# Oracle实例安装

cd /oracle/product/12cR2/db/network/admin/ 相关地址修改，修改监听信息

12c 工具连接不上配置文件sqlnet.ora，没有测创建

|  |
| --- |
| SQLNET.ALLOWED\_LOGON\_VERSION\_SERVER=8  SQLNET.ALLOWED\_LOGON\_VERSION\_CLIENT=8 |

export ORACLE\_SID=DCITS 设置实例名

# 一些sql操作

## 一些系统自带表的说明

all\_tab\_comments 查询所有用户下表视图等

user\_tab\_comments 查询当前用户下表视图等

all\_col\_comments 查询所有用户下表的列注释信息

user\_col\_comments 查询本用户下的列名和注释信息

all\_tab\_columns 查询所有用户下的列名类型信息

all\_tab\_columns 查询当前用户下的列名类型信息

### 查询表的注释信息

查询指定用户指定表的信息

|  |
| --- |
| select a.comments,a.table\_name,b.column\_name,b.comments as col\_desc from all\_tab\_comments a  inner join all\_col\_comments b on b.owner='ENS\_CBANK' and a.table\_name=b.table\_name where a.table\_name='FM\_BRANCH' |

查询当前用户下指定表信息

|  |
| --- |
| select a.comments,a.table\_name,b.column\_name,b.comments as col\_desc from user\_tab\_comments a  inner join user\_col\_comments b on a.table\_name=b.table\_name where a.table\_name='FM\_BRANCH' |

## 存储过程

### IF

|  |
| --- |
| IF 条件  THEN  ELSIF 条件  THEN  ELSE  END IF; |

### FOR

|  |
| --- |
| FOR 起始变量 IN 结束变量  LOOP  END LOOP; |

### merge into

|  |
| --- |
| MERGE INTO [target-table] A USING [source-table sql] B ON([conditional expression] and [...]...)  WHEN MATCHED THEN  [UPDATE sql]  WHEN NOT MATCHED THEN  [INSERT sql] |

作用:判断Ｂ表和Ａ表是否满足ON中条件，如果满足则用B表去更新A表，如果不满足，则将B表数据插入A表但是有很多可选项，如下:

1.正常模式

2.只update或者只insert

3.带条件的update或带条件的insert

4.全插入insert实现

5.带delete的update(觉得可以用3来实现)

例如：

|  |
| --- |
| MERGE INTO T T1  USING (SELECT '1001' AS a,**2** AS b FROM dual) T2  ON ( T1.a=T2.a)  WHEN MATCHED THEN  UPDATE SET T1.b = T2.b  WHEN NOT MATCHED THEN  INSERT (a,b) VALUES(T2.a,T2.b); |

### 异常捕获

|  |
| --- |
| BEGIN    EXCEPTION  WHEN OTHERS  THEN  ***NULL***  END; |

### 其他

CURSOR 游标

DROP TYPE PT2\_ENTRUSTGATHERING\_O\_T FORCE 取消关联

EXECUTE IMMEDIATE 执行sql 字符串

Rownum 行

CREATE UNIQUE INDEX 索引名称 ON 表名 (列名); 创建索引

select to\_char(1,'9999990.00') as a from dual 保留2位小数

trunc(10.119, 2) 舍去后面保留2位

ROUND(1.115,2) 四舍五入保留2位

DISTINCT 去重复

MINUS 减去的部分

全库模糊搜索用到的存储过程

select \* from user\_source where upper(text) like '%RB\_SEAL\_RELATION%'

