**Practice 5-1: Start Oracle Database**

### Overview

In this practice, you will start oracle database which has been already created as follows:

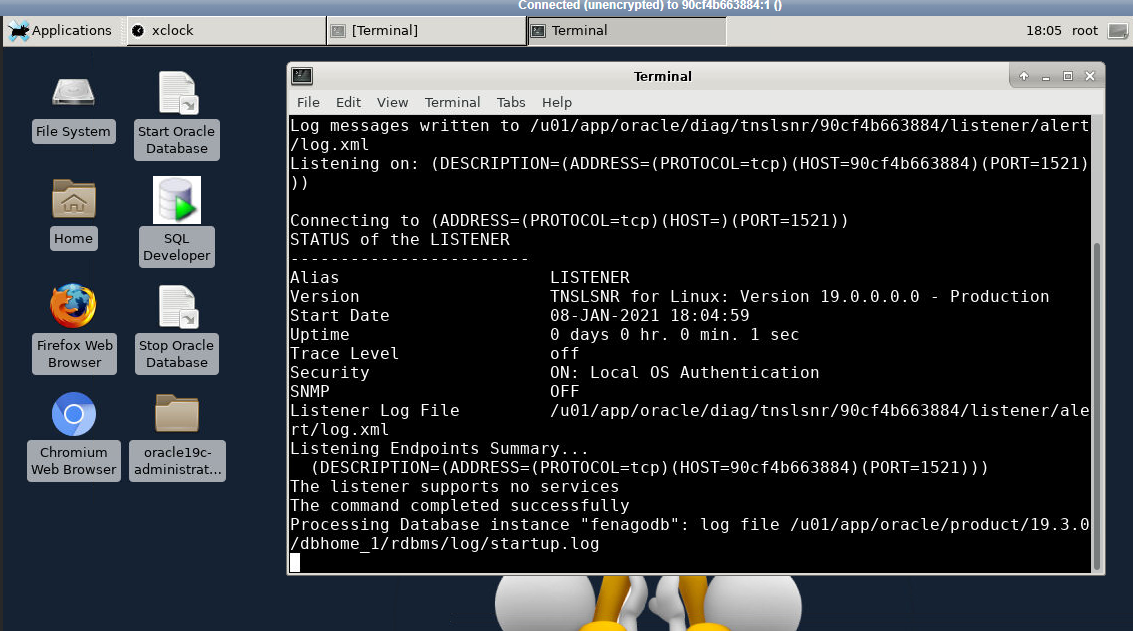
ORACLE\_SID=fenagodb

ORACLE\_PDB=fenagodb1

ORACLE\_PWD=fenago

### Tasks

1. Log in to your lab environment and double click “Start Oracle Database” shortcut.



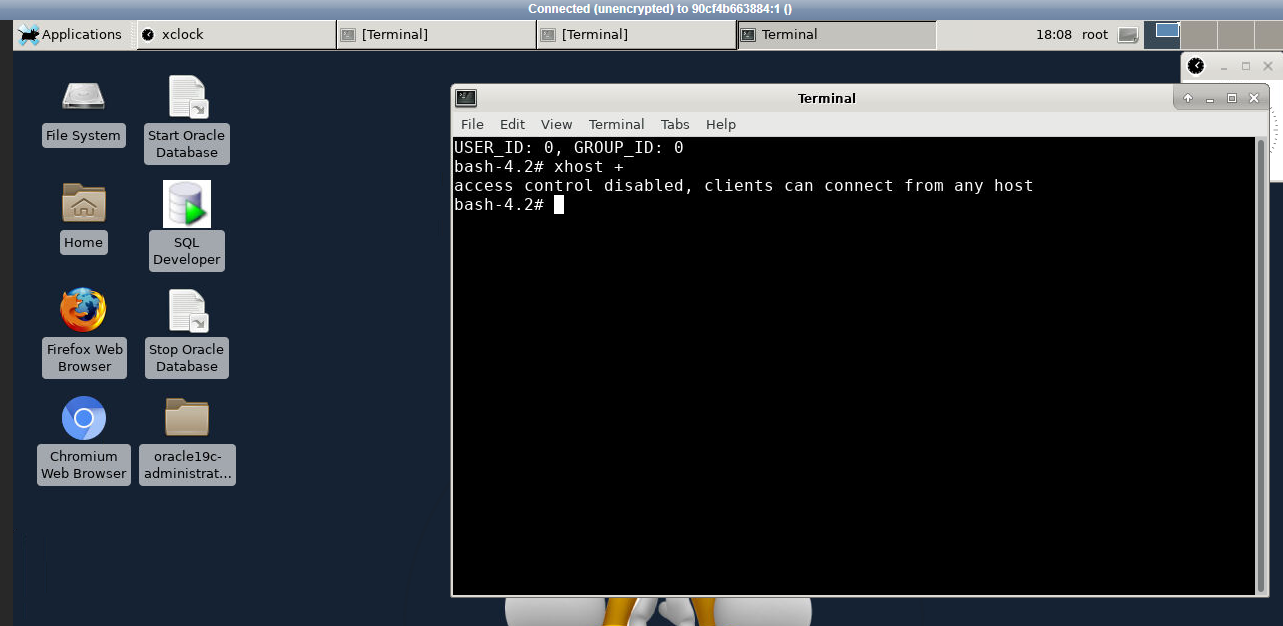
**Practice 5-2: Switch to oracle user from terminal**

