

# 实验 嵌套查询/基于感动中国年度人物和应急预案数据库

## —弘扬中华传统美德 共建和谐美好社会

### 1. 实验目的

- (1) 掌握 SELECT、WHERE、WITH 和 FROM 等子句嵌套的多种嵌套查询。
- (2) 弘扬中华传统美德 共建和谐美好社会

### 2. 实验环境

- (1) 硬件：

处理器：Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz 2.11 GHz

机带 RAM：8.00 GB (7.55 GB 可用)

系统类型：64 位操作系统，基于 x64 的处理器

- (2) 操作系统：Windows 10

- (3) 数据库管理系统：MySQL 8.0 MySQL workbench

### 3. 实验内容

- (1) 使用 SELECT 子句嵌套、FROM 子句嵌套、WHERE 子句嵌套、WITH 子句嵌套等实现嵌套查询。

### 4. 实验数据

(1) 本次实验使用“感动中国年度人物”数据库，主要涉及人物获奖表 award，包括获奖编号 (aid)，获奖称号 (atitle)，获奖人姓名 (aname)，获奖人性别 (asex)，获奖时的年龄 (aage)，获奖年份 (ayear)，颁奖辞 (description)。

(2) 应急预案指面对突发事件如自然灾害、重特大事故、环境公害及人为破坏的应急管理、指挥、救援计划等，是一种公文。通常一个应急预案由多个不同的编制单位协同编写，才能编制完成。应急预案包含预案编号 (plan\_id)，预案名 (plan\_name)，针对的灾害类型 (plan\_disatype)，针对的区域 (plan\_area)，针对的灾害等级 (plan\_level)，发布时间 (plan\_date)。应急预案编制的参与单位包含单位编号 (depart\_id)，单位名称 (depart\_name)，单位联系方式 (depart\_tel)。一个参与单位可能参与多个预案的编制，一个预案需要多个参与单位协作完成。当参与单位完成编写应急预案时，会记录该单位在应急预案编制中的职责 (depart\_respon) 和工作量 (workload)。

### 5. 实验作业

- (1) 登录 MySQL，用 SQL 语句创建数据表 award，并插入数据，展示表的结构和表中数据的当前状态。

表 award 结构

属性	描述	类型	长度
aid	获奖编号	varchar	20
atitle	获奖称号	varchar	20
aname	获奖人姓名	varchar	20
asex	获奖人性别	varchar	4

aage	获奖时的年龄	smallint	
ayear	获奖年份	smallint	
description	颁奖辞	text	

表 award 中记录

aid	atitle	aname	asex	aage	ayear	description
2018001	临危岂顾生	杜富国	男	27	2018	你退后/让我来/六个字铁骨铮铮/以血肉挡住危险/哪怕自己坠入深渊/无法还给妈妈一个拥抱/无法再见妻子明媚的笑脸/战友们拉着手趟过雷场/你听/那嘹亮的军歌/是对英雄的礼赞
2018002	见义勇为为	吕保民	男	49	2018	NULL
2018003	孤云心浩然	王仕花	女	56	2018	浪的执着/礁的顽强/民的本分/兵的责任/岛再小也是国土/家未立也要国先安/三十二年驻守/三代人无言付出/两百面旗帜收藏了太多风雨/涛拍孤岛岸/风颂赤子心
2017001	天意怜幽草， 人间重晚晴	卢永根	男	87	2017	种得桃李满天下，心唯大我育青禾。是春风，是春蚕，更化作护花的春泥。热爱祖国，你要把自己燃烧。稻谷有根，深扎在泥土，你也有根 扎根在人们心里。
2017002	际会中兴日， 四海两心知	卢丽安	女	49	2017	台湾的女儿有大气概，祖国为大乡愁不改，把握现在开创未来。分离再久，改不了我们的血脉。海峡再深，挡不住人民追求福祉的路。
2017003	芳兰生贵里， 山河澄正气	廖俊波	男	49	2017	人民的樵夫，不忘初心。上山寻路，扎实工作，廉洁奉公，牢记党的话，温暖群众的心。春茶记住你的目光，青山留下你的足迹，谁把人民扛在肩上，人民就把谁装进心里。
2016001	星斗焕文章	孙家栋	男	87	2016	少年勤学，青年担纲，你是国家的栋梁。导弹、卫星、嫦娥、北斗，满天星斗璀璨，写下你的传奇。年过古稀未伏枥，犹向苍穹寄深情。
2016002	芳兰振蕙叶	支月英	女	55	2016	你跋涉了许多路，总是围绕大

						山。吃了很多苦，但给孩子们的都是甜。坚守才有希望，这是你的信念。三十六年，绚烂了两代人的童年，花白了你的麻花辫。
2016003	君子通大道	秦玥飞	男	31	2016	NULL