(CASE WHEN dr\_amt > 0 THEN dr\_amt ELSE cr\_amt END) AS tran\_amt

alter table 表名 modify column\_name varchar2(32) 增加表列的长度

删除用户会同时删除用户下的所有资源

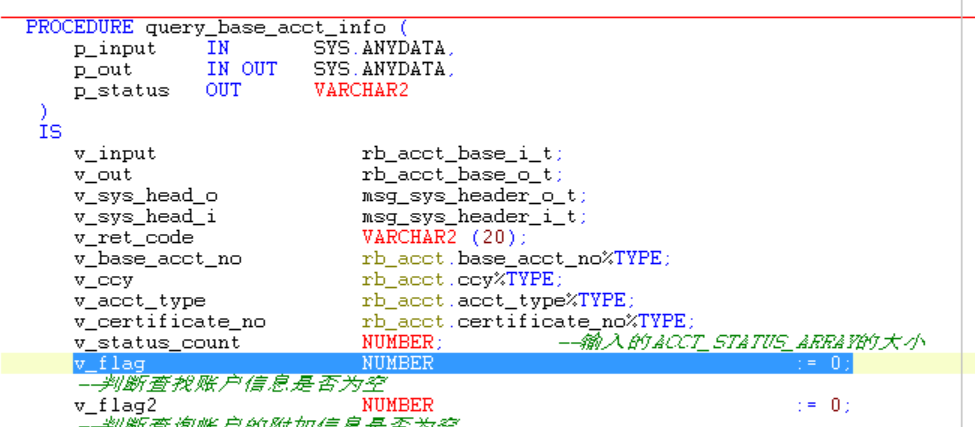
替换表中相关数据

update FM\_BRANCH set branch\_name=replace(branch\_name,'重庆银行','神州银行');

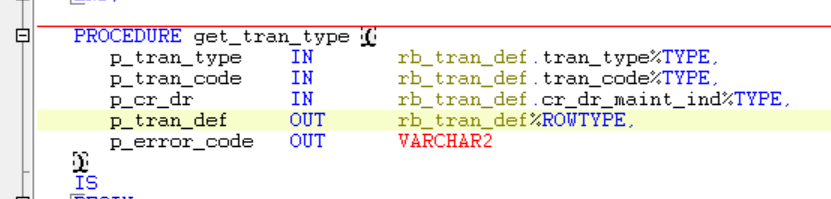
### 给方法或者过程设置默认值

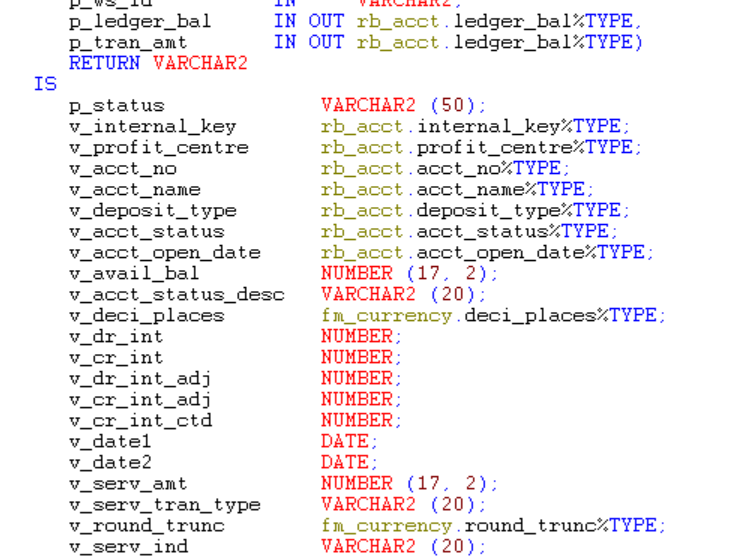
|  |
| --- |
| \_orig\_channal\_seq\_no IN VARCHAR2 DEFAULT NULL, |

|  |
| --- |
| v\_flag NUMBER := 0; |



### 设置过程的入参出参





### 创建用户

|  |
| --- |
| DROP USER ENS\_CBANK CASCADE;--删除用户  CREATE USER ENS\_CBANK  IDENTIFIED BY <password>  DEFAULT TABLESPACE CORE\_DATA  TEMPORARY TABLESPACE CORE\_DATA\_TEMP  PROFILE DEFAULT  ACCOUNT UNLOCK;  -- 2 Roles for ENS\_CBANK  GRANT CONNECT TO ENS\_CBANK WITH ADMIN OPTION;--连接权限  GRANT RESOURCE TO ENS\_CBANK WITH ADMIN OPTION;  ALTER USER ENS\_CBANK DEFAULT ROLE ALL;--默认所有角色  -- 11 System Privileges for ENS\_CBANK  GRANT ALTER ANY TABLE TO ENS\_CBANK;--改表权限  GRANT ALTER SESSION TO ENS\_CBANK WITH ADMIN OPTION;  GRANT CREATE TABLE TO ENS\_CBANK WITH ADMIN OPTION;--创建表权限  GRANT DELETE ANY TABLE TO ENS\_CBANK;--删除表权限  GRANT DROP ANY INDEX TO ENS\_CBANK;--删除索引权限  GRANT DROP ANY TABLE TO ENS\_CBANK WITH ADMIN OPTION;--删除表权限  GRANT INSERT ANY TABLE TO ENS\_CBANK;--插入数据权限  GRANT LOCK ANY TABLE TO ENS\_CBANK;--锁表权限  GRANT SELECT ANY TABLE TO ENS\_CBANK WITH ADMIN OPTION;--查询权限  GRANT UNLIMITED TABLESPACE TO ENS\_CBANK WITH ADMIN OPTION;  GRANT UPDATE ANY TABLE TO ENS\_CBANK;--修改权限 |