### Overview

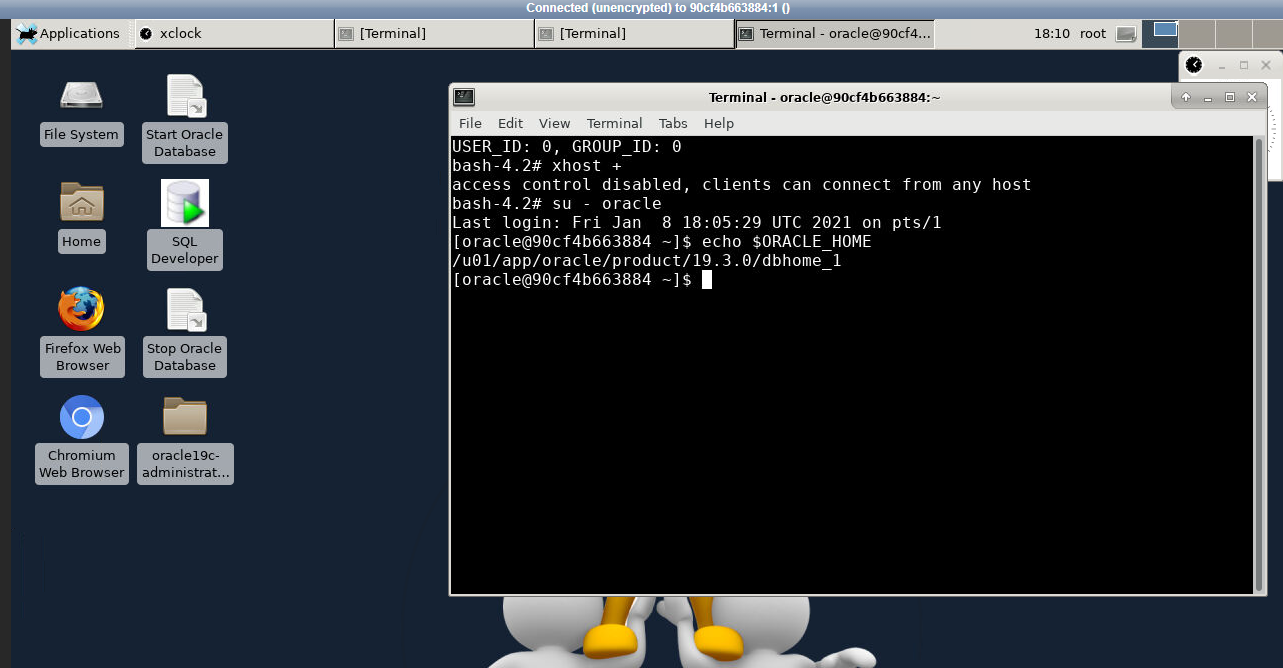
In this practice, you will switch to oracle user from terminal

### Tasks

* 1. Open terminal and run “xhost +” command as root user:



b. Run and run “su - oracle” command in the terminal to switch to **oracle** user:



**Practice 5-3: Exploring a CDB by Using SQL\*Plus**

### Overview

In this practice, you will learn how to do the following things:

* Set the Oracle environment variables
* Connect to the root container by using SQL\*Plus
* Query the data dictionary to view information about the containers, data files, users, instance, and services in a CDB
* List the services created automatically for each container

Some things to remember when you want to query the data dictionary for multiple PDBs or the whole CDB:

* Log in to the root container as a common user. A CDB common user is a database account created in the root container and is inherited by all PDBs in the CDB.
* Query container data objects, such as views whose names begin with V$ and CDB\_. For more information, refer to the following sections in *Oracle Database Administrator's Guide*:
* About Viewing Information When the Current Container is the CDB Root
* Viewing Information About the Containers in a CDB

In some of the steps below, you will format columns by using the COLUMN command. For example, applying the format A55 specifies an alphabetic format of 55 characters wide. Format 999 is an example of a numeric format.

