

# 数据库第四次上机

## 本次上机任务

本次的上机任务是熟悉SQL语言的基本操作：

- 外部表导入
- DQL复习
- DDL DML
- 视图

本次实验只需使用 **SQL SERVER / MySQL / openGauss** 中**一种**数据库完成即可  
**数据库管理工具不限**

## TASK 1 导入数据/ 复习DQL

新建数据库，将附件中的三个表导入到该数据库中 (导入数据教程在下页)。  
并完成以下查询任务：

Q1: 查询与CID=1的顾客同一个城市的所有顾客ID

Q2: 查询购买过所有省份（Food表中出现过的City）的食物的顾客ID

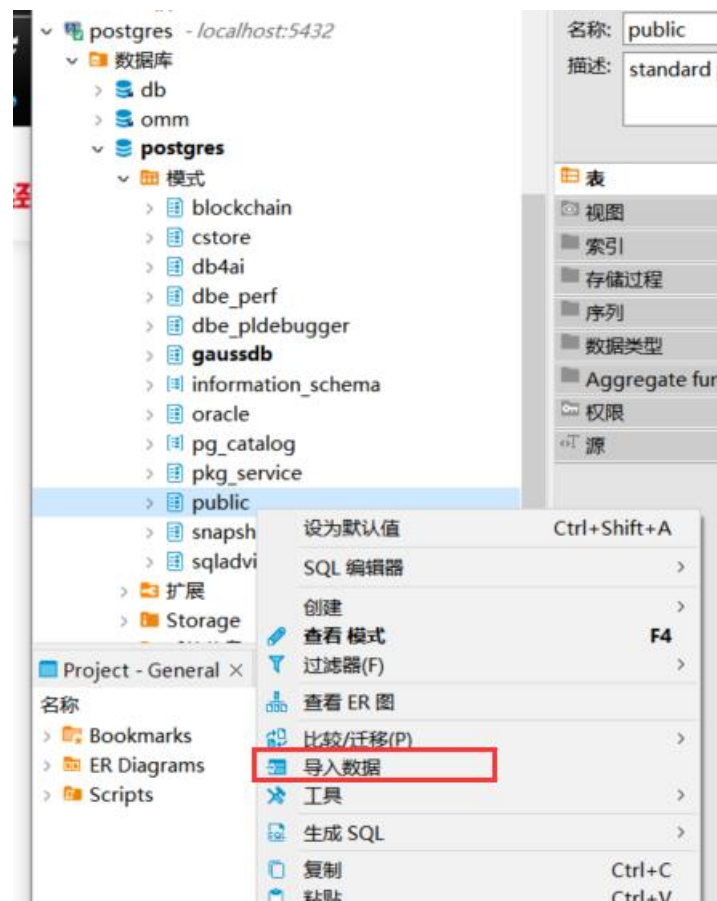
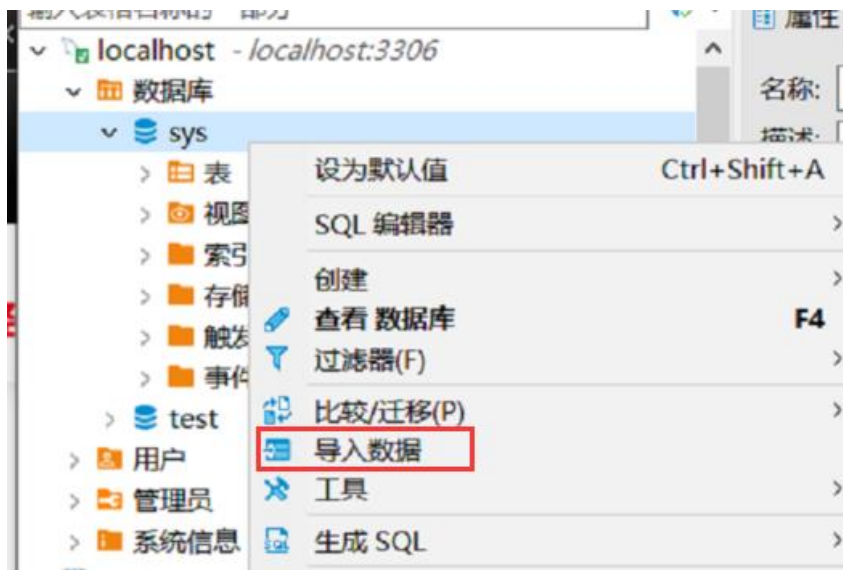
Q3: 查询至少购买过ID为4的顾客买过的全部食物的顾客ID

**提交要求：**请在PDF/WORD等任何方便助教阅读查看的文档中完成以下提交内容：需要提交每个查询的SQL语句和查询结果截图，请标好题号。

## TASK DBeaver 导入数据

使用mySQL和openGauss的导入第一步略有不同（主要是不同数据库对schema定义不同）  
区别在于opengauss需要右键数据库下某个模式（下右图选择了默认模式public）才能导入数据

