

**数据库系统原理**



**第一次实验**

**姓 名 鄭毓恒**

**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 10 月**

**实验一GaussDB(for openGauss)数据库创建与维护实验**

**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。

**实验平台及环境**

平台：华为云数据库数据管理服务DAS面向个体开发工具

环境：GaussDB(for openGauss)数据库，语言PostgreSQL

**实验内容**

1. 熟悉GaussDB(for openGauss)数据库的创建过程；

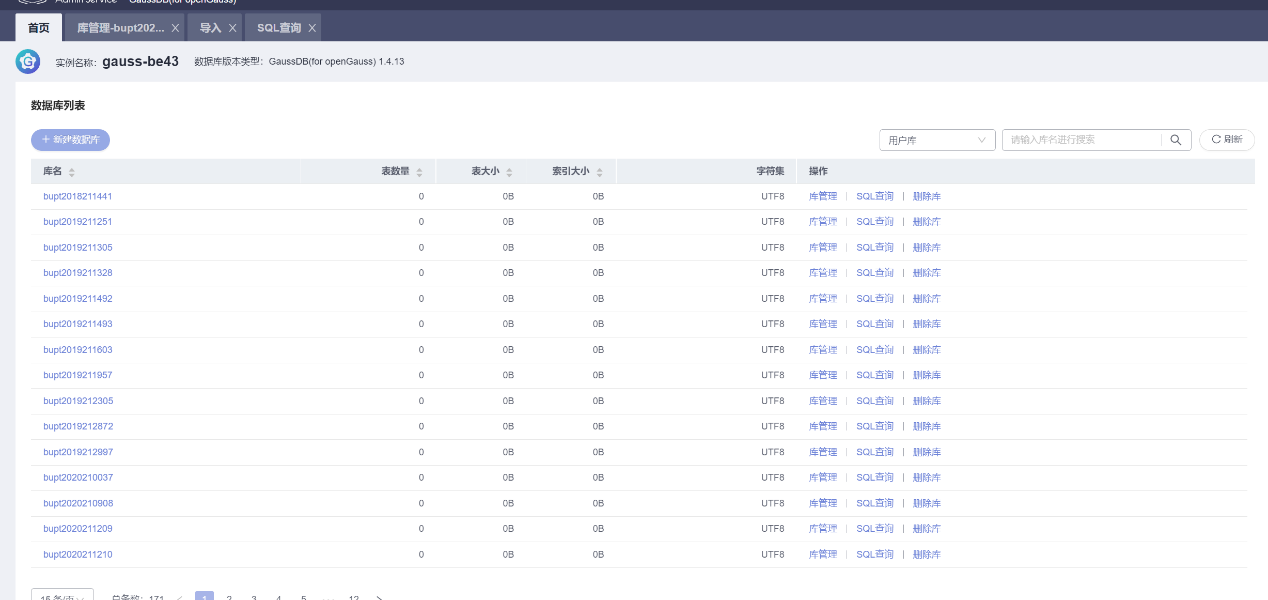
2. 创建一个名为“疫情数据”的数据库；

3. 删除“疫情数据”数据库。

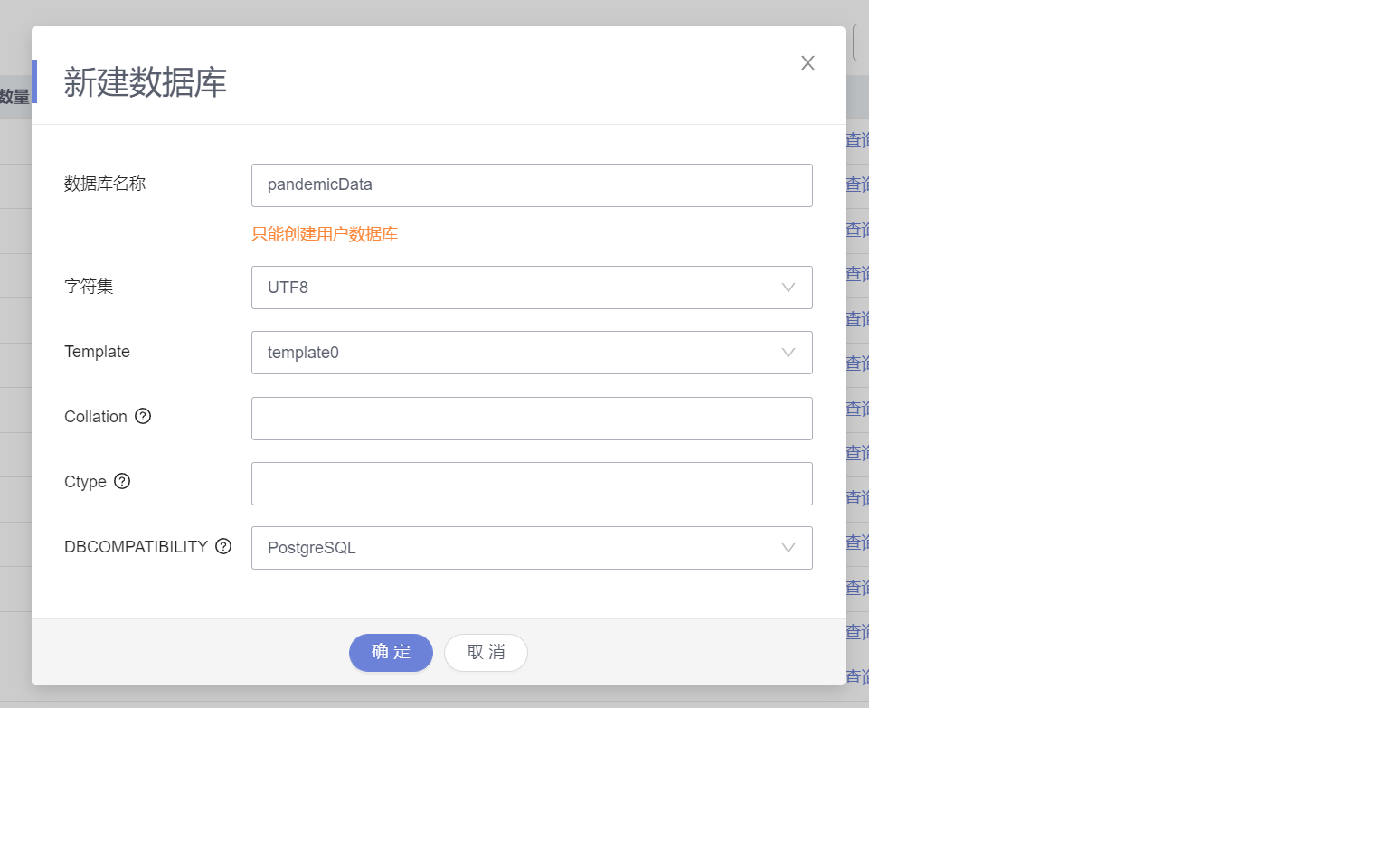
**实验步骤**



首先登录华为云，进入数据管理服务DAS个人开发工具，登录先前创建的数据库实例。



登录实例后，点击左上角新建数据库。



新建数据库，由于数据库名称只能为英文，因此使用’pandemicData’代替。字符集选择UTF8，Template选择template0，DBCOMPATIBILITY选择PostgreSQL。点击确定完成创建。创建完毕后，再选择该数据库，进行删除。