Commands in the practices are in uppercase and variables are in lower case. Any commands that you need to enter are bolded, for example:

SQL> **SELECT regions FROM hr.departments;**

### Assumptions

You are connected to the compute node as the oracle user. See Practice 5-2 for detail.

### Tasks

1. Set the Oracle environment variables. You need to set these each time you open a new terminal window.
   1. In the terminal window, list the search path that holds the oraenv script.

[oracle@MYDBCS ~]$ **which oraenv**

/u01/app/oracle/product/19.3.0/dbhome\_1/bin/oraenv

[oracle@MYDBCS ~]$

* 1. Source the oraenv script. oraenv sets the required environment variables needed for you to connect to your database instance. The oraenv script sets the ORACLE\_SID and ORACLE\_HOME environment variables and includes the $ORACLE\_HOME/bin directory in the PATH environment variable setting. Environment variables that this

script sets will persist in the terminal window until you close it. For the ORACLE\_SID

value, enter FENAGODB.

[oracle@MYDBCS ~]$ **. oraenv**

ORACLE\_SID = [FENAGODB] ? **FENAGODB**

The Oracle base has been set to /u01/app/oracle [oracle@MYDBCS ~]$

* 1. View the environment variables set by the oraenv command.

[oracle@MYDBCS ~]$ **set | grep ORACLE** OLD\_ORACLE\_BASE=/u01/app/oracle ORACLE\_BASE=/u01/app/oracle ORACLE\_HOME=/u01/app/oracle/product/19.3.0/dbhome\_1

ORACLE\_HOSTNAME=MYDBCS.compute-588436052.oraclecloud.internal ORACLE\_SID=FENAGODB

ORACLE\_UNQNAME=FENAGODB

[oracle@MYDBCS ~]$

**Note:** Remember that from this point on, each time you open a terminal window you will need to source the oraenv script to set the environment variables for your CDB.

1. Connect to the root container by using SQL\*Plus.
   1. Start SQL\*Plus and log in to the root container of your CDB as the SYS user with the SYSDBA privilege. You can connect to a database without a password when you have a local connection (on the same machine) and the current operating system user is a member of the privileged OSDBA group.

[oracle@MYDBCS ~]$ **sqlplus / as sysdba**

SQL\*Plus: Release 18.0.0.0.0 Production on Tue May 29 20:18:18

2018

Version 19.3.0.0.0

Copyright (c) 1982, 2017, Oracle. All rights reserved.

Connected to:

Oracle Database 19c EE High Perf Release 18.0.0.0.0 - Production Version 19.3.0.0.0

SQL>

* 1. Verify that you are logged in to the root container as the SYS user by using the SHOW USER command.

SQL> **SHOW user**

USER is "SYS"

SQL>

1. View information about the containers in your CDB.
   1. Verify that you have a container database by querying the V$DATABASE view. The NAME column should contain FENAGODB, the CDB column should contain YES, and the ID should be 0 (zero). A value of zero is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs.

--------- --- ---------- FENAGODB YES 0

SQL>

CON\_ID

CDB

NAME

SQL> **SELECT name, cdb, con\_id FROM v$database;**

* 1. Show the current container name. Because you're currently connected to the root container, the name should be CDB$ROOT.

SQL> **SHOW con\_name**

CON\_NAME

------------------------------ CDB$ROOT

SQL>

* 1. Show the current container ID. Because you're currently connected to the root container, the ID should be 1.

SQL> **SHOW con\_id**

CON\_ID

------------------------------ 1

SQL>

* 1. Determine the version of Oracle Database by querying the V$VERSION view. This view displays version numbers of core library components in Oracle Database.