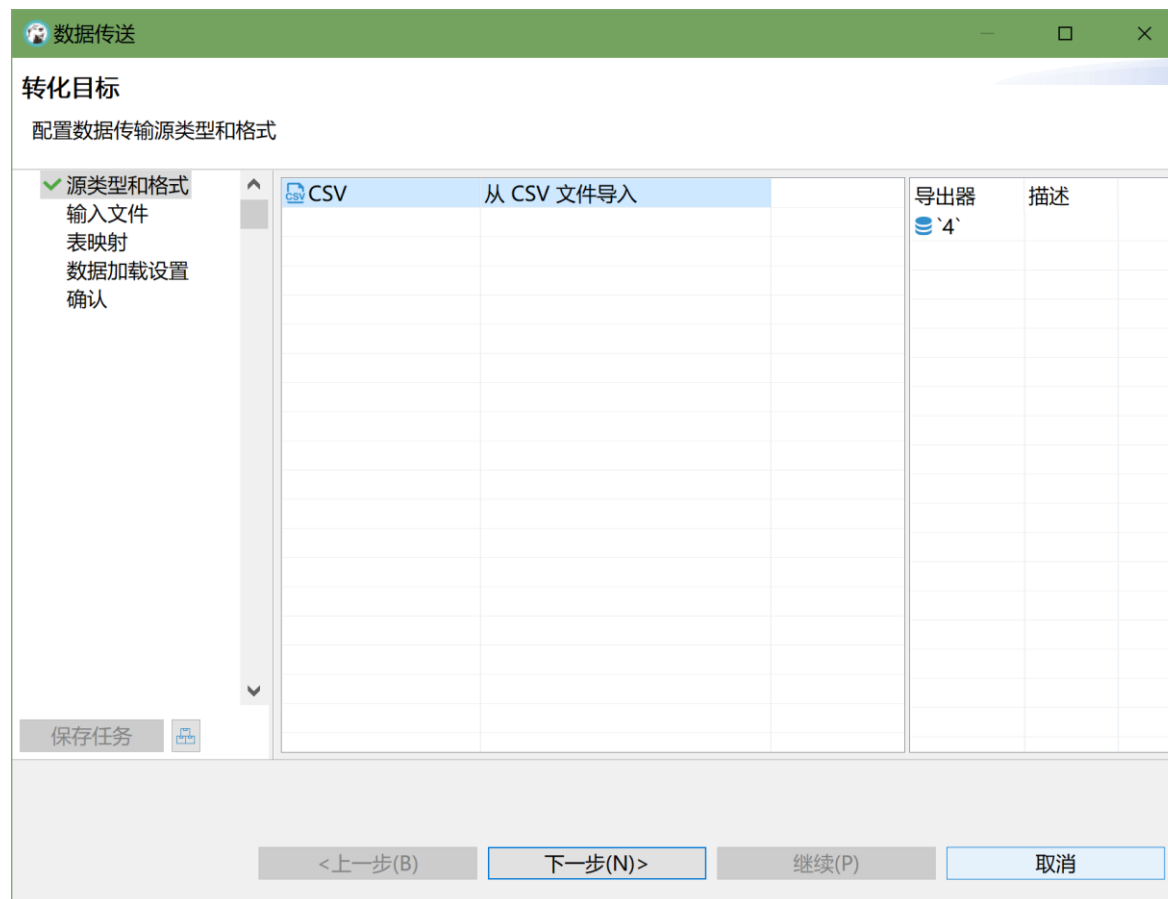
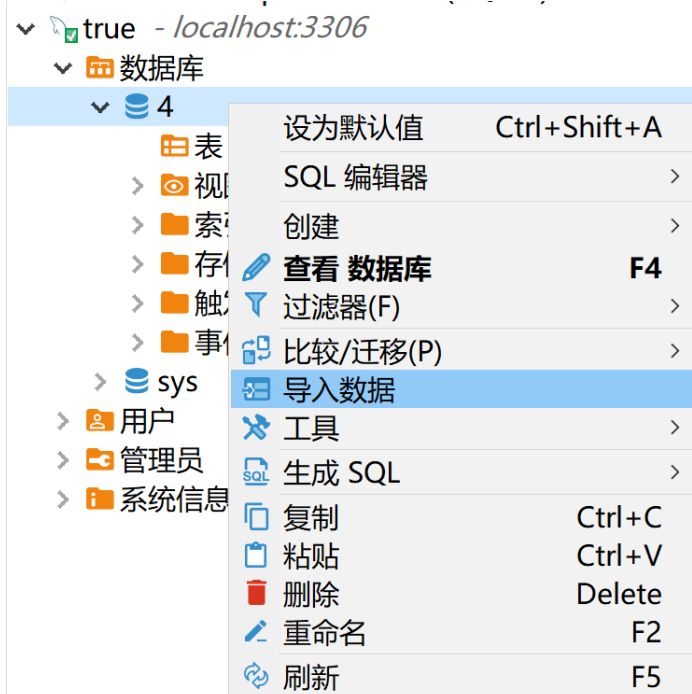
使用mySQL的同学只需右键想要导入的数据库即可（下左图）



# TASK MySQL导入数据

[DBeaver导入Excel数据\\_小小渔夫的博客-CSDN博客](#)

- 1.选中数据库表，右键，然后点击导入数据
- 2.双击CSV,选择待导入的文件



# TASK MySQL导入数据

[DBeaver导入Excel数据\\_小小渔夫的博客-CSDN博客](#)

3.修改编码格式（可选，不乱码不用）

数据传送

输入文件

配置输入文件或目录

✓ 源类型和格式

✓ 输入文件

表映射

数据加载设置

确认

输入文件:

源端	目标端
Customer.csv	?

