1. **offset and limit查询**

对于rownum来说它是oracle系统顺序分配为从查询返回的行的编号，返回的第一行分配的是1，第二行是2，依此类推，这个伪字段可以用于限制查询返回的总行数，而且rownum不能以任何表的名称作为前缀。

设定一个表company，有13行记录，要查3到10行，怎么查呢？都知道在mysql里面有limit，很好用，在oracle时就要使用子查询了。

如: SELECT \* FROM (SELECT ROWNUM NO FROM COMPANY) WHERE NO BETWEEN 3 AND 10

这样就ok了

如使用SELECT ROWNUM FROM COMPANY WHERE ROWNUM BETWEEN 3 AND 10 这样就是不对的。很小的问题，大家记住。

2. **使用nulls first 或者nulls last 语法**

Nulls first和nulls last是Oracle Order by支持的语法

如果Order by 中指定了表达式Nulls first则表示null值的记录将排在最前(不管是asc 还是 desc)

如果Order by 中指定了表达式Nulls last则表示null值的记录将排在最后 (不管是asc 还是 desc)

使用语法如下：

--将nulls始终放在最前

select \* from zl\_cbqc order by cb\_ld nulls first

--将nulls始终放在最后

select \* from zl\_cbqc order by cb\_ld desc nulls last

3. **日期字段格式化**

对日期字段进行格式化时(to\_char)，以24小时格式hh24，以12小时hh12

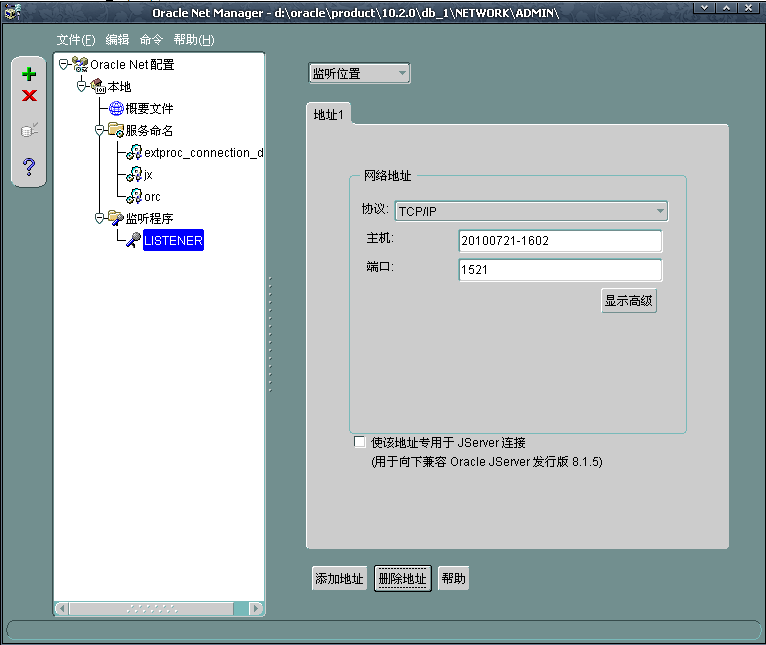
4. **decode()函数**

decode()函数用于实现多路分支结构：decode(deptno, 10, '财务部', 20, '研发部', '未知部门') dept

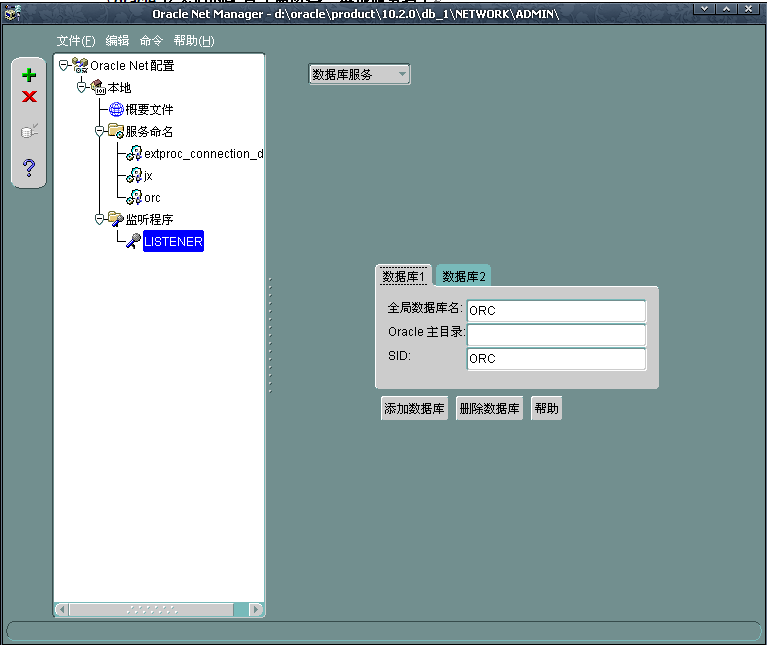
5.**oracle安装后的配置（监听器、本地服务名）**

开始/程序/ Oracle - OraDb10g\_home1 /Configuration and Migration Tools/Net Manager