Step 1.输入 SQL 语句创建“感动中国年度人物”数据库，并创建数据表 award，然后向数据表 award 中插入数据。如下图：

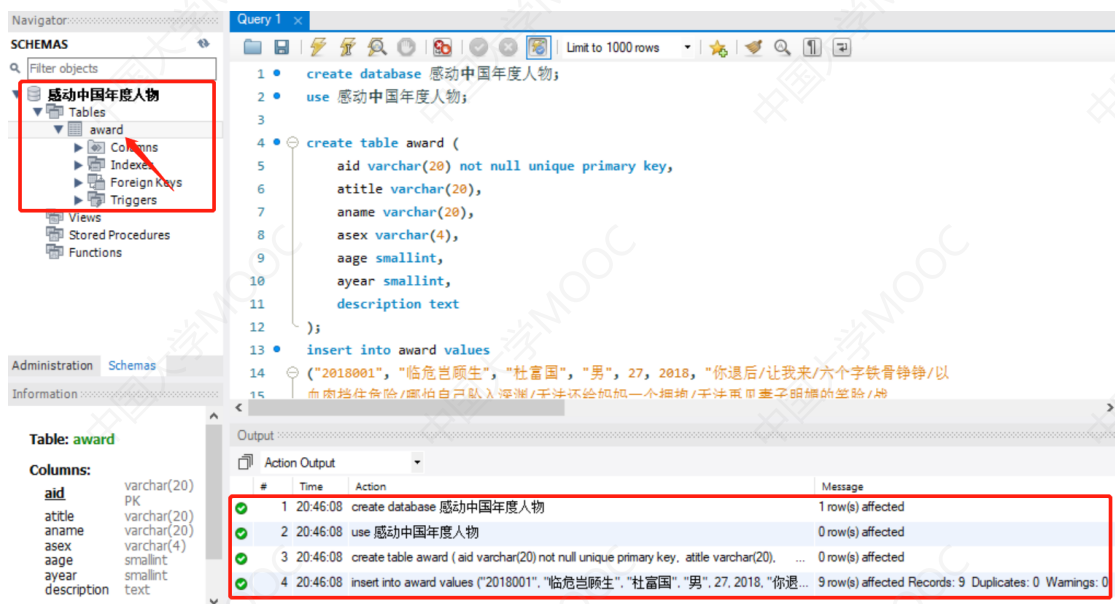
```

• create database 感动中国年度人物;
• use 感动中国年度人物;

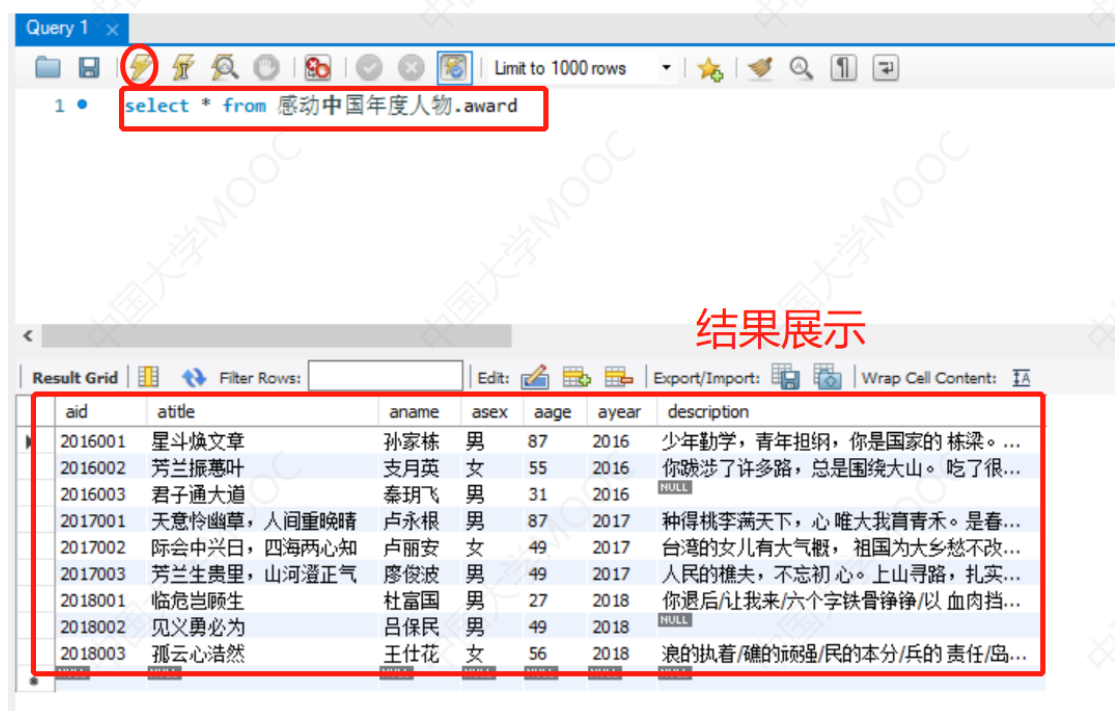
• create table award (
    aid varchar(20) not null unique primary key,
    atitle varchar(20),
    aname varchar(20),
    asex varchar(4),
    aage smallint,
    ayear smallint,
    description text
);
• insert into award values
    ("2018001", "临危岂顾生", "杜富国", "男", 27, 2018, "你退后/让我来/六个字铁骨铮铮/以血肉挡住危险/哪怕自己坠入深渊/无法还给妈妈一个拥抱/无法再见妻子明媚的笑脸/战友们拉着手趟过雷场/你听/那嘹亮的军歌/是对英雄的礼赞"),
    ("2018002", "见义勇为为", "吕保民", "男", 49, 2018, NULL),
    ("2018003", "孤云心浩然", "王仕花", "女", 56, 2018, "浪的执着/礁的顽强/民的本分/兵的责任/岛再小也是国土/家未立也要国先安/三十二年驻守/三代人无言付出/两百年旗帜收藏了太多风雨/涛拍孤岛岸/风颂赤子心"),
    ("2017001", "天意怜幽草，人间重晚晴", "卢永根", "男", 87, 2017, "种得桃李满天下，心唯大我育青禾。是春风，是春蚕，更化作护花的春泥。热爱祖国，你要把自己燃烧。稻谷有根，深扎在泥土；你也有根，那是对人民的爱。")