导入程序设置:

名称	值
▼ 属性	
扩展	csv,tsv,txt
编码	utf-8
列分隔符	,
标题位置	top
引用字符	"
转义字符	\
NULL 值标记	
将空字符串设置为 NULL	[ ]
日期/时间格式	yyyy-MM-dd[ HH:mm:ss[.SSS]]
修剪空白	[ ]
时区ID	
▼ 采样	
列样本数量	100
最小列长	50
列使用字节长度	[ ]

保存任务

<上一步(B)

下一步(N)>

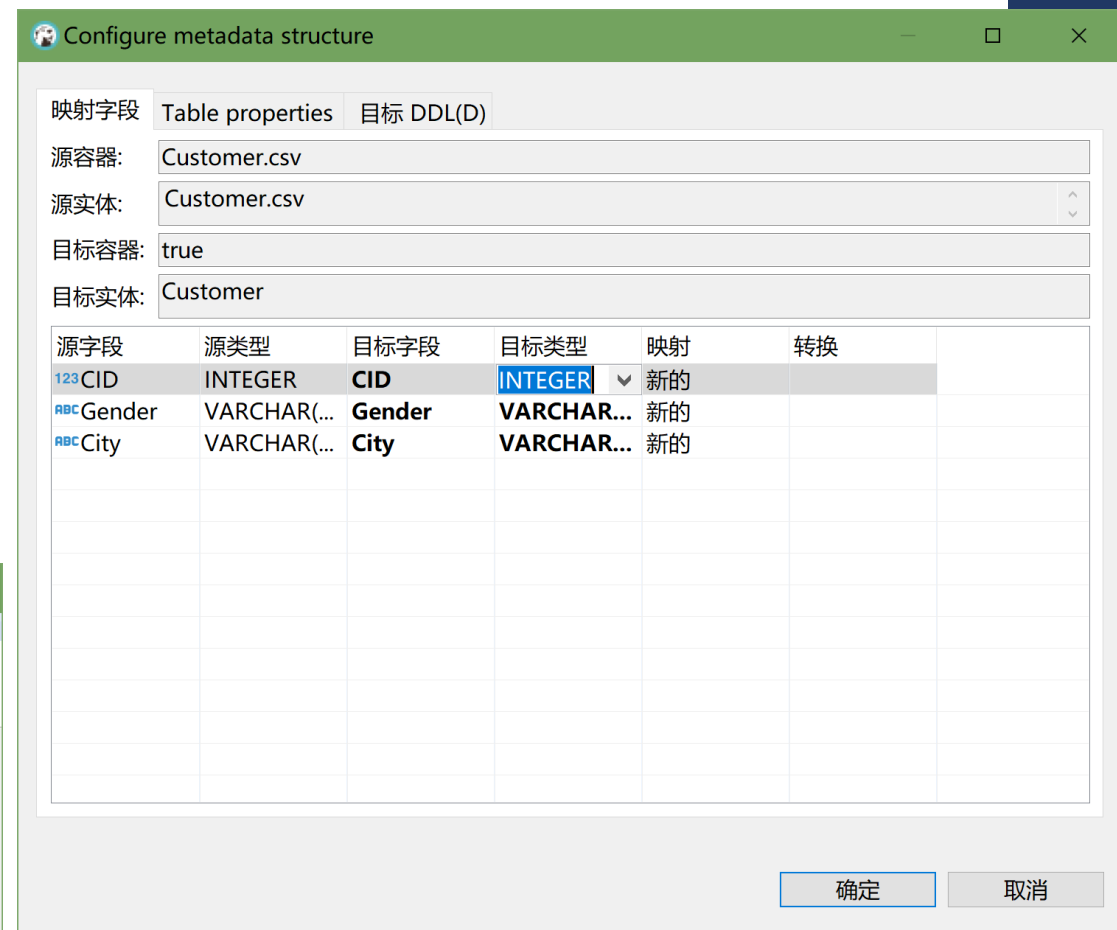
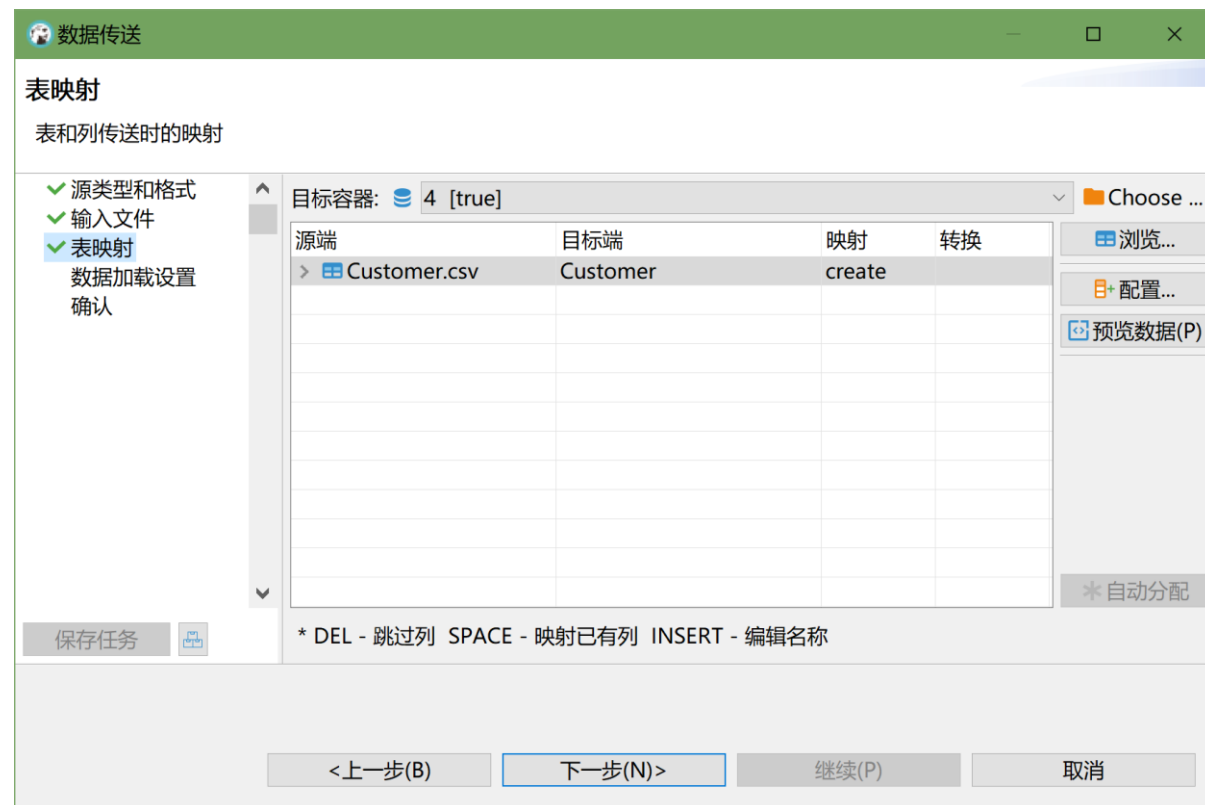
继续(P)