**实验结果及分析**



在所有数据库中搜索pandemicData，可以找到刚刚创建的数据库，证明数据创建成功。



点击删除库再点击‘是’，将该数据库删除。

**实验小结**

本次实验内容简单，只是简单的数据库创建和删除，并没有遇到什么问题。但是，从本次实验了解到华为云数据管理系统可以创建和使用的数据库类型。

**实验二 数据库表的创建与维护实验**

**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。

**实验平台及环境**

平台：华为云数据库数据管理服务DAS面向个体开发工具

环境：GaussDB(for openGauss)数据库，语言PostgreSQL

**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。

**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。

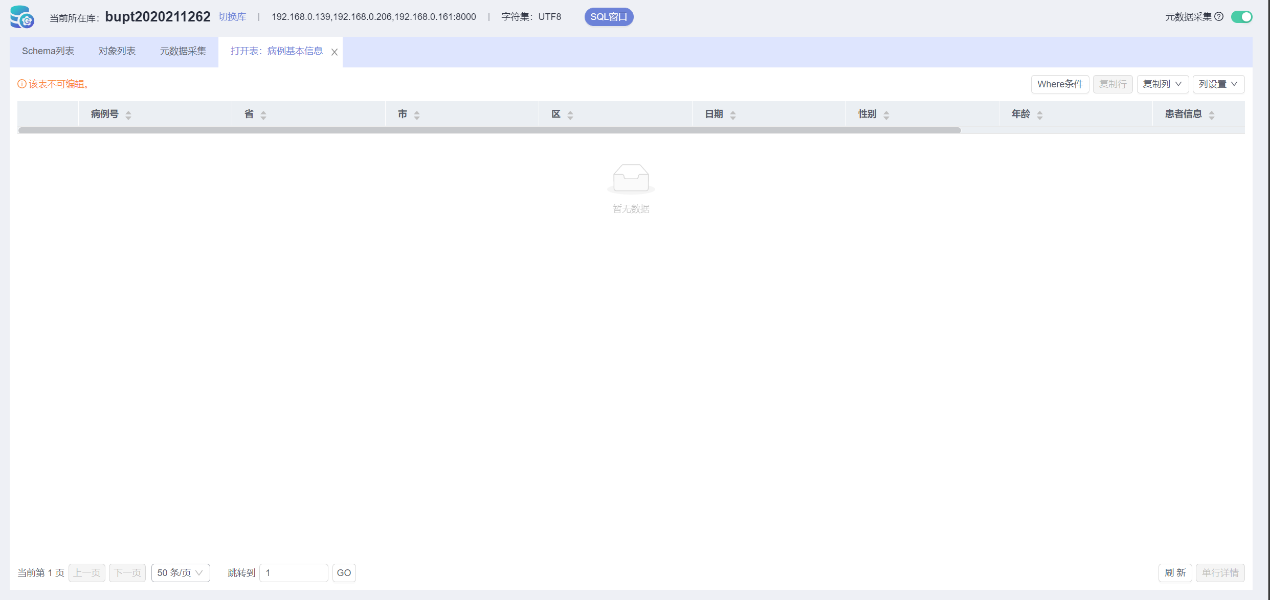
1. 修改 “病例基本信息”表数据，增加名为“备注”的列,数据类型为varchar()型；
2. 修改 “病例基本信息”表数据，将 “备注”列的数据类型改为int；
3. 修改 “病例基本信息”表数据，删除“备注”列；
4. 删除“病例基本信息”数据表。

**实验结果及分析**

|  |
| --- |
| CREATE TABLE 病例基本信息(  病例号 INT,  省 VARCHAR (20),  市 VARCHAR (20),  区 VARCHAR (20),  日期 VARCHAR (10),  性别 VARCHAR (10),  年龄 INT,  患者信息 VARCHAR (1000),  其它信息 VARCHAR (1000),  信息来源 VARCHAR (100)  ); |