```

Step 2.结果如图。运行成功，表 award 已出现。



Step 3.输入 SQL 语句对刚刚设置好的表进行展示。



代码：

```
create database 感动中国年度人物;
use 感动中国年度人物;
create table award (
    aid varchar(20) not null unique primary key,
    atitle varchar(20),
    aname varchar(20),
```

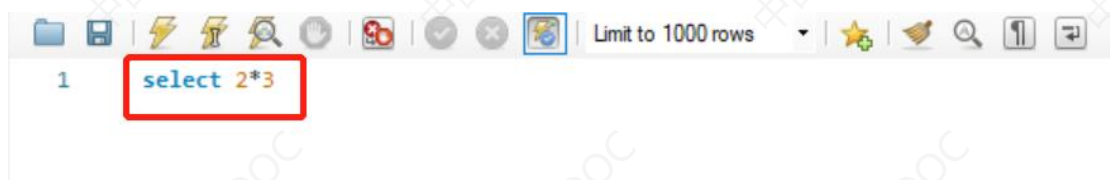
```

    asex varchar(4),
    aage smallint, ayear smallint,
    description text
);
insert into award values
("2018001", "临危岂顾生", "杜富国", "男", 27, 2018, "你退后/让我来/六个字铁骨铮铮/以
血肉挡住危险/哪怕自己坠入深渊/无法还给妈妈一个拥抱/无法再见妻子明媚的笑脸/战
友们拉着手趟过雷场/你听/那嘹亮的军歌/是对英雄的礼赞"),
("2018002", "见义勇为为", "吕保民", "男", 49, 2018, NULL),
("2018003", "孤云心浩然", "王仕花", "女", 56, 2018, "浪的执着/礁的顽强/民的本分/兵
的责任/岛再小也是国土/家未立也要国先安/三十二年驻守/三代人无言付出/两百面旗帜收
藏了太多风雨/涛拍孤岛岸/风颂赤子心"),
("2017001", "天意怜幽草，人间重晚晴", "卢永根", "男", 87, 2017, "种得桃李满天下，心
唯大我育青禾。是春风，是春蚕，更化作护花的春泥。热爱祖国，你要把自己燃烧。
稻谷有根，深扎在泥土，你也有根 扎根在人们心里。"),
("2017002", "际会中兴日，四海两心知", "卢丽安", "女", 49, 2017, "台湾的女儿有大气
概，祖国为大乡愁不改，把握现在开创未来。分离再久，改不了我们的血脉。海峡再
深，挡不住人民追求福祉的路。"),
("2017003", "芳兰生贵里，山河澄正气", "廖俊波", "男", 49, 2017, "人民的樵夫，不忘初
心。上山寻路，扎实工作，廉洁奉公，牢记党的话，温暖群众的心。春茶记住你的目
光，青山留下你的足迹，谁把人民扛在肩上，人民就把谁装进心里。"),
("2016001", "星斗焕文章", "孙家栋", "男", 87, 2016, "少年勤学，青年担纲，你是国家的
栋梁。导弹、卫星、嫦娥、北斗，满天星斗璀璨，写下你的传奇。年过古稀未伏枥，
犹向苍穹寄深情。"),
("2016002", "芳兰振蕙叶", "支月英", "女", 55, 2016, "你跋涉了许多路，总是围绕大山。
吃了很多苦，但给孩子们的都是甜。坚守才有希望，这是你的信念。三十六年，绚烂
了两代人的童年，花白了你的麻花辫。"),
("2016003", "君子通大道", "秦玥飞", "男", 31, 2016, null);

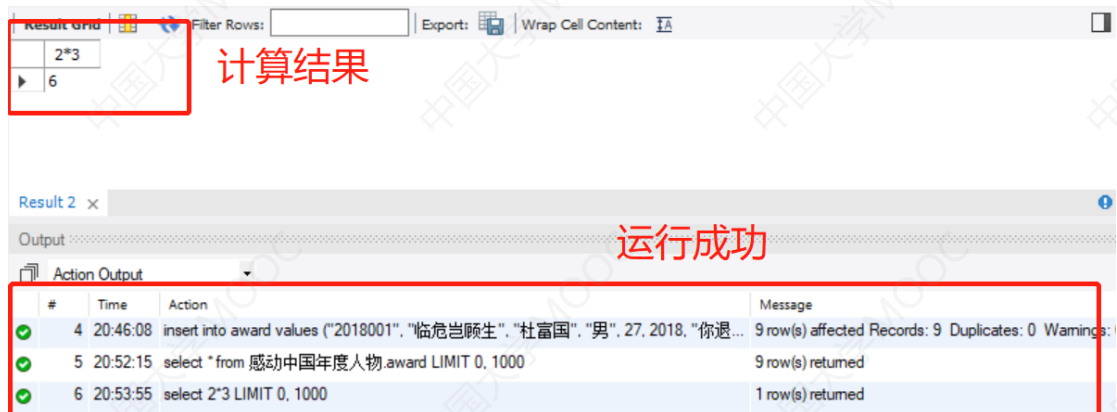
```