取消

## TASK MySQL导入数据

[DBeaver导入Excel数据\\_小小渔夫的博客-CSDN博客](#)

#### 4.点击配置，修改列的类型



## TASK MySQL导入数据

[DBeaver导入Excel数据\\_小小渔夫的博客-CSDN博客](#)

5.一直下一步，点击继续，等待导入完成即可

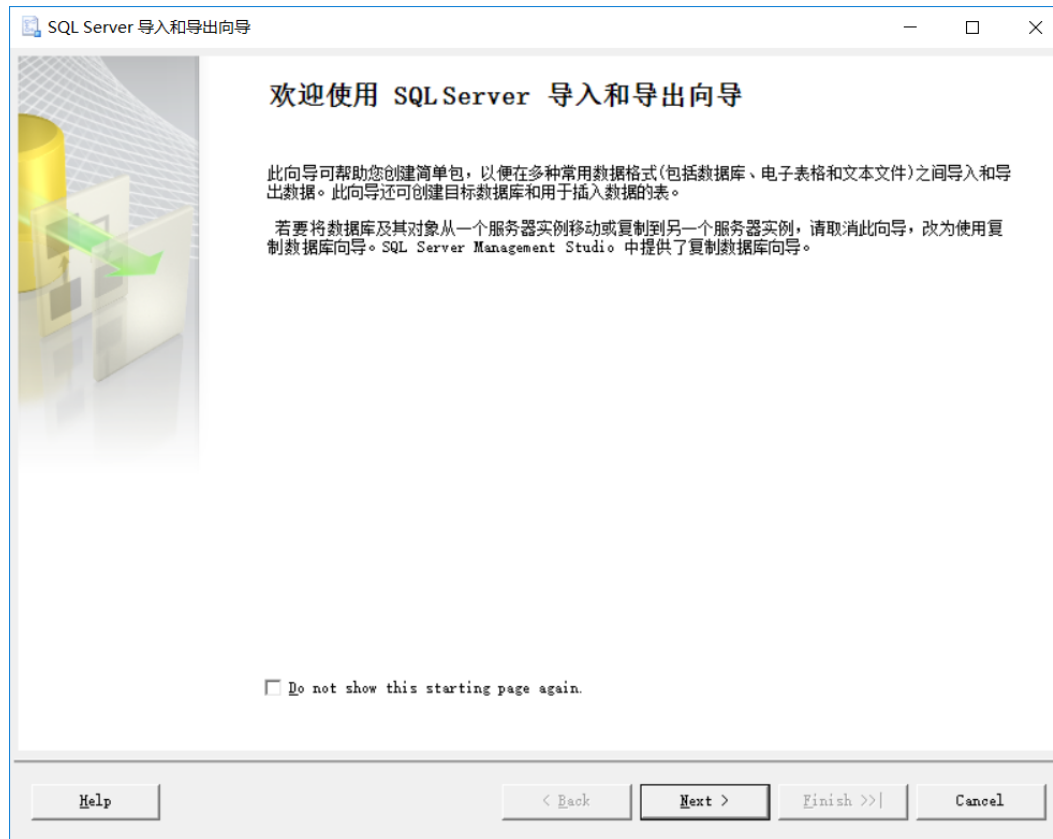
属性 数据 ER 图				
customer 输入一个 SQL 表达式来过滤结果 (使用 Ctrl+Space)				
	123 CID	ABC Gender	ABC City	
1	1	男	北京	
2	2	男	重庆	
3	3	女	北京	
4	4	女	广东	
5	5	女	湖南	
6	6	男	广东	
7	7	男	广东	
8	8	女	江西	
9	9	女	甘肃	
10	10	女	吉林	
11	11	男	游末邦	
12	12	男	水晶都	
13	13	女	水晶都	
14	14	男	游末邦	
15	15	男	水晶都	



