22920192204097 吴雨娟 数据库实验

实验二 基本查询语句操作

实验目的：熟悉 SQL SERVER2019 的操作

实验环境：WINDOWS10、SQL SERVER 2019

实验内容：

2.1 数据查询

1)查询年级为 2001 的所有学生的姓名并按学号升序排列。

```
select sname from STUDENTS where grade = 2001 order by sid;
```

结果 消息

	sname
1	ztozk
2	pgmrkdhh
3	nkdnfq
4	qtxblqzsv
5	ctvxn
6	airnnfv
7	vaesalave
8	xiraegdlg
9	dwjny
10	vqjfxfc
11	ebwxl
12	wtljfkvzh

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 6,754 行

2)查询学生的选课成绩合格的课程成绩，并把成绩换算为绩点（60 分对应绩点为 1，每增加 1 分，绩点增加 0.1）。

```
select score, 1+(score-60)*0.1 as 'point' from CHOICES where score >=
60;
```

结果 消息

	score	point
1	76	2.6
2	87	3.7
3	92	4.2
4	82	3.2
5	77	2.7
6	67	1.7
7	90	4.0
8	60	1.0
9	70	2.0
10	82	3.2
11	76	2.6
12	88	3.8

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:01 234,706 行

3)查询课时是 48 或 64 的课程的名称。

```
select cname from COURSES where hour = '48' or hour = '64';
```

结果 消息

	cname
1	computer graphics
2	java
3	design pattern
4	real-time system
5	c
6	computer interface

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 6 行

4)查询所有课程名称中含有 data 的课程编号。

```
select cid from COURSES where cname like '%data%';
```

结果 消息

	cid
1	10001
2	10008

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 2 行

5)查询所有选课记录的课程号（不重复显示）。

```
select distinct cid from CHOICES;
```

结果 消息

	cid
1	10008
2	10019
3	10018
4	10040
5	10011
6	10028
7	10035
8	10021
9	10046
10	10032
11	10005
12	10050
13	10034

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 49 行

6)统计所有教师的平均工资。

```
select AVG(salary) from TEACHERS;
```

结果 消息

	(无列名)
1	3417

7)查询所有教师的编号及选修其课程的学生的平均成绩，按平均成绩降序排列。

```
select t.tid ,
(select AVG(score) from CHOICES where tid = t.tid) as 'avgscore'
from TEACHERS as t
order by avgscore desc;
```

结果 消息

	tid	avgscore
1	259380467	90
2	272507136	89
3	241568555	89
4	266524091	88
5	260955336	88
6	265299360	88
7	294109744	88
8	267952388	87
9	230059993	87
10	250911925	87
11	268322984	87
12	252273969	87
13	254667943	87

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 15,001 行

8)统计各个课程的选课人数和平均成绩。

```
select cid, COUNT(sid) as 'count', AVG(score) as 'avg'
from CHOICES group by cid;
```

结果 消息

	cid	count	avg
1	10008	5985	75
2	10019	6074	75
3	10018	5969	76
4	10040	6102	75
5	10011	6086	76
6	10028	6042	75
7	10035	6104	76
8	10021	5916	76
9	10046	6069	76
10	10032	6073	75
11	10005	6031	76
12	10050	6077	75
13	10034	6000	76

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 49 行

9)查询至少选修了三门课程的学生编号。

```
select sid from CHOICES
group by sid
having COUNT(*) >=3;
```

结果 消息

	sid
1	875058993
2	872331801
3	820928013
4	868752380
5	816522768
6	851274287
7	825441836
8	820970361
9	827797155
10	825319033
11	800921538
12	878637448

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 58,619 行

10)查询编号 800009026 的学生所选的全部课程的课程名和成绩。

```
select cname,score from CHOICES,COURSES
where CHOICES.cid = COURSES.cid
and sid = '800009026';
```

结果 消息

	cname	score
1	tcp/ip protocol	94
2	c	83

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 2 行

11)查询所有选修了 database 的学生的编号。

```
select sid from CHOICES,COURSES
where CHOICES.cid = COURSES.cid
and cname = 'database';
```

结果 消息

	sid
1	870899566
2	830652286
3	818285935
4	891145052
5	882649811
6	896389791
7	875474472
8	885336151
9	813866325
10	827173338
11	830131870
12	840249984

