下：

	省
1	吉林省
2	辽宁省

3-5 根据病例基本信息表查询一月份国内新增患者病例最多的城市

执行 SQL 语句为:

SELECT 市, 病例数

```
FROM (
    SELECT 市, COUNT(*) AS 病例数
    FROM 病例基本信息表
    WHERE 日期 >= '2021-01-01' AND 日期 <= '2021-01-31'
    GROUP BY 市
) AS 一月各市确诊
WHERE 病例数 = (
    SELECT MAX(病例数)
    FROM (
        SELECT 市, COUNT(*) AS 病例数
        FROM 病例基本信息表
        WHERE 日期 >= '2021-01-01' AND 日期 <= '2021-01-31'
        GROUP BY 市
    ) AS 内部统计
);
```

执行结果如下:

	市	病例数
1	石家庄市	788

3-6 查询除中美两国以外的其余国家中, 进入 2021 年以来单日新增确诊病例始终不低于一万例的国家

由于学校提供的数据说明不够清晰, 我认为此处存在两种解答:

1 各国疫情数据表中的省州, 为 null 表示为国家级的数据, 不为 null 则表示省州级的数据。故计算国家的每日新增时, 需要排除省州。

执行 SQL 语句为:

```
WITH 每日新增 AS (
    SELECT 国家, 日期,
           累计确诊 - LAG(累计确诊) OVER (PARTITION BY 国家 ORDER BY 日期) AS 新增确诊
    FROM 各国疫情数据表
    WHERE 日期 >= '2021-01-01'
           AND 国家 NOT IN ('China', 'US')
           -- AND 省州 IS NULL -- 确保只统计国家级别的数据
)
SELECT DISTINCT 国家
FROM 每日新增
WHERE 国家 NOT IN (
    SELECT DISTINCT 国家
```

```
FROM 每日新增
WHERE 新增确诊 IS NOT NULL AND 新增确诊 < 10000
);
```

执行结果如下：

	国家
1	Russia
2	Brazil

- 2 不考虑省州的属性。因为查询了几个国家的数据：Canada、Singapore、France 发现：Canada 的省州不存在 null，Singapore 的省州全为 null，France 既存在 null，也存在具体的省州。故认为是数据未说明具体指向。

执行 SQL 语句为：

```
WITH 每日新增 AS (
    SELECT 国家, 日期,
           累计确诊 - LAG(累计确诊) OVER (PARTITION BY 国家 ORDER BY 日期) AS 新增确诊
    FROM 各国疫情数据统计表
    WHERE 日期 >= '2021-01-01'
           AND 国家 NOT IN ('China', 'US')
           AND 省州 IS NULL -- 确保只统计国家级别的数据
)
SELECT DISTINCT 国家
FROM 每日新增
WHERE 国家 NOT IN (
    SELECT DISTINCT 国家
    FROM 每日新增
    WHERE 新增确诊 IS NOT NULL AND 新增确诊 < 10000
);
```

执行结果如下：

	国家
1	Russia
2	United Kingdom
3	Brazil

六、实验总结

本次实验让我对 sql 语句的掌握得到了很大的提高，也让我学习到了新的知识点。同时，我学会了如何根据实际需求选择合适的查询方式，比如小型数据集使用 `except`，大型数据集更倾向于 `not in`。也让我认识到写 sql 语句时，不仅仅要关注表的属性，更需要关注存储的数据的具体内容。

除此之外，我也想提出我的建议：加强数据的格式化、统一化，或者在当前的基础上，增添数据格式的说明，避免因数据的问题造成解题时的困扰。