输入以上SQL语句，创建“病例基本信息”表。除了病例号和年龄设为int类型，其他设为varchar，方便导入数据表格。





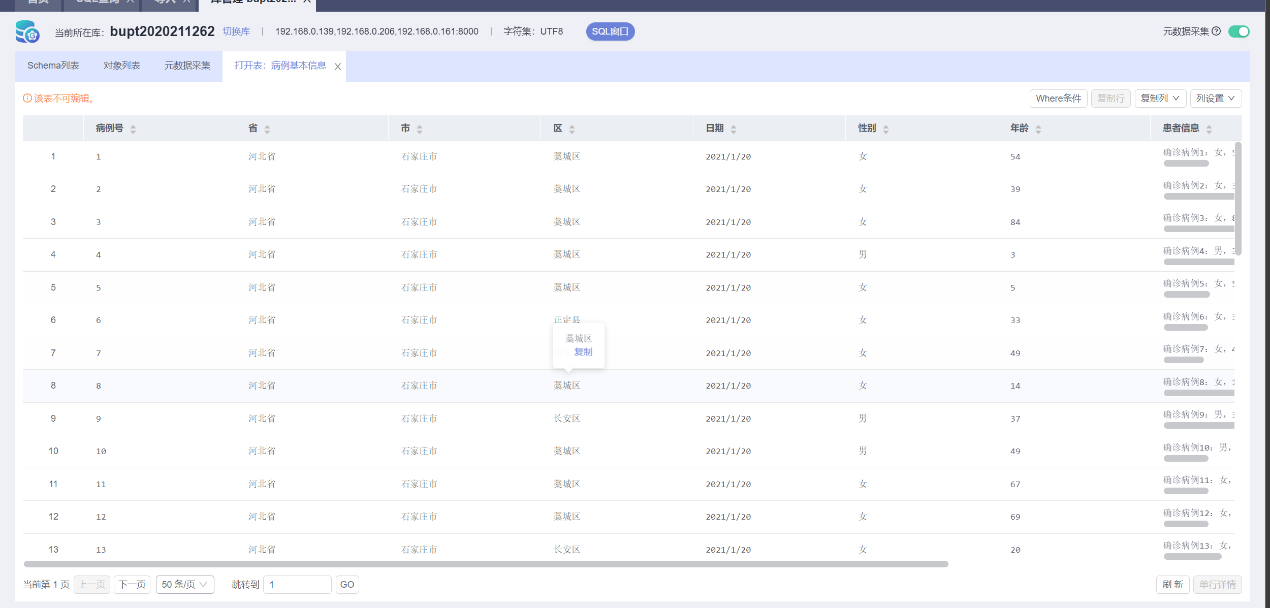
查看库中的表，可以看到“病例基本信息”表。点击查看，每列名称与SQL语句中设置的一致。