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 5,898 行

12)求出选择了同一个课程的学生数。

```
select cid,count(sid) as 'count' from CHOICES
```

```
group by cid;
```

结果 消息

	cid	count
1	10008	5985
2	10019	6074
3	10018	5969
4	10040	6102
5	10011	6086
6	10028	6042
7	10035	6104
8	10021	5916
9	10046	6069
10	10032	6073
11	10005	6031
12	10050	6077

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 49 行

13)求出至少被两名学生选修的课程编号。

```
select cid from CHOICES
group by cid
having COUNT(sid) >= 2;
```

结果 消息

	cid
1	10008
2	10019
3	10018
4	10040
5	10011
6	10028
7	10035
8	10021
9	10046
10	10032
11	10005
12	10050

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 49 行

14)查询选修了编号 800009026 的学生所选的某个课程的学生编号。

```
select sid from CHOICES
where cid in ( select cid from CHOICES where sid = '800009026' )
and sid != '800009026';
```

结果 消息

	sid
1	850955252
2	801967882
3	820248718
4	893821981
5	826800279
6	828743446
7	884859993
8	846251641
9	889127830
10	841348939
11	881781896
12	828875165

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 17,914 行

15)查询学生的基本信息及选修课程编号和成绩。

```
select STUDENTS.*, CHOICES.cid, CHOICES.score
from STUDENTS,CHOICES
where STUDENTS.sid = CHOICES.sid;
```

结果 消息

	sid	sname	email	grade	cid	score
1	823069829	pxfys	nplb4@xsf.gov	1994	10037	76
2	850955252	baqzmo	tzlk@wed.com	2001	10021	54
3	875434315	rekmgdbo	_ou4lh@mjvkp.edu	1993	10048	82
4	834091581	xxamhe	cphwsy@saz.edu	1994	10049	72
5	838043180	ligga	7_k3o@oga.org	1993	10035	86
6	820248718	zcpkgjxr	j_inl@qtmyg.edu	2003	10015	69
7	860404789	dxyjl	01sp4@jlbch.com	1991	10048	75
8	832640245	kfkpssnzp	5q1dk0@xnp.org	1991	10018	50
9	870049524	pkuapbvjs	lrcjlui@oggo.org	1994	10050	95
10	834236449	gqimd	27j6@orje.net	1996	10041	NULL
11	847572212	rgbpxe	qjjjtaa@zen.edu	2001	10040	60
12	889073206	weerc	u45evh@xvvtn.edu	1991	10023	82

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:02 293,440 行

16)查询学号 850955252 的学生的姓名和选修的课程名及成绩。

```
select STUDENTS.sname, COURSES.cname, CHOICES.score
from STUDENTS, COURSES, CHOICES
where STUDENTS.sid = CHOICES.sid
and CHOICES.cid = COURSES.cid
and STUDENTS.sid = '850955252';
```

结果 消息

sname	cname	score

17)查询与学号 850955252 的学生同年级的所有学生资料。

```
select * from STUDENTS where grade in (
select grade from STUDENTS where sid = '850955252' )
and sid != '850955252';
```

结果 消息

	sid	sname	email	grade
1	800028044	ztozk	r36919m@lmykh.gov	2001
2	800041569	pgmrkdhh	xpqi2wc@hrjtp.edu	2001
3	800070739	nkdnfq	pto7n@sci.com	2001
4	800152632	qtxblqzsv	tdvh@mneu.edu	2001
5	800166448	ctvxn	5b6zz@fpplh.org	2001
6	800169970	airnnfv	ytkyo@xeh.org	2001
7	800177146	vaesalave	tla7d4@ypzxr.gov	2001
8	800202438	xiraegdlg	_fow_i@hmbc.edu	2001
9	800268599	dwjny	9pax@ejjk.com	2001
10	800269975	vqjfxfc	i5w9ba@spu.com	2001
11	800270084	ebwxl	vp02qx@ekze.gov	2001
12	800284630	wtljfkvzh	cc33gi@tqos.org	2001

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:03 6,753 行

18)查询所有有选课的学生的详细信息。

```
select * from STUDENTS where sid in(
select sid from CHOICES );
```

