1. **Oracle简介**

如果想要在Linux系统下开启orancl的服务的话，则要用命令开启

Windows也是可以的 cmd----> listener control 监听控制的简写



**lsnrctl stop关闭监听服务**

**lsnrctl start开启监听服务**

OracleOraDb11g\_home1TNSListener-------------监听服务

**net stop oracleserviceorcl**



如果想要让自己的电脑服务器端能够向外提供服务，那么一定要开启这两项服务

只开启OracleServiceORCL的的话，是能够sqlPlus连接成功的。

客户端访问服务器端，那么服务器端一定要开启Listener服务，监听服务

**企业管理器**

<https://192.168.100.97:1158/em>

[https://127.0.0.1:1158/em](https://192.168.100.97:1158/em)

[https://localhost:1158/em](https://192.168.100.97:1158/em)

https://192.168.100.97:1158/em/console/logon/logon



**不过必须要开启 OracleDBConsoleorcl**



开发工具

SQL\*Plus

EM

SQL Developer

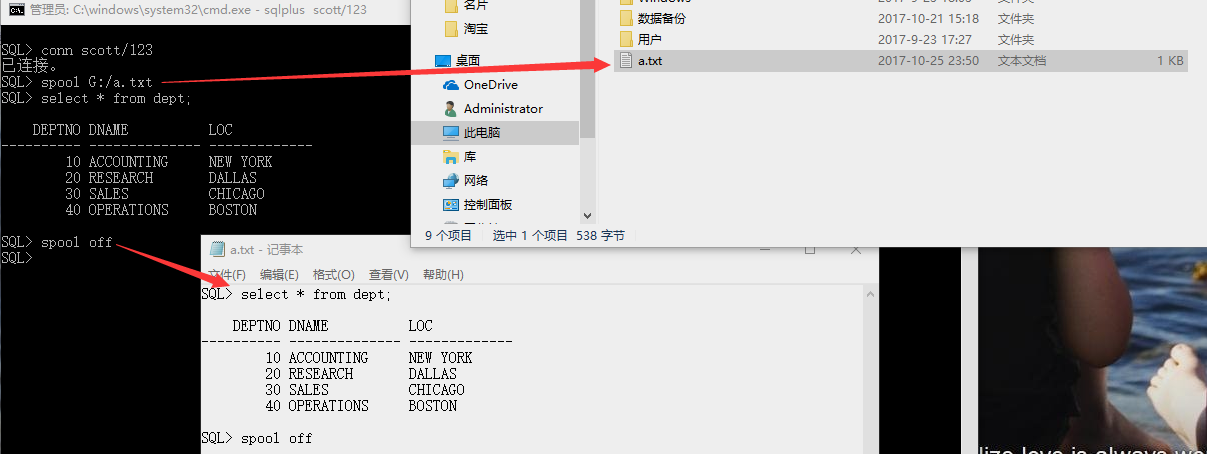
PL Sql/developer

TOAD（显示信息量大） 需要下载

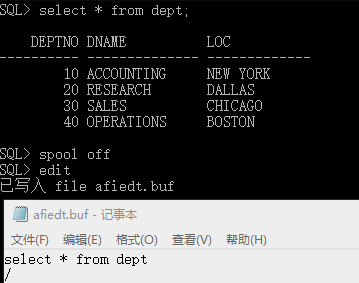
**Quit 退出当前用户**

**Clear screen 屏幕清空**

Spool----保存sql命令为脚本文件的命令



**Edit 命令 ----- 输入后显示最后输入的语句，目的是修改这个语句，然后提交，或者进行复制等操作，最后不要忘了输入 r 进行提交**



**Set time on/off 显示关闭时间**



**Show error 显示上一个命令的错误**



**Host 命令，直接回到操作系统的dos命令行**

**Host mspatin 直接打开操作系统的画图**

也可以作为临时使用操作系统的dos命令

**@命令 执行写好的脚本**

**假如D盘下面有一个a.sql的脚本文件，可以直接在sqlplus中这样执行**

**@d:/a.sql;**

**其实就和start 命令一样**

1. **oracle11g的基本SQL语句和函数**

SQL是**Structured Query Language（结构化查询语言）**的首字母缩写词

SQL是数据库语言，Oracle使用该语言存储利检索信息

表是主要的数据库对象，用于存储数据

**char 和 varchar不同之处在于char是固定长度的varchar是可变长度的**

char数据类型的列长度可以是1到2千个字节

VARCHAR2数据类型的大小在144000个字节

