**Practices for Lesson 1:**

**Basics of Multitenant**

**Container Database and**

**Pluggable Databases**

**Practices for Lesson 1: Overview**

**Practices Overview**

In previous Oracle Database versions, you used to create, configure, and manage non-CDBs.

In Oracle Database 12*c*, you need to know how to create, configure, and manage multitenant

container databases (CDBs) and pluggable databases (PDBs – PDB1\_1.

In this practice, you will explore new types of databases and get familiar with the architecture

and structures of multitenant container databases (CDBs) and pluggable databases (PDBs).

**Practice 1-1: Exploring CDB Architecture and Structures**

**Overview**

In this practice, you will explore the architecture and structures of cdb1 and its pluggable

databases. Create a Multitenant database called cdb1 with a pluggable database called pdb1\_1. Use the Advanced Option so you can select the sample schemas. Instructor Led.

**Tasks**

1. Explore the cdb1 instance, the background processes and the multitenant container

database.

a. Use the **ps –ef|grep** Unix command.

$ **ps -ef|grep cdb1**

oracle 378 375 0 18:05 ? 00:00:00 oraclecdb1

(DESCRIPTION=(LOCAL=YES)(ADDRESS=(PROTOCOL=beq)))

oracle 390 1 0 18:05 ? 00:00:00 ora\_w001\_cdb1

oracle 2711 2686 0 18:32 pts/2 00:00:00 grep cdb1

oracle 27530 1 0 13:24 ? 00:00:02 ora\_pmon\_cdb1

oracle 27534 1 0 13:24 ? 00:00:04 ora\_psp0\_cdb1

oracle 27538 1 1 13:24 ? 00:05:01 ora\_vktm\_cdb1

oracle 27544 1 0 13:24 ? 00:00:00 ora\_gen0\_cdb1

oracle 27548 1 0 13:24 ? 00:00:00 ora\_mman\_cdb1

oracle 27556 1 0 13:24 ? 00:00:00 ora\_diag\_cdb1

oracle 27560 1 0 13:24 ? 00:00:00 ora\_ofsd\_cdb1

oracle 27564 1 0 13:24 ? 00:00:00 ora\_dbrm\_cdb1

oracle 27568 1 0 13:24 ? 00:00:11 ora\_dia0\_cdb1

oracle 27572 1 0 13:24 ? 00:00:02 ora\_dbw0\_cdb1

oracle 27576 1 0 13:24 ? 00:00:01 ora\_lgwr\_cdb1

oracle 27580 1 0 13:24 ? 00:00:03 ora\_ckpt\_cdb1

oracle 27584 1 0 13:24 ? 00:00:00 ora\_smon\_cdb1

oracle 27630 1 0 13:25 ? 00:00:00 ora\_tmon\_cdb1

oracle 27634 1 0 13:25 ? 00:00:00 ora\_tt00\_cdb1

oracle 27638 1 0 13:25 ? 00:00:00 ora\_fbda\_cdb1

oracle 27642 1 0 13:25 ? 00:00:00 ora\_aqpc\_cdb1

oracle 27651 1 0 13:25 ? 00:00:00 ora\_p000\_cdb1

oracle 27659 1 0 13:25 ? 00:00:00 ora\_p001\_cdb1

oracle 27666 1 0 13:25 ? 00:00:00 ora\_p002\_cdb1

oracle 27670 1 0 13:25 ? 00:00:00 ora\_p003\_cdb1

oracle 27682 1 0 13:25 ? 00:00:09 ora\_cjq0\_cdb1

oracle 27734 1 0 13:25 ? 00:00:00 ora\_qm01\_cdb1

oracle 27738 1 0 13:25 ? 00:00:00 ora\_q001\_cdb1

oracle 27742 1 0 13:25 ? 00:00:00 ora\_q002\_cdb1

oracle 27750 1 0 13:25 ? 00:00:00 ora\_smco\_cdb1

oracle 31695 1 0 17:05 ? 00:00:00 ora\_w002\_cdb1

$

b. Connect to the multitenant container database cdb1.

$ **. oraenv**

ORACLE\_SID = [orcl] ? **cdb1**

The Oracle base remains unchanged with value /u01/app/oracle

$ **sqlplus / as sysdba**

Connected to:

Oracle Database 12*c* Enterprise Edition Release 12.1.0.0.2 -

64bit Production

With the Partitioning, OLAP, Data Mining, Real Application

Testing

SQL>

c. Check if the database is a multitenant container database.

