

- MusicShop报告
 - 查询
 - 1.查找收藏歌曲在2首及以上的用户及其收藏的歌曲数
 - 2.查找所包含的歌曲数大于等于10首的专辑，显示专辑名和所包含的歌曲数
 - 3.查找最近10年来（now（））的专辑销售情况，列出专辑ID、专辑名称、总销售额，按总销售从高到低排名。
 - 4.查询没有被收藏过的歌曲
 - 5.在购买了歌手刘欢专辑的客户中查询一次购买数量最多的客户的用户名
 - 6.查找被所有用户收藏的歌曲（两种实现方式：相关嵌套查询、不相关嵌套查询）
 - 7.查找一首歌曲都没有收藏的用户（两种实现方式：相关嵌套查询、连接查询）
 - 实验中遇到的问题与解决方法
 - 查询中没有对应数据
 - 编入数据

MusicShop报告

查询

1.查找收藏歌曲在2首及以上的用户及其收藏的歌曲数

```
-- 查找收藏歌曲在2首及以上的用户及其收藏的歌曲数
select UserName, count(SongID) as 收藏数
from collections
group by UserName
HAVING 收藏数 >= 2;
```

| | UserName | 收藏数 |
|---|----------|-----|
| 1 | wt | 3 |
| 2 | wyh | 6 |

2.查找所包含的歌曲数大于等于10首的专辑，显示专辑名和所包含的歌曲数

-- 查找所包含的歌曲数大于等于10首的专辑，显示专辑名和所包含的歌曲数

```
select AlbumName, count(SongNumber) as 歌曲数
from albums, songs
where albums.AlbumID = songs.AlbumID
group by AlbumName
HAVING 歌曲数 >= 10;
```

| | AlbumName | 歌曲数 |
|---|-----------|-----|
| 1 | F.ONE | 10 |

3.查找最近10年来（now（））的专辑销售情况，列出专辑ID、专辑名称、总销售额，按总销售从高到低排名。

-- 查找最近10年来（now（））的专辑销售情况，列出专辑ID、专辑名称、总销售额，按总销售从高到低排名。

```
select sales.AlbumID, albums.AlbumName, sum(sales.TotalPrice) as total
from albums, sales
where sales.AlbumID = albums.AlbumID
group by albums.AlbumID
order by total desc ;
```

| | AlbumID | AlbumName | total |
|---|---------|-----------|--------|
| 1 | 19 | 好好好 | 880.00 |
| 2 | 18 | 世纪好歌 | 720.00 |
| 3 | 2 | F.ONE | 255.00 |
| 4 | 5 | 三十而立 | 200.00 |
| 5 | 17 | 最炫民族风 | 80.00 |

4.查询没有被收藏过的歌曲

-- 查询没有被收藏过的歌曲

```
select *
from songs
where SongID not in (select SongID from collections);
```

| | SongID | SongNumber | AlbumID | SongTitle | SongDuration | SongContent | SongUploadDate |
|---|--------|------------|---------|-----------|--------------|-------------|------------------|
| 2 | 7 | 2 | 2 | 想知道现在你好不好 | 0:4:10 | 暂无 | 2009-07-10 00:00 |
| 3 | 8 | 3 | 2 | 起风 | 0:4:10 | 暂无 | 2009-07-10 00:00 |
| 4 | 9 | 4 | 2 | 灰色的彩虹 | 0:4:10 | 暂无 | 2009-07-10 00:00 |
| 5 | 10 | 5 | 2 | 傻的可以 | 0:4:10 | 暂无 | 2009-07-10 00:00 |
| 6 | 11 | 6 | 2 | 对不起 | 0:4:10 | 暂无 | 2009-07-10 00:00 |
| 7 | 12 | 7 | 2 | 别再生了 | 0:4:10 | 暂无 | 2009-07-10 00:00 |
| 8 | 13 | 8 | 2 | 没把握 | 0:4:10 | 暂无 | 2009-07-10 00:00 |
| 9 | 14 | 9 | 2 | 爱的盲点 | 0:4:10 | 暂无 | 2009-07-10 00:00 |

5.在购买了歌手刘欢专辑的客户中查询一次购买数量最多的客户的用户名

-- 在购买了歌手刘欢专辑的客户中查询一次购买数量最多的客户的用户名

```
SELECT orders.username
FROM orders
WHERE OrderID IN (
    SELECT sales.OrderID
    FROM sales
    WHERE sales.AlbumID IN (
        SELECT albums.AlbumID
        FROM albums
        WHERE AlbumSinger = '刘欢'
    ) AND sales.Quantity IN (
        SELECT MAX(sales.Quantity)
        FROM sales
        WHERE sales.AlbumID IN (
            SELECT albums.AlbumID
            FROM albums
            WHERE AlbumSinger = '刘欢'
        )
    )
);
```

| username |
|----------|
| 1 wyh |

6. 查找被所有用户收藏的歌曲（两种实现方式：相关嵌套查询、不相关嵌套查询）

The screenshot shows a SQL IDE with two queries and their results.

