## **1下载镜像**

docker pull klwang/oracle10g

# docker run 6056da9e524f env

|  |
| --- |
| PATH=/u01/app/oracle/product/10.2.0/dbhome2/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin  HOSTNAME=59035e7d1efa  ORACLE\_BASE=/u01/app/oracle  CONFIG\_BASE=/config  LOGS\_BASE=/logs  CVUQDISK\_GRP=oinstall  ORACLE\_HOME=/u01/app/oracle/product/10.2.0/dbhome2  DISABLE\_HUGETLBFS=1  ORACLE\_SID=orcl  ORACLE\_PASS=oracle  HOME=/root |

## **2查看镜像**

docker images

## **3 运行镜像**

docker run -d -p 1521:1521 -v oracle-u02:/u02 -v oracle-10g-vol:/u01/app/oracle -v /etc/localtime:/etc/localtime --privileged --name oracle10g klwang/oracle10g

### 3.1拷贝容器内的文件

docker run -d -p 1521:1521 --privileged --name oracle10g klwang/oracle10g

docker cp 3db108d6f40f:/u01 ./

docker cp 3db108d6f40f:/u02 ./

docker cp 3db108d6f40f:/config ./

docker cp 3db108d6f40f:/data ./

docker cp 3db108d6f40f:/etc ./

docker cp 3db108d6f40f:/logs ./

docker cp 3db108d6f40f:/tmp ./

#其中-d表示后台运行，这样退出终端docker容器不会挂掉，--name给容器起一个名字

docker exec -it oracle10g bash #进入容器

#-i表示交互方式，t表示终端

**docker-compose-oracle**

|  |
| --- |
| version: '2'  services:  oracle102:  container\_name: oracle10g  image: klwang/oracle10g:latest  #restart: always  environment:  - TZ=Asia/Shanghai  volumes:  - "./u01:/u01"  - "./u02:/u02"  - "./config:/config"  - "./data:/data"  - "./etc:/etc"  #- "./home:/home"  - "./logs:/logs"  - "./tmp:/tmp"  ports:  - "1521:1521"  privileged: true |

## **4 进入容器**

docker exec -it oracle10g bash

## **5 切换到oracle 用户**

不是必需，适合于没有DBA密码时使用，可以不用密码来进入sqlplus界面

su - oracle

## **6修改密码**

sqlplus /nolog

connect sys/oracle as sysdba

sqlplus / as sysdba

# alter user username identified by password;

alter user system identified by "123456";

alter user sys identified by "123456";

## **6授权**

grant sysdba to sys;

## **7 创建内部管理员账号**

创建一个用户名为oracle10的用户，密码为 123456

create user oracle10 identified by "123456";

## **8 将dba权限授权给内部管理员账号**

grant connect,resource,dba to oracle10;

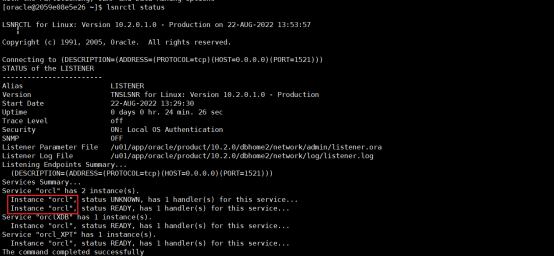
## **9修改密码规则策略为密码永不过期**

#oracle密码默认有效期是180天好像，

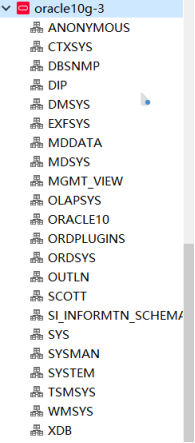
ALTER PROFILE DEFAULT LIMIT PASSWORD\_LIFE\_TIME UNLIMITED;

## **10 进入容器之后，使用如下命令查看数据库提供监听服务的名字**

lsnrctl status







## **11 其他常见命令**

查看当前的所有数据库: select \* from v$database;

查看哪些用户拥有SYSDBA、SYSOPER权限:

SQL>select \* from V\_$PWFILE\_USERS;

查看所有的数据库实例：

SQL>select \* from v$instance；

用户授权:  
grant connect,resource,dba to test11;  
grant sysdba to test11;  
commit;

