目录

[1. 删除重复记录 2](#_Toc532314838)

[2. 导出导入数据 2](#_Toc532314839)

[3. 日期 3](#_Toc532314840)

[4. 查询前N条记录 3](#_Toc532314841)

[5. oracle账户过期修改 4](#_Toc532314842)

[6. oracle授权 4](#_Toc532314843)

[7. 查看用户（实例）所占空间大小 5](#_Toc532314844)

[8. oracle复制表数据，复制表结构 6](#_Toc532314845)

[9. 使用flashback（闪回）恢复误删除的数据 或 误删除的表 7](#_Toc532314846)

[10. delete 删除数据恢复：（必须9i或10g以上版本支持，flashback无法恢复全文索引） 8](#_Toc532314847)

[11. 表被锁定 9](#_Toc532314848)

[12. Oracle表分区 9](#_Toc532314849)

[13. archive log 日志已满处理方法 11](#_Toc532314850)

# ****删除重复记录****

* 1. 释放表删除数据空间

truncate table tablename DROP STORAGE;

* 1. 查找表中多余的重复记录，重复记录是根据单个字段（Id）来判断

select \* from 表 where Id in (select Id from 表 group byId having count(Id) > 1

* 1. 删除表中多余的重复记录，重复记录是根据单个字段（Id）来判断，只留有rowid最小的记录

DELETE from 表 WHERE (id) IN ( SELECT id FROM 表 GROUP BY id HAVING COUNT(id) > 1) AND ROWID NOT IN (SELECT MIN(ROWID) FROM 表 GROUP BY id HAVING COUNT(\*) > 1);

* 1. 查找表中多余的重复记录（多个字段）

select \* from 表 a where (a.Id,a.seq) in(select Id,seq from 表 group by Id,seq having count(\*) > 1)

* 1. 删除表中多余的重复记录（多个字段），只留有rowid最小的记录

delete from 表 a where (a.Id,a.seq) in (select Id,seq from 表 group by Id,seq having count(\*) > 1) and rowid not in (select min(rowid) from 表 group by Id,seq having count(\*)>1)

* 1. 查找表中多余的重复记录（多个字段），不包含rowid最小的记录

select \* from 表 a where (a.Id,a.seq) in (select Id,seq from 表 group by Id,seq having count(\*) > 1) and rowid not in (select min(rowid) from 表 group by Id,seq having count(\*)>1)

delete from 表 a where (a.Id,a.seq) in (select Id,seq from 表 group by Id,seq having count(\*) > 1) and rowid not in (select min(rowid) from 表 group by Id,seq having count(\*)>1)

delete from BT\_SCALE\_MAIN a

where (a.intime) in (select intime from BT\_SCALE\_MAIN group by intime having count(intime) > 1)

and rowid not in (select min(rowid) from BT\_SCALE\_MAIN group by intime having count(intime)>1)

# ****导出导入数据****

第一种：删除用户

--删除用户

drop user scts cascade;

--创建用户并授权（根据实际权限）

create user scts3 identified by "scts3";

grant create session to scts3;

grant connect,resource,imp\_full\_database,exp\_full\_database,dba to scts3;

-- 导入导出直接在命令行里执行（不需要登录oracle用户）

exp scts/scts owner=scts file=scts\_2016-07-26.dmp

imp scts3/scts3 file=scts2016-07-31\_qiu.dmp full=y

第二种：直接导入导出

导出

exp lsreport/lsreport@129db file=E:\导出.dmp

①完全导出 full=y

②用户1和用户2的表导出 owner=(用户1,用户2)

③将表1和表2导出 tables=(table1,table2)

④将表1中字段1以“00” 开头的数据导出 tables=(table1) query=\"where 字段1 like '00%'\"

导入

imp ylglxt/ylglxt@orcl fromuser=lsreport touser=ylglxt file=e:\20160425data.dmp

fromuser=lsreport 导出用户名

touser=ylglxt 导入用户名

# ****日期****

delete from BT\_SCALE\_MAIN a where to\_date(a.intime , 'yyyy-mm-dd hh24:mi:ss') <(sysdate-4)

输出当月的所有天数

SELECT TRUNC (SYSDATE, 'MM') + ROWNUM - 1 FROM DUAL CONNECT BY ROWNUM <= TO\_NUMBER (TO\_CHAR (LAST\_DAY (SYSDATE), 'dd'))

select to\_char(sysdate,'yyyy-mm-dd hh24:mi:ss') as nowTime from dual;

# ****查询前N条记录****

* 1. 查询前10条记录

SELECT \*　FROM torderdetail a WHERE ROWNUM <= 10

* 1. 查询10到20条（先排序后取数据）

SELECT \* FROM (SELECT a.\*, ROWNUM rn FROM torderdetail a)　WHERE rn >= 10 AND rn <= 20