Query 1: 不相关嵌套 (Non-correlated Subquery)

```
select songs.SongTitle
from songs
where SongID in (select SongID
                 from collections
                 group by SongID
                 having count(Username) = (select count(Username)
                                           from users));
```

Result 1:

| SongTitle |
|-----------|
| 1 微笑说再见 |

Query 2: 相关嵌套 (Correlated Subquery)

```
SELECT s.SongID, s.SongTitle
FROM Songs s
WHERE NOT EXISTS (
    SELECT u.UserName
    FROM Users u
    WHERE NOT EXISTS (
        SELECT c.SongID
        FROM Collections c
        WHERE c.UserName = u.UserName AND c.SongID = s.SongID
    )
);
```

Result 2:

| SongID | SongTitle |
|--------|-----------|
| 1 | 15 微笑说再见 |

7. 查找一首歌曲都没有收藏的用户（两种实现方式：相关嵌套查询、连接查询）

-- 查找一首歌曲都没有收藏的用户（两种实现方式：相关嵌套查询、连接查询）

-- 连接查询

```
select users.UserName
from users
left join collections on users.UserName = collections.UserName
where collections.SongID is NULL;
```

| UserName | |
|----------|---|
| 1 喆King | > |

-- 嵌套查询

```
SELECT u.UserName
FROM Users u
WHERE NOT EXISTS (
    SELECT c.UserName
    FROM Collections c
    WHERE c.UserName = u.UserName
);
```

| UserName | |
|----------|---|
| 1 喆King | > |

实验中遇到的问题与解决方法

查询中没有对应数据

编入数据

-- 创造假数据

```
insert into albums(AlbumName, AlbumIssueCompany, AlbumIssueDate, AlbumType,
AlbumIntroduce, AlbumImageUrl, AlbumSinger, AlbumLanguage, AlbumMarketPrice,
AlbumMemberPrice, AlbumIsRecommend)
values ('好好好', '金大福', '2009-07-01', 8, '暂无', '无', '刘欢', '华语', 100, 80,
true);
```

```
INSERT Sales (OrderID, AlbumID, Quantity, TotalPrice) VALUES (5, 19, 1, 80.00);
```

```
INSERT Orders (OrderDate, UserName, GoodsFee, DeliverFee, DeliverID, AreaID,
PaymentID, ReceiverName, ReceiverAddress, ReceiverPostCode, ReceiverPhone,
ReceiverEmail, OrderIsPayment, OrderIsConsignment, OrderIsConfirm,
OrderIsPigeonhole)
```

```
VALUES ('2020-12-04', 'wyh', 80.00, 4.00, 1, 1, 1, '李晓', '北京市 朝阳区',
'100024', '65758696', NULL, 1, 1, 1, 1);
```

```
INSERT Sales (OrderID, AlbumID, Quantity, TotalPrice) VALUES (6, 19, 10, 800.00);
```

```
INSERT Orders (OrderDate, UserName, GoodsFee, DeliverFee, DeliverID, AreaID,
PaymentID, ReceiverName, ReceiverAddress, ReceiverPostCode, ReceiverPhone,
ReceiverEmail, OrderIsPayment, OrderIsConsignment, OrderIsConfirm,
OrderIsPigeonhole)
```

```
VALUES ('2021-12-04', 'wyh', 800.00, 4.00, 1, 1, 1, '李晓', '北京市 朝阳区',
```

```
'100024', '65758696', NULL, 1, 1, 1, 1);
```

```
-- 继续编数据
```

```
INSERT INTO Collections VALUES(15,'wt','2014-05-01');
```

```
INSERT INTO Collections VALUES(15,'wyh','2016-05-01');
```

```
INSERT INTO Collections VALUES(15,'丫丫','2017-05-01');
```

```
INSERT INTO Collections VALUES(15,'小龙女','2018-05-01');
```

```
INSERT INTO Collections VALUES(15,'郭大侠','2019-05-01');
```

```
INSERT INTO
```

```
Users(Username,UserPassword,UserSex,UserRealName,UserAgeRange,UserAddress,UserPostCode,UserPhone,UserEmail)
```

```
VALUES('喆King','666','女','sj','无可奉告','北京市 朝阳区','100024','13600088855','woshinvtong@cuc.edu.cn');
```