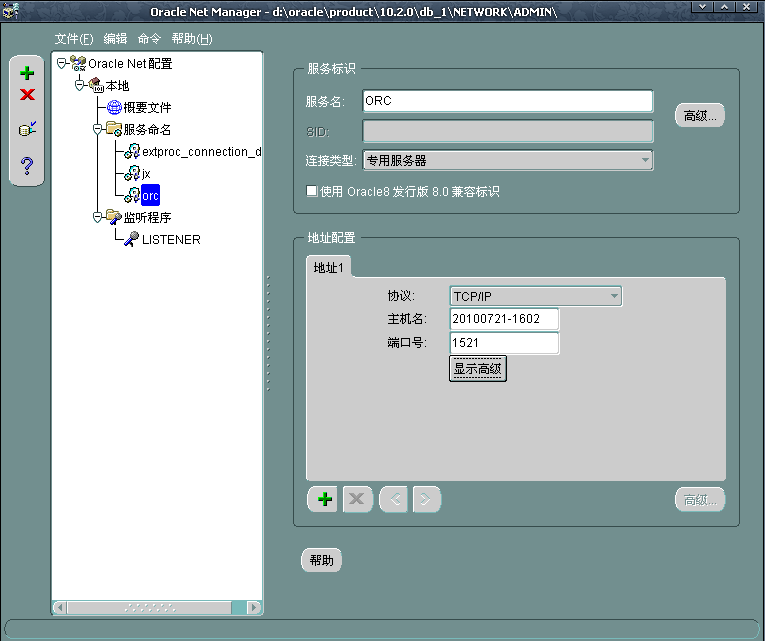
创建监听器：主机可以是计算机名称或IP地址，端口为默认的1521，完成后，会将监听器添加到服务中（例OracleOraDb10g\_home1TNSListener），如下图



创建数据库服务：在下拉表中选择数据库服务，如下图



创建数据库连接服务名：“服务名”就是使用SQL Plus的主机连接符，或使用PL/SQL Developer工具的Database，一般是用Oracle的全局数据库名；主机名可以是计算机名称或IP；端口号为默认的1521；如下图，



注：安装完Oracle数据库服务后，要进行以上配置，否则使用SQL Plus或PL/SQL Developer等工具连接数据库时会报“tns无法解析指定的连接标识符”的错误。在使用PL/SQL Developer，登录时的Database列出的就是上面第三步创建的“连接服务名”，这个就像MySQL里的创建连接，只是MySQL创建的是服务器的连接，这里是针对每个数据库都要创建。一般新建数据库都会创建这个服务名。

6.**oracle命令**

|  |  |
| --- | --- |
| 查看表空间 | select \* from v$tablespace; |
| 查看所有用户 | select \* from dba\_users; select \* from all\_users; select \* from user\_users; |
| 创建用户并分配Temp表空间 | create user username identified by pwd default tablespace users Temporary TABLESPACE Temp; |
| 为用户授权 | grant 权限 to user; |
| 查看表空间名 | select username,default\_tablespace from dba\_users; |
| 删除用户命令 | drop user user\_name cascade; |
| 删除表空间，即使表空间中有数据 | drop tablespace TBS\_TEST including contents |
| 查看用户表空间的限额 | select \* from user\_ts\_quotas; |
| 查看当前用户权限 | select \* from session\_privs; |