**不要再用varchar类型了，oracle在以后的版本里会逐渐取消对varchar的支持**

**注意number类型**

**create table shun(s number(3,2);**

**insert into shun values(1.2156) 插入的结果为 1.22**

**insert into shun values(10) 插入错误，值大于为此列指定的允许精度**

**因为10后面还有两位小数，加起来就是4位，超过了3，小数位超过就是四舍五入。**

**日期时间数据类型**存储日期和时间值，包括年、月、日，小时、分钟、秒

主要的日期时问类型有：

**DATE -存储日期和时间部分，精确到整个的秒**

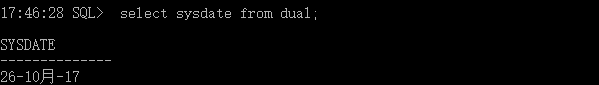
**TIMESTAMP -存储日期、时间和时区信息，秒值精确到小数点后6位**

**Dual是Oracle中的一个实际存在的表，任何用户均可读取，常用在没有目标表的Select语句块中**

**--查看当前连接用户SQL> select user from dual;  
--查看当前日期、时间SQL> select sysdate from dual;  
--当作计算器用SQL> select 1+2 from dual;**

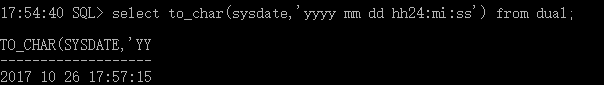
**sysdate**在oracle中是一个取到当前系统时间的函数

17:46:28 SQL> **select sysdate from dual;**

（在sql server中直接 select getdate()回车）

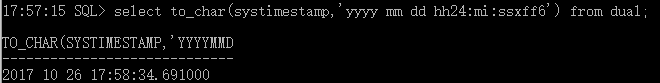
来个正常的时间

17:54:40 SQL> select **to\_char**(sysdate,'yyyy mm dd hh24:mi:ss') from dual;



精确到小数点后六位

**select to\_char(systimestamp,'yyyy mm dd hh24:mi:ssxff6') from dual;**



RAW数据类型用于存储二进制数据

RAW数据类型最多能存储2000字节

LONG RAW数据类型用于存储可变长度的二进制数据

LONG RAW数据类型最多能存储2 GB

CLOB 即 Character LOB（字符 LOB），它能够存储大量字符数据

BLOB 即 Binary LOB（二进制 LOB），可以存储较大的二进制对象，如图形、视频剪辑和声音文件

BFILE 即 Binary File（二进制文件），它用于将二进制数据存储在数据库外部的操作系统文件中

操作表

创建表：create table shun(sno number(6),sname varchar2(10),birthday date);

修改添加列：alter table shun add tele varchar2(11); **添加不要加column**

修改列字段长度：alter table shun **modify** tele varchar2(20); --modify：修改

查看表结构：**desc tablename;** --desc：也是order by 里面的降序

删除表字段：**alter table shun drop column tele;** 值得注意的是删除要**加column**

插入表内容：insert into shun values(001,'a','17-1月-01'); ----插入2017-1-1

注意插入date类型值的时候需要的格式为 '17-1月-01'