SQL> **select name, cdb, con\_id from v$database;**

NAME CDB CON\_ID

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

CDB1 YES 0

SQL>

d. Check the instance name.

SQL> **select INSTANCE\_NAME, STATUS, CON\_ID from v$instance;**

INSTANCE\_NAME STATUS CON\_ID

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

cdb1 OPEN 0

SQL> **EXIT**

$

2. Explore the services.

a. Start the listener if not yet started.

$ **lsnrctl status**

LSNRCTL for Linux: Version 12.1.0.0.2 - Production on 09-JUL-

2012 02:57:38

Copyright (c) 1991, 2012, Oracle. All rights reserved.

Connecting to

(DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC1521)))

STATUS of the LISTENER

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

Alias LISTENER

Version TNSLSNR for Linux: Version 12.1.0.0.2

- Production

Start Date 10-JUL-2012 00:15:19

Uptime 0 days 2 hr. 42 min. 19 sec

Trace Level off

Security ON: Local OS Authentication

SNMP OFF

Listener Parameter File

/u01/app/oracle/product/12.1.0/dbhome\_1/network/admin/listener.o

ra

Listener Log File

/u01/app/oracle/diag/tnslsnr/*yourserver*/listener/alert/log.xml

Listening Endpoints Summary...

(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=*yourserver*)(PORT=1521)

))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=*yourserver*)(PORT=5500

))(Security=(my\_wallet\_directory=/u01/app/oracle/admin/orcl/xdb\_

wallet)) (Presentation=HTTP)(Session=RAW))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=*yourserver*)(PORT=5502

))(Security=(my\_wallet\_directory=/u01/app/oracle/admin/cdb1/xdb\_

wallet)) (Presentation=HTTP)(Session=RAW))

Services Summary...

Service "**cdb1**" has 1 instance(s).

Instance "**cdb1**", status READY, has 1 handler(s) for this

service...

Service "cdb1XDB" has 1 instance(s).

Instance "cdb1", status READY, has 1 handler(s) for this

service...

Service "em12rep" has 1 instance(s).

Instance "em12rep", status READY, has 1 handler(s) for this

service...

Service "em12repXDB" has 1 instance(s).

Instance "em12rep", status READY, has 1 handler(s) for this

service...

Service "orcl" has 1 instance(s).

Instance "orcl", status READY, has 1 handler(s) for this

service...

Service "orcl2" has 1 instance(s).

Instance "orcl2", status READY, has 1 handler(s) for this

service...

Service "orcl2XDB" has 1 instance(s).

Instance "orcl2", status READY, has 1 handler(s) for this

service...

Service "orclXDB" has 1 instance(s).

Instance "orcl", status READY, has 1 handler(s) for this

service...

Service "**pdb1\_1**" has 1 instance(s).

Instance "**cdb1**", status READY, has 1 handler(s) for this

service...

The command completed successfully

$

The listener is already started. If it were not started, you would use the following command

to start the listener:

$ **lsnrctl start**

LSNRCTL for Linux: Version 12.1.0.0.2 - Production on 09-JUL-

2012 03:08:50

Copyright (c) 1991, 2012, Oracle. All rights reserved.

Starting /u01/app/oracle/product/12.1.0/dbhome\_1/bin/tnslsnr:

please wait...

TNSLSNR for Linux: Version 12.1.0.0.2 - Production

System parameter file is

/u01/app/oracle/product/12.1.0/dbhome\_1/network/admin/listener.o

ra

Log messages written to

/u01/app/oracle/diag/tnslsnr/*yourserver*/listener/alert/log.xml

Listening on:

(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))

Listening on:

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=*yourserver*)(PORT=1521)

))

Connecting to

(DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC1521)))

STATUS of the LISTENER

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

Alias LISTENER

Version TNSLSNR for Linux: Version 12.1.0.0.2

- Production

Start Date 09-OCT-2012 03:08:50

Uptime 0 days 0 hr. 0 min. 0 sec

Trace Level off

Security ON: Local OS Authentication

SNMP OFF

Listener Parameter File

/u01/app/oracle/product/12.1.0/dbhome\_1/network/admin/listener.o

ra

Listener Log File

/u01/app/oracle/diag/tnslsnr/*yourserver*/listener/alert/log.xml

Listening Endpoints Summary...

