

(一) oracle+centos7 环境配置:

1. 创建用户组 oinstall

```
[ root@localhost ~ ] # groupadd oinstall
```

2. 创建用户组 dba

```
[ root@localhost ~ ] # groupadd dba
```

3. 创建 oracle 用户, 并加入到 oinstall 和 dba 用户组

```
[ root@localhost ~ ] # useradd -g oinstall -g dba -m oracle
```

4. 设置用户 oracle 的登陆密码, 不设置密码, 在 CentOS 的图形登陆界面没法登陆, 并查看新建的 oracle 用户

```
[ root@localhost ~ ] # passwd oracle
更改用户 oracle 的密码 。
新的 密码 :
重新输入新的 密码 :
passwd: 所有的身份验证令牌已经成功更新。
[ root@localhost ~ ] # id oracle
uid=1002(oracle) gid=1004(dba) 组=1004(dba)
```

5. 创建 oracle 数据库安装目录

```
[ root@localhost ~ ] # cd ..
[ root@localhost / ] # mkdir -p /data/oracle
[ root@localhost / ] # mkdir -p /data/oraInventory
[ root@localhost / ] # mkdir -p /data/database
[ root@localhost / ] # cd /data
[ root@localhost data ] # ls
database oracle oraInventory
```

6. 设置目录所有者为 oinstall 用户组的 oracle 用户

```
[ root@localhost data ] # chown -R oracle:oinstall /data/oraInventory
[ root@localhost data ] # chown -R oracle:oinstall /data/oracle
[ root@localhost data ] # chown -R oracle:oinstall /data/database
```

7. 修改 OS 系统标识 (oracle 默认不支持 CentOS 系统安装, 但是 centos 其实就是 redhat)

```
[ root@localhost data ] # cat /proc/version
Linux version 3.10.0-957.el7.x86_64 (mockbuild@kbuilder.bsys.centos.org) (gcc version 4.8.5 20150623 (Red Hat 4.8.5-36) (GCC) ) #1 SMP Thu Nov 8 23:39:32 UTC 2018
[ root@localhost data ] # cat /etc/redhat-release
CentOS Linux release 7.6.1810 (Core)
```

```
[root@localhost data]# vi /etc/redhat-release
[root@localhost data]# cat /etc/redhat-release
redhat-7
```

8. 安装 oracle 数据库所需要的软件包

```
[root@localhost data]# yum -y install binutils* compat-libcap1* compat-libstdc++* gcc*
gcc-c++* glibc* glibc-devel* ksh* libaio* libaio-devel* libgcc* libstdc++* libstdc++-de
vel* libXi* libXtst* make* sysstat* elfutils* unixODBC*
已加载插件：fastestmirror, langpacks
Loading mirror speeds from cached hostfile
epel/x86_64/metalink | 6.2 kB 00:00:00
* base: mirrors.163.com
* epel: hkg.mirror.rackspace.com
* extras: mirrors.cn99.com
* updates: mirrors.163.com
* webtatic: us-east.repo.webtatic.com
base | 3.6 kB 00:00:00
epel | 5.4 kB 00:00:00
extras | 3.4 kB 00:00:00
mysql-connectors-community | 2.5 kB 00:00:00
mysql-tools-community | 2.5 kB 00:00:00
mysql57-community | 2.5 kB 00:00:00
updates | 3.4 kB 00:00:00
webtatic | 3.6 kB 00:00:00
```

9. 查看防火墙状态

```
[root@localhost data]# systemctl status firewalld.service
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: e
nabled)
   Active: active (running) since — 2019-08-05 14:19:24 CST; 1h 58min ago
     Docs: man: firewalld(1)
    Main PID: 828 (firewalld)
    CGroup: /system.slice/firewalld.service
            └─828 /usr/bin/python -Es /usr/sbin/firewalld --nofork --nopid

8月 05 14:19:16 localhost.localdomain systemd[1]: Starting firewalld - dynamic fir....
8月 05 14:19:24 localhost.localdomain systemd[1]: Started firewalld - dynamic fire....
Hint: Some lines were ellipsized, use -l to show in full.
```