选择以上设置，将病例基本信息数据表格导入。



导入成功。



打开表格可见以上内容，数据正确导入。

|  |
| --- |
| ALTER TABLE 病例基本信息 ADD 备注 VARCHAR(1000); |

使用以上语句，给“病例基本信息”表加入类型为varchar(100)的备注列。



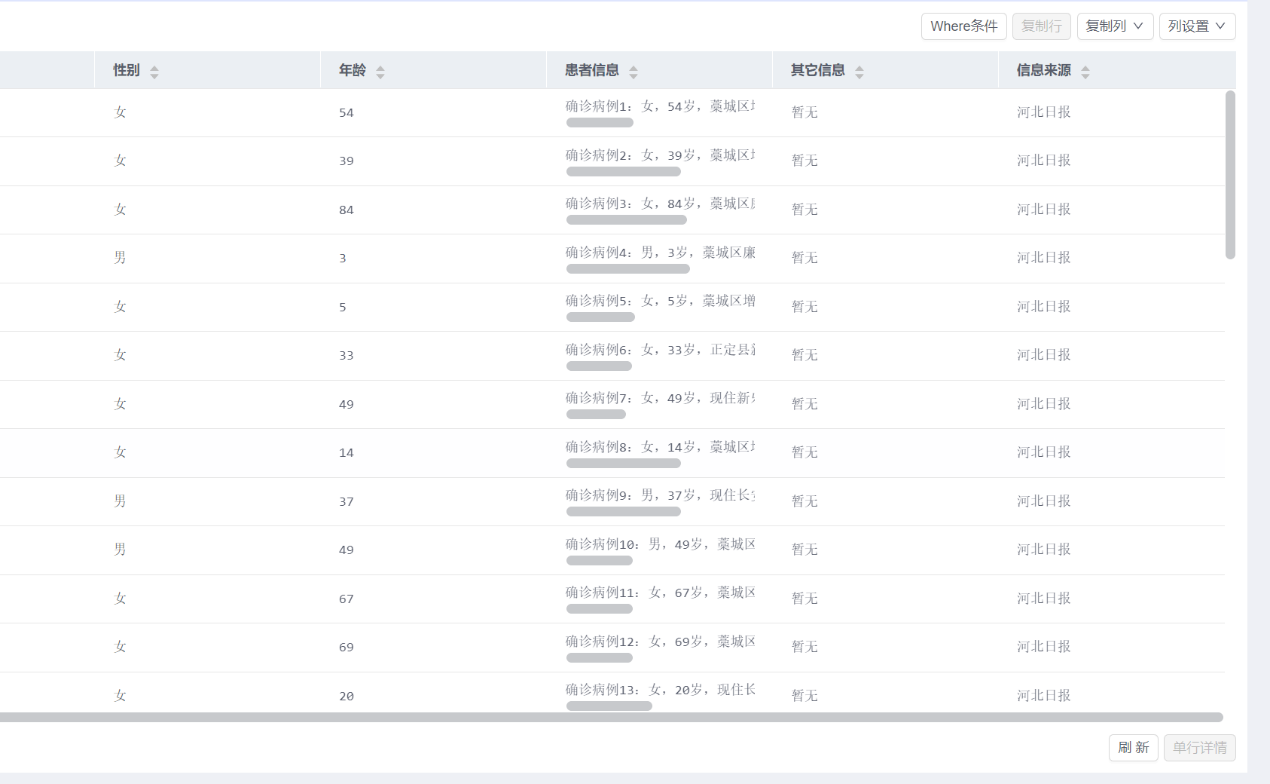
打开病例基本信息表，可以看到新增的“备注”列，新增成功。

|  |
| --- |
| ALTER TABLE 病例基本信息  MODIFY 备注 INT ; |

执行以上SQL语句，将“备注”列更改为int数据类型。

|  |
| --- |
| ALTER TABLE 病例基本信息  DROP COLUMN 备注; |

执行以上SQL语句，将“备注”列删除。



打开病例基本信息表，可见“备注”列已被删除。

|  |
| --- |
| DROP TABLE 病例基本信息; |

执行以上SQL语句，将“病例基本信息”表删除。



查看数据库中的表，没有“病例基本信息”，删除成功。

**实验小结**

实验在导入过程中遇到了问题，首先是导入的表格中的列和数据库中的目标表格的列要一致，不仅是它们的名称，还有它们的数据类型。比如，一开始在“患者信息”列只设置了varchar(100)，长度只有100个字符，大约为50个中文字符。然而，在原数据表格中，患者信息存在长度超过50个中文字符的项，因此导入失败。将该列的数据类型改为varchar(1000)，导入成功。从这点学习到，假如不能确定原数据表格的数据格式、类型或者规范性，可以先将数据类型设为足够长的varchar或表示范围够大的其他数据类型。

导入表格完成后，剩余的删改列和删除表操作没有遇到问题。从本次实验学习到利用SQL语句对表格进行基本的创建、修改和删除操作，以及如何将外部数据导入进数据库中，为后续实验建立了基础。

