

数据库 day02 随堂笔记

今日内容

- 数据库的查询语句[重点]
 - 基本查询(不带条件(where))
 - 条件查询(where)

SQL 语句

查询语句

基本查询

```
-- 需求1: 准备商品数据, 查询所有数据, 查询部分字段, 起字段别名, 去重
-- 查询所有数据: select * from 表名;
select * from goods;
-- 查询部分字段: select 字段名1, 字段名2 from goods;
select goodsName, price from goods;
-- 起字段别名: select 字段名 as '别名' from goods;
select goodsName as '商品名称', price as '价格' from goods;
-- 注意: 别名的引号可以省略
select goodsName as 商品名称, price as 价格 from goods;
-- 注意: as 关键字也可以省略[掌握]
select goodsName 商品名称, price 价格 from goods;
-- 起别名的作用: 1> 美化数据结果的显示效果 2> 可以起到隐藏真正字段名的作用
-- 另: 除了可以给字段起别名以外, 还可以给数据表起别名(连接查询时使用)
```

```
-- 去重: select distinct(字段名) from goods;  
-- 效果: 将目标字段内重复出现的数据只保留一份显示  
-- 小需求: 显示所有的公司名称  
select distinct(company) from goods;
```

条件查询

比较运算符&逻辑运算符

```
-- 需求2: 查询价格等于30并且出自并夕夕的所有商品信息  
select * from goods;  
-- 查询价格等于30 : 比较运算符(特殊: (大于等于)>=/(小于等于)<=/(不等于)!=/<>)  
select * from goods where price=30;  
-- 并且出自并夕夕的所有商品信息 : 逻辑运算符(and(与)/or(或)/not(非))  
-- 注意: 作为查询条件使用的字符串必须带引号!  
select * from goods where price=30 and company='并夕夕';  
-- 补充需求: 查询价格等于30但不出自并夕夕的所有商品信息  
select * from goods where not company='并夕夕' and price=30;  
-- 注意: not 与 and 和 or (左右两边连接条件)不同之处在于, not 只对自己右侧的条件有作用(右边连接条件)  
select * from goods where price=30 and not company='并夕夕';
```

模糊查询

```
-- 需求3：查询全部一次性口罩的商品信息
-- 模糊查询：like 和符号 %(任意多个字符)/_(任意一个字符)
-- 注意：作为查询条件使用的字符串必须带引号！
-- 注意：如果需要控制字符数量，需要使用_，并且有几个字符就使用几个_
-- %关键词% ：关键词在中间
select * from goods where remark like '%一次性%';
-- %关键词 ：关键词在末尾
select * from goods where remark like '%一次性';
-- 关键词% ：关键词在开头
select * from goods where remark like '一次性%';
```

范围查询

```
-- 需求4：查询所有价格在30-100的商品信息
-- 范围查询：1> 非连续范围：in 2> 连续范围：between ... and ...
select * from goods where price between 30 and 100;
-- 注意：between ... and ... 的范围必须是从小到大
select * from goods where price between 100 and 30;
```

判断空

```
-- 需求5：查询没有描述信息的商品信息
-- 注意：在 MySQL 中，只有显示为 NULL 的才为空！其余空白可能是空格/制表符(tab)/换行符(回车键)等空白符号
-- 判断空：1> 为空：is null 2> 不为空(双重否定表肯定)：is not null
select * from goods where remark is null;
-- 补充需求：查询有描述信息的所有商品
select * from goods where remark is not null;
```

其他复杂查询

排序

```
-- 需求6：查询所有商品信息，按照价格从大到小排序，价格相同时，按照数量少到多排序
-- select * from 表名 order by 列1 asc|desc,列2 asc|desc,...
-- 说明：order by 排序，asc：升序，desc：降序
-- 注意：排序过程中，支持连续设置多条排序规则，但离 order by 关键字越近，排序数据的范围越大！
select * from goods order by price desc;
select * from goods order by price desc, count asc;
-- 注意：默认排序为升序，asc 可以省略
select * from goods order by price desc, count;
```