10. 关闭防火墙，再次查看

```
[root@localhost data]# systemctl stop firewalld.service
[root@localhost data]# systemctl status firewalld.service
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: e
nabled)
   Active: inactive (dead) since — 2019-08-05 16:18:31 CST; 4s ago
     Docs: man: firewalld(1)
    Process: 828 ExecStart=/usr/sbin/firewalld --nofork --nopid $FIREWALLD_ARGS (code=exi
ted, status=0/SUCCESS)
    Main PID: 828 (code=exited, status=0/SUCCESS)

8月 05 14:19:16 localhost.localdomain systemd[1]: Starting firewalld - dynamic fir....
8月 05 14:19:24 localhost.localdomain systemd[1]: Started firewalld - dynamic fire....
8月 05 16:18:26 localhost.localdomain systemd[1]: Stopping firewalld - dynamic fir....
8月 05 16:18:31 localhost.localdomain systemd[1]: Stopped firewalld - dynamic fire....
Hint: Some lines were ellipsized, use -l to show in full.
```

11. 禁止使用防火墙，重启也是禁止

```
[root@localhost data]# systemctl disable firewalld.service
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
[root@localhost data]# █
```

12. 关闭 selinux, 将 selinux 改为 disabled

```
[root@localhost data] # vi /etc/selinux/config
[root@localhost data] # cat /etc/selinux/config

# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
#SELINUX=enforcing
SELINUX=disabled
# SELINUXTYPE= can take one of three values:
#   targeted - Targeted processes are protected,
#   minimum - Modification of targeted policy. Only selected processes are protected
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

13. 修改内核参数, 优化运行效果

```
[root@localhost data] # vi /etc/sysctl.conf
[root@localhost data] # cat /etc/sysctl.conf
# sysctl settings are defined through files in
# /usr/lib/sysctl.d/, /run/sysctl.d/, and /etc/sysctl.d/.
#
# Vendors settings live in /usr/lib/sysctl.d/.
# To override a whole file, create a new file with the same in
# /etc/sysctl.d/ and put new settings there. To override
# only specific settings, add a file with a lexically later
# name in /etc/sysctl.d/ and put new settings there.
#
# For more information, see sysctl.conf(5) and sysctl.d(5).
net.ipv4.icmp_echo_ignore_broadcasts = 1
net.ipv4.conf.all.rp_filter = 1
fs.file-max = 6815744 #设置最大打开文件数
fs.aio-max-nr = 1048576
kernel.shmall = 2097152 #共享内存的总量, 8G内存设置: 2097152*4k/1024/1024
kernel.shmmax = 2147483648 #最大共享内存的段大小
kernel.shmmni = 4096 #整个系统共享内存端的最大数
kernel.sem = 250 32000 100 128
net.ipv4.ip_local_port_range = 9000 65500 #可使用的IPv4端口范围
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
```

14. 配置用户的环境变量

```
[root@localhost data] # vi /home/oracle/.bash_profile
[root@localhost data] # cat /home/oracle/.bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/.local/bin:$HOME/bin

export PATH
export ORACLE_BASE=/data/oracle #oracle数据库安装目录
export ORACLE_HOME=$ORACLE_BASE/product/11.2.0/db_1 #oracle数据库路径
export ORACLE_SID=orcl #oracle启动数据库实例名
export ORACLE_UNQNAME=orcl
export ORACLE_TERM=xterm #xterm窗口模式安装
export PATH=$ORACLE_HOME/bin:/usr/sbin:$PATH #添加系统环境变量
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:/lib:/usr/lib #添加系统环境变量
export LANG=C #防止安装过程出现乱码
export NLS_LANG=AMERICAN_AMERICA.ZHS16GBK #设置Oracle客户端字符集, 必须与Oracle安装时设置的字符集保持一致
[root@localhost data] #
```

15. 使用户的环境变量配置立即生效