7. **imp命令导入数据表**时出现了

IMP-00013: 只有 DBA 才能导入由其它 DBA 导出的文件   
IMP-00000: 未成功终止导入   
的错误，在网上找了一些解决方案，现找到一种解决方案，分享如下：   
  
imp时连接用户没有DBA权限或者default role不是DBA   
你可以换个有DBA role的用户IMP。   
如果你还想用这个用户，如果这个用户没有DBA权限就grant dba to 用户；   
如果default role不是DBA，那就 alter user 用户 default role DBA;   
如：grant dba to scott

8.在数据库下新建的用户只属于该数据库，除了默认的那几个用户外。

9. left join以左表为准，如果不满足on条件，左表的数据会列出。Inner join，如果on条件不满足，双方的数据都不会列出。

10. 进行字符串连接时，字符串要用单引号(‘’)，例如’hello’ || ‘luffy’

11. 双引号与单引号区别

双引号的作用是：假如建立对象的时候，对象名、字段名加双引号，则示意 Oracle将严格区分大小写，否则Oracl都默认大写。

单引号则示意：这个加了单引号的字段是一个字类似字符串，并不区分大小写。

单引号用于标识字符与数字的区别

当指定字符串文本时，必须用单引号将字符串文本引住。

Oracle10g之前，假如字符串文本里含有单引号，那么必须运用两个单引号示意。例如：I'm a String!

字符串文本则为：string\_var:='I''m a String!'

例如

　　select \* from table\_name where id=1;这种代表查询的是数字

　　select \* from table\_name where id='1';这种代表查询的是字符

两个'' 示意了一个'

12. Oracle中NVARCHAR2与VARCHAR2的区别

VARCHAR2是Oracle提供的特定数据类型，Oracle可以保证VARCHAR2在任何版本中该数据类型都可以向上和向下兼容。  
VARCHAR在Oracle中不建议使用。

具体到NVARCHAR2和VARCHAR2的区别，从使用角度来看区别在于：NVARCHAR2在计算长度时和字符集相关的，例如数据库是中文字符集时以长度10为例，则

1、NVARCHAR2(10)是可以存进去10个汉字的，如果用来存英文也只能存10个字符。

2、而VARCHAR2(10)的话，则只能存进5个汉字，英文则可以存10个

13. [对Oracle里NULL的理解](http://www.cnblogs.com/xryyforver/archive/2010/09/27/1836724.html)

(1). 在Oracle里，null是个相对独立的东西。它和所有非空的数据对立。对于字符串''来说，null既不等于'',也不不等于'';对于数值型来说，null既不等于0，也不不等于0;它不等于任何值，也不不等于任何值。也就是说null和所以其他的值没有可比性，即不能用等于、不等于、大 于或小于和其它数值比较，当然也包括空值本身，只能用is null,is not null比较。

(2). 如果使用带有其它比较操作符的条件表达式，并且其结果依赖于空值(如，在全量中找到某字段为空的所有记录，然后在此基础上再查找时，结果肯定为空。)，那么其结果必定是NULL。在where条件中，Oracle认为结果为NULL的条件为FALSE，带有这样条件的select语句不返回行， 也不返回错误信息。

(3). 往Oracle里插入''时，不会和sqlserver一样存入''，而是将这个字段设置为null，那么在查询这条记录的时候，如果查询''，则不会返回结果，条件里使用null才能找到。

执行select \* from t\_xy t where vald!=0 and vald is null，不返回记录,因为where条件依赖于null，则结果为null，整个where条件为false，select语句不返回行

执行select \* from t\_xy t where vald!=0 and vald is not null 返回不带空字段,并且vald!=0的记录因为where条件不依赖于null

