

# 中山大学数据科学与计算机学院本科生实验报告

(2020学年秋季学期)

课程名称：数据库系统 任课教师：刘玉葆

年级&班级	18级 计科一班	专业(方向)	计科 (大数据与人工智能方向)
学号	18340014	姓名	陈嘉宁
电话	18475934419	Email	<a href="mailto:734311072@qq.com">734311072@qq.com</a>
开始日期	2020/11/13	完成日期	2020/11/13

## 一、实验题目

SQL对空值和空集的处理

## 二、实验目的

1. 认识NULL值在数据库中的特殊含义；
2. 了解SQL对空值和空集的处理；
3. 熟练使用SQL语句进行对空置、空集相关的操作。

## 三、实验内容

1. 通过实验验证 SQL SERVER对NULL的处理，包括：
  - 在查询的目标表达式中包含空值的运算。
  - 在查询条件中空值与比较运算符的运算结果。
  - 使用“IS NULL”或“IS NOT NULL”来判断元组该列是否为空值。
  - 对存在取空值的列按值进行 ORDER BY排序。
  - 使用保留字 DISTINCT对空值的处理。
  - 使用 GROUP BY对存在取空值的属性值进行分组。
  - 结合分组考察空值对各个集合函数的影响，特别注意对 COUNT(\*)和 COUNT(列名)的不同影响。
  - 考察结果集是空集时，各个集函数的处理情况。
  - 验证嵌套查询中返回空集的情况下与各个谓词的运算结果。
  - 进行与空值有关的等值连接运算。

## 四、实验过程

1. 通过查询选修课程C++的学生的人数，其中成绩合格的学生人数，不合格的学生人数，讨论NULL值的特殊含义：

查询“成绩合格的学生人数”的查询语句：

```
1 select COUNT(sid)
2 from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
3 where cname = 'C++'
```

查询结果为：

```
SQLQuery1.sql - (I:\KLS675B\dell (56))*
select COUNT(sid)
from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
where cname = 'c++'
```

(无列名)	
1	6031

共有6031人次选择了C++课程，其中包含重复选择（重修）的人数。

查询“成绩合格的学生人数”的查询语句为：

```
1 select COUNT(sid)
2 from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
3 where cname = 'c++' and score >= 60
```

查询结果为：

```
SQLQuery1.sql - (I:\KLS675B\dell (56))*
select COUNT(sid)
from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
where cname = 'c++' and score >= 60
```

(无列名)	
1	4817

共有4817人次及格。

查询“成绩不合格的学生人数”的查询语句为：

```
1 select COUNT(sid)
2 from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
3 where cname = 'c++' and score < 60
```

查询结果为：

```
SQLQuery1.sql - (I:\KLS675B\dell (56))*
select COUNT(sid)
from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
where cname = 'c++' and score < 60
```

(无列名)	
1	724

共有724人次不及格，其中包含补考仍不及格的人数。

score属性中的NULL值表示选修C++课程却没有成绩。

2. 查询选修课程C++的学生的编号和成绩，使用 ORDER BY按成绩进行排序时，取NULL的项是否出现在结果中？如果有，在什么位置？

查询语句为：

```

1 select sid,score
2 from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
3 where cname = 'C++'
4 order by score

```

查询结果为：

SQLQuery1.sql - (L:\KLS675B\del1 (56))\*

```

select sid,score
from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
where cname = 'C++'
order by score

```

结果 消息

	sid	score
1	845947855	NULL
2	890918686	NULL
3	898137922	NULL
4	867715893	NULL
5	872519782	NULL

NULL结果出现在所查询结果的开头处。

### 3. 在上面的查询的过程中，如果加上保留字 DISTINCT会有什么效果呢？

查询语句为：

```

1 select distinct sid,score
2 from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
3 where cname = 'C++'
4 order by score

```

查询结果为：

SQLQuery1.sql - (L:\KLS675B\del1 (56))\*

```

select distinct sid,score
from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
where cname = 'C++'
order by score

```

结果 消息

	sid	score
1	800554079	NULL
2	800575453	NULL
3	800579690	NULL
4	800758208	NULL
5	800895421	NULL

插入distinct前，查询结果共有6031项，插入后仅有6024项。说明存在部分学生多次C++考试成绩相同的情况。

### 4. 按年级对所有学生进行分组，能得到多少个组？与现实的情况有什么不同？

查询语句为：

```

1 select grade
2 from STUDENTS
3 group by grade

```

查询结果为：

SQLQuery1.sql - (I:\KLS675B\deli (56))\*

```

select grade
from STUDENTS
group by grade

```

结果 消息

	grade
1	1998
2	2004
3	2001
4	1996
5	2002
6	1999
7	1993
8	2000
9	1994
10	1997
11	1991
12	NULL
13	2003
14	1992
15	1995