```
[root@localhost data] # source /home/oracle/.bash_profile
```

16. 移动到含有 oracle 压缩包的目录下，并使用 unzip 解压

```
[oracle@localhost ~]$ cd /usr/local/src
[oracle@localhost src]$ ls
p13390677_112040_Linux-x86-64_1of7.zip  phpredis
p13390677_112040_Linux-x86-64_2of7.zip
[oracle@localhost src]$ unzip p13390677_112040_Linux-x86-64_1of7.zip -d /data/database/
Archive:  p13390677_112040_Linux-x86-64_1of7.zip
  creating: /data/database/database/
  inflating: /data/database/database/readme.html
```

17. 进入解压后的 database 文件目录下，运行 runInstaller

```
[root@localhost database]# ./runInstaller
```

```
The user is root. Oracle Universal Installer cannot continue installation if the
user is root.
```

```
: No such file or directory
```

```
[root@localhost database]# su oracle
```

```
[oracle@localhost database]$ ./runInstaller
```

```
Starting Oracle Universal Installer...
```

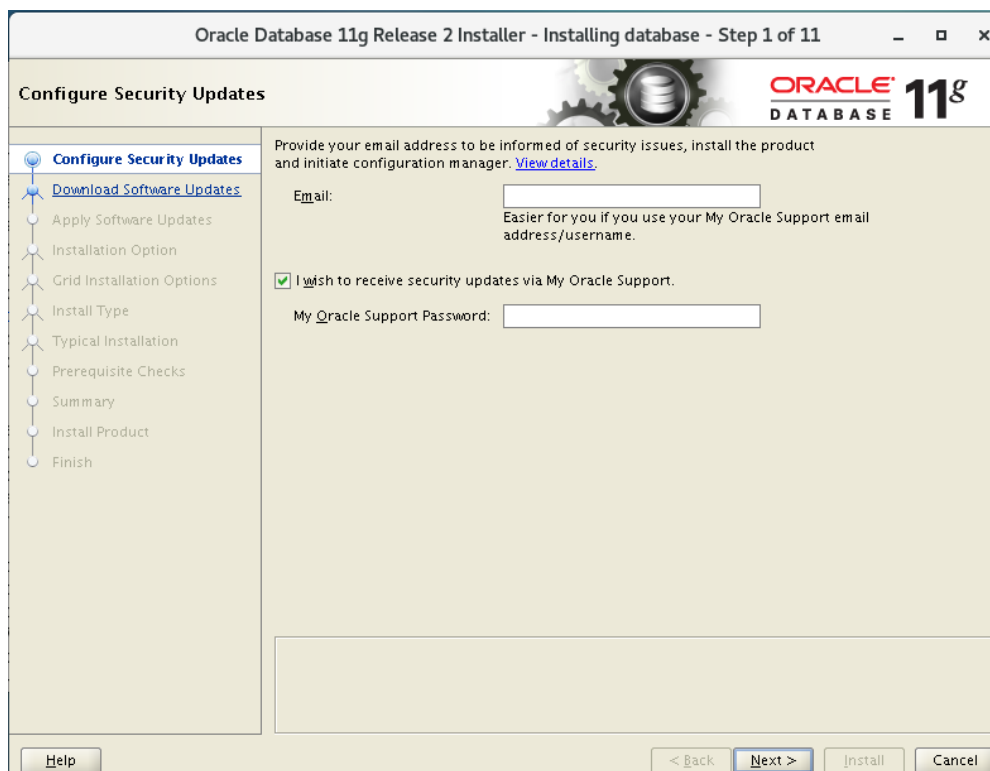
```
Checking Temp space: must be greater than 120 MB.    Actual 17719 MB    Passed
```