无用的格式为'17-1-01'---'2017-1-01' **必须要加月字**

由此想到**select sysdate from dual;返回系统当前时间为27-10月-17**没？

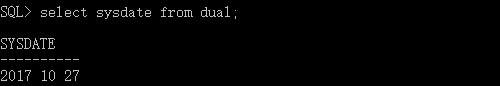
关于表字段长度由大到小的alter

varchar2(1)------>varchar(8)，是完全没有问题的

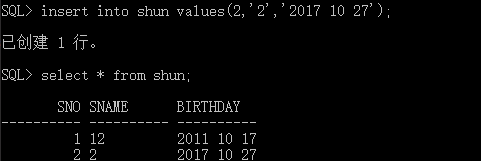
varchar2(8)------>varchar(5)，**如果该字段里面没有值，或者值小于没有超出修改后字段的大小也是可以的。但是值大小超出修改后字段的大小，修改便会失败。**

**设置时间格式：alter session set nls\_date\_format='yyyy-mm-dd hh24:mi:ss';**





然后插入date类型的字段时候，那么可以inset into tname values(‘19980713’);了



插date还可以insert into tname(birthday) values(to\_date(‘19980713’,’yyyymmdd’))

是**一旦退出了会话，那么又会变成格式 ，也就是说只在当前会话时有效。这一点一定要注意。**

**插入null值直接写Null;**

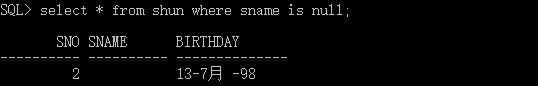
insert into tname(**na,age,bday**) values(‘na’,null,’to\_date(’19980812,,‘yyyymmdd’));

**或者直接不写**

insert into tname(**na,bday**) values(‘na’,’to\_date(’19980812’,‘yyyymmdd’));

只有一种方式查询null值,这是**唯一的**方式。

select \* from shun where **sname is null;**



**克隆表结构，复制表**

**sql server ----> select into 新表名 from 原表名**

**oracle ----> create table 新表名as select \* from 原表名;**

但是如何只创建表的结构而不进行全表的复制呢？

oracle ----> create table 新表名as select \* from 原表名 **where 1>2;**

清空表的全部内容 tuncate：缩短

delete tname; ------这个要commit **有日志，可回滚**

delete from tname; ------这个要commit **有日志，可回滚**

tuncate table tname; ------这个不用commit **无日志，要小心**

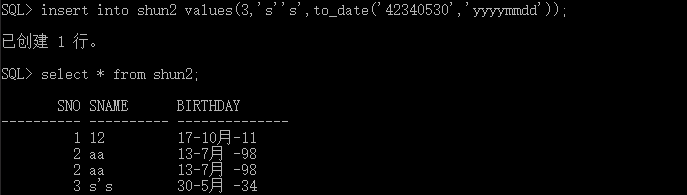
**如何向一个空表中复制某个表中的所有数据？**

**sql server ------> insert into 新表 select \* from 原表**

**oracle -------> insert into 新表 select \* from 原表**

**没错，都是一样的......**

想要为varchar2或者其它字符串类型的字段里面插入“‘ ”的话oracle里面规定两个“ ‘’ ”代表一个“‘ ”就**必须打两个’’**



**匹配符，通配符**

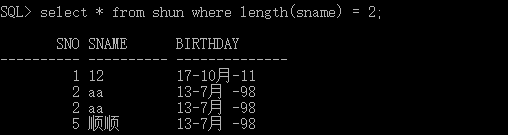
**%是多个，\_是一个**

sql server 和 oracle 都是一样的

select \* from shun where name like ‘a%’;找出第一个是a\*的

select \* from shun where name like ‘\_a\_’;找出第一个是长度为3的并且中间是a的

**长度函数select \* from \* where length(\*) = \*;**



排序

select \* from shun **order by column desc(顺序);**

select \* from shun **order by column asc(倒序);**

还可以加多列

**select \* from shun order by column desc(顺序) , column;**

还可以用列数字来进行代替

比如name在第3列

select \* from shun **order by 3 desc;**

别名的使用

select sno 学号 ,sname "姓名" from shun;