# TASK SQL SERVER 导入数据

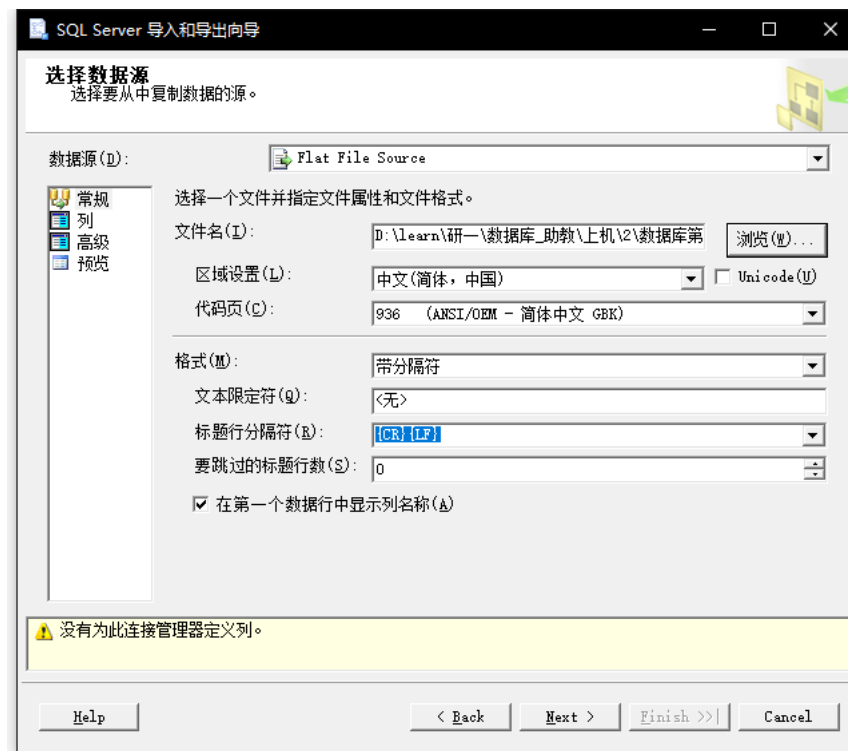
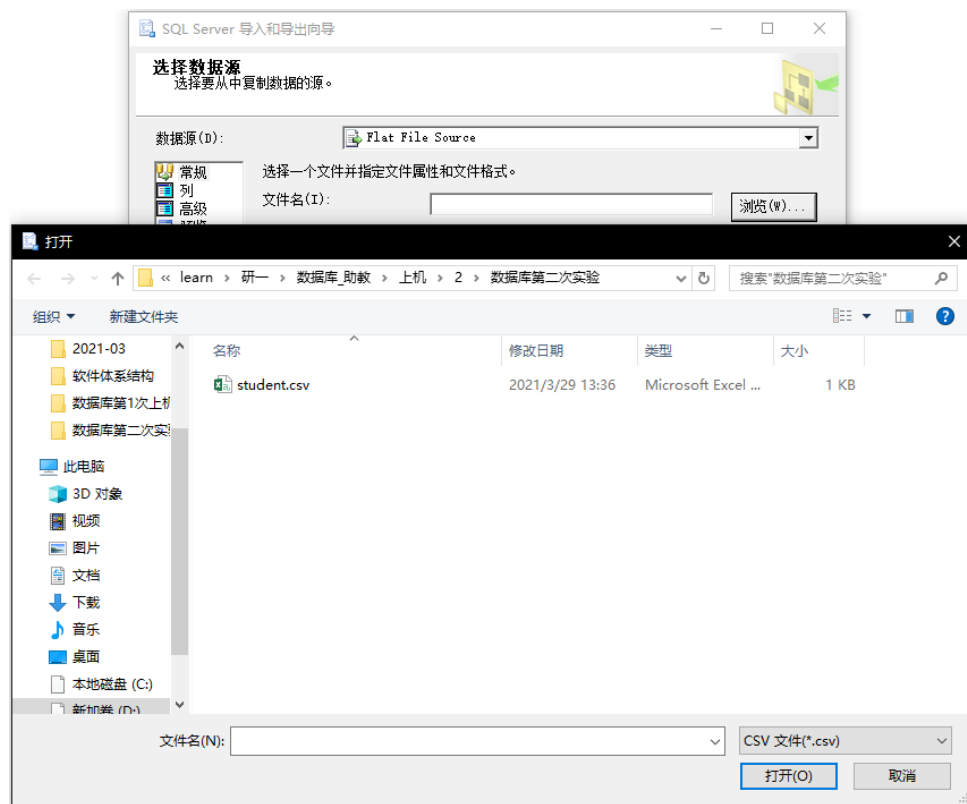
[https://blog.csdn.net/weixin\\_43870646/article/details/90602204](https://blog.csdn.net/weixin_43870646/article/details/90602204)