结果 消息

	sid	sname	email	grade
1	800005753	waqcj	hlhq0h8@jdba.gov	1992
2	800010666	uwphrw	emb7k@ipp.com	1992
3	800013889	nahhluoe	w6org6@maq.com	2000
4	800014004	aoaahudi	ftl0oci@fits.edu	1994
5	800023031	aqyty	qfiqdl@tlq.com	1991
6	800031798	oenbdg	c0cjho@kesxd.org	2000
7	800041459	hbywnfxzp	x0ss@isb.com	1996
8	800043108	ttyyb	u6cvso@dacx.gov	1997
9	800045856	szytx	dhu5dd@ess.net	1995
10	800052228	vfkmnuw	ijt_@llqq.com	1994
11	800061741	sfsoonuwe	x5h4mpa@logyp.org	1996
12	800062157	tulnfr	hntkymv@skdmb.gov	1991

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 99,569 行

19)查询没有学生选的课程的编号。

```
select cid from COURSES where cid not in (
select cid from CHOICES );
```

结果 消息

	cid
1	10004

20)查询选修了与C++的课时一样课程的学生名称。

```
select sname from STUDENTS where sid in(
select sid from CHOICES where cid in(
select cid from COURSES where hour in(
select hour from COURSES where cname = 'C++')
)
);
```

结果 消息

	sname
1	vnbqzsvv
2	waqcj
3	aoaahudi
4	cxjnq
5	efmgfprz
6	ocofw
7	ptqno
8	goqjy
9	hfpmwvs
10	hfxjzne
11	vfkmnuw
12	plmoeif

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 16,498 行

21)找出选修课程成绩最好的选课记录。

```
select * from CHOICES where score in(
select MAX(score) from CHOICES);
```

结果 消息

	no	sid	tid	cid	score
1	500031838	818494200	296153375	10013	99
2	500038225	816287601	277204992	10049	99
3	500087553	885816196	209268381	10028	99
4	500101548	892506415	228162635	10027	99
5	500115185	812357245	270853050	10039	99
6	500119539	831046924	290649756	10030	99
7	500144246	831447161	274776068	10015	99
8	500174259	852249511	246511513	10009	99
9	500193481	863407023	282238876	10013	99
10	500207163	868437479	231106864	10008	99
11	500232260	832866714	206952327	10007	99
12	500238836	865169832	295456153	10036	99

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 5,839 行

22)找出和课程 UML 或课程 C++的课时一样课程名称。

```
select cname from COURSES where hour in(
select hour from COURSES where cname = 'UML' or cname = 'C++')
and cname !='UML' and cname != 'C++';
```

结果 消息

	cname
1	data structure
2	computer network

23)查询所有选修编号 10001 的课程的学生的姓名。

```
select sname from STUDENTS where sid in(
select sid from CHOICES where cid = '10001');
```

结果 消息

	sname
1	tkbzqduq
2	efmgfprz
3	msqybykxi
4	jnbluzg
5	lmdjfyync
6	cadcsskc
7	minujyxfb
8	narsw
9	asczkv
10	drmpit
11	isbkjek
12	bhxghanhb

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:05 5,757 行

24)查询选修了所有课程的学生姓名。

```
select sname from STUDENTS where not exists (
select * from COURSES where cid not in (
select cid from CHOICES where sid = STUDENTS.sid ) );
```

结果 消息

sname

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:01 0 行

25) 利用集合运算，查询选修课程 C++或选修课程 Java 的学生的编号。

```
select sid from STUDENTS where sid in(
select sid from CHOICES where cid in(
select cid from COURSES where cname = 'C++' ) )
union
select sid from STUDENTS where sid in(
select sid from CHOICES where cid in(
select cid from COURSES where cname = 'JAVA' ) );
```

结果 消息

	sid
1	800045629
2	800273370
3	800649494
4	800855498
5	800898608
6	800901859
7	800988010
8	800988515
9	801045247
10	801166961
11	801197660
12	801716411
13	801932599

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 5,876 行

26) 实现集合交运算，查询既选修课程 C++又选修课程 Java 的学生的编号。

```
select sid from STUDENTS where sid in(
select sid from CHOICES where cid in(
select cid from COURSES where cname = 'C++' ) )
intersect
select sid from STUDENTS where sid in(
select sid from CHOICES where cid in(
```

```
select cid from COURSES where cname = 'JAVA' ) );
```