如果想要使别名带空格必须要使用 ”a a” 的格式。

查询当前用户所拥有的表名

**select table\_name from user\_tables;**

**select \* from tab;**

create table student(sno number(6),sname varchar2(10),birthday date, male char(2),tele char(11));

insert into student values(1,'张三','21-1月-80','男','0101');

insert into student values(2,'李四',to\_date('19830125','yyyymmdd'),'男'，'0102');

insert into student values(3,'王五',to\_date('19830125','yyyymmdd'),'男'，'0103');

create table address(sno number(6),zz varchar2(10));

insert into address values(1,'郑州');

insert into address values(2,'开封');

insert into address values(3,'洛阳');

insert into address values(4,'新乡');

create table score(sno number,km varchar2(10),score number);

insert into score values(1,'语文',60);

insert into score values(1,'数学',60);

insert into score values(1,'英语',60);

insert into score values(2,'语文',70);

insert into score values(2,'数学',70);

insert into score values(3,'语文',80);

create table student(sno number(6) ,birthday date ,sname varchar2(10));

insert into student values(1,'11-1月-81','张三');

insert into student values(2,'10-3月-82','李四');

insert into student values(3,'06-1月-83','王五');

commit;

要求1：查出每个学生的成绩的总和

select sno as 学号,sum(score) 总成绩 from score group by sno;

要求2：查出每门课程的平均成绩

select km as 课名,avg(score) 平均成绩 from score group by km;

要求3：查出平均成绩大于60的学生的学号和平均成绩

select sno as 学号,avg(score) 平均成绩 from score group by sno having(avg(score)>=60);

要求4：查出平均成绩大于所有学生的平均成绩的学生的学号和成绩

select sno 学号,avg(score) 平均成绩 from score group by sno

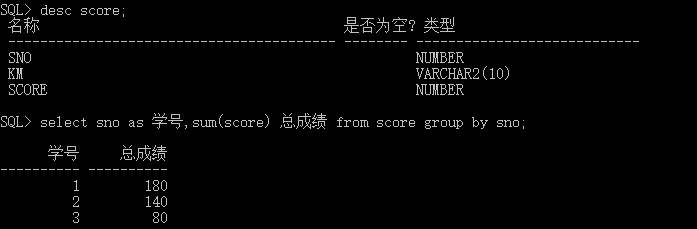
having(avg(score)>(select avg(score)from score));

要求5：查出平均成绩大于60的学生的学号和平均成绩，并按照学号的降序排列（说明了几个谓词的顺序）

select sno,avg(score) from score group by sno having(avg(score)>60) order by sno desc;

分组查询学生的总成绩

select sno as 学号,sum(score) 总成绩 from score group by sno;



之所以没有select \* form.........是因为已经有sno作为分组条件了，分组只能显示一条，而加上科目之后就没办法进行分组了

分组查询每科的平均成绩

select km as 课名,avg(score) 平均成绩 from score group by km;

SQL> select sno as 学号,avg(score) 平均成绩 from score group by sno having(avg(score)>60); --查询平均成绩大于60分的