将准备的csv文件导入数据库



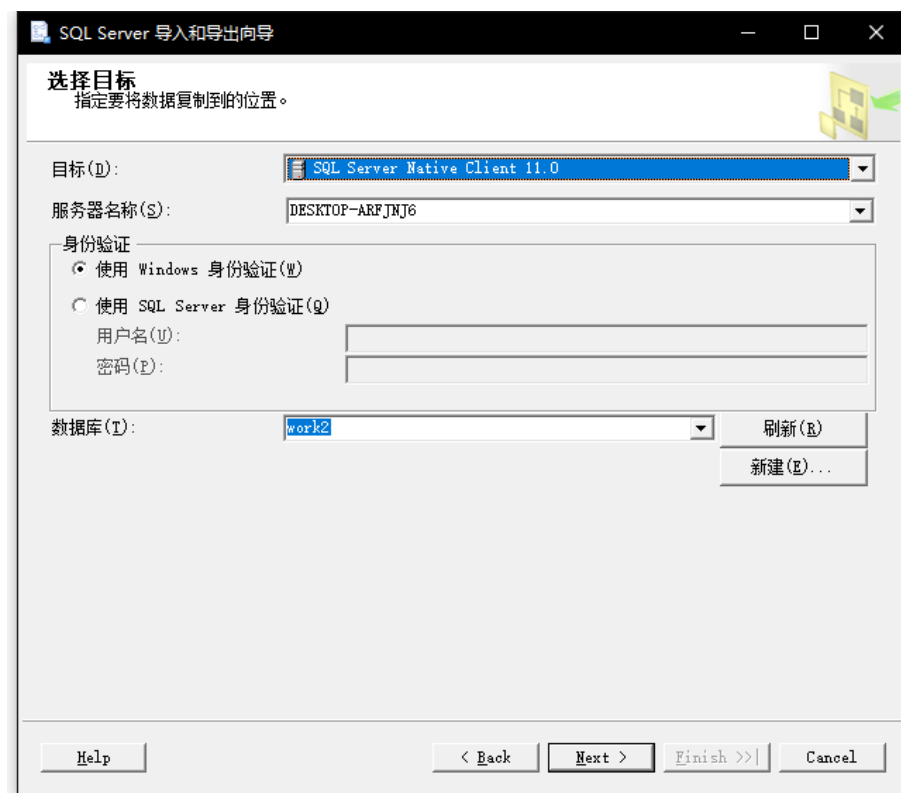
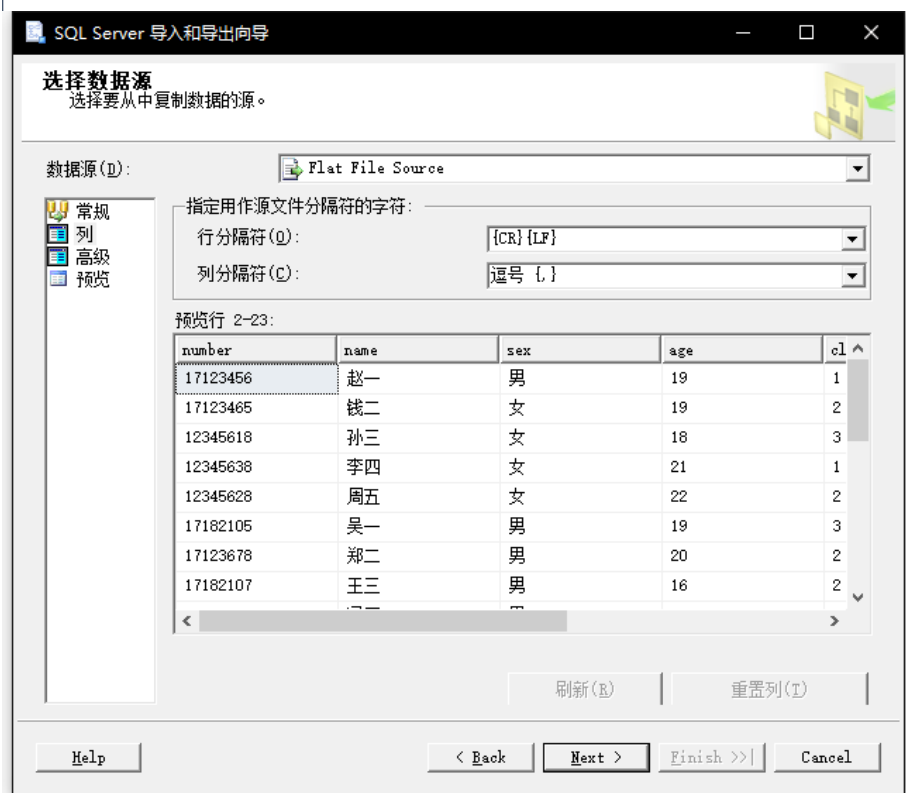
# TASK SQL SERVER 导入数据