<https://www.cnblogs.com/hzhida/archive/2012/08/01/2618825.html>

## **12参考文档**

<https://www.freesion.com/article/30581476722/>

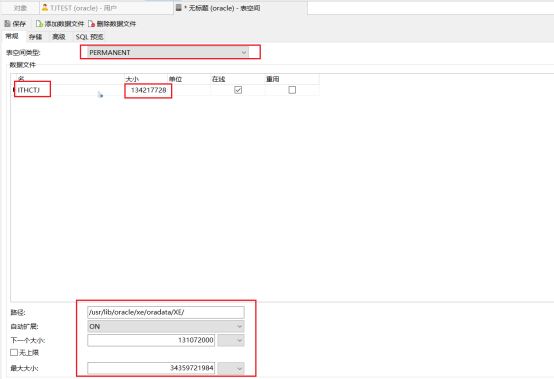
<https://blog.csdn.net/weixin_30562025/article/details/116389557>

<https://blog.csdn.net/sinat_33157758/article/details/103579660>

# **二** [navicat for oracle 导入dmp文件](https://www.liuyixiang.com/post/113222.html" \o "navicat for oracle 导入dmp文件)

## **2.1创建表空间**

1. 使用带有dba权限的用户登录oracle，最开始的用户是system和sys
2. 新建表空间，为表空间设置大小(大小:134217728,下一个大小:131072000,最大大小:34359721984 不选单位)和路径(F:\ORACLE\TABLE)，一定要开启自动扩展on，下一个大小指的是下次扩展大小



/u02/orcl/orcl/

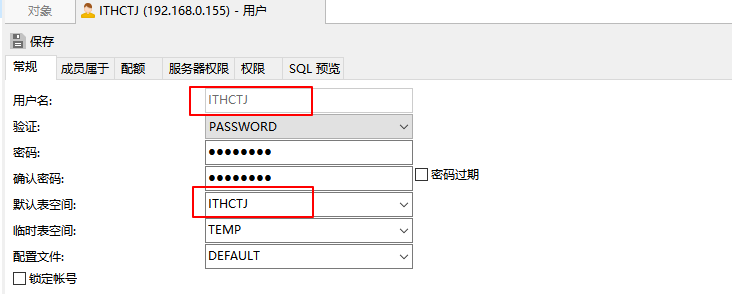
1. 创建成功后可以在所填的路径下面看到新建的表空间



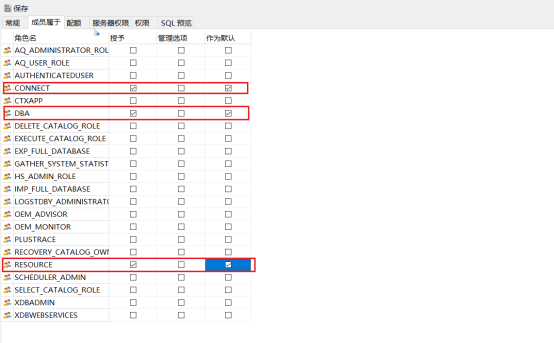
## **2.2创建用户：**

1. 在navicat顶部菜单栏点击用户，新建用户，这里注意用户名一定要是大写

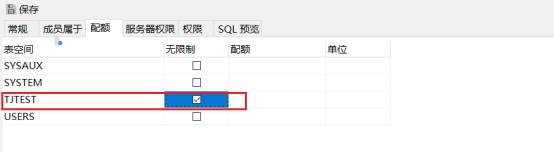
2、常规里面默认表空间选择我们刚建的表空间，不选择的话系统默认users表空间



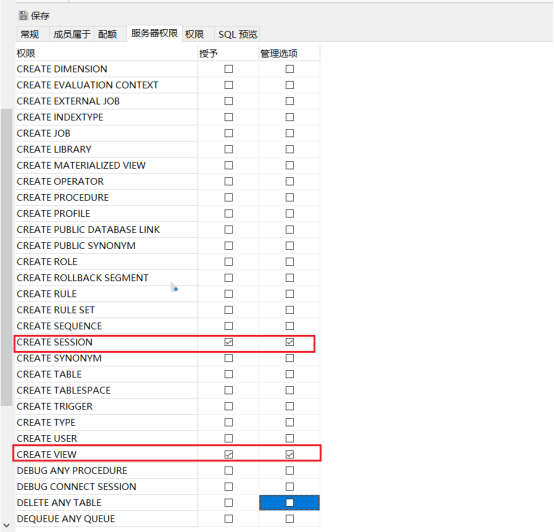
3、成员属于里面选择connect resource和dba，dba是最高权限，一般情况下都要选择(授予，作为默认)