聚合函数

```
-- 需求7：查询以下信息：商品信息总条数；最高商品价格；最低商品价格；商品平均价格；一次性口罩的总数量
-- 聚合函数：系统提供的一些可以直接用来获取统计数据的函数
-- 商品信息总条数：count(字段)：查询总记录数
select count(*) from goods;
-- 注意：统计数据总数，建议使用*，如果使用某一特定字段，可能会造成数据总数错误！
select count(remark) from goods;
-- 最高商品价格：max(字段)：查询最大值
select max(price) from goods;
-- 最低商品价格：min(字段)：查询最小值
select min(price) from goods;
-- 商品平均价格：avg(字段)：求平均值
select avg(price) from goods;
-- 一次性口罩的总数量：sum()：求和
```

```
-- 注意：此处的 count 是数据表中字段名！
select sum(count) from goods where remark like '%一次性%';
-- 扩展：在需求允许的情况下，可以一次性在一条 SQL语句中，使用所有的聚合函数
select count(*), max(price), min(price), avg(price) from goods;
```

分组

```
-- 需求8：查询每家公司的商品信息数量
-- 分组：select 字段1,字段2,聚合... from 表名 group by 字段1,字段2...
-- 说明：group by : 分组
-- 注意：
-- 1> 一般情况，使用哪个字段进行分组，那么只有该字段可以在 * 的位置处使用，其他字段没有实际意义(只要一组数据中的一条)
-- 2> 分组操作多和聚合函数配合使用
select count(*) from goods group by company;
select * from goods;
select company, count(*) from goods group by company;
-- 说明：其他字段没有实际意义(只要一组数据中的一条)
select price, count(*) from goods group by company;

-- 扩充：分组后条件过滤
-- 说明：group by 后增加过滤条件时，需要使用 having 关键字
-- 注意：
-- 1. group by 和 having 一般情况下需要配合使用
-- 2. group by 后边不推荐使用 where 进行条件过滤
-- 3. having 关键字后侧可以使用的内容与 where 完全一致(比较运算符/逻辑运算符/模糊查询/判断空)
-- 3. having 关键字后侧允许使用聚合函数

-- where 和 having 的区别：
```

```
-- where 是对 from 后面指定的表进行数据筛选，属于对原始数据的筛选
-- having 是对 group by 的结果进行筛选
-- having 后面的条件中可以用聚合函数，where 后面不可以
```

分页查询

```
-- 需求9：查询当前表当中第5-10行的所有数据
-- 分页查询：select * from 表名 limit start,count
-- 说明：limit 分页；start：起始行号；count：数据行数
-- 注意：计算机的计数从 0 开始，因此 start 默认的第一条数据应该为 0，
后续数据依次减1
-- 过渡需求：获取前 5 条数据
select * from goods limit 0, 5;
-- 注意：如果默认从第一条数据开始获取，则 0 可以省略！
select * from goods limit 5;
-- 需求：
select * from goods limit 4, 6;

-- 扩展 1：根据公式计算显示某页的数据
-- 已知：每页显示m条数据，求：显示第n页的数据
-- select * from 表名 limit (n-1)*m, m
-- 示例：每页显示 4 条数据，求展示第 2 页的数据内容
select * from goods limit 0, 4; -- 第1页(有数据)
select * from goods limit 4, 4; -- 第2页(有数据)
select * from goods limit 8, 4; -- 第3页(有数据)
select * from goods limit 12, 4; -- 第4页(一共 12 条数据，每页显示
4 条，没有第 4 页数据)

-- 扩展 2：分页的其他应用
-- 需求：要求查询商品价格最贵的数据信息
select * from goods order by price desc limit 1;
-- 进阶需求：要求查询商品价格最贵的前三条数据信息
```

