

一、查询案例

1. 创建数据表
2. 插入数据

二、SQL查询演练

1. SQL语句的强化
2. 创建 "商品分类" 表
3. 同步表数据
4. 创建 "商品品牌表" 表
5. 同步数据
6. 修改表结构
7. 外键

一、查询案例

1. 创建数据表

```
1  -- 创建 "京东" 数据库
2  create database jing_dong charset=utf8;
3
4  -- 使用 "京东" 数据库
5  use jing_dong;
6
7  -- 创建一个商品goods数据表
8  create table goods(
9      id int unsigned primary key auto_increment not null,
10     name varchar(150) not null,
11     cate_name varchar(40) not null,
12     brand_name varchar(40) not null,
13     price decimal(10,3) not null default 0,
14     is_show bit not null default 1,
15     is_saleoff bit not null default 0
16 );
```

2. 插入数据

```
1  -- 向goods表中插入数据
2
3  insert into goods values(0,'r510vc 15.6英寸笔记本','笔记本','华
4  硕','3399',default,default);
5  insert into goods values(0,'y400n 14.0英寸笔记本电脑','笔记本','联
6  想','4999',default,default);
7  insert into goods values(0,'g150th 15.6英寸游戏本','游戏本','雷
8  神','8499',default,default);
9  insert into goods values(0,'x550cc 15.6英寸笔记本','笔记本','华
10  硕','2799',default,default);
```

```
7 insert into goods values(0,'x240 超极本','超级本','联想','4880',default,default);
8 insert into goods values(0,'u330p 13.3英寸超极本','超级本','联想','4299',default,default);
9 insert into goods values(0,'svp13226scb 触控超极本','超级本','索尼','7999',default,default);
10 insert into goods values(0,'ipad mini 7.9英寸平板电脑','平板电脑','苹果','1998',default,default);
11 insert into goods values(0,'ipad air 9.7英寸平板电脑','平板电脑','苹果','3388',default,default);
12 insert into goods values(0,'ipad mini 配备 retina 显示屏','平板电脑','苹果','2788',default,default);
13 insert into goods values(0,'ideacentre c340 20英寸一体电脑','台式机','联想','3499',default,default);
14 insert into goods values(0,'vostro 3800-r1206 台式电脑','台式机','戴尔','2899',default,default);
15 insert into goods values(0,'imac me086ch/a 21.5英寸一体电脑','台式机','苹果','9188',default,default);
16 insert into goods values(0,'at7-7414lp 台式电脑 linux )','台式机','宏碁','3699',default,default);
17 insert into goods values(0,'z220sff f4f06pa工作站','服务器/工作站','惠普','4288',default,default);
18 insert into goods values(0,'poweredge ii服务器','服务器/工作站','戴尔','5388',default,default);
19 insert into goods values(0,'mac pro专业级台式电脑','服务器/工作站','苹果','28888',default,default);
20 insert into goods values(0,'hmq-t3w 头戴显示设备','笔记本配件','索尼','6999',default,default);
21 insert into goods values(0,'商务双肩背包','笔记本配件','索尼','99',default,default);
22 insert into goods values(0,'x3250 m4机架式服务器','服务器/工作站','ibm','6888',default,default);
23 insert into goods values(0,'商务双肩背包','笔记本配件','索尼','99',default,default);
```

二、SQL查询演练

1. SQL语句的强化

- 查询类型cate_name为 '超极本' 的商品名称、价格

```
1 select name,price from goods where cate_name = '超极本';
```

- 显示商品的种类

```
1 select cate_name from goods group by cate_name;
```

- 求所有电脑产品的平均价格,并且保留两位小数

```
1 select round(avg(price),2) as avg_price from goods;
```

- 显示每种商品的平均价格

```
1 select cate_name,avg(price) from goods group by cate_name;
```

- 查询每种类型的商品中 最贵、最便宜、平均价、数量

```
1 select cate_name,max(price),min(price),avg(price),count(*) from  
goods group by cate_name;
```

- 查询所有价格大于平均价格的商品, 并且按价格降序排序

```
1 select id,name,price from goods  
2 where price > (select round(avg(price),2) as avg_price from goods)  
3 order by price desc;
```

- 查询每种类型中最贵的电脑信息

```
1 select * from goods  
2 inner join  
3 (  
4     select  
5     cate_name,  
6     max(price) as max_price,  
7     min(price) as min_price,  
8     avg(price) as avg_price,  
9     count(*) from goods group by cate_name  
10 ) as goods_new_info  
11 on goods.cate_name=goods_new_info.cate_name and  
goods.price=goods_new_info.max_price;
```