SQL> **SELECT banner FROM v$version;**

BANNER

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

Oracle Database 19c EE High Perf Release 18.0.0.0.0 - Production

SQL>

* 1. List all the containers in your CDB by querying the V$CONTAINERS view. The results should show three containers—the root container (CDB$ROOT), the seed PDB (PDB$SEED), and PDB1.

-------- ---------- CDB$ROOT 1

PDB$SEED 2

PDB1 3

SQL>

CON\_ID

NAME

SQL> **COLUMN name FORMAT A8**

SQL> **SELECT name, con\_id FROM v$containers ORDER BY con\_id;**

* 1. List the PDBs in the CDB by using the SHOW command. The result should show two PDBs—the seed PDB (PDB$SEED) and PDB1. You can also list PDBs by querying the V$PDBS view. The SHOW command includes information about the open mode of each PDB and whether the PDB is restricted. The open mode for a PDB determines what type of activities a PDB will allow at that time. PDB$SEED is in READ ONLY mode and PDB1 is in READ WRITE mode. The RESTRICTED column indicates whether only users possessing the RESTRICTED SESSION privilege can connect to the PDB.

SQL>

READ WRITE NO

NO

READ ONLY

1. PDB$SEED
2. PDB1

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

OPEN MODE RESTRICTED

SQL> **SHOW pdbs**

CON\_ID CON\_NAME

* 1. View the status of all PDBs in the CDB by querying the CDB\_PDBS view. The status of a PDB describes the state of the PDB. For example, if the PDB is new, but never opened, the status is NEW. If it is available and ready for use, the status is NORMAL.

SQL> **COLUMN pdb\_name FORMAT A8**

SQL> **SELECT pdb\_name, status FROM cdb\_pdbs ORDER BY 1;**

PDB\_NAME STATUS

-------- ---------- PDB1 NORMAL PDB$SEED NORMAL

SQL>

1. View information about the data files in your CDB.
   1. List all the data files in the CDB (for the root container and all PDBs) by querying the

CDB\_DATA\_FILES view. The order of your results may vary.

SQL> **COLUMN file\_name FORMAT A50**

SQL> **COLUMN tablespace\_name FORMAT A10**

SQL> **SELECT file\_name, tablespace\_name FROM cdb\_data\_files;**

FILE\_NAME TABLESPACE

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

/u02/app/oracle/oradata/FENAGODB/users01.dbf USERS

/u02/app/oracle/oradata/FENAGODB/undotbs01.dbf UNDOTBS1

/u02/app/oracle/oradata/FENAGODB/system01.dbf SYSTEM

/u02/app/oracle/oradata/FENAGODB/sysaux01.dbf SYSAUX

/u02/app/oracle/oradata/FENAGODB/PDB1/system01.dbf SYSTEM

/u02/app/oracle/oradata/FENAGODB/PDB1/sysaux01.dbf SYSAUX

/u02/app/oracle/oradata/FENAGODB/PDB1/undotbs01.dbf UNDOTBS1

/u02/app/oracle/oradata/FENAGODB/PDB1/PDB1\_users01.dbf USERS

8 rows selected.

SQL>

* 1. List all the tablespaces in the CDB (for both the root container and all the PDBs) by querying the V$DATAFILE and V$TABLESPACE views.

SQL> **COL name FORMAT A12**

SQL> **SELECT d.file#, ts.name, ts.ts#, ts.con\_id**

1. **FROM v$datafile d, v$tablespace ts**
2. **WHERE d.ts#=ts.ts# AND d.con\_id=ts.con\_id**
3. **ORDER BY 4;**

12 rows selected.

SQL>

|  |  |  |  |
| --- | --- | --- | --- |
| FILE# | NAME | TS# | CON\_ID |
| ---------- | ------------ | ---------- | ---------- |
| 1 | SYSTEM | 0 | 1 |
| 3 | SYSAUX | 1 | 1 |
| 4 | UNDOTBS1 | 2 | 1 |
| 7 | USERS | 4 | 1 |
| 6 | SYSAUX | 1 | 2 |
| 13 | USERS | 5 | 2 |
| 8 | UNDOTBS1 | 2 | 2 |
| 5 | SYSTEM | 0 | 2 |
| 9 | SYSTEM | 0 | 3 |
| 10 | SYSAUX | 1 | 3 |
| 11 | UNDOTBS1 | 2 | 3 |
| 12 | USERS | 5 | 3 |

* 1. List all temp files in the CDB (for the root container and all PDBs) by querying the

CDB\_TEMP\_FILES view.