(2) 计算 `select 2*3`。

Step 1. 输入 SQL 语句进行计算。



Step 2. 运行结果如图。

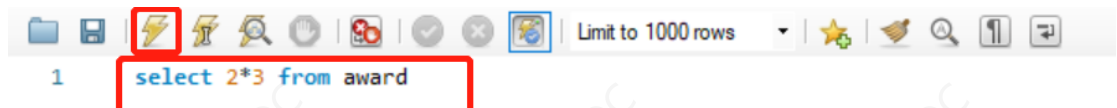


代码:

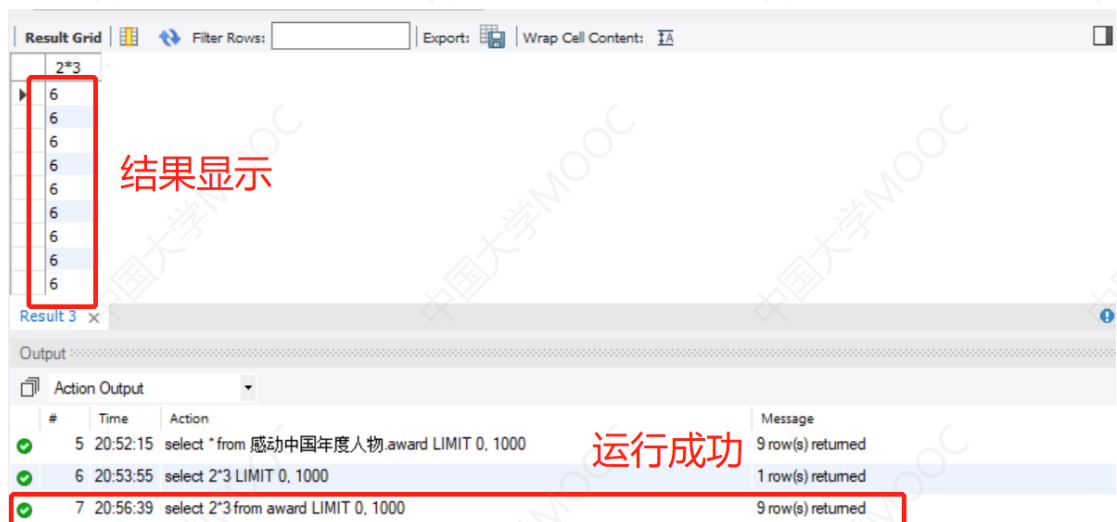
```
select 2 * 3;
```

(3) 计算  $select\ 2 \times 3\ from\ award$ 。

Step 1. 输入 SQL 语句进行计算。



Step 2. 运行结果如图。



代码:

```
select 2 * 3 from award;
```

(4) 使用 SQL 语句查询同年份获奖人的平均年龄。

Step 1. 输入下列 SQL 语句进行查询。



```
1 select ayear , avg(aage)
2 from award
3 group by ayear
4 order by ayear
```

Step 2. 运行结果如图。



运行结果

ayear	avg(aage)
2016	57.6667
2017	61.6667
2018	44.0000

执行成功

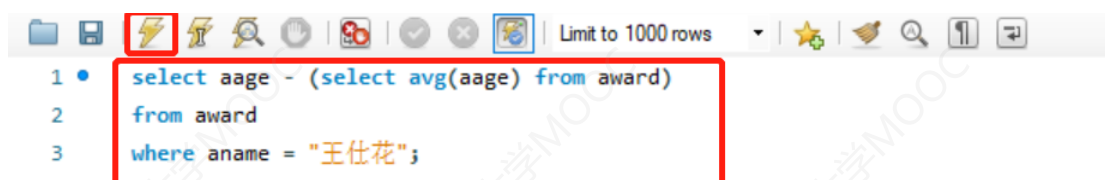
#	Time	Action	Message
1	20:59:39	select ayear , avg(aage) from award group by ayear order by ayear LIMIT 0, 1000	3 row(s) returned

代码：

```
select ayear, avg(aage)
from award
group by ayear
order by ayear;
```

(5) 使用 select 子句嵌套查询王仕花与获奖人平均年龄差。

Step 1. 输入下列 SQL 语句进行嵌套查询。



```
1 select aage - (select avg(aage) from award)
2 from award
3 where aname = "王仕花";
```

Step 2. 运行结果如图。



Result Grid

	aage - (select avg(aage) from award)
1.5556	

查询结果

Result 6 x

Output

Action Output

执行成功

#	Time	Action	Message
1	20:59:39	select ayear , avg(aage) from award group by ayear order by ayear LIMIT 0, 1000	3 row(s) returned
2	21:02:28	select aage - (select avg(aage) from award) from award where aname = "王仕花" LIMIT...	1 row(s) returned

代码:

```
select aage - (select avg(aage) from award)
from award
where aname = "王仕花";
```

(6) 使用 select 子句嵌套查询所有获奖人与平均年龄差。

Step 1. 输入下列 SQL 语句进行嵌套查询。

Limit to 1000 rows

```
1 • select aid,aname,aage - (select avg(aage) from award)
2 from award;
```

Step 2. 运行结果如图。

Result Grid

	aid	aname	aage - (select avg(aage) from award)
2016001	孙家栋	32.5556	
2016002	支月英	0.5556	
2016003	秦玥飞	-23.4444	
2017001	卢永根	32.5556	
2017002	卢丽安	-5.4444	
2017003	廖俊波	-5.4444	
2018001	杜富国	-27.4444	
2018002	吕保民	-5.4444	
2018003	王仕花	1.5556	

