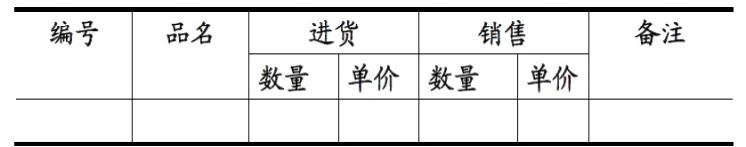
三大范式：

第一范式：符合1NF的关系中的每个属性都不可再分，范式一强调数据表的原子性。



第二范式：2NF在1NF的基础之上，消除了非主属性对于码的部分函数依赖。

学号，课程号，成绩，学生姓名。

部分依赖：当主键由两个或两个以上字段构成，而表中的某些信息通过主键的一个字段就能唯一确定。

第三范式：第三范式（3NF）3NF在2NF的基础之上，消除了非主属性对于码的传递函数依赖。也就是说， 如果存在非主属性对于码的传递函数依赖，则不符合3NF的要求。

在满足第二范式的基础上，在实体中不存在非主键属性传递函数依赖于主键属性。



8、说明：创建索引：create [unique] index idxname on tabname(col….)   
删除索引：drop index idxname  
注：索引是不可更改的，想更改必须删除重新建。  
9、说明：创建视图：create view viewname as select statement   
删除视图：drop view viewname  
10、说明：几个简单的基本的sql语句  
选择：select \* from table1 where 范围  
插入：insert into table1(field1,field2) values(value1,value2)  
删除：delete from table1 where 范围  
更新：update table1 set field1=value1 where 范围  
查找：select \* from table1 where field1 like ’%value1%’ ---like的语法很精妙，查资料!  
排序：select \* from table1 order by field1,field2 [desc]  
总数：select count as totalcount from table1  
求和：select sum(field1) as sumvalue from table1  
平均：select avg(field1) as avgvalue from table1  
最大：select max(field1) as maxvalue from table1  
最小：select min(field1) as minvalue from table1  
11、说明：几个高级查询运算词  
A： UNION 运算符   
UNION 运算符通过组合其他两个结果表（例如 TABLE1 和 TABLE2）并消去表中任何重复行而派生出一个结果表。当 ALL 随 UNION 一起使用时（即 UNION ALL），不消除重复行。两种情况下，派生表的每一行不是来自 TABLE1 就是来自 TABLE2。   
B： EXCEPT 运算符   
EXCEPT 运算符通过包括所有在 TABLE1 中但不在 TABLE2 中的行并消除所有重复行而派生出一个结果表。当 ALL 随 EXCEPT 一起使用时 (EXCEPT ALL)，不消除重复行。   
C： INTERSECT 运算符  
INTERSECT 运算符通过只包括 TABLE1 和 TABLE2 中都有的行并消除所有重复行而派生出一个结果表。当 ALL 随 INTERSECT 一起使用时 (INTERSECT ALL)，不消除重复行。   
注：使用运算词的几个查询结果行必须是一致的。   
12、说明：使用外连接   
A、left （outer） join：   
左外连接（左连接）：结果集几包括连接表的匹配行，也包括左连接表的所有行。   
SQL: select a.a, a.b, a.c, b.c, b.d, b.f from a LEFT OUT JOIN b ON a.a = b.c  
B：right （outer） join:   
右外连接(右连接)：结果集既包括连接表的匹配连接行，也包括右连接表的所有行。   
C：full/cross （outer） join：   
全外连接：不仅包括符号连接表的匹配行，还包括两个连接表中的所有记录。  
12、分组:Group by:  
   一张表，一旦分组 完成后，查询后只能得到组相关的信息。  
    组相关的信息：（统计信息） count,sum,max,min,avg  分组的标准)  
    在SQLServer中分组时：不能以text,ntext,image类型的字段作为分组依据  
   在selecte统计函数中的字段，不能和普通的字段放在一起；