(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=*yourserver*)(PORT=1521)

))

The listener supports no services

The command completed successfully

$

b. Check services.

$ **lsnrctl services**

LSNRCTL for Linux: Version 12.1.0.0.2 - Production on 06-SEP-

2012 23:29:20

Copyright (c) 1991, 2012, Oracle. All rights reserved.

Connecting to

(DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC1521)))

Services Summary...

Service "**cdb1**" has 1 instance(s).

Instance "**cdb1**", status READY, has 1 handler(s) for this

service...

Handler(s):

"DEDICATED" established:2 refused:0 state:ready

LOCAL SERVER

Service "cdb1XDB" has 1 instance(s).

Instance "cdb1", status READY, has 1 handler(s) for this

service...

Handler(s):

"D000" established:0 refused:0 current:0 max:1022

state:ready

DISPATCHER <machine: *yourserver*, pid: 27840>

(ADDRESS=(PROTOCOL=tcp)(HOST=*yourserver*)(PORT=29863))

Service "em12rep" has 1 instance(s).

Instance "em12rep", status READY, has 1 handler(s) for this

service...

Handler(s):

"DEDICATED" established:2748 refused:0 state:ready

LOCAL SERVER

Service "em12repXDB" has 1 instance(s).

Instance "em12rep", status READY, has 1 handler(s) for this

service...

Handler(s):

"D000" established:0 refused:0 current:0 max:1022

state:ready

DISPATCHER <machine: *yourserver*, pid: 18255>

(ADDRESS=(PROTOCOL=tcp)(HOST=*yourserver*)(PORT=59402))

Service "orcl" has 1 instance(s).

Instance "orcl", status READY, has 1 handler(s) for this

service...

Handler(s):

"DEDICATED" established:251 refused:0 state:ready

LOCAL SERVER

Service "orcl2" has 1 instance(s).

Instance "orcl2", status READY, has 1 handler(s) for this

service...

Handler(s):

"DEDICATED" established:4 refused:0 state:ready

LOCAL SERVER

Service "orcl2XDB" has 1 instance(s).

Instance "orcl2", status READY, has 1 handler(s) for this

service...

Handler(s):

"D000" established:6 refused:0 current:0 max:1022

state:ready

DISPATCHER <machine: *yourserver*, pid: 23615>

(ADDRESS=(PROTOCOL=tcp)(HOST=*yourserver*)(PORT=50200))

Service "orclXDB" has 1 instance(s).

Instance "orcl", status READY, has 1 handler(s) for this

service...

Handler(s):

"D000" established:0 refused:0 current:0 max:1022

state:ready

DISPATCHER <machine: *yourserver*, pid: 30821>

(ADDRESS=(PROTOCOL=tcp)(HOST=*yourserver*)(PORT=27384))

Service "**pdb1\_1**" has 1 instance(s).

Instance "**cdb1**", status READY, has 1 handler(s) for this

service...

Handler(s):

"DEDICATED" established:2 refused:0 state:ready

LOCAL SERVER

The command completed successfully

$

c. List the services automatically created for each container.

$ **sqlplus / as sysdba**

Connected to:

Oracle Database 12*c* Enterprise Edition Release 12.1.0.0.2 -

64bit Production

With the Partitioning, OLAP, Advanced Analytics and Real

Application Testing options

SQL> **col name format A20**

SQL> **select name, con\_id from v$services;**

NAME CON\_ID

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

pdb1\_1 3

cdb1XDB 1

cdb1 1

SYS$BACKGROUND 1

SYS$USERS 1

SQL>

Notice that the PDB$SEED service is not listed. No one should connect to this service

because there should be no operations performed on this container. It is reserved as a

template to create other PDBs.

3. Display the pluggable databases.

a. Use the new view V$PDBS.

SQL> **select CON\_ID, NAME, OPEN\_MODE from v$pdbs;**

CON\_ID NAME OPEN\_MODE

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

2 PDB$SEED READ ONLY

3 PDB1\_1 READ WRITE

SQL>

Notice that the seed PDB is in READ ONLY open mode.

b. Use the new command SHOW CON\_NAME and CON\_ID to know which container you

are connected to.