* 1. 对于分组后取最近的10条纪录，则是rownum无法实现的，这时只有row\_number可以实现，row\_number() over(partition by 分组字段 order by 排序字段)就能实现分组后编号，比如说要取近一个月的每天最后10个订单纪录

SELECT \* FROM (SELECT a.\*, ROW\_NUMBER () OVER (PARTITION BY TRUNC (order\_date) ORDER BY order\_date DESC) rn FROM torderdetail a)　WHERE rn <= 10

# ****oracle账户过期修改****

* 1. 查看用户的proifle是哪个，一般是default：

SELECT username,PROFILE FROM dba\_users;

* 1. 查看指定概要文件（如default）的密码有效期设置：

SELECT \* FROM dba\_profiles s WHERE s.profile='DEFAULT' AND resource\_name='PASSWORD\_LIFE\_TIME';

* 1. 修改为无限期；修改之后不需要重启动数据库，会立即生效。

ALTER PROFILE DEFAULT LIMIT PASSWORD\_LIFE\_TIME UNLIMITED;

* 1. 修改后，还没有被提示ORA-28002警告的帐户不会再碰到同样的提示； 已经被提示的帐户必须再改一次密码

alter user smsc identified by <原来的密码> ----不用换新密码

# ****oracle授权****

系统权限授权命令：grant connect, resource, dba to 用户名1 [,用户名2]...;

系统权限传递：增加WITH ADMIN OPTION选项，则得到的权限可以传递。

grant connect, resorce to user50 with admin option; //可以传递所获权限。

系统权限回收：系统权限只能由DBA用户回收

Revoke connect, resource from user50;

实体权限分类：select, update, insert, alter, index, delete, all //all包括所有权限

user01:

SQL> grant select, update, insert on product to user02;

SQL> grant all on product to user02;

将表的操作权限授予全体用户：

SQL> grant all on product to public; // public表示是所有的用户，这里的all权限不包括drop。

实体权限传递(with grant option)：

user01:

SQL> grant select, update on product to user02 with grant option; // user02得到权限，并可以传递。

实体权限回收：

user01:

SQL>Revoke select, update on product from user02; //传递的权限将全部丢失。

# ****查看用户（实例）所占空间大小****

* 1. select tablespace\_name ,sum(bytes) / 1024 / 1024 as MB　from dba\_data\_files group by tablespace\_name;
  2. select tablespace\_name,file\_name from dba\_data\_files;
  3. select b.file\_name 物理文件名,

b.tablespace\_name 表空间,

b.bytes/1024/1024 大小M,

(b.bytes-sum(nvl(a.bytes,0)))/1024/1024 已使用M,

substr((b.bytes-sum(nvl(a.bytes,0)))/(b.bytes)\*100,1,5) 利用率

from dba\_free\_space a,dba\_data\_files b

where a.file\_id=b.file\_id

group by b.tablespace\_name,b.file\_name,b.bytes

order by b.tablespace\_name;

* 1. 各个实例所占空间

select \*

from (select owner || '.' || tablespace\_name name, sum(b) g

from (select owner,

t.segment\_name,

t.partition\_name,

round(bytes / 1024 / 1024 / 1024, 2) b,

tablespace\_name

from dba\_segments t)

where owner not in

('SYS', 'OUTLN', 'SYSTEM', 'TSMSYS', 'DBSNMP', 'WMSYS')

group by owner || '.' || tablespace\_name)

order by name

# oracle复制表数据，复制表结构

* 1. 不同用户之间的表数据复制

对于在一个数据库上的两个用户A和B，假如需要把A下表old的数据复制到B下的new，请使用权限足够的用户登入sqlplus：

insert into B.new(select \* from A.old);

如果需要加条件限制，比如复制当天的A.old数据

insert into B.new(select \* from A.old where date=GMT);

蓝色斜线处为选择条件

* 1. .同用户表之间的数据复制?

用户B下有两个表：B.x和B.y，如果需要从表x转移数据到表y，使用用户B登陆sqlpus即可：

insert into?目标表y select \* from x where log\_id>'3049' --?复制数据?

注意：要示目标表y必须事先创建好

如insert into bs\_log2 select \* from bs\_log where log\_id>'3049'

* 1. B.x中个别字段转移到B.y的相同字段?

