Practice 5: Creating a PDB by Cloning from Local and Remote PDBs

Practice Overview

In this practice you will perform the following:

- Create a PDB by cloning a local PDB
- Create a PDB by cloning a remote PDB

Practice Assumptions

- You have the `srv1` and its CDB database up and running.

Creating a PDB by cloning a local PDB

In the following sections, you will create a new PDB by cloning a local PDB. Doing this process online requires to have the CDB running in ARCHIVELOG mode.

A. Enabling Archivelog Mode

1. Create a Putty session to `srv1` and login as `oracle` user.

2. Login to the CDB as `sysdba` in SQL*Plus.

   ```
   sqlplus / as sysdba
   ```

3. Check out whether the CDB is running in ARCHIVELOG mode.

   When you created the CDB, you did not enable the ARCHIVELOG mode.

   ```
   SELECT LOG_MODE FROM V$DATABASE ;
   ```

4. Check out if any of the archive destinations is pointing to any location.

   ```
   SELECT NAME, VALUE
   FROM V$SYSTEM_PARAMETER
   WHERE UPPER(NAME) LIKE 'LOG_ARCHIVE_DEST_%'
     AND VALUE IS NOT NULL AND VALUE <>'enable';
   ```

5. Set the archive destination to the fast recovery area.

   ```
   ALTER SYSTEM SET LOG_ARCHIVE_DEST_1 = 'LOCATION=USE_DB_RECOVERY_FILE_DEST'
   SCOPE=BOTH;
   ```

6. Enable the ARCHIVELOG mode.

   ```
   SHUTDOWN IMMEDIATE
   STARTUP MOUNT
   ALTER DATABASE ARCHIVELOG;
   ALTER DATABASE OPEN;
   ALTER SYSTEM SWITCH LOGFILE;
   SELECT NAME FROM V$ARCHIVED_LOG;
   ```

7. Confirm the CDB has the local undo property enabled.

   This is a requirement to perform the cloning online.

   ```
   SELECT PROPERTY_VALUE FROM DATABASE_PROPERTIES
    WHERE PROPERTY_NAME = 'LOCAL_UNDO_ENABLED';
   ```

B. Create a new PDB by cloning a local PDB

In this section you will create a new `PDB`, named `PDB3`, by cloning `PDB2`.

8. Verify the you are connected to the root.

```
SHOW CON_NAME
```

9. Verify the OMF is enabled.

If OMF is not enabled, you need to specify the destination of the new PDB data files.

```
SHOW PARAMETER DB_CREATE_FILE_DEST
```

10. Create a new PDB named PDB3 from cloning `PDB2`.

In Oracle 12.1, you need to close pdb2 first.

```
CREATE PLUGGABLE DATABASE pdb3 FROM pdb1;
```

11. Open the `PDB` in read/write mode

```
ALTER PLUGGABLE DATABASE pdb3 OPEN;
```

12. Retrieve list of the newly created PDB datafiles

```
set linesize 100
col name format a100
SELECT NAME
FROM   V$DATAFILE
WHERE  CON_ID = ( SELECT CON_ID FROM V$PDBS WHERE NAME='PDB3' );
```

C. Create a new PDB by cloning a remote PDB

In this section you will create a new PDB, named PDB4, by cloning a remote DPB. The source PDB is PDB3 and to consider it as a remote PDB, you will create database link in the root and use it to clone the PDB.

13. Create a database link to the CDB root container.

```
CREATE DATABASE LINK CDB1_LINK
 CONNECT TO system IDENTIFIED BY oracle
 USING 'CDB1';
```

14. Test the database link.

```
SELECT SYSDATE FROM DUAL@CDB1_LINK;
```

15. Create a new PDB named PDB4 from cloning PDB3 as a remote PDB.

```
CREATE PLUGGABLE DATABASE pdb4 FROM PDB3@CDB1_LINK;
```

16. Open the PDB in read/write mode

```
ALTER PLUGGABLE DATABASE pdb4 OPEN;
```

17. Test connecting to the new PDB

```
connect sys/oracle@srv1:1521/PDB4.localdomain as SYSDBA
SHOW CON_NAME
```

Summary

You can create a PDB by cloning a local or remote PDB. In 12.2, this can be done online, i.e. you do not have to close the source pdb.