2. 创建 "商品分类" 表

```
1 -- 创建商品分类表  
2 create table if not exists goods_cates(  
3     id int unsigned primary key auto_increment,  
4     name varchar(40) not null  
5 );
```

- 查询goods表中商品的种类

```
1 select cate_name from goods group by cate_name;
```

- 将分组结果写入到goods_cate数据表

```
1 insert into goods_cate (name) select cate_name from goods group by  
cate_name;
```

3. 同步表数据

- 通过goods_cate数据表来更新goods表

```
1 update goods as g inner join goods_cate as c on g.cate_name=c.name  
set g.cate_name=c.id;
```

4. 创建 "商品品牌表" 表

- 通过create...select来创建数据表并且同时写入记录,一步到位

```
1 -- select brand_name from goods group by brand_name;  
2  
3 -- 在创建数据表的时候一起插入数据  
4 -- 注意: 需要对brand_name 用as起别名, 否则name字段就没有值  
5 create table goods_brands (  
6     id int unsigned primary key auto_increment,  
7     name varchar(40) not null) select brand_name as name from goods  
group by brand_name;
```

5. 同步数据

- 通过goods_brands数据表来更新goods数据表

```
1 update goods as g inner join goods_brands as b on  
g.brand_name=b.name set g.brand_name=b.id;
```

6. 修改表结构

- 查看 goods 的数据表结构,会发现 cate_name 和 brand_name对应的类型为 `varchar` 但是存储的都是数字

```
1 desc goods;
```

- 通过alter table语句修改表结构

```
1 alter table goods  
2 change cate_name cate_id int unsigned not null,  
3 change brand_name brand_id int unsigned not null;
```

7. 外键

- 分别在 goods_cates 和 goods_brands 表中插入记录

```
1 insert into goods_cates(name) values ('路由器'),('交换机'),('网卡');
2 insert into goods_brands(name) values ('海尔'),('清华同方'),('神舟');
```

- 在 goods 数据表中写入任意记录

```
1 insert into goods (name,cate_id,brand_id,price)
2 values('LaserJet Pro P1606dn 黑白激光打印机', 12, 4, '1849');
3
```

- 查询所有商品的详细信息 (通过内连接)

```
1 select g.id,g.name,c.name,b.name,g.price from goods as g
2 inner join goods_cates as c on g.cate_id=c.id
3 inner join goods_brands as b on g.brand_id=b.id;
4
```

- 查询所有商品的详细信息 (通过左连接)

```
1 select g.id,g.name,c.name,b.name,g.price from goods as g
2 left join goods_cates as c on g.cate_id=c.id
3 left join goods_brands as b on g.brand_id=b.id;
4
```

- 如何防止无效信息的插入,就是可以在插入前判断类型或者品牌名称是否存在呢? 可以使用之前讲过的外键来解决
- 外键约束:对数据的有效性进行验证
- 关键字: foreign key,只有 innodb数据库引擎 支持外键约束
- 对于已经存在的数据表 如何更新外键约束

```
1 -- 给brand_id 添加外键约束成功
2 alter table goods add foreign key (brand_id) references
   goods_brands(id);
3 -- 给cate_id 添加外键失败
4 -- 会出现1452错误
5 -- 错误原因:已经添加了一个不存在的cate_id值12,因此需要先删除
6 alter table goods add foreign key (cate_id) references
   goods_cates(id);
7
8
```

- 如何在创建数据表的时候就设置外键约束呢?
- 注意: goods 中的 cate_id 的类型一定要和 goods_cates 表中的 id 类型一致

```
1 create table goods(  
2     id int primary key auto_increment not null,  
3     name varchar(40) default '',  
4     price decimal(5,2),  
5     cate_id int unsigned,  
6     brand_id int unsigned,  
7     is_show bit default 1,  
8     is_saleoff bit default 0,  
9     foreign key(cate_id) references goods_cates(id),  
10    foreign key(brand_id) references goods_brands(id)  
11 );  
12
```

- 如何取消外键约束

```
1 -- 需要先获取外键约束名称,该名称系统会自动生成,可以通过查看表创建语句来获取名称  
2 show create table goods;  
3 -- 获取名称之后就可以根据名称来删除外键约束  
4 alter table goods drop foreign key 外键名称;  
5
```

- 在实际开发中,很少会使用到外键约束,会极大的降低表更新的效率