结果 消息

sid

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:05 0 行

27) 实现集合减运算，查询选修课程 C++而没有选修课程 Java 的学生的编号。

```
select sid from STUDENTS where sid in(
select sid from CHOICES where cid in(
select cid from COURSES where cname = 'C++' ) )
except
select sid from STUDENTS where sid in(
select sid from CHOICES where cid in(
select cid from COURSES where cname = 'JAVA' ) );
```

结果 消息

	sid
1	822863200
2	875448836
3	818699970
4	804503469
5	824244242
6	835859985
7	862430162
8	832313883
9	848838912
10	855474205
11	865758495
12	866207466
13	859869249

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 5,876 行

2.2 空值和空集的处理

1)查询所有选课记录的成绩并将它换算为五分制（满分 5 分，合格 3 分），注意 SCORE 取 NULL 值的情况。

```
select score/20 as 'score' from CHOICES;
```

结果 消息

	score
1	3
2	4
3	2
4	4
5	4
6	3
7	3
8	4
9	3
10	NULL
11	3
12	4

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:01 293,440 行

SCORE 取 NULL 值的时候，运算后仍为 NULL。

2)通过查询选修编号 10028 的课程的学生的人数，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。

```
select
( select COUNT(sid) from CHOICES where cid = '10028') as 'total',
( select COUNT(sid) from CHOICES where cid = '10028' and score >=
60 ) as 'pass',
( select COUNT(sid) from CHOICES where cid = '10028' and score < 60 )
as 'fail';
```

结果 消息

	total	pass	fail
1	6042	4812	755

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 1 行

通过结果可以看到，成绩合格的人数加上不合格的人数小于总人数，所以在筛选成绩合不合格时，NULL 值不参与。

3)通过实验检验在使用 ORDER BY 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？

将学生选课记录按成绩升序排序：

```
select * from CHOICES order by score;
```

结果 消息

	no	sid	tid	cid	score
1	504633744	810944133	213188190	10014	NULL
2	516882306	854732119	292667129	10041	NULL
3	507160198	878729934	254521631	10047	NULL
4	502168378	845239394	259972452	10031	NULL
5	509546580	835343030	276071516	10038	NULL
6	514447255	843296996	240408939	10008	NULL
7	512100061	832384804	296902320	10024	NULL
8	500005519	813520169	226385492	10023	NULL

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:02 293,440 行

从结果可以看到，取 NULL 的项会出现在结果中，NULL 值排序时被当做最小值。

4)在上面的查询过程中如果加上保留字 DISTINCT 会有什么效果？

```
select distinct * from CHOICES order by score;
```

结果 消息

	no	sid	tid	cid	score
1	500005519	813520169	226385492	10023	NULL
2	502168378	845239394	259972452	10031	NULL
3	509546580	835343030	276071516	10038	NULL
4	514447255	843296996	240408939	10008	NULL
5	507160198	878729934	254521631	10047	NULL
6	516882306	854732119	292667129	10041	NULL
7	504633744	810944133	213188190	10014	NULL
8	512100061	832384804	296902320	10024	NULL

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:02 293,440 行

从结果的行数看来，加上保留字 DISTINCT 后，多个 NULL 值并不会被当做重复值删掉。

5)通过实验说明使用分组 GROUP BY 对取值为 NULL 的项的处理。

```
select score, count(score) as 'count' from CHOICES group by score;
```

结果 消息

	score	count
29	62	5765
30	99	5839
31	76	5877
32	NULL	0
33	82	5915
34	88	5848
35	53	5788
36	71	5881

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:02 47 行

从结果可以看到，NULL 值会被分为一组，但是不会进行计数。

6)结合分组，使用集合函数求每个同学的平均分、总的选课记录、最高成绩、最低成绩和总成绩。

```
select sid, AVG(score) as 'avg', COUNT(cid) as 'count', MAX(score) as
'max',
MIN(score) as 'min', SUM(score) as 'sum' from CHOICES where sid in (
```

```
select sid from STUDENTS ) group by sid;
```

结果 消息

	sid	avg	count	max	min	sum
1	828667933	80	3	90	68	242
2	886690009	83	5	95	67	415
3	858350370	73	2	73	73	73
4	812917218	77	5	99	53	386
5	801855166	87	4	94	76	349
6	888277410	73	5	90	65	365
7	807064377	88	4	99	77	354
8	808113482	97	1	97	97	97