共有15项（即15组）。现实中不会出现年级为“NULL”的情况。

##### 5. 结合分组,使用集合函数求每个课程选修的学生的平均分,总的选课记录数,最高成绩,最低成绩,讨论考察取空值的项对集合函数的作用的影响

查询平均分的查询语句为:

```

1 select cname, AVG(score) as avarage_score
2 from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
3 group by cname

```

查询结果为:

SQLQuery1.sql - (I:\KLS675B\deli (56))\*

```

select cname, AVG(score) as avarage_score
from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
group by cname

```

结果 消息

	cname	avarage_score
1	algorithm	76
2	architectonics	75
3	artifical intelligence	75
4	asp	75
5	basic	76
6	c	75
7	c#	76
8	c++	76

查询选课记录数的语句为:

```

1 select cname, count(score) as num_of_choices
2 from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
3 group by cname

```

查询结果为:

SQLQuery1.sql - (I:\KLS675B\del (56))*		
<pre> select cname, count(score) as num_of_choices from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid group by cname </pre>		
<div> <div>结果</div> <div>消息</div> </div>		
	cname	num_of_choices
1	algorithm	5528
2	architectonics	5567
3	artifical intelligence	5474
4	asp	5382
5	basic	5547
6	c	5549
7	c#	5625
8	c++	5541

查询最高分数的语句为：

```

1 select cname, max(score) as max_score
2 from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
3 group by cname

```

查询结果为：

SQLQuery1.sql - (I:\KLS675B\del (56))*		
<pre> select cname, max(score) as max_score from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid group by cname </pre>		
<div> <div>结果</div> <div>消息</div> </div>		
	cname	max_score
1	algorithm	99
2	architectonics	99
3	artifical intelligence	99
4	asp	99
5	basic	99
6	c	99
7	c#	99
8	c++	99

查询最低分数的语句为：

```

1 select cname, min(score) as min_score
2 from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid
3 group by cname

```

查询结果为：

SQLQuery1.sql - (I:\KLS675B\del (56))*		
<pre> select cname, min(score) as min_score from CHOICES inner join COURSES on CHOICES.cid = COURSES.cid group by cname </pre>		
<div> <div>结果</div> <div>消息</div> </div>		
	cname	min_score
1	algorithm	50
2	architectonics	50
3	artifical intelligence	50
4	asp	50
5	basic	50
6	c	50
7	c#	50
8	c++	50

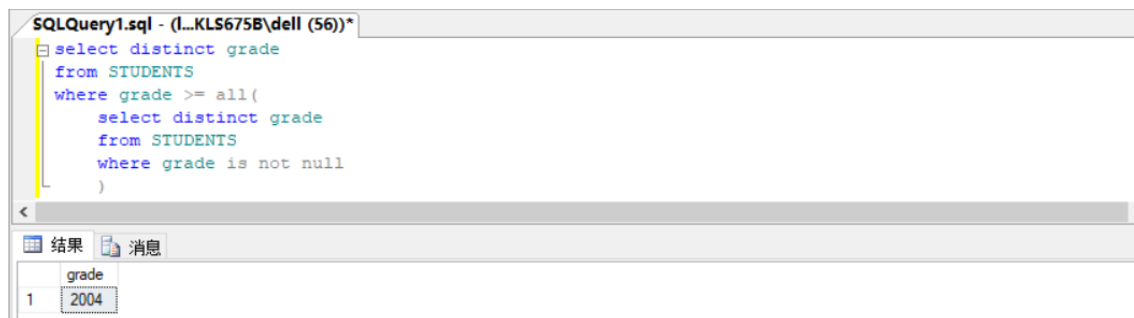
经过验证（即在每个查询语句中都加入“where score is not null”指令）发现NULL值对上述结果均无影响。

6. 采用嵌套查询的方式,利用比较运算符和谓词ALL的结合来查询表 STUDENTS中最晚入学的学生年级。当存在 GRADE取空值的项时,考虑可能出现的情况,并解释

查询语句为:

```
1 select distinct grade
2 from STUDENTS
3 where grade >= all(
4     select distinct grade
5     from STUDENTS
6     where grade is not null
7 )
```

查询结果为:



The screenshot shows a SQL query window titled "SQLQuery1.sql - (I:\KLS675B\del1 (56))\*". The query is:   
select distinct grade  
from STUDENTS  
where grade >= all(  
 select distinct grade  
 from STUDENTS  
 where grade is not null  
)  
The results pane shows a table with one column "grade" and one row with the value "2004".

grade
2004

去掉“where grade is not null”的限制后的查询结果:



The screenshot shows the same SQL query window, but the query is:   
select distinct grade  
from STUDENTS  
where grade >= all(  
 select distinct grade  
 from STUDENTS  
)  
The results pane shows a table with one column "grade".

grade
-------

因为2004与NULL无法进行比较, 所以没有输出值。