SQL> **show con\_name**

CON\_NAME

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

CDB$ROOT

SQL> **show con\_id**

CON\_ID

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

1

SQL>

You can also use SYS\_CONTEXT function to view the CON\_NAME and CON\_ID attributes

of your session context.

**SELECT sys\_context('userenv','CON\_NAME') from dual;**

**SELECT sys\_context('userenv','CON\_ID') from dual;**

4. View some of the new family of views CDB\_*xxx*:

SQL> **col PDB\_NAME format a8**

SQL> **col CON\_ID format 99**

SQL> **select PDB\_ID, PDB\_NAME, DBID, GUID, CON\_ID**

2 **from cdb\_pdbs;**

PDB\_ID PDB\_NAME DBID GUID CON\_ID

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

3 PDB1\_1 3624951709 C3920…..0C3 1

2 PDB$SEED 4029862422 C2CBF…….B141 1

SQL>

The PDB\_ID number 2 is always assigned to the seed PDB because it is the second

container to be created after the root container (CON\_ID 1).

5. Check all files of the CDB.

a. View the redo log files of the CDB.

SQL> **col MEMBER format A40**

SQL> **select GROUP#, CON\_ID, MEMBER from v$logfile;**

GROUP# CON\_ID MEMBER

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

3 0 /u01/app/oracle/oradata/cdb1/redo03.log

2 0 /u01/app/oracle/oradata/cdb1/redo02.log

1 0 /u01/app/oracle/oradata/cdb1/redo01.log

SQL>

b. View the control files of the CDB.

SQL> **col NAME format A60**

SQL> **select NAME , CON\_ID from v$controlfile;**

NAME CON\_ID

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

/u01/app/oracle/oradata/cdb1/control01.ctl 0

/u01/app/oracle/fast\_recovery\_area/cdb1/control02.ctl 0

SQL>

c. View all data files of the CDB, including those of the root and all PDBs.

1) With CDB\_DATA\_FILES view:

SQL> **col file\_name format A50**

SQL> **col tablespace\_name format A8**

SQL> **col file\_id format 9999**

SQL> **col con\_id format 999**

SQL> **select FILE\_NAME, TABLESPACE\_NAME, FILE\_ID, con\_id**

2 **from cdb\_data\_files order by con\_id ;**

FILE\_NAME TS NAME FILE\_ID CON\_ID

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

/$ORACLE\_BASE/oradata/cdb1/users01.dbf USERS 6 1

/$ORACLE\_BASE/oradata/cdb1/undotbs01.dbf UNDOTBS1 4 1

/$ORACLE\_BASE/oradata/cdb1/sysaux01.dbf SYSAUX 3 1

/$ORACLE\_BASE/oradata/cdb1/system01.dbf SYSTEM 1 1

/$ORACLE\_BASE/oradata/cdb1/pdbseed/system01.dbf SYSTEM 5 2

/$ORACLE\_BASE/oradata/cdb1/pdbseed/sysaux01.dbf SYSAUX 7 2

/$ORACLE\_BASE/oradata/cdb1/pdb1\_1/system01.dbf SYSTEM 8 3

/u01/app/oracle/oradata/cdb1/pdb1\_1/sysaux01.dbf SYSAUX 9 3

/$ORACLE\_BASE/oradata/cdb1/pdb1\_1/SAMPLE\_SCHEMA\_ USERS 10 3

users01.dbf

/$ORACLE\_BASE/oradata/cdb1/pdb1\_1/example01.dbf EXAMPLE 11 3

10 rows selected.

SQL>

2) With **ls** Unix command:

SQL> !**ls -l $ORACLE\_BASE/oradata/cdb1**

total 2575988

-rw-r----- 1 oracle oinstall 17874944 Sep 6 23:38

control01.ctl

drwxr-xr-x 2 oracle oinstall 4096 Sep 5 10:54 pdb1\_1

drwxr-x--- 2 oracle oinstall 4096 Sep 5 10:37 pdbseed

-rw-r----- 1 oracle oinstall 52429312 Sep 6 23:38 redo01.log

-rw-r----- 1 oracle oinstall 52429312 Sep 6 20:01 redo02.log

-rw-r----- 1 oracle oinstall 52429312 Sep 6 22:23 redo03.log

-rw-r----- 1 oracle oinstall 849354752 Sep 6 23:35 sysaux01.dbf

-rw-r----- 1 oracle oinstall 828383232 Sep 6 23:35 system01.dbf

-rw-r----- 1 oracle oinstall 571482112 Sep 6 23:18 temp01.dbf

-rw-r----- 1 oracle oinstall 246423552 Sep 6 23:36

undotbs01.dbf

-rw-r----- 1 oracle oinstall 5251072 Sep 6 22:29 users01.dbf

SQL> !**ls -l $ORACLE\_BASE/oradata/cdb1/pdbseed**

total 985064

-rw-r----- 1 oracle oinstall 88088576 Sep 5 10:50

pdbseed\_temp01.dbf

-rw-r----- 1 oracle oinstall 671096832 Sep 5 10:50 sysaux01.dbf

-rw-r----- 1 oracle oinstall 262152192 Sep 5 10:50 system01.dbf

SQL>

There are only the SYSTEM and SYSAUX data files and a temp file for the seed

PDB.

d. Still connected to the root, now use DBA\_DATA\_FILES view.

SQL> **col file\_name format A42**

SQL> **select FILE\_NAME, TABLESPACE\_NAME, FILE\_ID**

2 **from dba\_data\_files;**

FILE\_NAME TABLESPACE NAME FILE\_ID

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

/u01/app/oracle/oradata/cdb1/users01.dbf USERS 6

/u01/app/oracle/oradata/cdb1/undotbs01.dbf UNDOTBS1 4

/u01/app/oracle/oradata/cdb1/sysaux01.dbf SYSAUX 3

/u01/app/oracle/oradata/cdb1/system01.dbf SYSTEM 1

SQL>

Notice that only the root data files are listed.

e. Now use V$TABLESPACE and V$DATAFILE view.

SQL> **col NAME format A12**

SQL> **select FILE#, ts.name, ts.ts#, ts.con\_id**

**2 from v$datafile d, v$tablespace ts**

**3 where d.ts#=ts.ts#**

**4 and d.con\_id=ts.con\_id**

**5 order by 4,3;**

FILE# NAME TS# CON\_ID

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

1 SYSTEM 0 1

3 SYSAUX 1 1

4 UNDOTBS1 2 1

6 USERS 4 1

5 SYSTEM 0 2

7 SYSAUX 1 2

8 SYSTEM 0 3

9 SYSAUX 1 3

10 USERS 3 3

11 EXAMPLE 4 3

10 rows selected.

SQL>

f. List the temp files of the CDB.

SQL> **col file\_name format A47**

SQL> **select FILE\_NAME, TABLESPACE\_NAME, FILE\_ID**

2 **from cdb\_temp\_files;**

FILE\_NAME TABLESPA

FILE\_ID

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

/u01/app/oracle/oradata/cdb1/temp01.dbf TEMP

1

/u01/app/oracle/oradata/cdb1/pdbseed/pdbseed\_temp01.dbf TEMP

2

/u01/app/oracle/oradata/cdb1/pdb1\_1/pdb1\_1\_temp01.dbf TEMP

3

SQL>

6. List all users created.

a. Verify that the SYSTEM user is created.

SQL> **col username format A22**

SQL> **select username, common, con\_id from cdb\_users**

2 **where username ='SYSTEM';**

USERNAME COM CON\_ID

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

SYSTEM YES 1

SYSTEM YES 2

SYSTEM YES 3

SQL>

Notice that the user SYSTEM exists in all containers as a common user.

b. List all common users of the CDB.

SQL> **select distinct username from cdb\_users**

2 **where common ='YES';**

USERNAME

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

DVF

MDSYS

XS$NULL

SYSKM

APEX\_040100

DIP

SPATIAL\_WFS\_ADMIN\_USR

FLOWS\_FILES

SYSBACKUP

CTXSYS

OUTLN

SPATIAL\_CSW\_ADMIN\_USR

GSMUSER

OLAPSYS

SYSTEM

ORACLE\_OCM

DVSYS

AUDSYS

ORDSYS

DBSNMP

OJVMSYS

GSMADMIN\_INTERNAL

MDDATA

APEX\_PUBLIC\_USER

ORDPLUGINS

APPQOSSYS

GSMCATUSER

ORDDATA

SYSDG

XDB

SYS

WMSYS

LBACSYS

ANONYMOUS

SI\_INFORMTN\_SCHEMA

35 rows selected.

