# Oracle Database 19C Installation

* Download oracle Database 19c Enterprise Edition from <https://edelivery.oracle.com/>

**Note:** If you don’t have an Oracle account, please create a oracle account to download the Database Software

* Find the pictures below for your reference.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

* Download and move the zip file to **/tmp** in target server.

# Connect To the Server Through Mobaxterm

* Before starting the Database installation, we need to install the following packages.

$ curl -o oracle-database-preinstall-19c-1.0-1.el9.x86\_64.rpm https://yum.oracle.com/repo/OracleLinux/OL9/appstream/x86\_64/getPackage/oracle-database-preinstall-19c-1.0-1.el9.x86\_64.rpm

$ sudo dnf -y localinstall oracle-database-preinstall-19c-1.0-1.el9.x86\_64.rpm

* Now create a password to ORACLE user

$ passwd oracle

* Go to SSH configuration file and uncomment and change the value from NO to YES to ForwardX11
* Now create the following mount points(Directory’s)

$ sudo mkdir /u01 /u02

$ sudo chown -R oracle:oinstall /u01 /u02

$ sudo touch /home/oracle/.Xauthority

$ echo $DISPLAY

$ xauth list

$ sudo su - oracle

$ xauth add <Enter Xauth list output>

$ export DISPLAY=<Enter echo $DISPLAY output>

$ sudo cp /tmp/V982063-01.zip /u02

$ unzip V982063-01.zip

$ export ORACLE\_BASE=/u01/app/oracle

$ export ORACLE\_HOME=/u01/app/oracle/product/19.0.0/db1

$ export CV\_ASSUME\_DISTID=OEL7.8

$ cd /u02

$ ./runInstaller

A red box with white text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, Word

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

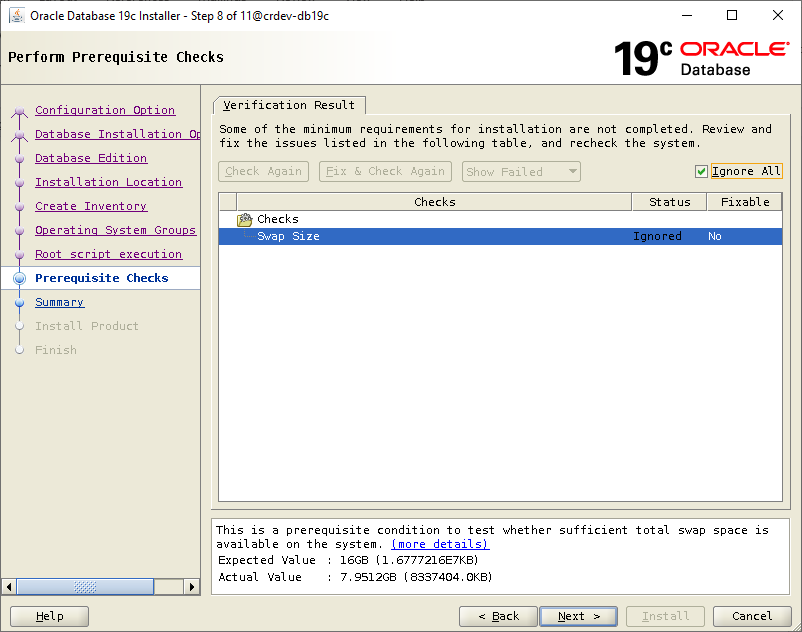
Description automatically generated

Graphical user interface, text

Description automatically generated

Graphical user interface

Description automatically generated



|  |
| --- |
| **Output:**   * This is a prerequisite condition to test whether sufficient total swap space is available on the system. (more details) * Expected Value : 16GB (1.6777216E7KB) * Actual Value : 7.9512GB (8337404.0KB) |