**实验三 数据查询实验**

**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。

**实验平台及环境**

平台：华为云数据库数据管理服务DAS面向个体开发工具

环境：GaussDB(for openGauss)数据库，语言PostgreSQL

**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。

**实验步骤**

|  |
| --- |
| CREATE TABLE 病例基本信息(  病例号 INT,  省 VARCHAR(50),  市 VARCHAR(50),  区 VARCHAR(50),  日期 VARCHAR(10),  性别 VARCHAR(10),  年龄 INT,  患者信息 VARCHAR(1000),  其它信息 VARCHAR(1000),  信息来源 VARCHAR(100)  );  CREATE TABLE 病例行程信息(  行程号 INT,  病例号 INT,  日期信息 VARCHAR(1000),  行程信息 VARCHAR(5000)  );  CREATE TABLE 参考信息(  组合码 VARCHAR(100),  国家 VARCHAR(100),  省州 VARCHAR(500),  市县 VARCHAR(500),  纬度 FLOAT,  经度 FLOAT,  人口数 INT  );  CREATE TABLE 各国疫情数据统计(  日期 VARCHAR(100),  国家 VARCHAR(100),  省州 VARCHAR(500),  累计确诊 INT,  累计治愈 INT,  累计死亡 INT  );  CREATE TABLE 美国各州县确诊与死亡数统计(  日期 VARCHAR(100),  州 VARCHAR(100),  县 VARCHAR(100),  累计确诊 INT,  累计死亡 INT  );  CREATE TABLE 全国城市风险等级(  省 VARCHAR(100),  市 VARCHAR(100),  区 VARCHAR(100),  地址详情 VARCHAR(1000),  风险等级 VARCHAR(20)  );  CREATE TABLE 全国各省参考信息(  中文名称 VARCHAR(100),  英文名称 VARCHAR(100),  组合码 VARCHAR(100),  人口数 INT,  省会城市 VARCHAR(100),  纬度 FLOAT,  经度 FLOAT  );  CREATE TABLE 全国各省累计数据统计(  日期 VARCHAR(100),  省 VARCHAR(100),  累计确诊 INT,  累计治愈 INT,  累计死亡 INT  ); |

首先执行以上SQL语句，为要导入的所有数据表格建立表，然后将原始表格导入。



此处显示全部导入成功，可以开始实验。

对实验内容的每一步，分别执行以下SQL语句完成任务。

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。 |
| SELECT 信息来源  FROM 病例基本信息; |

|  |
| --- |
| 1-2给出河南省、西藏自治区、台湾省的英文名称和人口数。 |
| SELECT 中文名称, 英文名称, 人口数  FROM 全国各省参考信息  WHERE 中文名称 = '河南省' or 中文名称 = '西藏自治区' or 中文名称 = '台湾省'; |

|  |
| --- |
| 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。 |
| SELECT 省, CAST(累计确诊-累计死亡-累计治愈 AS INT) AS 现有确诊  FROM 全国各省累计数据统计  WHERE 日期 = '2021/1/20'  ORDER BY 现有确诊 DESC |

|  |
| --- |
| 1-4顺义区中风险地区的数量。 |
| SELECT count(\*)  FROM 全国城市风险等级  WHERE 区 = '顺义区' and 风险等级 = '中风险地区'; |

|  |
| --- |
| 1-5计算截至2021年1月20日全国累计确诊病例数。 |
| SELECT SUM(累计确诊)  FROM 全国各省累计数据统计  WHERE 日期 = '2021/1/20'; |

|  |
| --- |
| 1-6查询1005号病例确诊后，其所在市新增的所有确诊病例。 |
| ALTER TABLE 病例基本信息  MODIFY 日期 DATE;  SELECT \*  FROM 病例基本信息 AS A,  (SELECT 省, 市  FROM 病例基本信息  WHERE 病例号 = 1005  )AS B  WHERE A.日期 > '2021-01-06' and A.省 = B.省 and A.市 = B.市; |

\*创建“病例基本信息”时，将日期的类型设为varchar()，因此要先改为DATE类型才能进行比较。

|  |
| --- |
| 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。 |
| SELECT \*  FROM 病例基本信息  WHERE 日期 <= '2021-01-11' and 省 = '河北省' and 市 = '石家庄市' and 年龄 > 60; |

|  |
| --- |
| 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。 |
| ALTER TABLE 美国各州县确诊与死亡数统计  MODIFY 日期 date;  SELECT 州, SUM(累计确诊)  FROM 美国各州县确诊与死亡数统计  WHERE 日期 = '2020-12-30'  GROUP BY 州  ORDER BY SUM(累计确诊) DESC  LIMIT 10; |