执行select \* from t\_xy t where vald!=0 返回不带空字段,并且vald!=0的记录因为where条件不依赖于null，等同于上面的语句。(vald在和0比较的时候，Oracle自动排除掉 空值字段，所以可以不用加vald is not null条件。所以只要是满足vald!=0，则肯定是在not null的范围里相互比较,所以肯定查找不到is null的记录)

14. Oracle可以用Union all来合并两个表的数据，只要字段类型和个数一一对应即可。另外

可以用字段别名来作为最终结果集的字段名。例：

select a.F\_NB\_GLZDID sslkid, a.F\_VC\_ZDKSSJ sslkTime, a.F\_DT\_SJHFTC sjhftc from B\_GLZDXX a left join B\_GLZDLX b on a.F\_VC\_LB = b.F\_NB\_ZDLXID

union all

select x.F\_NB\_ID sslkid , x.F\_CR\_FABSJ sslkTime, sysdate sjhftc from B\_XWXX x

union all合并的结果集允许出来重复的记录，而union是无重复记录的。

15. Oracle排序的字段要在字段列表中有

16. Oracle安装目录\db\_1\NETWORK\ADMIN\tnsnames.ora，此文件中保存着oracle连接配置，也可以通过Net Manager进行管理。如果只有clint端，在对应的NETWORK目录中也有此文件配置。配置的连接在pl sql进行连接时，会出现在DataBase中。

17. [日期比较](http://www.blogjava.net/forker/archive/2007/09/07/143467.html)，首先将字符串转换为日期

在今天之前:

select \* from up\_date where update < to\_date('2007-09-07 00:00:00','yyyy-mm-dd hh24:mi:ss')  
select \* from up\_date where update <= to\_date('2007-09-07 00:00:00','yyyy-mm-dd hh24:mi:ss')

在今天只后:

select \* from up\_date where update > to\_date('2007-09-07 00:00:00','yyyy-mm-dd hh24:mi:ss')  
select \* from up\_date where update >= to\_date('2007-09-07 00:00:00','yyyy-mm-dd hh24:mi:ss')

精确时间:

select \* from up\_date where update = to\_date('2007-09-07 00:00:00','yyyy-mm-dd hh24:mi:ss')

在某段时间内:

select \* from up\_date where update between to\_date('2007-07-07 00:00:00','yyyy-mm-dd hh24:mi:ss') and to\_date('2007-09-07 00:00:00','yyyy-mm-dd hh24:mi:ss')  
select \* from up\_date where update < to\_date('2007-09-07 00:00:00','yyyy-mm-dd hh24:mi:ss') and update > to\_date('2007-07-07 00:00:00','yyyy-mm-dd hh24:mi:ss')  
select \* from up\_date where update <= to\_date('2007-09-07 00:00:00','yyyy-mm-dd hh24:mi:ss') and update >= to\_date('2007-07-07 00:00:00','yyyy-mm-dd hh24:mi:ss')

18. Oracle中start with...connect by prior子句用法

简单说来是将一个树状结构存储在一张表里，例：

select \* from table  
start with org\_id = 'HBHqfWGWPy'  
connect by prior org\_id = parent\_id;

详解请参照帮助文档oracle

19. 增大表空间一般通过两种方式: （一般在导入数据时，遇到表空间太小）

增大原有的数据文件:

alter database datafile 数据文件 resize 大小

为表空间增加新的数据文件:

alter tablespace 表空间名add datafile 数据文件 size 大小

20. ORA-01536: 超出表空间 'FOCUSYD' 的空间限额

例如，刚创建完表空间和对应的用户，然后程序在这个用户连接下，进行数据插入操作时会出现此问题，解决办法是增大用户对此空间的使用量或无限制的使用，如下：

**ALTER USER A QUOTA 50M ON users;** 或

**ALTER USER  A  QUOTA UNLIMITED ON users;**

详情参照oracle帮助文档：“超出了表空间users的空间限量.htm”

21. 随机查询记录

Oracle,随机查询20条

select \* from

(   
 select  \*  from 表名  
 order by dbms\_random.value

)

where rownum <= 20;