SQL> **SELECT file\_name, tablespace\_name FROM cdb\_temp\_files;**

FILE\_NAME TABLESPACE

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

/u04/app/oracle/oradata/temp/temp01.dbf TEMP

/u02/app/oracle/oradata/FENAGODB/PDB1/pdbseed\_temp0120 TEMP 18-02-19\_18-48-12-642-PM.dbf

SQL>

* 1. List all the redo log files in the CDB (for the root container and all PDBs) by querying the V$LOGFILE view.

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

3 /u04/app/oracle/redo/redo03.log 0

2 /u04/app/oracle/redo/redo02.log 0

1 /u04/app/oracle/redo/redo01.log 0

SQL>

CON\_ID

SQL> **COLUMN member FORMAT A42**

SQL> **SELECT group#, member, con\_id FROM v$logfile;**

GROUP# MEMBER

* 1. List the control files in the CDB by querying the V$CONTROLFILE view. There should be two—control01.ctl and control02.ctl.

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

/u02/app/oracle/oradata/FENAGODB/control01.ctl 0

/u03/app/oracle/fast\_recovery\_area/FENAGODB/control02.ctl 0

SQL>

CON\_ID

SQL> **COLUMN name FORMAT A55**

SQL> **SELECT name, con\_id FROM v$controlfile;**

NAME

1. View information about the pre-created users in your CDB.
   1. List only the common users in the CDB by querying the CDB\_USERS view.

SQL> **SELECT DISTINCT username FROM cdb\_users**

2 **WHERE common ='YES' ORDER BY 1;**

USERNAME

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

ANONYMOUS APPQOSSYS AUDSYS C##DBAAS\_BACKUP

… SYSTEM WMSYS XDB XS$NULL

38 rows selected.

SQL>

* 1. List all the users in every PDB in the CDB by querying the CDB\_USERS view. In the results, notice that the SYS, SYSTEM, and PDBADMIN user accounts are listed for PDB1. The root container's id is 1 and PDB1's id is 3.

SQL> **COLUMN username FORMAT A25**

SQL> **SELECT con\_id, username FROM cdb\_users**

2 **ORDER BY username, con\_id;**

CON\_ID USERNAME

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

1 ANONYMOUS

3 ANONYMOUS 3 APEX\_050100

3 APEX\_INSTANCE\_ADMIN\_USER

…

1 OUTLN

3 OUTLN

3 PDBADMIN

1 REMOTE\_SCHEDULER\_AGENT

3 REMOTE\_SCHEDULER\_AGENT

…

1 SYS

3 SYS

…

1 SYSRAC

3 SYSRAC

1 SYSTEM

3 SYSTEM

…

84 rows selected.

SQL>

1. View information about the database instance and the services.
   1. View the database instance name, its status, and which container database it is associated with by querying the V$INSTANCE view. The instance's status is OPEN, which means users can access the CDB and PDB.

---------------- ------------ ---------- FENAGODB OPEN 0

SQL>

CON\_ID

STATUS

INSTANCE\_NAME

SQL> **SELECT instance\_name, status, con\_id FROM v$instance;**

* 1. List the services for all the containers in the CDB by querying the V$SERVICES view. The query returns five services. The PDB$SEED service is not listed because no one should connect to it and no operation should be performed with it. It is reserved as a template to create other PDBs.

SQL> **SELECT con\_id, name FROM v$services ORDER BY 1;**

CON\_ID NAME

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

1 SYS$BACKGROUND

1 FENAGODB.588436052.oraclecloud.internal

1 FENAGODB.588436052.oraclecloud.internalXDB

1 SYS$USERS

3 pdb1

SQL>

1. Exit SQL\*Plus.

SQL > **exit**

Disconnected from Oracle Database 19c EE High Perf Release

18.0.0.0.0 - Production Version 19.3.0.0.0

[oracle@MYDBCS ~]$

**Practice 5-4: Exploring a PDB by Using SQL\*Plus**

### Overview

In this practice, you will learn how to do the following things:

* Connect to a PDB indirectly through a CDB
* Query the data dictionary to view information about data files, temp files, and users in a PDB
* Connect to a PDB directly by using the Easy Connect syntax To find data dictionary information specific to a root container or a PDB:
* Query DBA\_ views to return container-specific information.
* When you are logged in to a PDB, queries against the data dictionary return information about that PDB only, regardless of the view you query.
* When queried from a PDB, the DBA\_PDBS view returns the information related to the PDB to which you are connected. When queried from the root container, the DBA\_PDBS view provides information on all PDBs belonging to a given CDB.