学号 平均成绩

---------- ----------

2 70

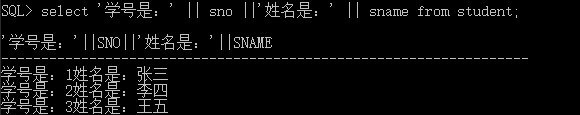
3 80

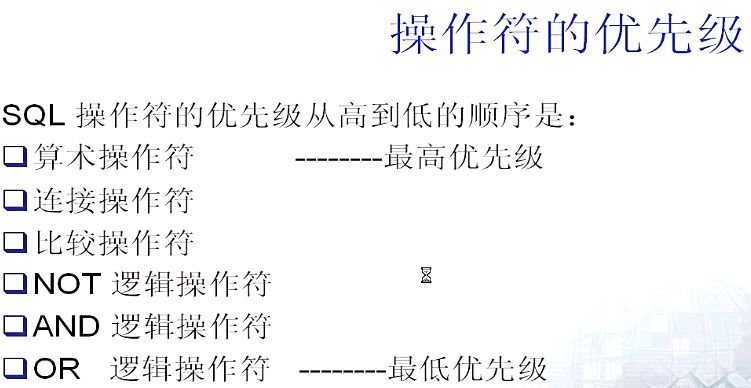
**分组后加限制条件只能通过加havaing的方式，where条件不行**

查询平均成绩大于所有的平均成绩的学号和成绩（就是没有拖后腿的）

select sno 学号,avg(score) 平均成绩 from score group by sno having(avg(score)>(select avg(score)from score));

连接符 select '学号是：' || sno ||'姓名是：' || sname from student;





oracle中的函数

单行函数

单行函数对于从表中查询的每一行只返回一个值

可以出现在SELECT子句中和WHERE子句中

单行函数可以大致划分为：

字符函数：字符函数接受字符输入并返回字符或数值

|  |  |  |
| --- | --- | --- |
| **函数** | **输入** | **输出** |
| **Initcap(char)（首字母大写）** | **Select initcap(‘hello’) from dual;** | **Hello** |
| **Lower(char)（全小写）** | **Select lower(‘FUN’) from dual;** | **fun** |
| **Upper(char)（全大写）** | **Select upper(‘sun’) from dual;** | **SUN** |
| **Ltrim(char,set)（减左）** | **Select ltrim( ‘xyzadams’,’xyz’) from dual;** | **adams** |
| **Rtrim(char,set)（减右）** | **Select rtrim(‘xyzadams’,’ams’) from dual;** | **xyzad** |
| **Translate(char, from, to)（替换）** | **Select translate(‘jack’,’j’ ,’b’) from dual;** | **back** |
| **Replace(char, searchstring,[rep string])** | **Select replace(‘jack and jue’ ,’j’,’bl’) from dual;** | **black and blue** |
| **Instr (char, m, n)（位置）** | **Select instr (‘worldwide’,’d’) from dual;** | **5** |
| **Substr (char, m, n)（截取）** | **Select substr(‘abcdefg’,3,2) from dual;** | **cd** |
| **Concat (expr1, expr2)（连接）** | **Select concat (‘Hello’,’ world’) from dual;** | **Hello world** |

chr函数的使用

**select chr(97) from dual; 打印a**

**select ascii('a') from dual; 打印97**

填充函数

select lpad('abcd',10,'s') from dual; lpad：左填充

不够10位，打印ssssssabcd

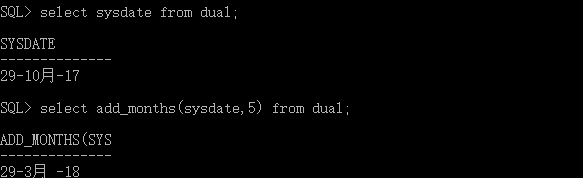
length函数：显示长度大于2的记录

select \* from shun2 where length(sname) > 2;

日期时问函数：

**select sysdate from dual;查询当前的系统时间**

select **add\_months(sysdate,5)** from dual; 查询系统时间加5个月的时间

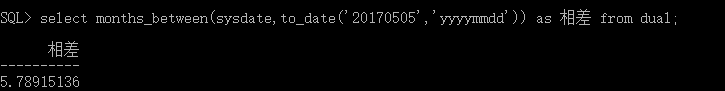


但是记住只有months月份有，years,days是没有的噢

计算相差多少个月

**select months\_between(sysdate,to\_date('20170505','yyyymmdd')) as 相差 from dual; （前面的月份减去后面的月份）**

**但是记住只有months月份有，years,days是没有的噢**



提取年月日份

select extract(**year from sysdate**) as 今年是 from dual;

select extract(**month from sysdate**) as 今年多少月 from dual;

select extract(**year from sysdate**) as 今年是 from dual;