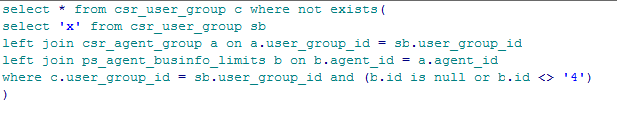
MS SQL Server，随机查询20条

select top 20  \* from  表名order by newid()

My SQL:，随机查询20条

select  \*  from  表名 order by rand() limit 20

22. 查询满足组中所有成员b.id不为空而且全是4条件的用户组



23. 将一列字符串转换为多行显示

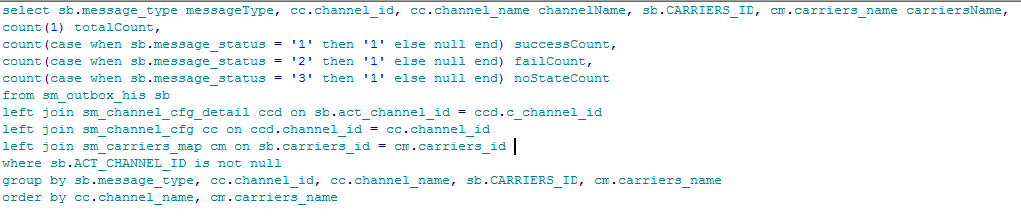


14. 删除重复记录：最高效的删除重复记录方法 (因为使用了ROWID)例子：

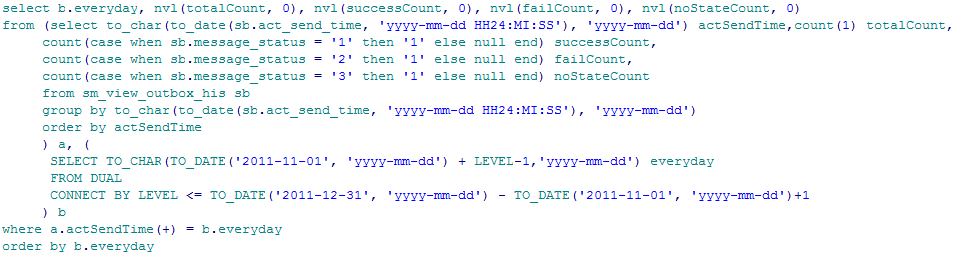
delete from sm\_draftbox a where a.rowid > (select min(b.rowid) from sm\_draftbox b where b.batch\_id = a.batch\_id)

注：rowid是oracle关键字，子查询中的where条件作用是以某个条件为基础判断哪些记录是重复记录。例子中是删除batch\_id字段值重复的记录

15.分组统计不同条件的数据



16.补全连续日期

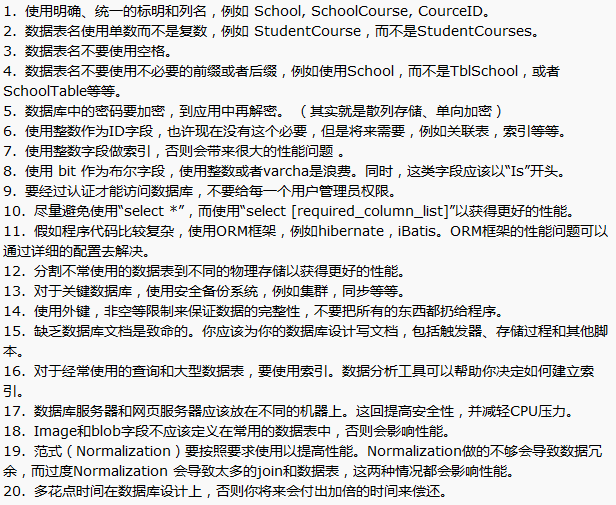


运用connect by level和nvl函数

17.连续数字



18. 20个数据库设计最佳实践



19. 取两张表不同的记录

select a.user\_id from wb\_friends\_followers a

minus

select b.user\_id from wb\_friends\_followers\_init b

20. Oracle 全文检索

详见帮助文档/oracle全文检索.htm

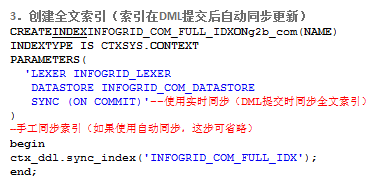
索引维护：对于CTXSYS.CONTEXT 索引，当应用程序对基表进行DML 操作后，对基表的