### Assumptions

You have a connection to the compute node through PuTTY or SSH and are logged in as the

oracle user.

### Tasks

1. Connect to PDB1 indirectly through the root container.
   1. Start SQL\*Plus and connect to the root container as the SYS user with the SYSDBA privilege. Oracle allows any DBA group user at the operating system level to log into SQL\*Plus without any authentication.

[oracle@MYDBCS ~]$ **sqlplus / as sysdba**

… SQL>

* 1. Verify that PDB1 is open. After DBCA creates a PDB, it opens it automatically. The results below indicate that the open mode is READ WRITE, which means PDB1 is open. PDB users with the SYSDBA, SYSOPER, SYSBACKUP, SYSDG, SYSKM, or SYSRAC privilege can connect to a closed PDB; however, all other PDB users can connect only when the PDB is open.

OPEN\_MODE

CON\_ID NAME

SQL> **COLUMN con\_id FORMAT 999**

SQL> **COLUMN name FORMAT A10**

SQL> **SELECT con\_id, name, open\_mode FROM v$pdbs;**

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

1. PDB$SEED READ ONLY
2. PDB1 READ WRITE

SQL>

* 1. If PDB1 is closed for some reason and its open mode was MOUNTED in the previous step, open it by using the ALTER PLUGGABLE DATABASE command.

SQL> **ALTER PLUGGABLE DATABASE PDB1 OPEN;**

Pluggable database altered. SQL>

* 1. Switch to PDB1. When logged in to a CDB as an appropriately privileged user, you can use the ALTER SESSION command to switch between containers within the CDB. From this point on, your queries against the data dictionary will retrieve information for PDB1 only.

SQL> **ALTER SESSION SET CONTAINER = PDB1;**

Session altered.

SQL>

* 1. Verify that the container name is PDB1.

SQL> **SHOW con\_name**

CON\_NAME

------------------------------ PDB1

SQL>

1. Query the data dictionary to list the data files and temp files for PDB1.
   1. List the data files for PDB1 and the tablespaces to which they belong by querying the

DBA\_DATA\_FILES view.

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

/u02/app/oracle/oradata/FENAGODB/PDB1/system01.dbf SYSTEM

/u02/app/oracle/oradata/FENAGODB/PDB1/sysaux01.dbf SYSAUX

/u02/app/oracle/oradata/FENAGODB/PDB1/undotbs01.dbf UNDOTBS1

TABLESPACE

FILE\_NAME

SQL> **col file\_name format a60**

SQL> **col tablespace\_name format a10**

SQL> **SELECT file\_name, tablespace\_name FROM dba\_data\_files;**

SQL>

USERS

/u02/app/oracle/oradata/FENAGODB/PDB1/PDB1\_users01.dbf

* 1. List the temp files for PDB1 and the tablespaces to which they belong by querying the

DBA\_TEMP\_FILES view.

/u02/app/oracle/oradata/FENAGODB/PDB1/pdbseed\_temp012018-02-19\_1 TEMP 8-48-12-642-PM.dbf

SQL>

TABLESPACE

----------

FILE\_NAME

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

SQL> **SELECT file\_name, tablespace\_name FROM dba\_temp\_files;**

* 1. List the local users for PDB1 by querying the DBA\_USERS view.

SQL> **SELECT DISTINCT username FROM dba\_users WHERE common='NO';**

USERNAME

---------------------------------------------------------------- PDBADMIN

APEX\_LISTENER

APEX\_PUBLIC\_USER APEX\_REST\_PUBLIC\_USER FLOWS\_FILES APEX\_050100

APEX\_INSTANCE\_ADMIN\_USER

SCOTT

8 rows selected.

SQL>

1. Make a direct connection to PDB1 by using the Easy Connect syntax. The Easy Connect syntax enables you to connect to the PDB without 1) requiring a connection to the root container and 2) having to set up a net service name for the PDB.
   1. Disconnect from the PDB.

SQL > **DISCONNECT**

Disconnected from Oracle Database 19c EE High Perf Release

18.0.0.0.0 - Production Version 19.3.0.0.0

SQL>

* 1. Verify that you aren't connected as any user. The SHOW user command returns " " indicating that you are not connected.