4、配额里面对我们刚建的表空间打勾



5、服务器权限选择create session(授予/管理选项) creation view(授予/管理选项)



<https://www.cnblogs.com/zmxnb/articles/14821608.html>

<https://www.liuyixiang.com/post/113222.html>

<https://www.cnblogs.com/zhaoyudang/p/14986935.html>

## **2.3导入信息**

ITHCTJ 表空间

/u02/orcl/orcl

导入命令

imp ITHCTJ/123456@localhost:1521/orcl file=/tmp/tj.DMP log=./dmp.log full=y ignore=y

[navicat](https://so.csdn.net/so/search?q=navicat&spm=1001.2101.3001.7020)在展示oracle和mysql的数据时会有不一样的概念，mysql展示的是库，而oracle展示的是用户名下的表空间，虽然指定用户进去了，但是还是展示所有的用户表空间，所以需要找某个表，则根据当前的用户名去找对应的表空间

## 2.4删除表空间

drop tablespace ITHCTJ including contents and datafiles;

<https://www.csdn.net/tags/MtTaMg5sMzQzMi1ibG9n.html>

# **三 导入问题**

## **3.1 30 IMP-00019: row rejected due to ORACLE error 12899**

MP-00019: row rejected due to ORACLE error 12899

IMP-00003: ORACLE error 12899 encountered

ORA-12899: value too large for column "CRM"."BK\_ECS\_ORDER\_INFO\_00413"."POSTSCRIPT" (actual: 895, maximum: 765)导入日志报

IMP-00019: 由于 ORACLE 错误 12899 而拒绝行

IMP-00003: 遇到 ORACLE 错误 12899

ORA-12899: 列 "JACKEYJ"."JK\_REGISTER"."OPNAME" 的值太大 (实际值: 21, 最大值: 20)

出现这种问题时因为生产库的字符集和本机测试库的字符集不一样。

到生产库上用

SQL>select userenv('language') from dual;

查看到是ZHS16GBK，而本机库是utf-8用一下命令修改本机库字符集

|  |
| --- |
| SQL>SHUTDOWN IMMEDIATE  SQL>STARTUP MOUNT  SQL>ALTER SYSTEM ENABLE RESTRICTED SESSION;  SQL>ALTER SYSTEM SET JOB\_QUEUE\_PROCESSES=0;  SQL>ALTER SYSTEM SET AQ\_TM\_PROCESSES=0;  SQL>ALTER DATABASE OPEN;  SQL>ALTER DATABASE CHARACTER SET INTERNAL\_USE ZHS16GBK;  SQL>SHUTDOWN IMMEDIATE  SQL>STARTUP |

设置后导入成功！

<http://t.zoukankan.com/storymedia-p-4571211.html>

<https://www.cnblogs.com/chengxuyonghu/p/16419588.html>

<https://blog.csdn.net/qq_23859799/article/details/107498587>

http://t.zoukankan.com/KyleLi-p-7615506.html

<https://www.cnbugs.com/post-1010.html>

## **3.2 IMP-00003: ORACLE error 1658 encountered**

ALTER DATABASE DATAFILE '/u02/orcl/orcl/ITHCTJ.dbf' AUTOEXTEND ON NEXT 100M;

<https://blog.csdn.net/qq_42422526/article/details/123566741>

## 3.3 导入报错

IMP-00015: following statement failed because the object already exists:

IMP-00017: following statement failed with ORACLE error 14063:

IMP-00003: ORACLE error 14063 encountered

IMP-00017: following statement failed with ORACLE error 14048:

IMP-00003: ORACLE error 14048 encountered

. . importing table "SYS\_IMPORT" 0 rows imported

IMP-00017: following statement failed with ORACLE error 14063:

IMP-00003: ORACLE error 14063 encountered

IMP-00017: following statement failed with ORACLE error 14048:

IMP-00003: ORACLE error 14048 encountered

IMP-00017: following statement failed with ORACLE error 14063:

IMP-00003: ORACLE error 14063 encountered

IMP-00017: following statement failed with ORACLE error 14048:

IMP-00003: ORACLE error 14048 encountered

<https://www.jianshu.com/p/32534135846e/>

<https://www.cnblogs.com/wendelhuang/p/6933803.html>

<https://blog.csdn.net/ak_red007/article/details/84221452>