--如果两个表结构一样

insert into table\_name\_new select \* from table\_name\_old

如果两个表结构不一样：

insert into y(字段1,字段2) select?字段1,字段2 from x

* 1. .只复制表结构 加入了一个永远不可能成立的条件1=2，则此时表示的是只复制表结构，但是不复制表内容???

create table?用户名.表名?as select \* from?用户名.表名?where 1=2

如create table zdsy.bs\_log2 as select \* from zdsy.bs\_log where 1=2

* 1. 完全复制表(包括创建表和复制表中的记录)

create table test as select \* from bs\_log??--bs\_log是被复制表

* 1. 将多个表数据插入一个表中

insert into 目标表test(字段1。。。字段n) (select?字段1.。。。。字段n) from 表 union all select?字段1.....字段n from 表

* 1. 、创建用户budget\_zlgc，权限和budget相同,（A、只复制所有表结构

B、复制所有表所有信息）

创建用户budget\_zlgc，并导出budge用户数据

exp userid="\"sys/sys as sysdba"\" file='/backup/expdb/oa0824.dmp' log='/backup/expdb/oaex0825.log' owner=budget ignore=Y buffer=256000000

?

A、budget用户所有表，表结构和budget相同，（无数据）

imp userid="\"sys/sys as sysdba"\" file='/backup/expdb/oa0824.dmp' log='/backup/expdb/oa0825.log' fromuser=budget touser=budget\_zlgc ignore=Y buffer=256000000 rows=n

B、budget用户所有表，表结构、数据和budget相同

imp userid="\"sys/sys as sysdba"\" file='/backup/expdb/oa0824.dmp' log='/backup/expdb/oa0825.log' fromuser=budget touser=budget\_zlgc ignore=Y

# 使用flashback（闪回）恢复误删除的数据 或 误删除的表

1、从回收站恢复时重命名表

SQL> flashback table t2 to before drop rename to t4 ;

2、删除回收站指定的表

SQL> purge table t4;

3、清空回收站

SQL> purge recyclebin

4、删除表时不经过回收站直接删除

SQL> drop table t3 purge ;

5、从回收站看删除的表

select \* from user\_recyclebin

①恢复表中误删除的记录

前提：1.表的结构未改动;如果在删除后表结构发生改动则不能使用闪回；

2.用户必须有足够的权限

直接用个例子来说明更加直观：

表：

create table TEST1

(

ID NUMBER(10) not null,

CREATE\_DATE DATE

)

SQL> select count(\*) from test1;

COUNT(\*)

----------

57158

SQL> delete test1 where id<1000;

已删除999行。

SQL> select count(\*) from test1;

COUNT(\*)

----------

56159

SQL>select sysdate from dual;

SYSDATE

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

2013-06-03 15:35:57

//开始恢复数据

SQL> alter table test1 enable row movement;//在闪回前必须 启动行移动功能 否则会报错误： ORA-08189: 因为未启用行移动功能, 不能闪回表

表已更改。

SQL> FLASHBACK TABLE test1 TO TIMESTAMP to\_timestamp('2013-06-03 15:35:00','yyyy-mm-dd hh24:mi:ss');

//注意：恢复时间点必须是在删除数据之前 这里是2013-06-03 15:35:57 之前就可以

闪回完成。

SQL> select count(\*) from test1;

COUNT(\*)

----------

57158

数据已经恢复成功，57158行 跟未删除前是一样的，如果想看被删除了哪些行（在删除后闪回前）：

SELECT \* FROM test1 AS OF TIMESTAMP to\_timestamp('2013-06-03 15:35:00','yyyy-mm-dd hh24:mi:ss')

MINUS

SELECT \* FROM test1

②恢复被删除的表

SQL> drop table test1;

表已删除。

SQL> flashback table test1 to before drop;

闪回完成。

# delete 删除数据恢复：（必须9i或10g以上版本支持，flashback无法恢复全文索引）

利用Oracle提供的闪回方法，如果在删除数据后还没做大量的操作（只要保证被删除数据的块没被覆写），就可以利用闪回方式直接找回删除的数据

具体步骤为：

\*确定删除数据的时间（在删除数据之前的时间就行，不过最好是删除数据的时间点）

\*用以下语句找出删除的数据：select \* from 表名 as of timestamp to\_timestamp('删除时间点','yyyy-mm-dd hh24:mi:ss')

\*把删除的数据重新插入原表：

insert into 表名 (select \* from 表名 as of timestamp to\_timestamp('删除时间点','yyyy-mm-dd hh24:mi:ss'));注意要保证主键不重复。

# 表被锁定

ORACLE EBS操作某一个FORM界面，或者后台数据库操作某一个表时发现一直出于"假死"状态，可能是该表被某一用户锁定，导致其他用户无法继续操作

复制代码 代码如下:

--锁表查询SQL

SELECT object\_name, owner,machine, object\_type,s.sid, s.serial#

FROM gv$locked\_object l, dba\_objects o, gv$session s

WHERE l.object\_id = o.object\_id AND l.session\_id = s.sid;