```
Checking swap space: must be greater than 150 MB.    Actual 2047 MB    Passed
```

```
Checking monitor: must be configured to display at least 256 colors.    Actual 1
6777216    Passed
```

```
Preparing to launch Oracle Universal Installer from /tmp/OraInstall2019-08-06_02
-33-23PM. Please wait ...
```

18. 进入图形化界面安装步骤，oracle 安装完成



(二) oracle 与 sql

1. 开启 *lsnrctl*

```
[oracle@localhost ~]$ lsnrctl start

LSNRCTL for Linux: Version 11.2.0.4.0 - Production on 06-AUG-2019 16:39:09

Copyright (c) 1991, 2013, Oracle. All rights reserved.

Starting /home/oracle/app/oracle/product/11.2.0/dbhome_1/bin/tnslsnr: please wait...

TNSLSNR for Linux: Version 11.2.0.4.0 - Production
System parameter file is /home/oracle/app/oracle/product/11.2.0/dbhome_1/network/admin/listener.ora
Log messages written to /home/oracle/app/oracle/diag/tnslsnr/localhost/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=localhost)(PORT=1521)))

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC1521)))
STATUS of the LISTENER
```

2. 运行 *sqlplus*

```
[oracle@localhost ~]$ sqlplus /nolog

SQL*Plus: Release 11.2.0.4.0 Production on Tue Aug 6 16:39:24 2019

Copyright (c) 1982, 2013, Oracle. All rights reserved.
```

3. 使用 *sys* 超级账户连接 *oracle*

```
SQL> conn /as sysdba
Connected to an idle instance.
SQL> startup
ORACLE instance started.

Total System Global Area 759943168 bytes
Fixed Size 2257112 bytes
Variable Size 499126056 bytes
Database Buffers 251658240 bytes
Redo Buffers 6901760 bytes
Database mounted.
Database opened.
```

4. 查看当前数据库文件名

```
SQL> select name from v$database;

NAME
-----
ORCL

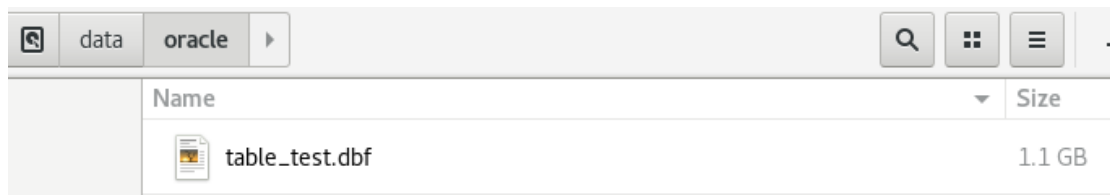
SQL>
```


5. 创建临时表空间 *table_test*

```
SQL> create temporary tablespace table_test tempfile  
2  '/data/oracle/table_test.dbf'  
3  size 1024m;
```

Tablespace created.

6. 去目录下查看，发现刚刚创建的数据库文件



Name	Size
 table_test.dbf	1.1 GB

7. 创建新用户 *user1*，密码 123，临时表空间指定 *table_test*， 永久性表空间指定 *data*

```
SQL> create USER user1 identified by 123 default tablespace data temporary table  
space table_test;
```

User created.

SQL>

8. 给 *user1* 新增权限，这样就可以使用 *user1* 进行 *oracle* 登陆

```
SQL> grant connect,resource,dba to user1;
```

Grant succeeded.

SQL> █

9. 新建 *student* 表，*id* 为 *int* 型，

age 为 *number (3)* 也就是 3 为的 *int* 型，

sex 为指定 6 位字符长度的字符且不是 *male* 就是 *female*

cno 为 *varchar (4)* 也就是 4 位的字符类型，区别于 *char*，

varchar 可以不定长

created_time 为默认的系统时间

```
SQL> create table student(
  2  id int,
  3  age number(3),
  4  name varchar(10),
  5  sex char(6) check(sex = 'male' or sex = 'female'),
  6  cno varchar(4) not null,
  7  created_time date default sysdate
  8 );
```

Table created.