索引维护是必须的。索引维护包括索引同步和索引优化。当基表中的被索引文档发生insert、

update、delete 操作的时候，基表的改变并不能马上影响到索引上直到同步索引。

可以使用Oracle 提供的ctx\_ddl 包同步和优化索引，使用job 定时同步和优化。

下面是在创建索引时，设置同步更新



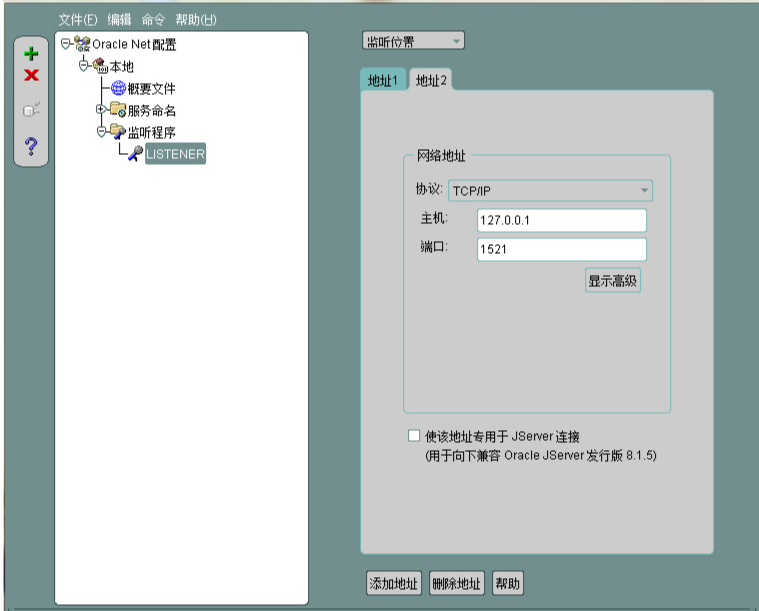
21. PLSQL **ORA-12514: TNS: 监听程序当前无法识别连接描述符中请求的服务**

第一种通过窗口配置方法：

打开配置和移植工具下的net manager->本地->监听程序

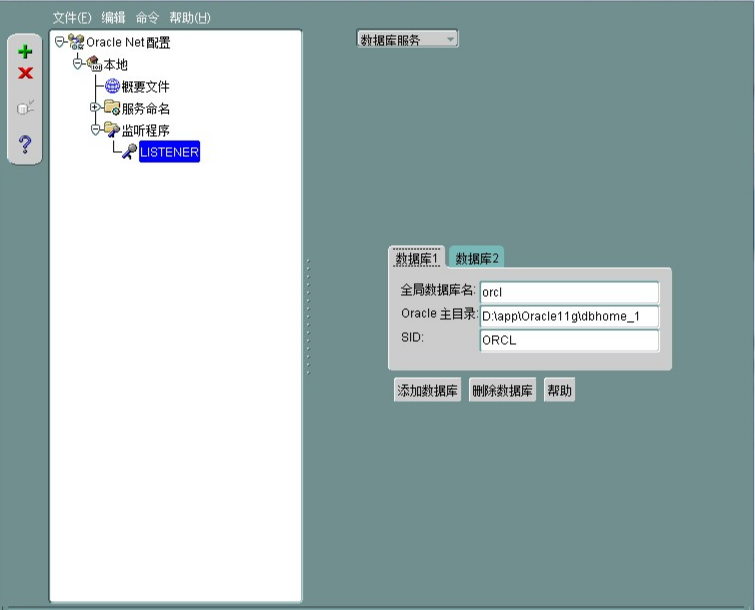
如果有默认的listener，选中右边下拉菜单选监听位置，选地址2，协议：TCP/IP，