* Ignore this error and continue Click on yes

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

* **Execute Root scripts in CLI:**

By default, root.sh file will store’s in $ORACLE\_HOME

$ /u01/app/oracle/product/19.0.0/db1/root.sh

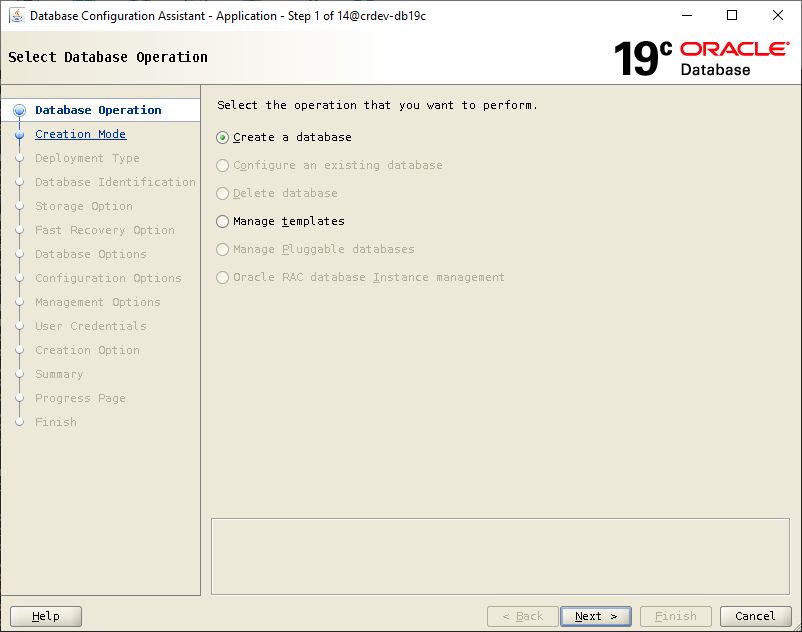
**Output:**

|  |
| --- |
| Performing root user operation.  The following environment variables are set as:  ORACLE\_OWNER= oracle  ORACLE\_HOME= /u01/app/oracle/product/19.0.0/db1  Enter the full pathname of the local bin directory: [/usr/local/bin]:  The contents of "dbhome" have not changed. No need to overwrite.  The contents of "oraenv" have not changed. No need to overwrite.  The contents of "coraenv" have not changed. No need to overwrite.  Entries will be added to the /etc/oratab file as needed by  Database Configuration Assistant when a database is created  Finished running generic part of root script.  Now product-specific root actions will be performed.  Oracle Trace File Analyzer (TFA - Standalone Mode) is available at :  /u01/app/oracle/product/19.0.0/db1/bin/tfactl  Note :  1. tfactl will use TFA Service if that service is running, and user has been granted access  2. tfactl will configure TFA Standalone Mode only if user has no access to TFA Service or TFA is not installed |