查询结果

Result 7 x

Output

Action Output

执行成功

#	Time	Action	Message
1	20:59:39	select ayear , avg(aage) from award group by ayear order by ayear LIMIT 0, 1000	3 row(s) returned
2	21:02:28	select aage - (select avg(aage) from award) from award where aname = "王仕花" LIMIT...	1 row(s) returned
3	21:04:19	select aid,aname,aage - (select avg(aage) from award) from award LIMIT 0, 1000	9 row(s) returned

代码:



```
select aid, aname, aage - (select avg(aage) from award)
from award;
```

(7) 使用 select 子句嵌套查询 award 表中杜富国与吕保民的年龄差。

Step 1. 输入下列 SQL 语句进行嵌套查询。



```
1 select
2 (select aage from award where aname = "杜富国")
3 -
4 (select aage from award where aname = "吕保民");
```

Step 2. 运行结果如图。



查询结果

执行成功

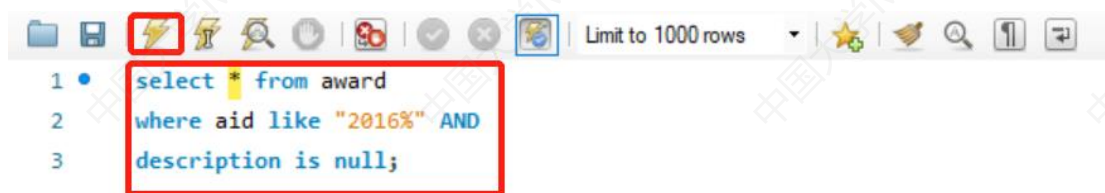
Time	Action	Message
1 21:07:58	select (select aage from award where aname = "杜富国") - (select aage from award wh...	1 row(s) returned

代码：

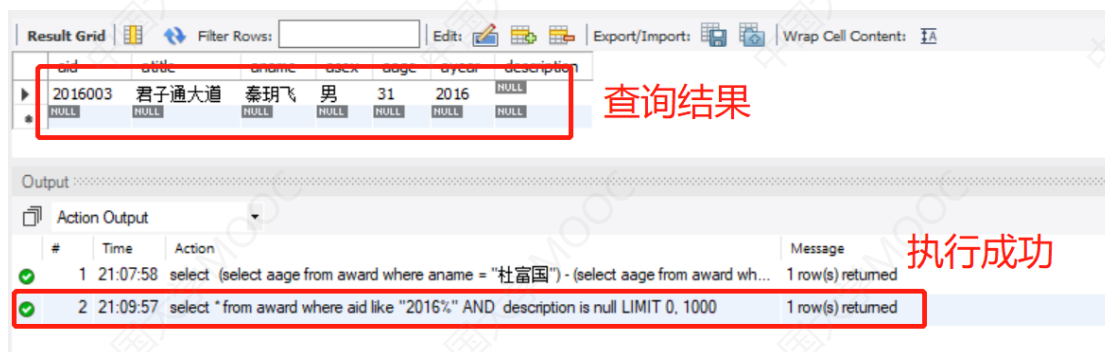
```
select
(select aage from award where aname = "杜富国")
-
(select aage from award where aname = "吕保民");
```

(8) 使用 SQL 查询 award 表中编号以 '2016' 开头并且 description 为空的行。

Step 1. 输入下列 SQL 语句进行查询。



Step 2. 运行结果如图。

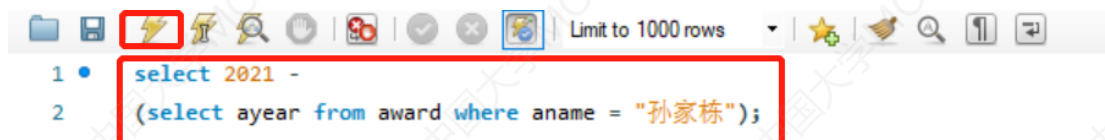


代码：

```
select *
from award
where aid like "2016%" AND
description is null;
```

(9) 使用 select 子句嵌套查询从孙家栋获奖的时间到现在过去了几年。

Step 1. 输入下列 SQL 语句进行查询。



Step 2. 运行结果如图。



代码:

```
select 2021 -  
(select ayear from award where aname = "孙家栋");
```

(10) 创建数据表 `depart`, `plan`, `record`, 并插入数据, 调整表的结构和表中数据到如下状态。

表 `depart` 结构

属性	类型	长度	是否为主键
<code>depart_id</code>	<code>int</code>		是
<code>depart_name</code>	<code>varchar</code>	50	否
<code>depart_tel</code>	<code>varchar</code>	30	否

表 `depart` 数据

<code>depart_id</code>	<code>depart_name</code>	<code>depart_tel</code>
1	教育部	58789087
2	应急部	58789768
3	民政部	58786272
4	国防部	58784585

表 `plan` 结构

属性	类型	长度	是否为主键
<code>plan_id</code>	<code>int</code>		是
<code>plan_name</code>	<code>varchar</code>	50	否
<code>plan_disatype</code>	<code>varchar</code>	20	否
<code>plan_area</code>	<code>varchar</code>	30	否
<code>plan_level</code>	<code>varchar</code>	10	否
<code>plan_date</code>	<code>date</code>		否

表 `plan` 数据

<code>plan_id</code>	<code>plan_name</code>	<code>plan_disatype</code>	<code>plan_area</code>	<code>plan_level</code>	<code>plan_date</code>
1	山东省台风应急预案	自然灾害	山东	三级	2019-08-21