#### 创建只读用户

|  |
| --- |
| CREATE USER READDER  IDENTIFIED BY READDER  DEFAULT TABLESPACE CORE\_DATA  TEMPORARY TABLESPACE CORE\_DATA\_TEMP  PROFILE DEFAULT  ACCOUNT UNLOCK;  -- 2 Roles for READDER  GRANT CONNECT TO READDER WITH ADMIN OPTION;  ALTER USER READDER DEFAULT ROLE ALL;  GRANT SELECT ANY TABLE TO READDER WITH ADMIN OPTION; |

## Oracle操作

### 查看日志

select \* from v$logfile 查看日志路径

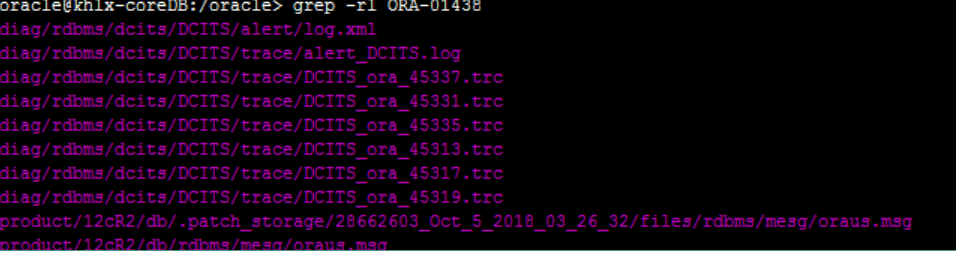
然后使用工具logMiner查看

#### 查看报错sql日志

Sqlplus / as sysdba 进入控制台设置如下

alter system set events='1438 trace name Errorstack forever,level 10'; 查看报错的sql

设置完之后执行sql可以查看对应的日志信息



### Oracle游标操作

查看游标的使用情况

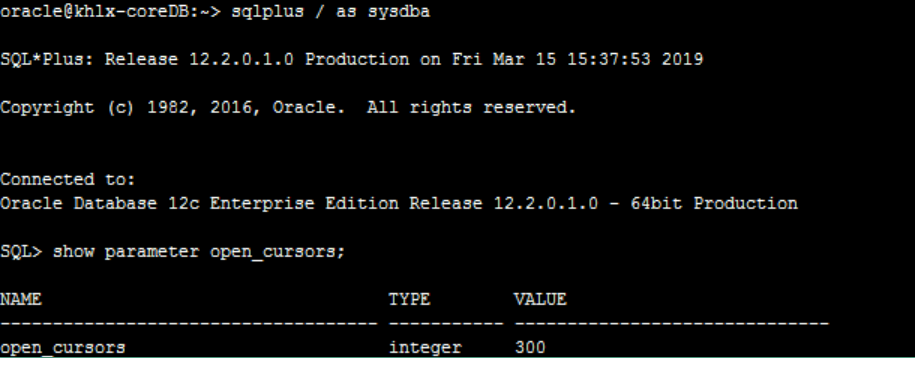
|  |
| --- |
| select user\_name,o.sid,osuser,machine,count(\*) num\_curs from v$open\_cursor o,v$session s  where user\_name ='ENS\_CBANK' and o.sid=s.sid group by user\_name,o.sid,osuser,machine order by num\_curs desc |

查看游标sid执行的sql执行明细

|  |
| --- |
| select q.sql\_text from v$open\_cursor o,v$sql q where Q.HASH\_VALUE=O.HASH\_VALUE and o.sid='837' |

查看当前Oracle最大游标数

show parameter open\_cursors;



查看当前打开的游标数目

select count(1) from v$open\_cursor

修改Oracle最大游标数

alter system set open\_cursors=1000 scope=both;