主机：服务端所在的机器IP，端口默认1521，（自己视情况而定也可）



选上面下拉菜单数据库服务，全局数据库名，填安装时写的全局数据库名，oracle主目录

填oracle安装所在硬盘上的目录，我的是D:\app\Oracle11g，SID填安装时的那个SID



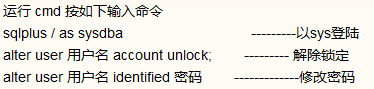
最后，保存并重启listener

第二种通过配置文件方法：

修改NETWORK\ADMIN\listener.ora文件，添加红色部分（根据自身配置而定），改完后重启Listener

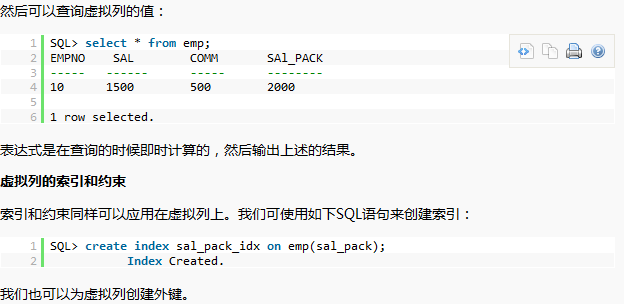


22. 忘记Oracle system用户密码，更改方式如下：



23. Oracle 11g 数据库的新特性 —— 虚拟列



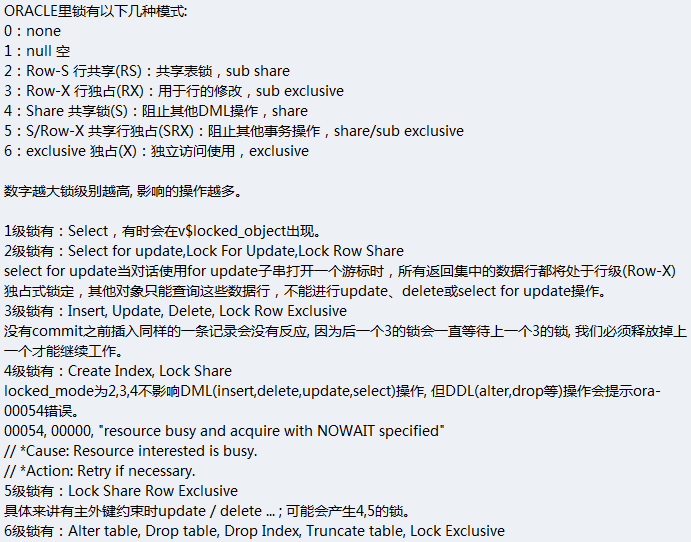


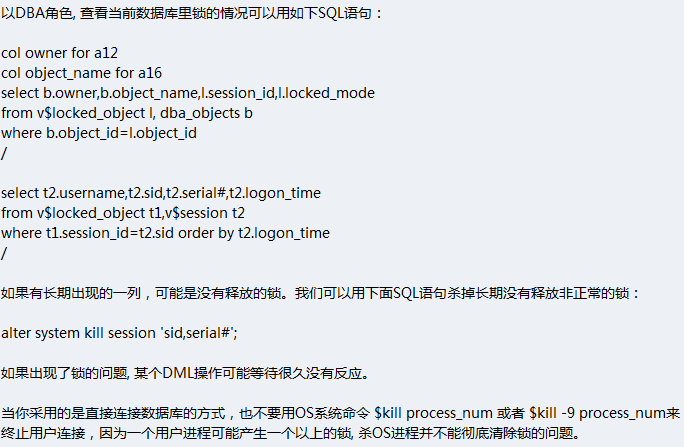
24．ORACLE监控最新执行过的SQL（操作），通过下面查看：

select t.SQL\_TEXT, t.LAST\_ACTIVE\_TIME from v$sqlarea t order by t.LAST\_ACTIVE\_TIME desc

25. ORACLE 锁表的解决方法及查找引起锁表SQL语句方法







获得这个连接正在执行的sql文

select b.sql\_text from v$session a, v$sql b where a.sid=12 and a.SQL\_ADDRESS=b.ADDRESS(+);