2	江西省暴雨应急预案	自然灾害	江西	一级	2018-07-19
3	汶川地震应急预案	自然灾害	汶川	二级	2008-05-12

表 record 结构

属性	类型	长度	是否为主键	是否外键
depart_id	int		是	是
plan_id	int		是	是
depart_respon	varchar	20	否	否
workload	varchar	30	否	否

表 record 数据

depart_id	plan_id	depart_respon	workload
2	1	安置受灾群众	10 天
3	2	负责灾后重建	30 天
4	1	教育人们加强防范	3 天

Step 1. 输入 SQL 语句创建数据库“应急预案”，并创建三个相关数据表，然后向数据表中插入数据。



```
1 create database 应急预案;
2 use 应急预案;
3 create table depart(
4     depart_id int not null unique primary key,
5     depart_name varchar(50),
6     depart_tel varchar(30)
7 );
8 create table plan(
9     plan_id int not null unique primary key,
10    plan_name varchar(50),
11    plan_disatype varchar(20),
12    plan_area varchar(30),
13    plan_level varchar(10),
14    plan_date date
15 );
16 create table record(
17     depart_id int,
18     plan_id int,
19     depart_respon varchar(20),
20     workload varchar(30),
21     primary key (depart_id,plan_id),
22     foreign key (depart_id) references depart(depart_id),
23     foreign key (plan_id) references plan(plan_id)
24 );
25 insert into depart values
```

Step 2. 结果如图。运行成功，表已出现。

**结果展示**

**执行成功**

#	Time	Action
1	21:12:03	select 2021 - (select ayear from award where aname = "孙家栋") LIMIT 0, 1000
2	21:30:55	create database 应急预案
3	21:30:55	use 应急预案
4	21:30:55	create table depart( depart_id int not null unique primary key, depart_name varchar(50), ...
5	21:30:55	create table plan(plan_id int not null unique primary key, plan_name varchar(50), plan_...
6	21:30:55	create table record( depart_id int, plan_id int, depart_respon varchar(20), workload var...
7	21:30:55	insert into depart values (1, '教育部', '58789087'), (2, '应急部', '58789768'), (3, '民政...
8	21:30:55	insert into plan values (1, '山东省台风应急预案', '自然灾害', '山东', '三级', '2019-08-2...
9	21:30:55	insert into record values (2, 1, '安置受灾群众', '10天'), (3, 2, '负责灾后重建', '30天'), ...

#### 代码:

```
create database 应急预案;
use 应急预案;
create table depart(
    depart_id int not null unique primary key,
    depart_name varchar(50),
    depart_tel varchar(30)
);
create table plan(
    plan_id int not null unique primary key,
    plan_name varchar(50),
    plan_disatype varchar(20),
    plan_area varchar(30),
    plan_level varchar(10),
    plan_date date
);
create table record(
    depart_id int,
    plan_id int,
    depart_respon varchar(20),
    workload varchar(30),
    primary key (depart_id, plan_id),
```

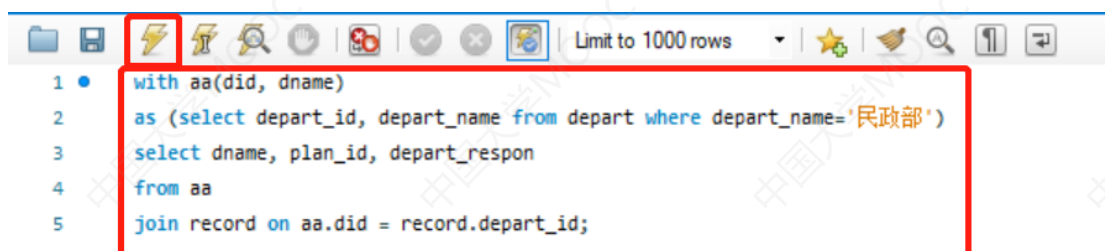
```

foreign key (depart_id) references depart(depart_id),
foreign key (plan_id) references plan(plan_id)
);
insert into depart values
(1, '教育部', '58789087'),
(2, '应急部', '58789768'),
(3, '民政部', '58786272'),
(4, '国防部', '58784585' );
insert into plan values
(1, '山东省台风应急预案', '自然灾害', '山东', '三级', '2019-08-21'),
(2, '江西省暴雨应急预案', '自然灾害', '江西', '一级', '2018-07-19'),
(3, '汶川地震应急预案', '自然灾害', '汶川', '二级', '2008-05-12' );
insert into record values
(2, 1, '安置受灾群众', '10 天'),
(3, 2, '负责灾后重建', '30 天'),
(4, 1, '教育人们加强防范', '3 天' );