### Oracle一些命令操作

Sqlplus / as sysdba

Startup 启动

Shutdown 停止

lsnrctl start 启动监听

impdp –help 查看参数

expdp -help 查看参数

history |grep impdp

impdp directory=DUMP\_FILE\_PATH dumpfile=eis\_20171117.dmp logfile=eis\_20171117.log schemas=eis

expdp directory=impdp dumpfile=ucbsdb01\_ac\_subject.dump logfile=ucbsdb01\_ac\_subject.log tables=ensemble.ac\_subject

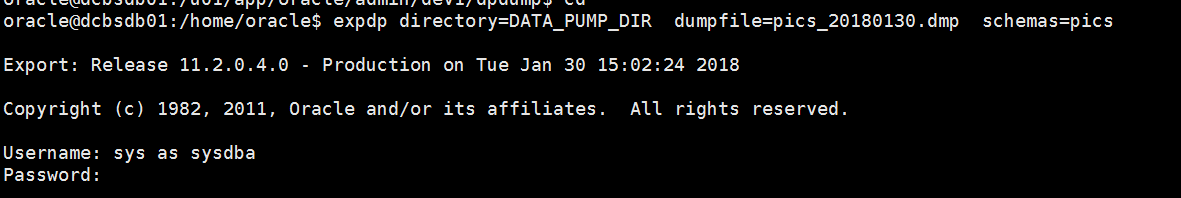
create directory dpdata as 'd:\test\dump';

select \* from dba\_directories;

select directory\_path,directory\_name from dba\_directories;

select tablespace\_name from dba\_tablespaces; 查看表空间都有哪些

select file\_name from dba\_data\_files; 查看dbf数据文件



alter user sys identified by oracle; 修改sys密码

### Oracle数据导入导出备份方式

使用EXPDP和IMPDP时应该注意的事项：

EXP和IMP是客户端工具程序，它们既可以在客户端使用，也可以在服务端使用。

EXPDP和IMPDP是服务端的工具程序，他们只能在ORACLE服务端使用，不能在客户端使用。

IMP只适用于EXP导出的文件，不适用于EXPDP导出文件；IMPDP只适用于EXPDP导出的文件，而不适用于EXP导出文件。

expdp或impdp命令时，可暂不指出用户名/密码@实例名 as 身份，然后根据提示再输入，如：

expdp schemas=scott dumpfile=expdp.dmp DIRECTORY=dpdata1;

一、创建逻辑目录，该命令不会在操作系统创建真正的目录，最好以system等管理员创建。

create directory dpdata1 as 'd:\test\dump';

二、查看管理理员目录（同时查看操作系统是否存在，因为Oracle并不关心该目录是否存在，如果不存在，则出错）

select \* from dba\_directories;

三、给scott用户赋予在指定目录的操作权限，最好以system等管理员赋予。

grant read,write on directory dpdata1 to scott;

四、导出数据

1)按用户导

expdp [scott/tiger@orcl](mailto:scott/tiger@orcl) schemas=scott dumpfile=expdp.dmp DIRECTORY=dpdata1;

2)并行进程parallel

expdp [scott/tiger@orcl](mailto:scott/tiger@orcl) directory=dpdata1 dumpfile=scott3.dmp parallel=40 job\_name=scott3

3)按表名导

expdp [scott/tiger@orcl](mailto:scott/tiger@orcl) TABLES=emp,dept dumpfile=expdp.dmp DIRECTORY=dpdata1;

4)按查询条件导

expdp [scott/tiger@orcl](mailto:scott/tiger@orcl) directory=dpdata1 dumpfile=expdp.dmp Tables=emp query='WHERE deptno=20';

5)按表空间导

expdp system/manager DIRECTORY=dpdata1 DUMPFILE=tablespace.dmp TABLESPACES=temp,example;

6)导整个数据库

expdp system/manager DIRECTORY=dpdata1 DUMPFILE=full.dmp FULL=y;

五、还原数据

1)导到指定用户下

impdp scott/tiger DIRECTORY=dpdata1 DUMPFILE=expdp.dmp SCHEMAS=scott;

2)改变表的owner

impdp system/manager DIRECTORY=dpdata1 DUMPFILE=expdp.dmp TABLES=scott.dept REMAP\_SCHEMA=scott:system;

3)导入表空间

impdp system/manager DIRECTORY=dpdata1 DUMPFILE=tablespace.dmp TABLESPACES=example;

4)导入数据库

impdb system/manager DIRECTORY=dump\_dir DUMPFILE=full.dmp FULL=y;

5)追加数据

impdp system/manager DIRECTORY=dpdata1 DUMPFILE=expdp.dmp SCHEMAS=system TABLE\_EXISTS\_ACTION