# Dbca Configuration

$ cd /u01/app/oracle/product/19.0.0/db1/bin/

$ ./dbca



Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

**DATABASE IDENTIFICATION DETAILS**:

**GLOBAL DATABASE NAME**: Hostname

**SID :** Give a SID

Graphical user interface, text, application, email

Description automatically generated

* **Database File location: /u02/oradata**

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text

Description automatically generated

Graphical user interface, text, application, email

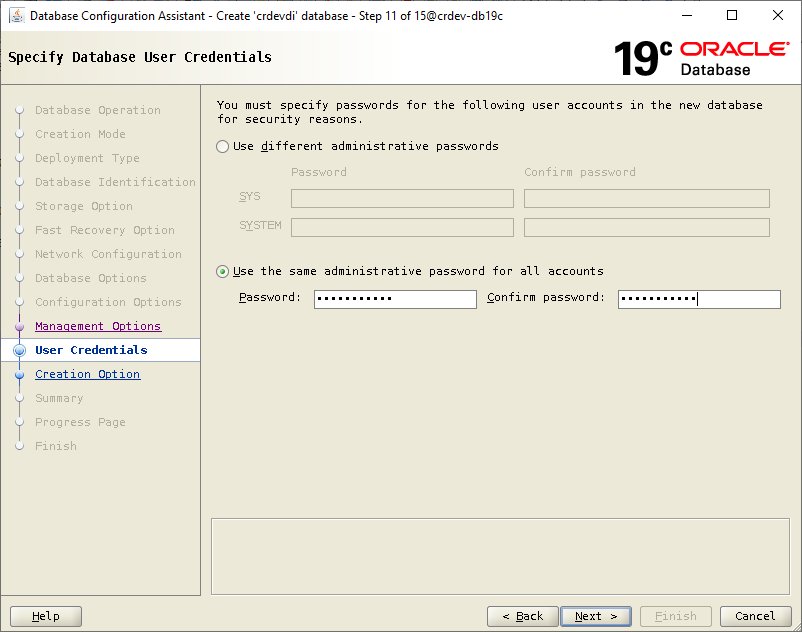
Description automatically generated

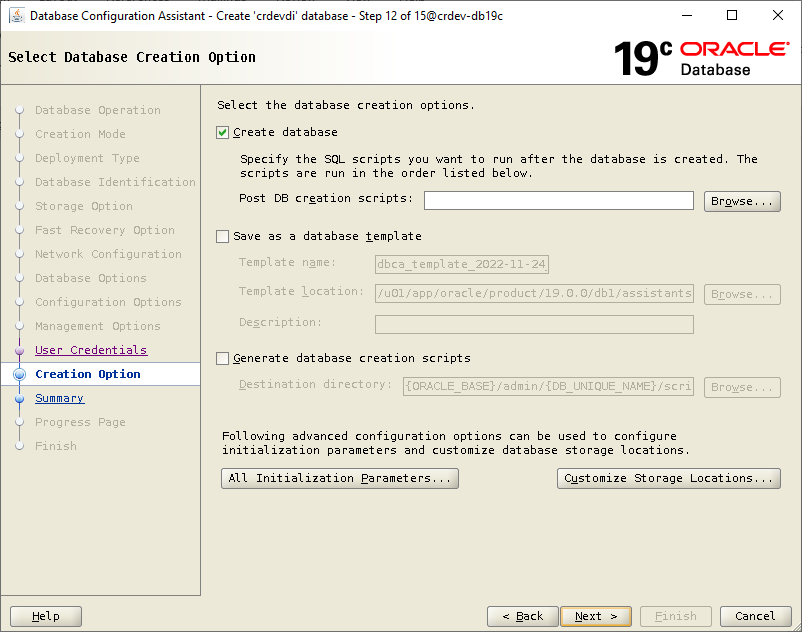
Graphical user interface

Description automatically generated with medium confidence

Graphical user interface, application, email

Description automatically generated



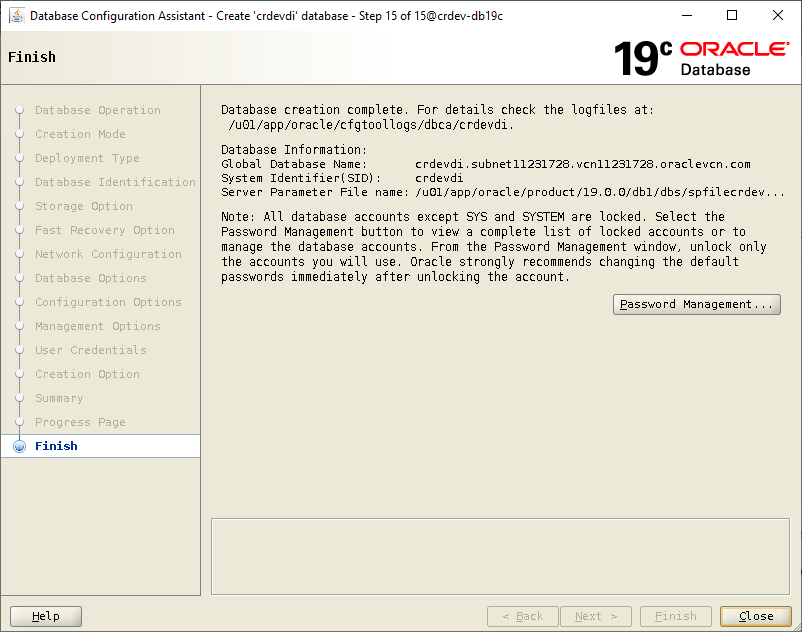


Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated



db\_create\_file\_dest string /u02/oradata/

* Connect to Oracle Database server and do the following steps.

$ sqlplus / as sysdba

SQL> ALTER SYSTEM SET DB\_CREATE\_FILE\_DEST = '/path/to/your/directory' SCOPE=SPFILE;

SQL> create tablespace CONVERTRITE datafile size 5G autoextend on next 1G maxsize unlimited;

SQL> CREATE USER CONVERTRITE IDENTIFIED BY CONVERTRITE;

SQL> ALTER USER CONVERTRITE DEFAULT TABLESPACE CONVERTRITE TEMPORARY TABLESPACE "TEMP" ACCOUNT UNLOCK;

# Create Database Directory

SQL> CREATE DIRECTORY G2N\_TAB\_MAIN AS '/your\_directory\_path';

SQL> GRANT READ, WRITE ON DIRECTORY G2N\_TAB\_MAIN TO CONVERTRITE;

# Create Database Quotas

SQL> ALTER USER CONVERTRITE QUOTA UNLIMITED ON CONVERTRITE;

# Grant System Privileges

GRANT CREATE TRIGGER TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT CREATE MATERIALIZED VIEW TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT DEBUG ANY PROCEDURE TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT CREATE VIEW TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT CREATE SESSION TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT CREATE TABLE TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT CREATE TYPE TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT SELECT ANY DICTIONARY TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT CREATE PUBLIC SYNONYM TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT CREATE SYNONYM TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT EXECUTE ANY PROGRAM TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT CREATE PUBLIC DATABASE LINK TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT DROP PUBLIC SYNONYM TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT CREATE PROCEDURE TO "CONVERTRITE" WITH ADMIN OPTION;