数据源选择“Flat File Source”，选择文件名时注意在右下角切换为csv文件，选择完成后点击Next



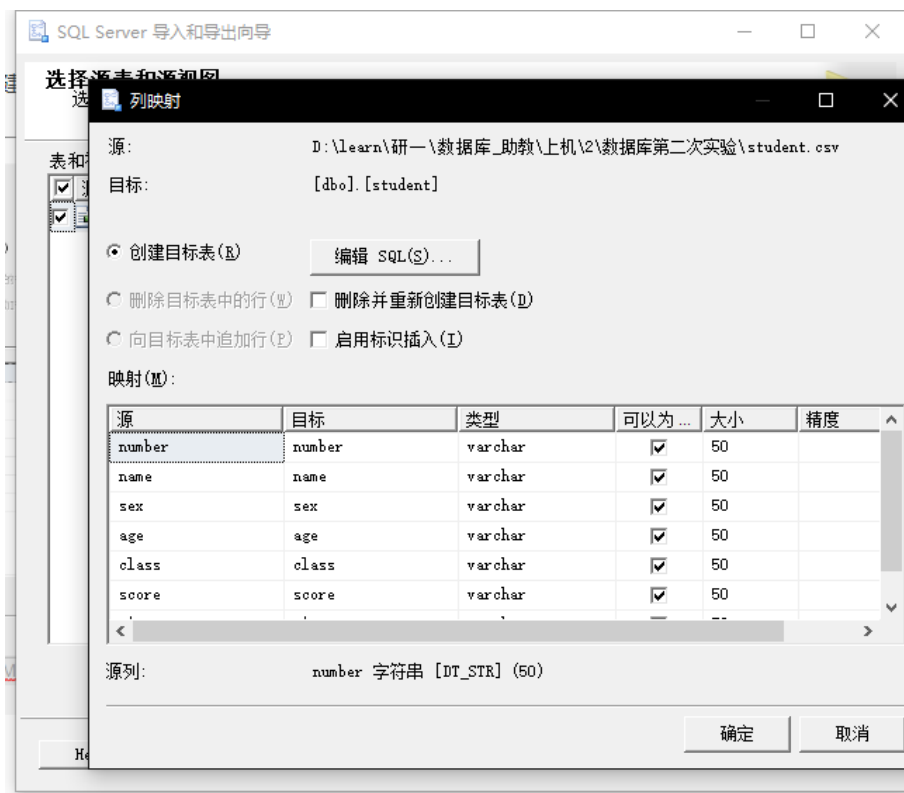
# TASK SQL SERVER 导入数据

上一页完成后进入左图的界面，点击Next，进入右图。目标选择如图，根据个人情况选择服务器名称与数据库（一般来说目标选好后会自动选择），点击Next



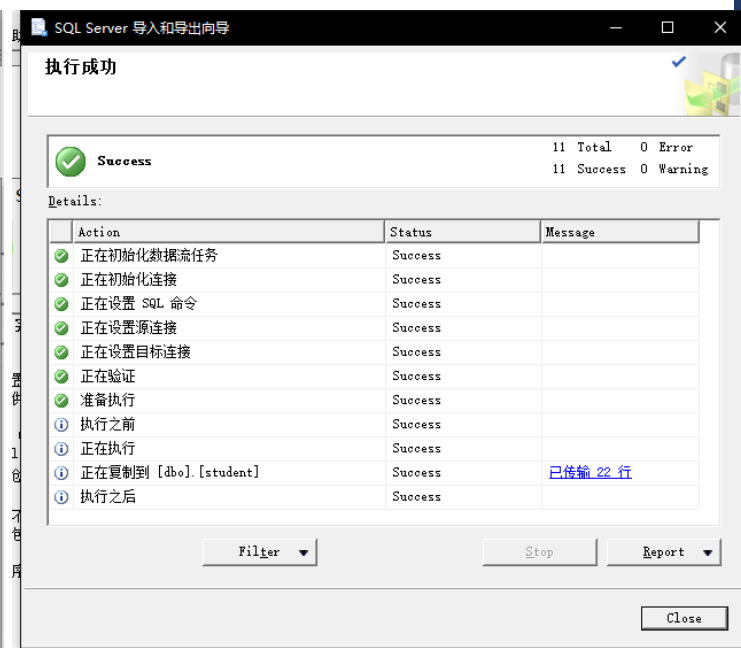
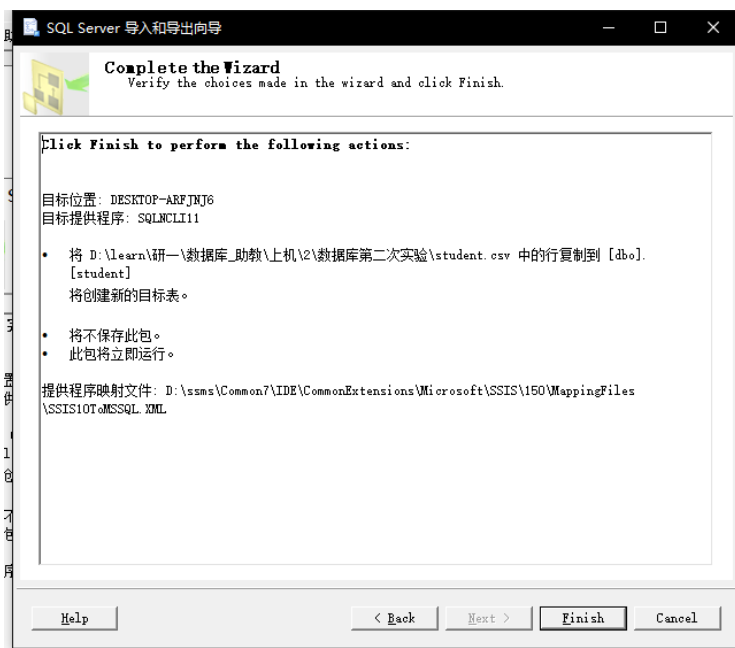
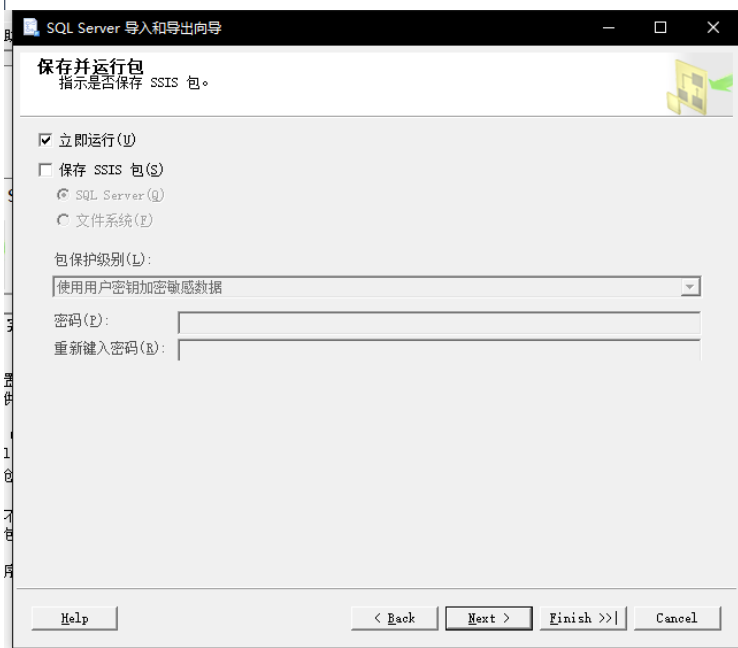
## TASK SQL SERVER 导入数据

