

# **Creating Pluggable Databases (PDBs)**

**By Ahmed Baraka**

# Objectives

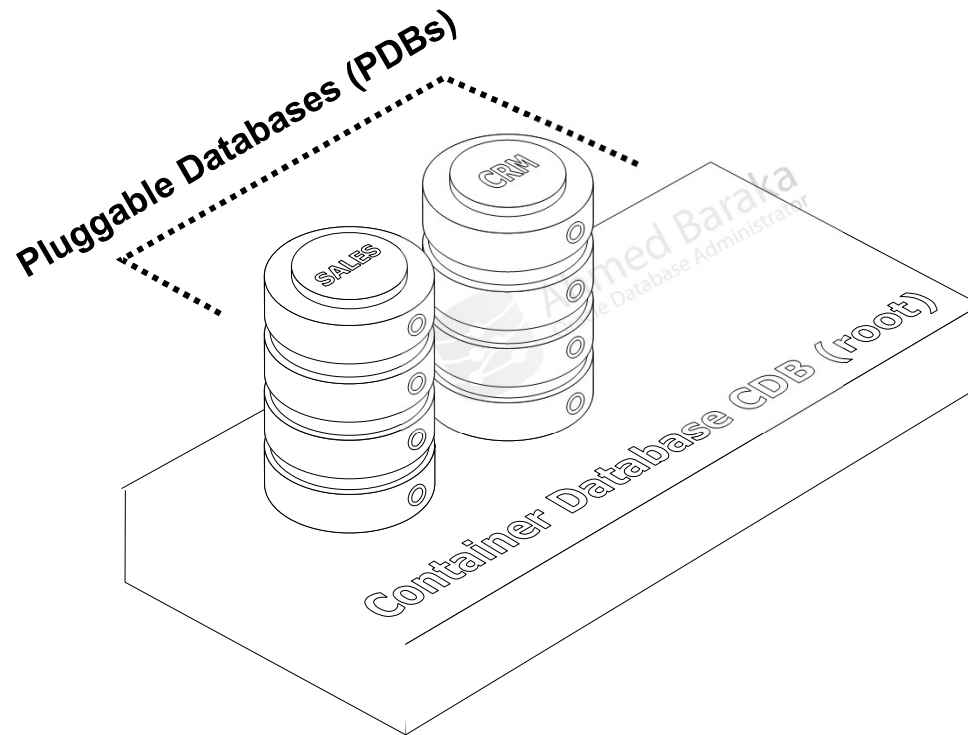
---

In this lecture, you will learn how to perform the following:

- Describe PDB creation process and its prerequisites
- Create a PDB using dbca and SQL Developer
- Create a PDB using **CREATE PLUGGABLE DATABASE** statement
- Create a PDB from the Seed Container
- Clone a Local PDB
- Drop a PDB

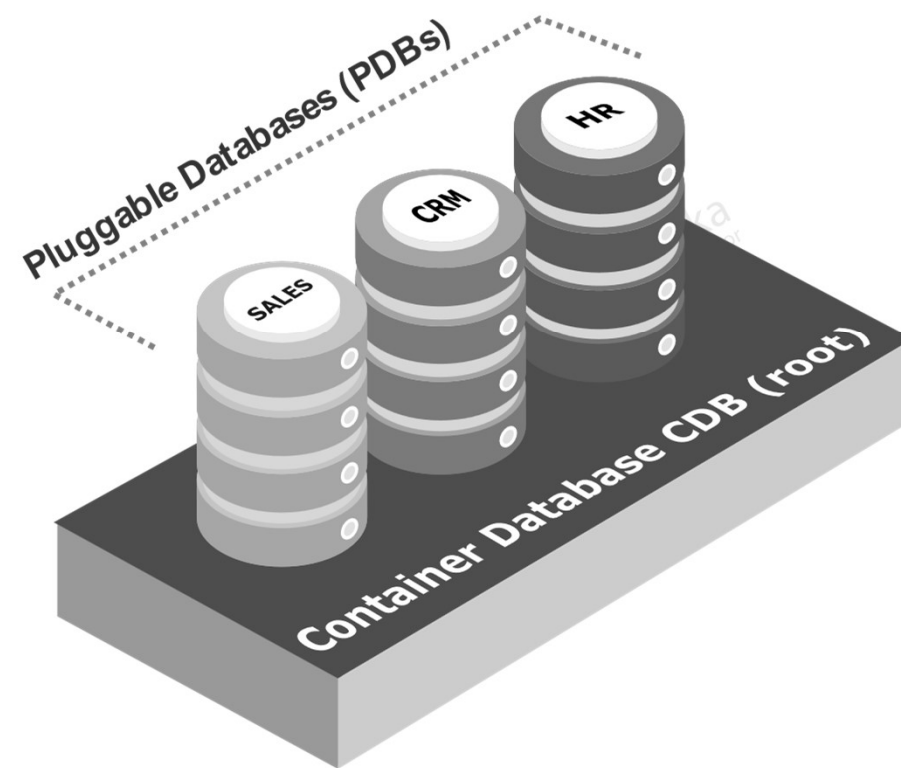
## A CD Database with Two application PDBs

---



## An Oracle DB Database with Three User PDBs

---



# Creating a PDB Prerequisites

---

- The CDB must be open in read/write mode
- The current user must be a common user
- The current user must have the **CREATE PLUGGABLE DATABASE** system privilege



# PDB Creation Tools

---

- SQL\*Plus
  - Using **CREATE PLUGGABLE DATABASE** statement
- **dbca**
  - Copy from seed
  - Plugging from supplied templates (schema templates)
  - Unplugging / plugging method
  - Cloning from a remote PDB (19c)
- SQL Developer
- EM Cloud Control

# Duplicating an Active PDB using DBCA

---

Select Database Operation

19<sup>c</sup> ORACLE<sup>®</sup>  
Database

☒ Database Operation

☐ Manage Pluggable Databases

☐ Select Database

☐ Create Pluggable Database

☐ PDB Identification

☐ Pluggable Database Options

☐ Summary

☐ Progress Page

☐ Finish

Select the operation that you want to perform.