SQL>

c. List all local users of the CDB.

SQL> **select distinct username, con\_id from cdb\_users**

2 **where common ='NO';**

USERNAME CON\_ID

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

SCOTT 3

BI 3

PM 3

IX 3

SH 3

OE 3

HR 3

PDBADMIN 3

SQL>

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

SCOTT 3

BI 3

PM 3

d. List local users in the root.

SQL> **select username, con\_id from cdb\_users**

2 **where common ='NO';**

USERNAME CON\_ID

IX 3

SH 3

OE 3

HR 3

PDBADMIN 3

8 rows selected.

SQL>

Notice that there is no local user in the root container because it is impossible to create any

local user in the root.

7. List all roles and privileges of the CDB.

a. List all roles of the CDB.

SQL> **col role format A30**

SQL> **select role, common, con\_id from cdb\_roles;**

ROLE COM CON\_ID

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

CONNECT YES 1

RESOURCE YES 1

DBA YES 1

AUDIT\_ADMIN YES 1

AUDIT\_VIEWER YES 1

SELECT\_CATALOG\_ROLE YES 1

EXECUTE\_CATALOG\_ROLE YES 1

DELETE\_CATALOG\_ROLE YES 1

PROF\_ADMIN YES 1

EXP\_FULL\_DATABASE YES 1

IMP\_FULL\_DATABASE YES 1

CDB\_DBA YES 1

…

DV\_XSTREAM\_ADMIN YES 2

DV\_GOLDENGATE\_REDO\_ACCESS YES 2

DV\_AUDIT\_CLEANUP YES 2

DV\_REALM\_OWNER YES 2

PDB\_DBA YES 2

…

DV\_AUDIT\_CLEANUP YES 3

DV\_REALM\_RESOURCE YES 3

DV\_REALM\_OWNER YES 3

PDB\_DBA YES 3

252 rows selected.

SQL>

Notice that there is no local role in the root container because it is impossible to create any

local role in the root.

b. Check that the privileges are neither common nor local by nature.

SQL> **desc sys.system\_privilege\_map**

Name Null? Type

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

PRIVILEGE NOT NULL NUMBER

NAME NOT NULL VARCHAR2(40)

PROPERTY NOT NULL NUMBER

SQL> **desc sys.table\_privilege\_map**

Name Null? Type

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

PRIVILEGE NOT NULL NUMBER

NAME NOT NULL VARCHAR2(40)

SQL>

Notice that there is no COMMON column.

c. Verify that the privilege, when granted, becomes a common or local privilege.

SQL> **desc CDB\_SYS\_PRIVS**

Name Null? Type

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

GRANTEE VARCHAR2 (128)

PRIVILEGE VARCHAR2(40)

ADMIN\_OPTION VARCHAR2(3)

**COMMON** VARCHAR2(3)

CON\_ID NUMBER

SQL> **desc CDB\_TAB\_PRIVS**

Name Null? Type

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

GRANTEE VARCHAR2(128)

OWNER VARCHAR2(128)

TABLE\_NAME VARCHAR2(128)

GRANTOR VARCHAR2(128)

PRIVILEGE VARCHAR2(40)

GRANTABLE VARCHAR2(3)

HIERARCHY VARCHAR2(3)

**COMMON** VARCHAR2(3)

TYPE VARCHAR2(24)

CON\_ID NUMBER

SQL>

There is a **COMMON** column.

d. Notice that the role, though common or local depending on how the role was created is

also, like privileges, either granted commonly or locally.

SQL> **col grantee format A10**

SQL> **col granted\_role format A28**

SQL> **select grantee, granted\_role, common, con\_id**

2 **from cdb\_role\_privs**

3 **where grantee='SYSTEM';**

GRANTEE GRANTED ROLE COM CON\_ID

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

SYSTEM DBA YES 1

SYSTEM AQ\_ADMINISTRATOR\_ROLE YES 1

SYSTEM DBA YES 2

SYSTEM AQ\_ADMINISTRATOR\_ROLE YES 2

SYSTEM DBA YES 3

SYSTEM AQ\_ADMINISTRATOR\_ROLE YES 3

6 rows selected.

SQL> **EXIT**

$