当前月的最后一天是

**select last\_day(sysdate) as 这月的最后一天 from dual;**

create table student (sno number(6),birthday date ,sname varchar2(10));

insert into student values(1,'11-1月-81','张三');

insert into student values(2,'10-3月-82','李四');

insert into student values(3,'06-1月-83','王五');

commit;

**除了drop table tnme可以删除表还有drop table tname purge;**

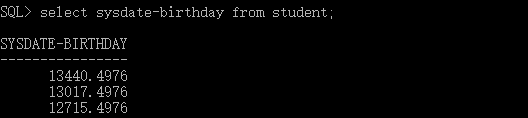
求出生日期加一年的日期

**select sno, add\_months(birthday,12) from student;**

查询到大于27岁的人

select \* from student where **sysdate>add\_months(birthday,27\*12);**

求生日到今天，活了多少天

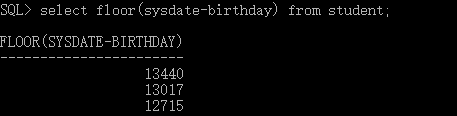


由于sysdate和birthday(date)都是日期型，精确到秒数级别所以有小数

取整函数floor() and ceil() 都是直接舍去小数

select floor(sysdate-birthday) from student;

select ceil(sysdate-birthday) from student;



数字函数：

|  |  |  |
| --- | --- | --- |
| 函数 | 输入 | 输出 |
| Abs(n) （绝对值） | Select abs(-15) from dual; | 15 |
| Ceil(n) （舍去小数） | Select ceil(44.778) from dual; | 45 |
| Cos(n) （求余弦值） | Select cos(180) from dual; | -.5984601 |
| Cosh(n) （双曲） | Select cosh(0) from dual; | 1 |
| Floor(n) （舍去小数） | Select floor(100.2) from dual; | 100 |
| Power(m,n) （次幂） | Select power(4,2) from dual; | 16 |
| Mod(m,n) （模以） | Select mod(10,3) from dual; | 1 |
| Round(m,n) （四舍五入） | Select round(100.256,2) from dual; | 100.26 |
| Trunc(m,n) （保留n位） | Select trunc(100.256,2) from dual; （不四舍五入） | 100.25 |
| Sqrt(n) （求平方根） | Select sqrt(4) from dual; | 2 |
| Sign(n) | Select sign(-30) from dual; | -1 |

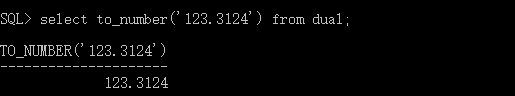
Sign(n)取数字n的符号,大于0返回1,小于0返回-1,等于0返回0

select round(1236.23,-1) from dual; 打印1240 因为 -1 是1236的整数位的倒数第一个数进行四舍五入

转换函数：

to\_char to\_date to\_number

select to\_char(33,'L99.99') from dual; 打印￥33.00



一个规则的日期格式

select to\_char(sysdate,'YYYY"年"fmMM"月"fmdd"日" HH24:MI:SS') from dual;



混合函数：

decode函数

NVL，第一为空返回二；否则返回一。

select nvl(2,1) from dual; 打印2

select nvl('',1) from dual; 打印1

select nvl(null,1) from dual; 打印1

NVL2，第一个不空则返回二；否则返回三。

select nvl2(2,4,3) from dual; 打印4

select nvl2(null,4,3) from dual; 打印3

NULLIF，两个表达式，相等则返回空；否则第一个。

select nullif(2,25)from dual; 打印2

select nullif(2,2)from dual; 打印‘ ’

create table student(sno number(6),sname varchar2(10),sage int);

insert into student values(1,'张三',21);

insert into student values(2,'张飞飞',22);

insert into student values(3,'王五',23);

insert into student values(4,'赵云',24);

commit;

create table address(sno number(6),zz varchar2(10));

insert into address values(1,'郑州');

insert into address values(2,'开封');

insert into address values(3,'洛阳');

insert into address values(5,'新乡');

commit;