### 表空间

#### 创建表空间以及字段说明

创建表空间

|  |
| --- |
| CREATE TABLESPACE CORE\_DATA DATAFILE  '/data/oracle/oradata/DCITS/CORE\_DATA.dbf' SIZE 2300M AUTOEXTEND ON NEXT 100M MAXSIZE UNLIMITED  LOGGING  ONLINE  EXTENT MANAGEMENT LOCAL AUTOALLOCATE  BLOCKSIZE 8K  SEGMENT SPACE MANAGEMENT AUTO  FLASHBACK ON; |

SIZE 表空间初始大小2300M

AUTOEXTEND 若表空间满了自动增长，ON为打开，OFF为关闭自动增长

NEXT 每次自动增长的步长

MAXSIZE 自动增长到的上限大小，UNLIMITED为无限大

BLOCKSIZE 数据块的大小，默认8kb

LOGGING 将日志写入日志文件

ONLINE,OFFLINE 联机，脱机

EXTENT MANAGEMENT LOCAL 区管理本地化

SEGMENT SPACE MANAGEMENT 段空间管理方式(AUTO 是表示 ASSM管理的标志)

FLASHBACK 表空间是否可闪回

(当表空间使用本地管理且段空间为字段管理时，表空间才能使用大文件表空间。(本地管理的撤销表空间和临时表空间除外))

AUTOALLOCATE或者UNIFORM 一种区的管理方式

|  |
| --- |
| AUTOALLOCATE 使表空间以64k的最小范围大小进行系统管理  UNIFORM SIZE 128k 指定大小128k，若省略SIZE则默认是1M |

创建临时表空间

|  |
| --- |
| CREATE TEMPORARY TABLESPACE CORE\_DATA\_TEMP TEMPFILE  '/data/oracle/oradata/DCITS/CORE\_DATA\_TEMP.dbf' SIZE 200M AUTOEXTEND ON NEXT 100M MAXSIZE UNLIMITED  TABLESPACE GROUP ''  EXTENT MANAGEMENT LOCAL UNIFORM SIZE 1M; |

#### 删除表空间

--删除空的表空间，但是不包含物理文件  
drop tablespace tablespace\_name;  
--删除非空表空间，但是不包含物理文件  
drop tablespace tablespace\_name including contents;  
--删除空表空间，包含物理文件  
drop tablespace tablespace\_name including datafiles;  
--删除非空表空间，包含物理文件  
drop tablespace tablespace\_name including contents and datafiles;  
--如果其他表空间中的表有外键等约束关联到了本表空间中的表的字段，就要加上CASCADE CONSTRAINTS  
drop tablespace tablespace\_name including contents and datafiles CASCADE CONSTRAINTS;

一般语句，包含物理文件

|  |
| --- |
| DROP TABLESPACE CORE\_DATA INCLUDING CONTENTS AND DATAFILES; |

#### 查看用户所属表空间

select username,default\_tablespace from dba\_users  where username='用户名';

#### 查看表空间情况

|  |
| --- |
| select t.tablespace\_name,  round(t.bytes / 1024 / 1024 / 1024, 2) || 'G' "总大小",  round((t.bytes - f.bytes) / 1024 / 1024 / 1024, 2) || 'G' "已使用",  round(100 \* (t.bytes - f.bytes) / t.bytes, 2) || '%' "使用率"  from (select tablespace\_name, sum(bytes) bytes  from dba\_data\_files  group by tablespace\_name) t,  (select tablespace\_name, sum(bytes) bytes  from dba\_free\_space  group by tablespace\_name) f  where f.tablespace\_name(+) = t.tablespace\_name |

查找数据文件指标及路径

|  |
| --- |
| select \* from (  select b.file\_id 文件ID,  B.TABLESPACE\_NAME 表空间,  B.FILE\_NAME 物理文件名,  round(B.BYTES / 1024 / 1024 / 1024, 2) || 'G' 总字大小,  round((b.bytes-sum(nvl(a.bytes,0)))/ 1024 / 1024 / 1024, 2) || 'G' 已使用,  round(sum(nvl(a.bytes,0))/ 1024 / 1024 / 1024, 2) || 'G' 剩余,  round(sum(nvl(a.bytes,0))/(b.bytes)\*100, 2) 剩余百分比  from dba\_free\_space a,dba\_data\_files b where a.file\_id=b.file\_id  group by b.file\_id,B.TABLESPACE\_NAME,B.FILE\_NAME,B.BYTES  order by b.tablespace\_name ) c where c.剩余百分比<15 |