找到被锁定的表，解锁

复制代码 代码如下:--释放SESSION SQL:

--alter system kill session 'sid, serial#';

# Oracle表分区

创建表及分区

create table DT\_ELECTRICITY\_MINUTEDATA

(

rtuid INTEGER,

mp\_id INTEGER,

rtudesc NVARCHAR2(100),

"date" INTEGER,

time INTEGER

)

PARTITION BY RANGE("date")

(

PARTITION elec\_min\_part201612 VALUES LESS THAN (20170101) ,

PARTITION elec\_min\_part201701 VALUES LESS THAN (20170201) ,

PARTITION elec\_min\_part201702 VALUES LESS THAN (20170301) ,

PARTITION elec\_min\_part201703 VALUES LESS THAN (20170401) ,

PARTITION elec\_min\_part201704 VALUES LESS THAN (20170501) ,

PARTITION elec\_min\_part201705 VALUES LESS THAN (20170601) ,

PARTITION elec\_min\_part201706 VALUES LESS THAN (20170701) ,

PARTITION elec\_min\_part201707 VALUES LESS THAN (20170801)

);

create index DT\_ELEC\_MINDATABACK\_INDEX on DT\_ELECTRICITY\_MINUTEDATA (RTUID, MP\_ID, "date", TIME)

tablespace USERS

pctfree 10

initrans 2

maxtrans 255

storage

(

initial 64K

next 1M

minextents 1

maxextents unlimited

)

--ALTER TABLE DT\_ELECTRICITY\_MINUTEDATA EXCHANGE PARTITION elec\_min\_part201602 WITH TABLE DT\_ELECTRICITY\_MINUTEDATA\_BACK;--交换分区

--alter table DT\_ELECTRICITY\_MIN add partition elec\_minute\_part201708 VALUES LESS THAN (20170901);--添加分区

-- SELECT COUNT(\*) FROM DT\_ELECTRICITY\_MINUTEDATA PARTITION (elec\_min\_part201601);

/\* alter table DT\_ELECTRICITY\_MINUTEDATA\_BACK

exchange partition elec\_min\_part201602

with table DT\_ELECTRICITY\_MINUTEDATA without validation\*/

--收集表的统计信息

--exec dbms\_stats.gather\_table\_stats('ENERGY', 'DT\_ELECTRICITY\_MINUTEDATA', cascade => true);

--检查重定义的合理性

--exec dbms\_redefinition.can\_redef\_table('ENERGY', 'DT\_ELECTRICITY\_MINUTEDATA');

SELECT \* FROM dba\_part\_tables where owner='ENERGY';--查看用户那些表有分区

SELECT \* FROM dba\_tab\_partitions WHERE table\_name = 'DT\_ELECTRICITY\_MINUTEDATA' and partition\_name='ELEC\_MIN\_PART201707';--查找表的分区

SELECT \* FROM dba\_ind\_partitions WHERE index\_name = 'DT\_ELECTRICITY\_MINUTEDATA';--查找索引

--INSERT INTO DT\_ELECTRICITY\_MINUTEDATA SELECT \* FROM DT\_ELECTRICITY\_MINUTEDATA\_BACK

-- ALTER TABLE DROP PARTITION --来删除分区，元数据和数据将被一并删除。-全删除

--ALTER TABLE DT\_ELECTRICITY\_MINUTEDATA DROP PARTITION ELEC\_MINUTE\_PART201708;--删除不需要的分区

--ALTER TABLE yourTable TRUNCATE PARTITION partionName1;

--语句虽简单、操作需谨慎。

# archive log 日志已满处理方法

* 1. 错误提示：ORA-00257: archiver error. Connect internal only, until freed
  2. Oracle archive log归档文件地址:

D:\app\Administrator\flash\_recovery\_area\YULANGDB\ARCHIVELOG

* 1. select \* from V$RECOVERY\_FILE\_DEST; //查询归档日志空间大小及路径

show parameter recover; //显示归档文件路径

进入rman进行删除归档，相关命令如下：

rman target sys/lsreport@yulangdb--登录rman

RMAN>list archive log --查看archivelog 列表

RMAN>show parameter log\_archive\_dest;--查看archivelog日志位置

RMAN>delete expired archivelog all; //删除所有归档日志

RMAN>DELETE ARCHIVELOG ALL COMPLETED BEFORE ‘SYSDATE-7’; //保留7天的归档日志

关闭归档

SQL> alter system set log\_archive\_start=false scope=spfile; #禁用自归档

SQL> shutdown immediate; //强制关闭数据库

SQL> startup mount; //重启数据库到mount模式

SQL> alter database noarchivelog; //修改为非归档模式

SQL> alter database open; //打数据文件

SQL> archive log list; //再次查看前归档模式