```
SQL> █
```

10. 查看刚刚创建的表的结构

```
SQL> describe student;
```

Name	Null?	Type
ID		NUMBER(38)
AGE		NUMBER(3)
NAME		VARCHAR2(10)
SEX		CHAR(6)
CNO	NOT NULL	VARCHAR2(4)
CREATED_TIME		DATE

```
SQL> █
```

11. 向 student 表中插入数据

```
SQL> insert into student(id,age,name,sex,cno)values(1,21,'jason','male',0211);
```

1 row created.

```
SQL> select * from student;
```

ID	AGE	NAME	SEX	CNO	CREATED_TIME
1	21	jason	male	211	07-AUG-19

```
SQL>
```

12. 全表查询 student

```
SQL> select * from student;
```

ID	AGE	NAME	SEX	CNO	CREATED_TIME
1	21	jason	male	211	07-AUG-19
2	22	jack	male	312	07-AUG-19
3	19	rose	female	133	07-AUG-19
4	20	marry	female	133	07-AUG-19

```
SQL> █
```

13. 使用 update 更新语句, 想更新 marry 的 sex, 第一次想变成 unkown, 被告知错误, 看来更新数据也要遵循建表时的

规则，第二次服从规则，更新成功

```
SQL> update student set sex = 'unkown' where name = 'marry';
update student set sex = 'unkown' where name = 'marry'
*
ERROR at line 1:
ORA-02290: check constraint (USER1.SYS_C0011372) violated

SQL> update student set sex = 'male' where name = 'marry';

1 row updated.

SQL>
```

14. 带条件的查询

```
SQL> select * from student where age > 20;
```

ID	AGE	NAME	SEX	CNO	CREATED_TIME
1	21	jason	male	211	07-AUG-19
2	22	jack	male	312	07-AUG-19

15. 使用 and/or 合并条件查询

```
SQL> select * from student where cno = 133 and age > 19;
```

ID	AGE	NAME	SEX	CNO	CREATED_TIME
4	20	marry	male	133	07-AUG-19

16. 使用 order by 对查询结果排序

```
SQL> select * from student order by age;
```

ID	AGE	NAME	SEX	CNO	CREATED_TIME
3	19	rose	female	133	07-AUG-19
4	20	marry	male	133	07-AUG-19
1	21	jason	male	211	07-AUG-19
2	22	jack	male	312	07-AUG-19

17. 使用 union 联合查询

```
SQL> select id,age,name,sex from student union select id,age,name,sex from student_unio
n;
```

ID	AGE	NAME	SEX
1	21	jason	male
2	22	jack	male
3	19	rose	female
4	20	marry	male
999	999	user	emmm

```
6 rows selected.
```

```
SQL> █
```


18. 使用 avg、count、sum 等聚合函数查询

```
SQL> select cno,sum(age),count(*),sum(age)/count(*),avg(age) from student group by cno;
```

CNO	SUM(AGE)	COUNT(*)	SUM(AGE)/COUNT(*)	AVG(AGE)
312	22	1	22	22
211	21	1	21	21
133	39	2	19.5	19.5

```
SQL> █
```

(三) 学习 sql 时的笔记

连接 oracle:

第一种方法:

```
Insrctl start
sqlplus /nolog
conn /as sysdba --以最高权限登陆
startup
```

第二种方法:

```
Insrctl start
sqlplus 用户名/密码 --直接进入
```

**** 当使用第二种方法时, 在用户名和密码都输入正确的情况下无法登陆, 且弹出 lack create session privilege, 尝试添加权限解决 ****

**** grant create session,resource to user2 --用户名 ****

****sys 用户是超级用户, 具有最高权限, 具有 sysdba 角色, 有 create database 的权限****

**** system 用户是管理操作员, 权限也很大。具有 sysoper 角色, 没有 create database 的权限 ****

创建临时表空间:

```
create temporary tablespace test --表空间名称
tempfile 'oracle/oracledata/tenp001.dbf' --文件路径
size 1024m --初始大小
autoextend on next 32m --每次自动扩展大小
maxsize 2048m; --最大多少 (可设置为 UNLIMITED,则最大为 32g)
```

创建永久性表空间：

```
create tablespace TEST
datafile '/oracle/*****'
size 32m
autoexten on next 32m
maxsize 2048m;
```

为表空间添加数据文件：

```
alter tablespace TEST add datafile '**/**/**'
size 50m
autoextend on next 5m
maxsize 100m;
```

**** 当表空间对应的数据文件已经增长到上限值（自己设置的上限值，或者 32g），此时则需要为该表空间新增一个数据文件 ****

对已存在的数据文件属性进行修改：

```
alter database datafile '**/**/**'
resize 100m;                                --重新调整大小
```

用户创建：

```
create user username          -- 用户名
identified by password        -- 密码
default tablespace TEST       -- 指定永久性表空间
temporary tablespace test     -- 指定临时的表空间
profile default;              -- 默认的 profile
```

**** 可以不指定表空间，系统会将该用户之后创建的数据放在默认表空间里，可能有默认表空间溢满的危险 ****

**** 当决定要指定的表空间时，临时表空间和永久表空间要同时指定 ****

用户授权：

****把角色授予用户，该用户也将拥有该角色的权限****

```
grant connect to username;           --连接数据库
grant resource to username;          --创建数据库实体（表，过程等）
grant dba to username;               --创建数据库结构
```

****针对表空间使用****

```
grant unlimited tablespace to username; --用户可以再其他表空间随意建表，且无
限额
```

****系统权限****

```
grant create cluster to username;
grant create procedure to username;
grant create synonym to username;
grant create trigger to username;
grant create view to username;
grant create JOB to username;
grant drop any table to username;
```

****对象权限****

```
grant delete any table to username;
grant update any table to username;
grant insert any table to username;
grant select any table to username;
```

查询已建立的表：

```
select * from user_tables;           --查自己的表;
select * from all_tables;            --查自己有权限的表;
select * from dba_tables;            --查系统里所有的表（需要 dba 权限）
```

查询列名：

```
Select  column_name,data_type,data_length  from  user_tab_columns  [where
table_name='表名'];
```

**** *column_name:列名;data_type:列的数据类型; data_length:列的长度; *****

*****只能使用单引号，表名要大写*****

查询表名：

```
select table_name,tablespace_name,temporary from user_tables[where  
table_name='表名'];
```

*****table_name:表名;tablespace_name:存储表名的表空间; temporary:是否是临时表;*****

*****只能使用单引号，表名要大写*****

查询表空间对应数据文件的路径：

```
select * from dba_data_files;
```

修改列属性：

```
alter table 表名 modify 列名 varchar(50); --varchar(50)是修改后的属性
```

新增数据：

```
insert into 表名 (column1,column2,column3,...)values(data1,data2,data3,...);
```

**** *插入多条数据 *****

```
insert into 表名 (column1,column2,column3,...) select 另一个表名;
```

**** *想插入多条数据必须使用 select 语句，从另一个表里检索数据 *****

**** *insert into 指定的列名可以与 select 子句指定的列名不同，但是数据类型必须兼容,且满足约束 *****

更新语句：

```
update 表名 set 列名 1 = 新数据 [where column1=data1];
```

联合查询语句：

```
select 列名 1, 列名 2 from 表名 1 [where column1=data1] union [all] select 列名  
1, 列名 2 from 表名 2
```

**** eg:select id from student where age > 19 union select id from test; ****

**** union 查询的列名数量必须与前面对应，数据类型也要兼容 ****

**** union all 与 union 的区别在于 union 会合并重复的行，而 union all 不会 ****