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 99,569 行

为了防止有学生没有选课的情况，按 STUDENTS 表中所有的学号来统计。

7)查询成绩小于 60 的选课记录，统计总数、平均分、最大值和最小值。

```
select COUNT(*) as 'count', AVG(score) as 'avg',
MAX(score) as 'max', MIN(score) as 'min' from CHOICES where score <
60;
```

结果 消息

	count	avg	max	min
1	35293	52	55	50

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 1 行

8）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 COURSES 中最少的课时。假设数据库中只有一个记录的时候，使用前面的方法会得到什么结果，为什么？

```
select distinct hour from COURSES where hour <= ALL(select hour from
COURSES);
```

结果 消息

	hour
1	48

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) School 00:00:00 1 行

建立一个只有一条课程记录的表 COURSES_2，如下：

LAPTOP-N3QO3T4...dbo.COURSES_2 | exp2.sql - LAPTOP..

	cid	cname	hour
	10001	computer neteork ...	66
▶*	NULL	NULL	NULL

执行查询语句如下：

```
select distinct hour from COURSES_2 where hour <= ALL(select hour from COURSES_2);
```

结果如下：

结果 | 消息

	hour
1	66

查询已成功执行。 LAPTOP-N3QO3T48 (15.0 RTM) wyj (65) TEST 00:00:00 1 行

所以当数据库中只有一个记录的时候，使用前面的方法同样可以得到正确的结果。因为使用的比较运算符是‘<=’，自己是满足小于等于自己的条件的，所以依然可以得到正确的查询结果。

9）创建一个学生表 S（NO， SID， SNAME），教师表 T（NO， TID， TNAME）作为实验用的表。其中 NO 分别是这两个表的主键，其他键允许为空。

向 S 插入元组（1， 0129871001， 王小明）、（2， 0129871002， 李兰）、（3， 0129871005， NULL）、（4， 0129871004， 关红）；

向 T 插入元组 1， 100189， 王小明）、（2， 100180， 李小）、（3， 100121， NULL）、（4， 100128， NULL）。

对这两个表作对姓名的等值连接运算，找出既是老师又是学生的人员的学生编号和老师编号。

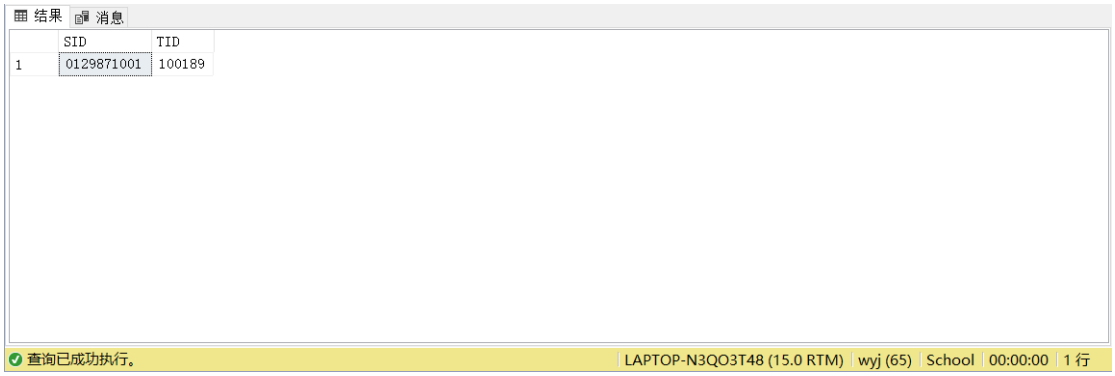
```
create table S
(NO int primary key,
 SID char(10),
 SNAME char(30)
 );
```

```sql
create table T
(NO int primary key,
TID char(10),
TNAME char(30)
);

insert into S values(1,'0129871001','王小明');
insert into S values(2,'0129871002','李兰');
insert into S values(3,'0129871005',NULL);
insert into S values(4,'0129871004','关红');

insert into T values(1,'100189','王小明');
insert into T values(2,'100180','李小');
insert into T values(3,'100121',NULL);
insert into T values(4,'100128',NULL);

select S.SID,T.TID from S,T where S.SNAME = T.TNAME;
```



实验总结：

通过本次实验，熟悉了基本查询语句的操作，在遇到报错时会查阅课本，查看在哪里发生了语法错误。在完成实验后，对查询语句的应用更加熟练，并且加深了对 NULL 值的理解，在一些特定的查询中，不能忽略 NULL 值带来的影响。