```

(11) 使用 with 子句嵌套查询民政部负责的预案号及职责。

Step 1. 输入下列 SQL 语句进行嵌套查询。

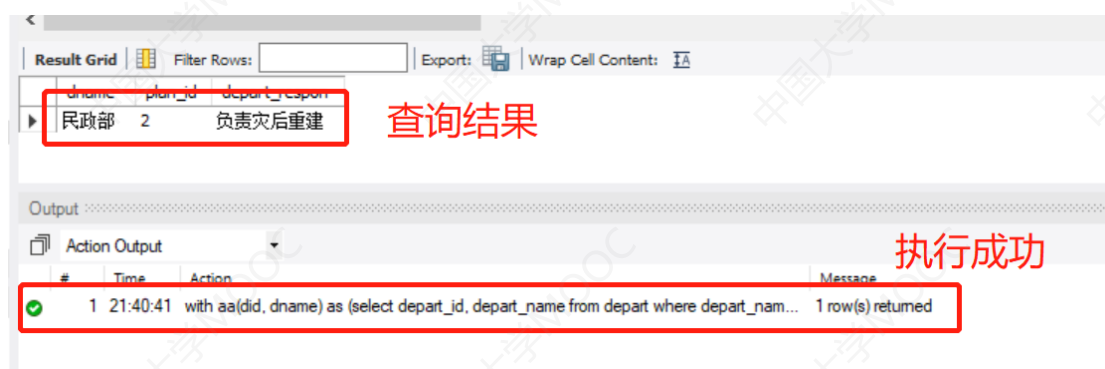


```

1 with aa(did, dname)
2 as (select depart_id, depart_name from depart where depart_name='民政部')
3 select dname, plan_id, depart_respon
4 from aa
5 join record on aa.did = record.depart_id;

```

Step 2. 运行结果如图。



dname	plan_id	depart_respon
民政部	2	负责灾后重建

**查询结果**

Output

Action Output

#	Time	Action	Message
1	21:40:41	with aa(did, dname) as (select depart_id, depart_name from depart where depart_name=...	1 row(s) returned

**执行成功**

代码:

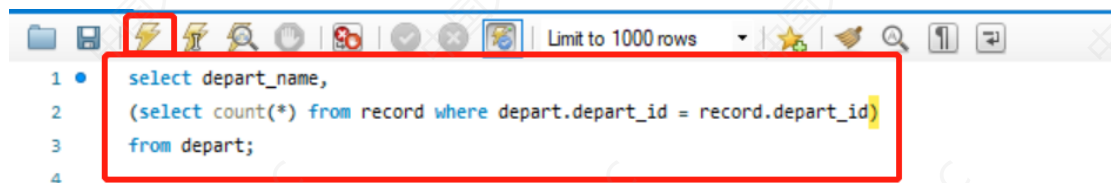
```
with aa(did, dname)
```



```
as (select depart_id, depart_name from depart where depart_name='民政部')
select dname, plan_id, depart_respon
from aa
join record on aa.did = record.depart_id;
```

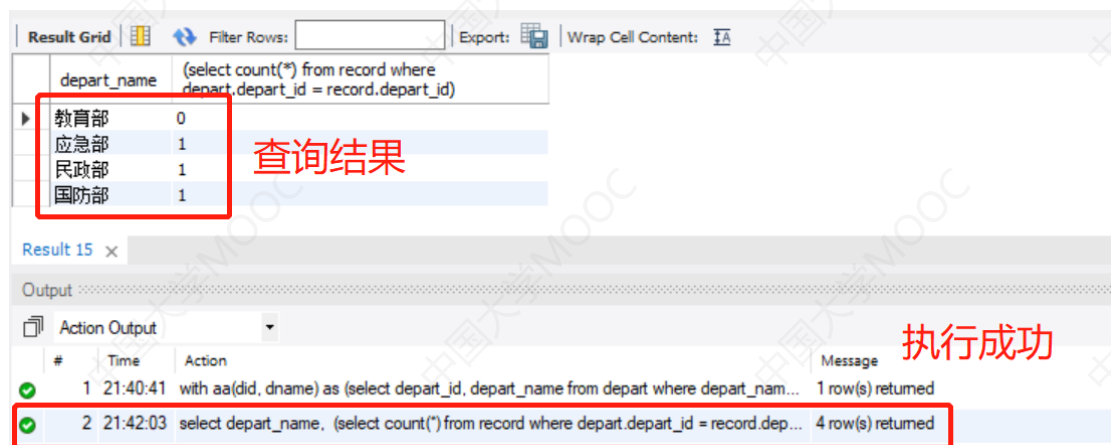
(12) 使用 select 子句嵌套查询各个单位名称及其负责的预案数量。

Step 1.输入下列 SQL 语句进行嵌套查询。



```
1 select depart_name,
2 (select count(*) from record where depart.depart_id = record.depart_id)
3 from depart;
4
```

Step 2. 查询结果如图。



depart_name	(select count(*) from record where depart.depart_id = record.depart_id)
教育部	0
应急部	1
民政部	1
国防部	1

查询结果

执行成功

2 21:42:03 select depart\_name, (select count(\*) from record where depart.depart\_id = record.dep... 4 row(s) returned

代码：

```
select depart_name, (select count(*) from record where depart.depart_id =
record.depart_id)
from depart;
```

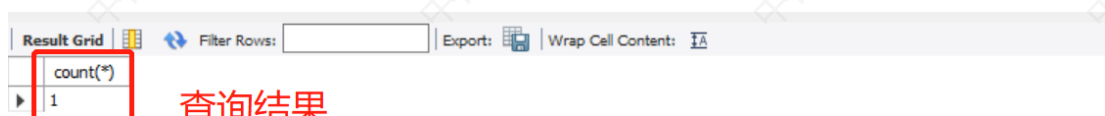
(13) 使用 from 子句嵌套查询应急部负责的预案个数，并将 from 子查询命名为 num