\*创建“美国各州县确诊与死亡数统计”时，将日期的类型设为varchar()，因此要先改为DATE类型才能进行比较。

|  |
| --- |
| 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）。 |
| ALTER TABLE 病例基本信息  ADD COLUMN 人群 VARCHAR(10);  UPDATE 病例基本信息  SET 人群 = '儿童'  WHERE 年龄 < 18;  UPDATE 病例基本信息  SET 人群 = '青壮年'  WHERE 年龄 >= 18 and 年龄 < 60;  UPDATE 病例基本信息  SET 人群 = '老年'  WHERE 年龄 >= 60;  SELECT 人群, COUNT(人群)  FROM 病例基本信息  WHERE 日期 <= '2021-01-20' AND 人群 IS NOT NULL  GROUP BY 人群  ORDER BY COUNT(人群) DESC  LIMIT 1; |
|  |

\*先给“病例基本信息”表加入“人群”列，并更新相应数据。

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。 |
| SELECT \*  FROM 病例基本信息  WHERE 病例号 IN (  SELECT DISTINCT 病例号  FROM 病例行程信息  WHERE 行程信息 LIKE '%源升品质生活坊%'); |

|  |
| --- |
| 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。 |
| SELECT 日期  FROM 病例基本信息  WHERE 病例号 IN (  SELECT DISTINCT 病例号  FROM 病例行程信息  WHERE 行程信息 LIKE '%家庭聚餐%'); |

|  |
| --- |
| 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)。 |
| SELECT C.日期, C.中国累计确诊, A.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  GROUP BY 日期  ORDER BY 日期  ) AS C,  (  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数统计  GROUP BY 日期  ORDER BY 日期  ) AS A  WHERE C.日期 = A.日期  ORDER BY 日期 |

|  |
| --- |
| 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。 |
| SELECT B.州, B.县, B.县累计确诊  FROM  (  SELECT 州, SUM(累计确诊) AS 州累计确诊  FROM 美国各州县确诊与死亡数统计  WHERE 日期 = '2021-01-20'  GROUP BY 州  ) AS A,  (  SELECT 州, 县, 累计确诊 AS 县累计确诊  FROM 美国各州县确诊与死亡数统计  WHERE 日期 = '2021-01-20'  ) AS B  WHERE A.州 = B.州 AND B.县累计确诊 >= 2 \* (A.州累计确诊 - B.县累计确诊) |

|  |
| --- |
| 2-5计算世界上人口数排名前10位的国家地区。 |
| SELECT 国家, SUM(人口数) AS 人口总数  FROM 参考信息  WHERE 人口数 IS NOT NULL AND 省州 IS NULL  GROUP BY 国家  ORDER BY 人口总数 DESC  LIMIT 10; |

|  |
| --- |
| 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。 |
| SELECT A.州, B.人口数, CAST(A.累计死亡/B.人口数 AS DECIMAL(10, 2)) AS 死亡率  FROM  (  SELECT 州, SUM(累计死亡) AS 累计死亡  FROM 美国各州县确诊与死亡数统计  WHERE 日期 = '2021-01-20'  GROUP BY 州  )AS A,  (  SELECT 省州, 人口数  FROM 参考信息  WHERE 国家 = 'US' AND 省州 IS NOT NULL AND 市县 IS NULL  )AS B  WHERE B.人口数 >= 10000000 AND A.州 = B.省州 AND A.累计死亡/B.人口数 > 0.02; |

|  |
| --- |
| 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。 |
| SELECT A.市, A.区  FROM  (  SELECT 市, 区  FROM 病例基本信息  WHERE 省 = '河北省'  )AS A  WHERE NOT EXISTS  (  SELECT 市, 区  FROM 全国城市风险等级 AS B  WHERE 省 = '河北省' AND A.市 = B.市 AND A.区 = B.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。 |
| SELECT A.\*, B.\*  FROM 病例基本信息 AS A, 病例行程信息 AS B  WHERE A.病例号 = B.病例号 AND A.省 = '河北省'; |