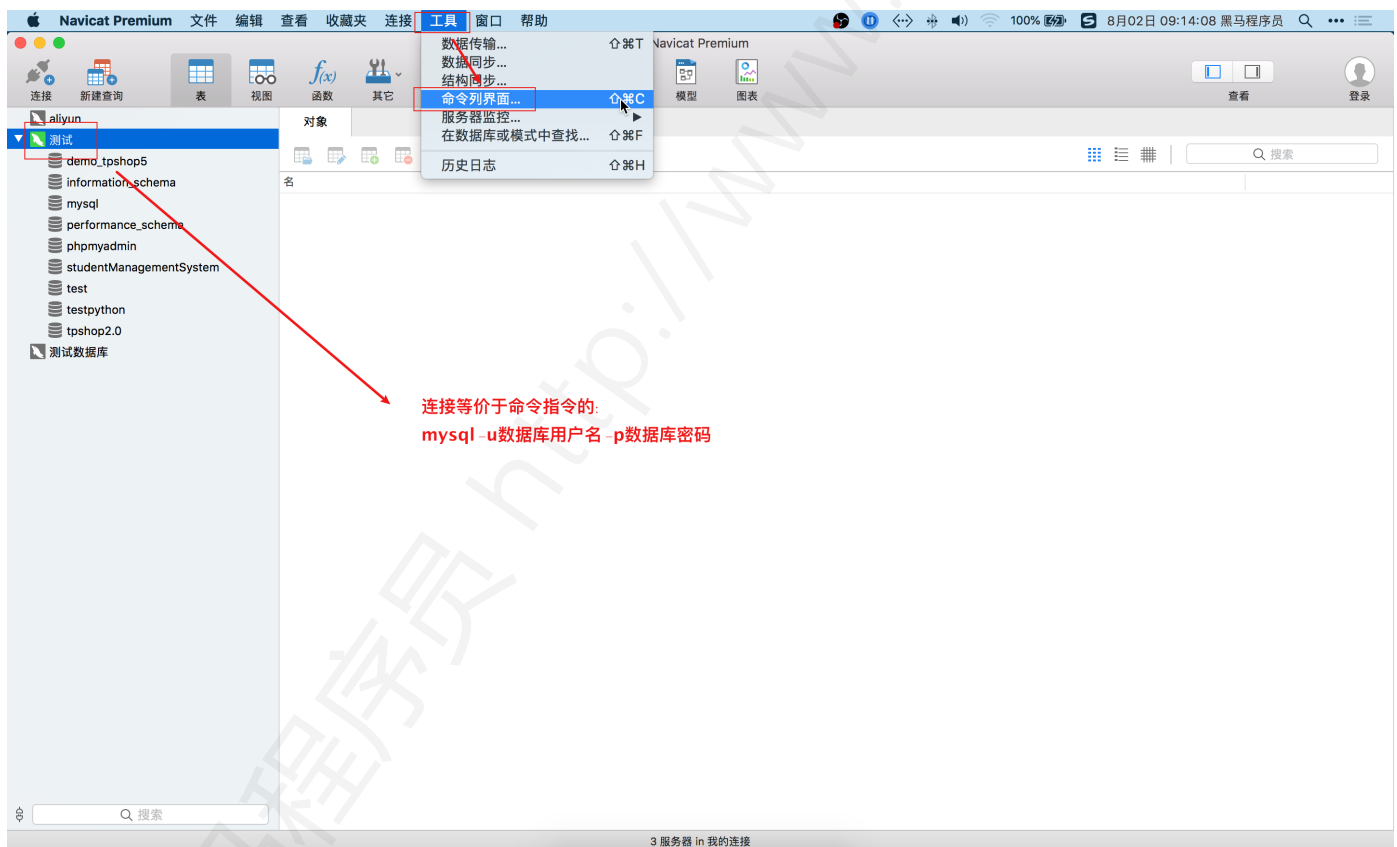
```
select * from goods order by price desc limit 3;
```

今日任务要求

- 语法不过关的 -> 优先处理课上案例(课件里的或上课案例)
- 语法还行的 -> 处理课下练习[自行安排 -> 出了本阶段也可以继续]

回顾: 命令行操作数据库的基本步骤

使用 Navicat 工具中的命令列



命令行基本操作步骤

命令列界面 - MySQL ()

mysql> show databases;

Database
demo_tpshop5
information_schema
mysql
performance_schema
phpmyadmin
studentManagementSystem
test
testpython
tpshop2.0

9 rows in set (0.00 sec)

mysql> use test;

Database changed

mysql> select database();

database()
test

1 row in set (0.00 sec)

mysql> show tables;

Tables_in_test
stu
students

2 rows in set (0.00 sec)

mysql> select * from stu;

Empty set

mysql> select * from students;

studentNo	name	sex	hometown	age	class	card
001	王昭君	女	北京	20	1班	340322199001247654
002	诸葛亮	男	上海	18	2班	340322199002242354
003	张飞	男	南京	24	3班	340322199003247654
004	白起	男	安徽	22	4班	340322199005247654
005	大乔	女	天津	19	3班	340322199004247654
006	孙尚香	女	河北	18	1班	340322199006247654
007	百里玄策	男	山西	20	2班	340322199007247654
008	小乔	女	河南	15	3班	NULL
009	百里守约	男	湖南	21	1班	
010	妲己	女	广东	26	2班	340322199607247654
011	李白	男	北京	30	4班	340322199005267754
012	孙膑	男	新疆	26	3班	340322199000297655

12 rows in set (0.00 sec)

查看所有数据库

使用数据库(切换数据库)

查看当前数据库名称

查看所有表

查询语句

desc 表名
查看表结构