SQL> **SHOW user**

USER is ""

SQL>

* 1. Connect to PDB1 directly as the SYSTEM user by using the Easy Connect syntax. See *Course Practice Environment: Security Credentials* for the SYSTEM user password. In Practice 5-3, step 6b, you queried V$SERVICES. Append the value in the query results following FENAGODB to pdb1 to create the service name as shown in this example.

SQL> **CONNECT**

**system/*password*@localhost:1521/pdb1.588436052.oraclecloud.intern al**

Connected.

SQL>

* 1. Verify that you are now connected as the SYSTEM user by using the SHOW USER

command again.

SQL> **SHOW user**

SQL> USER is "SYSTEM"

SQL>

1. Exit SQL\*Plus.

SQL> **EXIT**

…

[oracle@MYDBCS ~]$

**Practice 5-5: Installing the HR Sample Schema**

### Overview

In this practice, you will manually install the HR sample schema.

### Assumptions

You have a connection to the compute node through PuTTY or SSH and are logged in as the

oracle user.

### Tasks

1. In your terminal window, navigate to the

$ORACLE\_HOME/demo/schema/human\_resources directory.

[oracle@MYDBCS ~]$ **cd $ORACLE\_HOME/demo/schema/human\_resources**

[oracle@MYDBCS human\_resources]$

1. Use the ls command to view the contents of the human\_resources directory. In a later step, you will execute the hr\_main.sql to create the HR user, objects and load data into the HR tables.

|  |  |  |  |
| --- | --- | --- | --- |
| [oracle@MYDBCS human\_resour  hr\_analz.sql hr\_comnt.sql hr\_main.sql | | ces]$ **ls**  hr\_drop\_new.sql | hr\_idx.sql |
| hr\_code.sql hr\_popul.sql | hr\_cre.sql | hr\_drop.sql | hr\_main\_new.sql |
| [oracle@MYDBCS human\_resources]$ | | | |

1. Start SQL\*Plus and connect to the root container as the SYS user with the SYSDBA

privilege.

[oracle@MYDBCS human\_resources]$ **sqlplus / as sysdba**

…

SQL>

1. Switch to PDB1.

SQL> **ALTER SESSION SET CONTAINER = PDB1;**

Session altered.

SQL>

1. Execute the **hr\_main.sql** script and respond to the prompts as follows.
   1. Enter the password for the HR user as specified in the *Course Practice Environment: Security Credentials*.
   2. Enter **USERS** as the default tablespace for the HR user.
   3. Enter **TEMP** as the temporary tablespace for the HR user.
   4. Enter **$ORACLE\_HOME/demo/schema/log/** for the log directory.

SQL> **@hr\_main**

specify password for HR as parameter 1:

Enter value for 1: ***password***

specify default tablespeace for HR as parameter 2:

Enter value for 2: **USERS**

specify temporary tablespace for HR as parameter 3:

Enter value for 3: **TEMP**

specify log path as parameter 4:

Enter value for 4: **$ORACLE\_HOME/demo/schema/log/**

PL/SQL procedure successfully completed. User created.

User altered.

Grant succeeded.

…

Comment created.

Commit complete.

PL/SQL procedure successfully completed.

SQL>

* 1. Exit from SQL\*Plus.

SQL> **exit**

…

[oracle@MYDBCS human\_resources]$

1. Query the USER\_TABLES view as the HR user to verify that the user and tables were created.
   1. Connect as the HR user. Be sure to provide the correct service name for your PDB as you did in Practice 5-4, step 3c.

[oracle@MYDBCS human\_resources]$ **sqlplus hr/*password*@localhost:1521/PDB1.588436052.oraclecloud.internal**

…

SQL>

* 1. Query USER\_TABLES.

SQL> **SELECT table\_name FROM user\_tables;**

TABLE\_NAME

---------------------------------------------------------------- REGIONS

COUNTRIES

LOCATIONS DEPARTMENTS JOBS EMPLOYEES JOB\_HISTORY

7 rows selected.

SQL>

1. Exit from SQL\*Plus and close the connection to the compute node.

SQL> **exit**

Disconnected from Oracle Database 19c Enterprise Edition Release

18.0.0.0.0 - Production Version 19.3.0.0.0

[oracle@MYDBCS human\_resources]$ **exit**