Step 1.输入下列 SQL 语句进行查询。



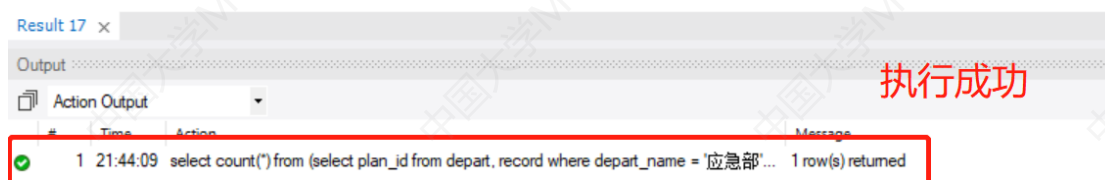
```
1 • select count(*)
2   from
3   (select plan_id from depart, record
4    where depart_name = '应急部' AND
5    depart.depart_id=record.depart_id) as num
6
```

Step 2.运行结果如图。



count(*)
1

查询结果



#	Time	Action	Message
1	21:44:09	select count(*) from (select plan_id from depart, record where depart_name = '应急部'...	1 row(s) returned

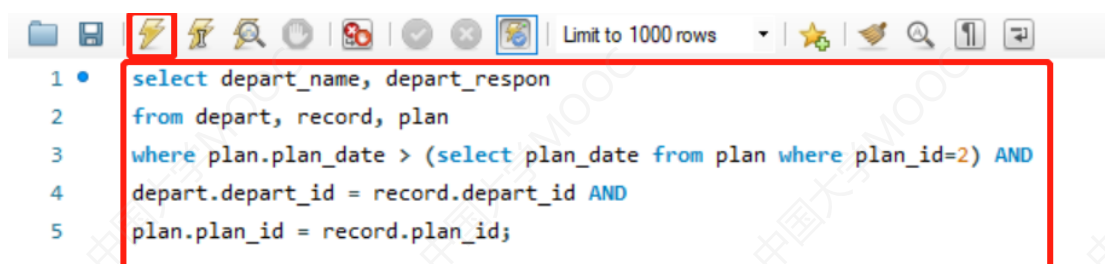
执行成功

代码：

```
select count(*)
from
(select plan_id from depart, record
where depart_name = '应急部' AND
depart.depart_id=record.depart_id) as num;
```

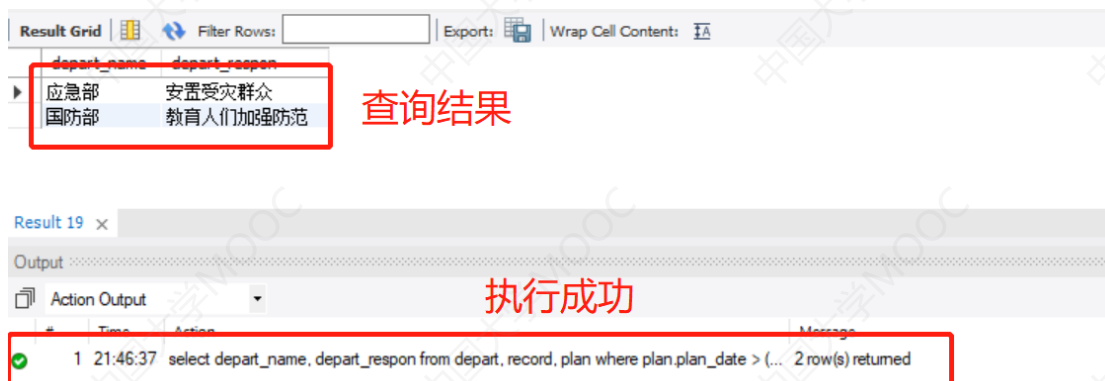
(14) 使用 where 子句嵌套查询：负责‘发布时间在预案编号为 2 的发布时间之后的预案’的单位名称及其在预案中的职责。

Step 1.输入下列 SQL 语句进行查询。



```
1 • select depart_name, depart_respon
2   from depart, record, plan
3   where plan.plan_date > (select plan_date from plan where plan_id=2) AND
4   depart.depart_id = record.depart_id AND
5   plan.plan_id = record.plan_id;
```

Step 2. 运行结果如图。



#### 代码：

```
select depart_name, depart_respon
from depart, record, plan
where plan.plan_date > (select plan_date from plan where plan_id=2) AND
depart.depart_id = record.depart_id AND
plan.plan_id = record.plan_id;
```

## 6. 问题与思考

(1) 从自己的体会来讲，联接查询和嵌套查询各有什么特点？

答：1. 在意义上： 嵌套查询是包含一个或多个子查询或子查询的另一个术语的 select 语句。在外部查询中包含内部查询的另一个子查询是 SQL 语句的扩展。

联接查询是关系数据库中最重要查询。在关系数据库管理系统中，数据之间的关系不需要确定表的建立时间，实体的所有信息通常都存储在表中。检索数据时，通过联接操作查询存储在多个表中的不同实体的信息。

2. 在特征上： 嵌套查询的主要特点是 sql 允许从内到外进行多层嵌套分析，并将子查询的结果作为主查询的查询条件。

联接是关系数据库模型的主要特征。多表查询可以通过联接运算符实现。联接操作给用户带来极大的灵活性，可以随时添加新的数据类型。

(2) 如果一个查询涉及的多个表中有同名字段，在写查询时应注意什么？

答：应注意避免发生歧义、覆盖等问题，并且由于不会报错而造成隐蔽错误。

解决方案：1. 给同名字段用 AS 起别名，在 select 时使用 as + 别名对表中字段进行映射，从而解决问题。

2. 用表名对同名字段进行区分，查询时使用“表.列名”的格式进行区分。