GRANT EXECUTE ON DBMS\_LOCK TO CONVERTRITE;

GRANT CREATE JOB, CREATE ANY JOB, EXECUTE ANY PROGRAM TO CONVERTRITE;

GRANT EXECUTE ON sys.dbms\_scheduler TO CONVERTRITE WITH GRANT OPTION;

GRANT DBA TO CONVERTRITE;

GRANT CREATE USER TO CONVERTRITE WITH ADMIN OPTION;

GRANT SELECT ANY TABLE TO CONVERTRITE WITH ADMIN OPTION;

GRANT DELETE ANY TABLE TO CONVERTRITE WITH ADMIN OPTION;

GRANT UPDATE ANY TABLE TO CONVERTRITE WITH ADMIN OPTION;

GRANT EXECUTE ANY TYPE TO CONVERTRITE WITH ADMIN OPTION;

GRANT EXECUTE ANY PROCEDURE TO CONVERTRITE WITH ADMIN OPTION;

GRANT CREATE ANY SEQUENCE TO CONVERTRITE WITH ADMIN OPTION;

GRANT SELECT ANY SEQUENCE TO CONVERTRITE WITH ADMIN OPTION;

GRANT DROP ANY TABLE TO CONVERTRITE WITH ADMIN OPTION;

GRANT INSERT ANY TABLE TO CONVERTRITE WITH ADMIN OPTION;

# 

# Below Steps are not required for new Installations.

# Create Public Database Link

# Things to do in the Postgres database.

* Login to postgres container and add the following line to the **pg\_hba.conf** file.

# 

# vi /var/lib/postgresql/data/pg\_hba.conf

host postgres postgres <Oracle-DB-IP>/32 md5

* If You are Not able to edit the file in Container copy the file to Docker host and edit, then copy back to the container.
* After making the change, reload the Postgres container.

# Things to do in the Oracle database.

* First, Login to the Oracle DB server and install the following package.

$ sudo yum install postgresql-odbc

* Now create a file in the following path and add the contents in the file according to the settings of our own Postgres server.

$ vi /etc/odbc.ini

[PG]

Description = PG

Driver = /usr/lib64/psqlodbc.so

ServerName = <Postgres-Hosted-server-IP>

Username = postgres

Password = 145913\_Mss

Port = 5000

Database = postgres

[Default]

Driver = /usr/lib64/libodbcpsqlS.so

* Save and exit the file.
* Create the initPG.ora file in the $ORACLE\_HOME/hs/admin directory and add the following lines to it.

$ vi $ORACLE\_HOME/hs/admin/initPG.ora

HS\_FDS\_CONNECT\_INFO = PG

HS\_FDS\_TRACE\_LEVEL = 4

HS\_FDS\_TRACE\_FILE\_NAME=/tmp/ora\_hs\_trace.log

HS\_FDS\_SHAREABLE\_NAME = /usr/lib64/psqlodbc.so

HS\_NLS\_NCHAR=UCS2

HS\_LANGUAGE=AMERICAN\_AMERICA.WE8ISO8859P1

set ODBCINI=/etc/odbc.ini

HS\_FDS\_TRACE\_LEVEL=DEBUG

* Add the following lines to the tnsnames.ora file.

vi $ORACLE\_HOME/network/admin/tnsnames.ora

PG =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = TCP) (HOST = <Oracle-DB-HostAddr>)(PORT = 1521))

)

(CONNECT\_DATA =

(SID = PG)

)

(HS = OK)

)

* Save and exit the file.
* Add the following lines to the listener.ora file.

vi $ORACLE\_HOME/network/admin/listener.ora

SID\_LIST\_LISTENER =

(SID\_LIST=

(SID\_DESC=

(SID\_NAME= PG)

(ORACLE\_HOME= /u01/app/oracle/product/19.0.0/db1)

(PROGRAM= dg4odbc)

)

)

* Save and exit the file.
* Now Reload the Listener.

$ lsnrctl reload

* Now connect to the Oracle Database as sys and Create the Database Link

SQL> CREATE PUBLIC DATABASE LINK PGD\_DBLINK CONNECT TO "postgres" IDENTIFIED BY "145913\_Mss" USING 'PG';

* Now Test the DB-LINK by checking the following quire.

SQL> **select \* from "public"."cr\_objects"@PGD\_DBLINK;**