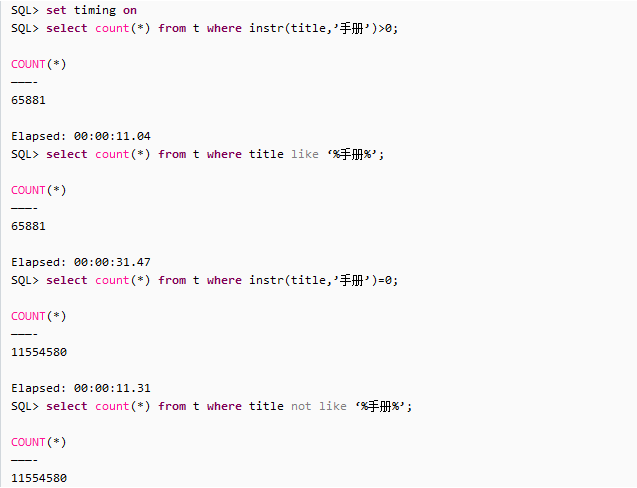
26．select count(distinct(a.customer\_id)) AS COUNT

去重后进行统计

27. [数据库导出备份的时候部分表不能导出](http://bicashy.iteye.com/blog/1064344)

问题就在于版本是11g的，因为11G中有个新特性，当表无数据时，不分配segment，以节省空间，所以导不出，如果重来没有使用过的空表默认情况下是导不出来的，详见帮助文档。或者可以使用expdp命令导，详见帮助文档。

28. oracle用instr代替like 效率比拼



29. Oracle添加、更改字段、类型

a. 添加字段

alter table SM\_CONTENT add (ATTACHIDS VARCHAR2(64), FILE\_PATHANDNAME VARCHAR2(2000));

b. 修改字段类型

alter table MYFAVORITE modify(MY\_ID varchar2(32));

c. 如果修改的类型差异很大（NUMBER->VARCHAR），上面那种方法行不通，需要：

/\*修改原字段名\*/

ALTER TABLE SM\_SEARCH\_HISTORY RENAME COLUMN SEARCH\_ID TO SEARCH\_ID1;

/\*添加一个和原字段同名的字段\*/

ALTER TABLE SM\_SEARCH\_HISTORY ADD SEARCH\_ID VARCHAR2(32);

/\*

将原来的数据更新到新字段中

这是要注意，一定要显示进行数据类型转换（不同于MSSQL）

\*/

UPDATE SM\_SEARCH\_HISTORY SET SEARCH\_ID = CAST(SEARCH\_ID1 AS VARCHAR2(32));

/\*删除原来的备份字段\*/

ALTER TABLE SM\_SEARCH\_HISTORY DROP COLUMN SEARCH\_ID1;

如果修改的字段是主键，还要：

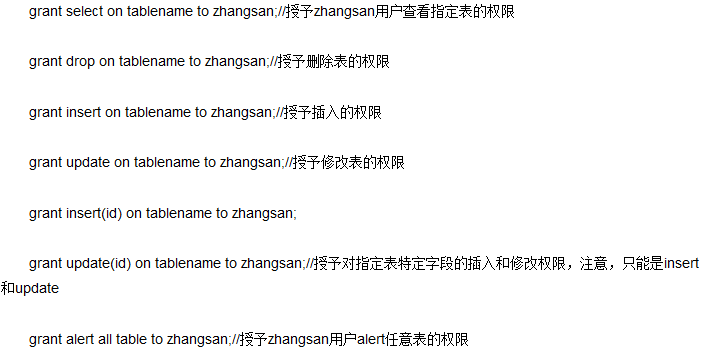
alter table SM\_SEARCH\_HISTORY modify(SEARCH\_ID VARCHAR2(32) not null);

alter table SM\_SEARCH\_HISTORY add constraint PK\_SM\_SEARCH\_HISTORY primary key (SEARCH\_ID);

30. 表空间无法创建INITIAL区



31. oracle对权限管理比较严谨，普通用户之间也是默认不能互相访问的，需要互相授权



32. [PL/SQL Developer 连接oracle 客户端与服务端字符编码不一致](http://aguu125.iteye.com/blog/482630)