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)。 |
| SELECT A.\*  FROM 病例基本信息 AS A  WHERE A.年龄 >= ALL (  SELECT MAX(年龄)  FROM 病例基本信息  ) |

|  |
| --- |
| 3-2查询2020年12月份新增确诊患者最多的城市。 |
| SELECT A.\*  FROM  (  SELECT 市, COUNT(\*) AS 十二月新增确诊数  FROM 病例基本信息  WHERE 日期 >= '2020-12-01' AND 日期 < '2021-01-01'  GROUP BY 市  )AS A  ORDER BY A.十二月新增确诊数 DESC  LIMIT 1; |

|  |
| --- |
| 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。 |
| SELECT \*  FROM  (  SELECT DISTINCT 中文名称 AS 省  FROM 全国各省参考信息  )AS A  WHERE NOT EXISTS(  SELECT DISTINCT 省  FROM 病例基本信息 AS B  WHERE A.省 = B.省  ) |

|  |
| --- |
| 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。 |
| SELECT \*  FROM  (  SELECT DISTINCT 省  FROM 全国城市风险等级  )AS A  WHERE NOT EXISTS(  SELECT DISTINCT 省  FROM 病例基本信息 AS B  WHERE A.省 = B.省 AND 日期 = '2021-01-20'  ) |

|  |
| --- |
| 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。 |
| SELECT A.\*  FROM  (  SELECT 市, COUNT(\*) AS 一月新增确诊数  FROM 病例基本信息  WHERE 日期 >= '2021-01-01'  GROUP BY 市  )AS A  ORDER BY A.一月新增确诊数 DESC  LIMIT 1 |

|  |
| --- |
| 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。 |
| ALTER TABLE 各国疫情数据统计  MODIFY 日期 DATE  SELECT A.国家  FROM  (  SELECT DISTINCT 国家  FROM 各国疫情数据统计  WHERE 国家 <> 'US' AND 国家 <> 'China'  )AS A  WHERE 国家 NOT IN  (  SELECT D.国家  FROM (  SELECT C.日期, C.国家, CAST(C.累计确诊 - B.累计确诊 AS INT) AS 单日新增确诊  FROM (  SELECT 日期, 国家, SUM(累计确诊) AS 累计确诊  FROM 各国疫情数据统计  GROUP BY 日期, 国家  ) AS B,  (  SELECT 日期, 国家, SUM(累计确诊) AS 累计确诊  FROM 各国疫情数据统计  GROUP BY 日期, 国家  ) AS C  WHERE B.日期 >= '2020-12-31' AND C.日期 - B.日期 = 1 AND C.国家 = B.国家 )AS D  WHERE D.单日新增确诊 < 10000  ) |

\*创建“各国疫情数据统计”时，将日期的类型设为varchar()，因此要先改为DATE类型才能进行比较。

**实验结果及分析**

以上SQL语句的执行结果如下。

|  |
| --- |
| 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）。 |
|  |

|  |
| --- |
| 2-1 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。 |
|  |

|  |
| --- |
| 3-1 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。 |
|  |

**实验小结**

本次实验主要以编写SQL代码完成特定查询任务为主，遇到的问题很多是有关这一方面的。例如，更改表的某一列的名称，在打完ALTER TABLE 表名称之后，无论是输入ALTER COLUMN还是MODIFY COLUMN都执行失败，后来才知道只需要打MODIFY就能成功。系统并没有自动补全，当时以为并没有MODIFY这个关键字。又例如SQL要求选择的列中的，除了使用函数处理的以外的所有列，都需要使用Group By，而有的情况并不需要。这里主要还是代码思路问题，错误使用了SUM()、COUNT()等函数。

在本次实验，通过对数据库的实践操作和SQL的编写，对数据库原理和SQL语句的使用有了更深刻的认识，并掌握了更多使用技巧，为后续实验建立了基础。同时，也了解到SQL处理数据表的便利性，激发了我对数据库的兴趣，希望能在后续课程和实验中，学习到更多知识，并在往后利用于程序设计编写中。