如果已经建表直接下一步，没有建表的话，点击编辑映射，可以修改列值类型  
图中类型并未做修改，不可直接模仿



# TASK SQL SERVER 导入数据

然后一直Next，直到Finish



## TASK 2 DDL DML

请使用SQL语句（DDL/DML），基于TASK1的三个表，完成以下需求：

Q1: 创建一个新表Sales，字段为：Food ID（主键） 数字型，食物名（非空）字符型 长度20，总销量 数字型。查询每种食物的总销量，将结果插入表中。（两条SQL语句，一条为create语句，一条为insert语句）。

Q2: 向Order表添加一条交易记录，内容自定义，并更新对应食物的总销量字段（两条SQL语句）

Q3: 为新表添加一个评分字段（数字型），要求分数范围限定为0-10，并设置默认值6

**提交要求：**请在PDF/WORD等任何方便助教阅读查看的文档中完成以下提交内容：需要提交每个查询的SQL语句和查询/操作结果截图，请标好题号。

# SQL语言基础操作

## — DDL

- 修改表

ALTER TABLE 表名

ADD 字段名 字段类型

| DROP COLUMN 字段名

| ALTER COLUMN 字段名 字段类型

| ADD CONSTRAINT 约束名 约束表达式

| DROP CONSTRAINT 约束名

# SQL语言基础操作

## — DML

DML (Data Manipulation Language) 数据操纵语言

INSERT

UPDATE

DELETE

MERGE

CALL

LOCK TABLE

用于：添加、修改、删除、锁定一些数据；调用过程



# SQL语言基础操作

## — DML

- 增加记录

```
INSERT INTO 表名(字段1, 字段2, 字段3)  
VALUES(值1, 值2, 值3);
```

```
INSERT INTO 表名 SELECT...
```

从一个表复制数据，然后把数据插入到一个已存在的表中

- 修改记录

```
UPDATE 表名  
SET 字段1=值1, 字段2=值2, 字段3=值3  
[WHERE 条件表达式];
```

- 删除记录

```
DELETE FROM 表名  
[WHERE 条件表达式];
```

## TASK 3 视图

Task 3. 建立视图，**基于TASK1的三个表**，解决下面的问题

Q1: 建立购买过重庆或四川食物的顾客视图Shu-view (包含Customer中CID, City)

Q2: 查询购买过重庆或四川食物的顾客中订单总消费最高的顾客CID (使用视图Shu-view, 思考使用视图的好处)

Q3: 向视图Shu-view加入表项 (16, 湖南), 能成功吗, 为什么?

Q4: 建立男性顾客的视图Male-view (包含Customer中CID, City), 并要求对该视图进行的更新操作只涉及男性顾客。 (**WITH CHECK OPTION**)

Q5: 向视图Male-view加入表项 (17, 湖南), 能成功吗, 为什么?

**提交要求:** 请在**PDF/WORD**等任何方便助教阅读查看的文档中完成以下提交内容: 需要提交每个查询的**SQL语句和查询结果截图**, 并回答相应的问题, 请标好题号。

# 视图

视图是基于 SQL 语句的结果集的可视化的表；视图看起来和用起来基本都像一个真实的表；而视图中的字段本来就是来自一个或多个数据库中的真实的表中的字段。

本质：

视图是虚表，只存储了SELECT的定义，并没有存储对应的数据；在用户使用视图时先通过定义从基表中搜集数据，再展现给用户。

简而言之，可以理解为执行了一串被保存起来的SELECT语句



# 视图

创建视图:

```
CREATE VIEW 视图名 AS  
    <SELECT语句>;
```

销毁视图

```
DROP VIEW 视图名;
```

例如:

```
CREATE VIEW TopScore AS  
    SELECT IDCourse, MAX(Score)  
    FROM GradeBook  
    GROUP BY IDCourse  
    ORDER BY IDCourse;
```

```
DROP VIEW TopScore;
```

# 视图

使用视图：

一般地，视图支持DQL和DML

DQL：基本无限制

DML（INSERT、UPDATE、DELETE等）：视图**定义**中含有下列语法成分时，DML会受到限制

JOIN、GROUP BY等子句

**DISTINCT**等修饰词

SELECT子句中含有聚合函数

WHERE的子查询引用了FROM句子中的表

# 视图

WITH CHECK OPTION:

为视图添加WITH CHECK OPTION，可以在通过视图进行增删改操作时，不破坏视图定义中的谓词条件（即子查询中的条件表达式）

[例] 建立信息系学生的视图，并要求透过该视图进行的更新操作只涉及信息系学生。

```
CREATE VIEW IS_Student AS  
    SELECT Sno, Sname, Sage  
    FROM Student  
    WHERE Sdept= 'IS'  
    WITH CHECK OPTION;
```

修改操作时：DBMS自动加上Sdept= 'IS'的条件

删除操作时：DBMS自动加上Sdept= 'IS'的条件

插入操作时：DBMS自动检查Sdept属性值是否为'IS'  
如果不是，则拒绝该插入操作  
如果没有提供Sdept属性值，则自动定义Sdept为'IS'

## 关于作业提交

### TASK1 2 3 提交

请在PDF/WORD等任何方便助教阅读查看的文档中按照各个作业要求提交相关内容，记得标清题号。

3. 若为PDF/WORD单文档文件直接提交即可，其他提交压缩包，命名为“**学号\_姓名\_第\*次实验**”。

4. 提交网址：软件学院云平台**第四次上机**[北航软件学院-云平台 \(buaa.edu.cn\)](http://buaa.edu.cn)  
( **按要求提交**)

作业截止时间为**本周日24:00之前**，提交方式为**提交到云平台**。