连接查询

内连接--相等连接

select s.sno 学号,s.sname 姓名,a.zz 住址 from student s,address a where s.sno=a.sno;

select s.sno 学号,s.sname 姓名,a.zz 住址 from student s inner join address a on s.sno=a.sno;

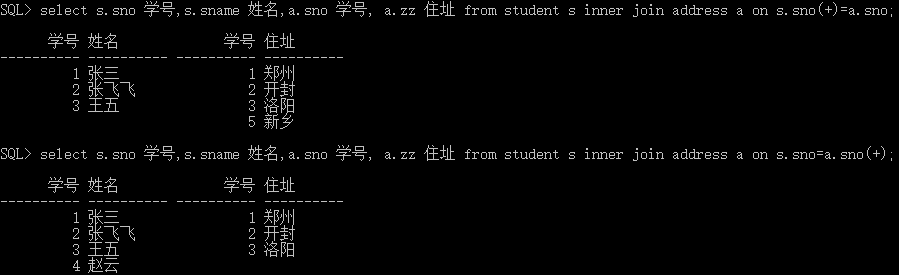
左右连接

select s.sno 学号,s.sname 姓名,a.sno 学号, a.zz 住址 from student s inner join address a on s.sno=a.sno(+); --右连接

select s.sno 学号,s.sname 姓名,a.sno 学号, a.zz 住址 from student s inner join address a on s.sno(+)=a.sno; --左连接

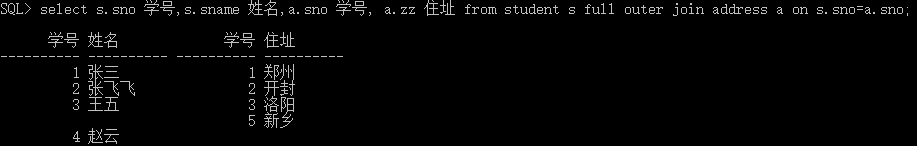
还有select s.sno 学号,s.sname 姓名,a.sno 学号, a.zz 住址 from student s right join address a on s.sno=a.sno;这种方式

left join right join



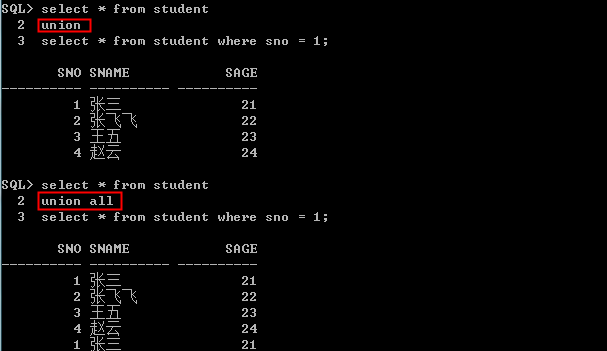
更有全连接的方式

select s.sno 学号,s.sname 姓名,a.sno 学号, a.zz 住址 from student s full outer join address a on s.sno=a.sno;

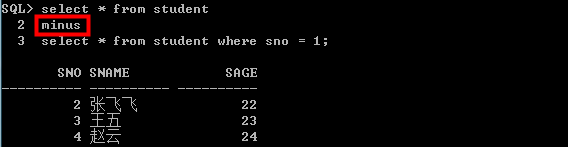


集合操作符

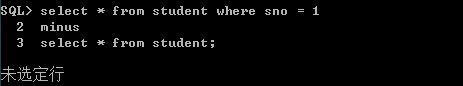
union



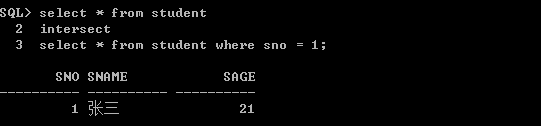
minus 减去显示



但是注意不能够小的减去大的



intersect 交叉，贯穿，，，查到一模一样的



重命名表

rename table to newtable;

重命名表字段名

alter table tablename rename column 列名 to new列名