#### 表空间大小修改

修改已有的表空间大小

|  |
| --- |
| ALTER DATABASE DATAFILE '/data/oracle/oradata/DCITS/CORE\_DATA.dbf'  RESIZE 5G; |

# Oracle报错问题解决方案

## ORA-01438

方案一，修改表字段类型

|  |
| --- |
| --备份表字段原数据类型  create table num\_old\_IrlGlInfo as  select aa.column\_name,aa.data\_length,aa.DATA\_PRECISION,aa.DATA\_SCALE  fro，m all\_tab\_columns aa  where aa.table\_name = 'IRL\_GL\_INFO'  and aa.owner = 'ENS\_CBANK'  AND aa.data\_type = 'NUMBER'  and aa.data\_scale > 0;  --生成修改字段脚本 然后拷贝执行  select 'alter table IRL\_GL\_INFO modify ('||t.column\_name||' number);' from num\_old\_IrlGlInfo t;      ---排序找到哪个字段长度超过22位  select MAX(LENGTH(INT\_ACCRUED)),  MAX(LENGTH(INT\_ACCRUED\_CTD)),  MAX(LENGTH(INT\_ACCRUED\_CALC\_CTD)),  MAX(LENGTH(INT\_ACCRUED\_DIFF)),  MAX(LENGTH(TAX\_ACCRUED)),  MAX(LENGTH(TAX\_ACCRUED\_CTD)),  MAX(LENGTH(TAX\_ACCRUED\_CALC\_CTD)),  MAX(LENGTH(TAX\_ACCRUED\_DIFF)),  MAX(LENGTH(TAX\_RATE)),  MAX(LENGTH(AGG)),  MAX(LENGTH(INT\_AMT)),  MAX(LENGTH(REAL\_RATE)),  MAX(LENGTH(FLOAT\_RATE)),  MAX(LENGTH(ACTUAL\_RATE)),  MAX(LENGTH(ACCT\_SPREAD\_RATE)),  MAX(LENGTH(ACCT\_PERCENT\_RATE)),  MAX(LENGTH(ACCT\_FIXED\_RATE)),  MAX(LENGTH(AGREE\_FIXED\_RATE)),  MAX(LENGTH(AGREE\_PERCENT\_RATE)),  MAX(LENGTH(AGREE\_SPREAD\_RATE))  from Irl\_Accr\_Info\_Main MM  WHERE MM.ACCR\_DATE = DATE '2019-5-19'  GROUP BY LENGTH(INT\_ACCRUED),  LENGTH(INT\_ACCRUED\_CTD),  LENGTH(INT\_ACCRUED\_CALC\_CTD),  LENGTH(INT\_ACCRUED\_DIFF),  LENGTH(TAX\_ACCRUED),  LENGTH(TAX\_ACCRUED\_CTD),  LENGTH(TAX\_ACCRUED\_CALC\_CTD),  LENGTH(TAX\_ACCRUED\_DIFF),  LENGTH(TAX\_RATE),  LENGTH(AGG),  LENGTH(INT\_AMT),  LENGTH(REAL\_RATE),  LENGTH(FLOAT\_RATE),  LENGTH(ACTUAL\_RATE),  LENGTH(ACCT\_SPREAD\_RATE),  LENGTH(ACCT\_PERCENT\_RATE),  LENGTH(ACCT\_FIXED\_RATE),  LENGTH(AGREE\_FIXED\_RATE),  LENGTH(AGREE\_PERCENT\_RATE),  LENGTH(AGREE\_SPREAD\_RATE)  ORDER BY 1 DESC,  2 DESC,  3 DESC,  4 DESC,  5 DESC,  6 DESC,  7 DESC,  8 DESC,  9 DESC,  10 DESC,  11 DESC,  12 DESC,  13 DESC,  14 DESC,  15 DESC,  16 DESC,  17 DESC,  18 DESC,  19 DESC,  20 DESC  ;  --定位有问题的账户  CREATE TABLE Irl\_Accr\_Info\_Main\_JINGDU AS  select\*  from Irl\_Accr\_Info\_Main MM  WHERE/\* MM.ACCR\_DATE = DATE '2019-5-19'  AND\*/ LENGTH(MM.INT\_ACCRUED\_DIFF)> 22; |

方案二，查看数据库日志，详见上面查看日志篇