☐ Create a database

☐ Configure an existing database

☐ Delete database

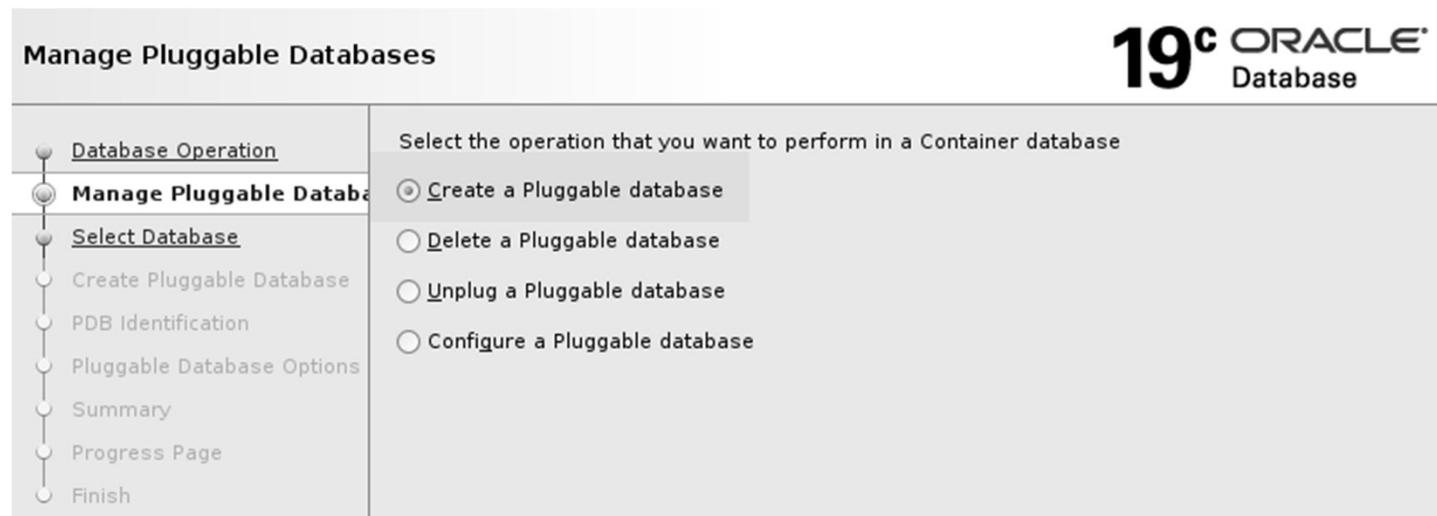
☐ Manage templates

☒ Manage pluggable databases

☐ Oracle RAC database instance management

# Duplicating an Active PDB using DBCA

---





# Duplicating an Active PDB using DBCA

Select Source Database

19<sup>c</sup> ORACLE<sup>®</sup>  
Database

Database Operation

Manage Pluggable Databases

**Select Database**

Create Pluggable Database

PDB Identification

Pluggable Database Options

Summary

Progress Page

Finish

Select a Container database within which Pluggable database needs to be created.

Database	Local instance	Type
<input checked="" type="radio"/> oradb	oradb	Single Instance

Ahmed Baraka  
Oracle Database Administrator

DBCA will connect to the database using OS based authentication. Database credentials may be needed if OS based authentication is disabled. Specify the credentials, if needed.

User name:

sys

Password:

.....

# Duplicating an Active PDB using DBCA

---

Create Pluggable Database

19<sup>c</sup> ORACLE<sup>®</sup>  
Database

Database Operation

Manage Pluggable Databases

Select Database

**Create Pluggable Database**

PDB Identification

Pluggable Database Options

Summary

Progress Page

Finish

☒ Create a new Pluggable database from another PDB

Select Pluggable database: 

PDB\$SEED  
PDB\$SEED  
PDB1

☐ Create as clone

☐ Create from PDB archive

Pluggable database archive:  

☒ Create using PDB file set

Pluggable database metadata file:  

Pluggable database datafile backup:

Oracle Database Administration from Zero to Hero - a course by Ahmed Baraka

# Duplicating an Active PDB using DBCA

---

**Pluggable Database Identification Options** **19<sup>c</sup>** ORACLE<sup>®</sup>  
Database

- Database Operation
- Manage Pluggable Databases
- Select Database
- Create Pluggable Database
- PDB Identification**
- Pluggable Database Options
- Summary
- Progress Page
- Finish

Pluggable database name:

☒ Create a new administrator

Administrator user name:

Administrator password:

Confirm administrator password:

☐ Lock all existing PDB users

# Duplicating an Active PDB using DBCA

---

### Pluggable Database Options

Database Operation

Manage Pluggable Databases

Select Database

Create Pluggable Database

**PDB Identification**

**Pluggable Database Options**

Summary

Progress Page

Finish

19c ORACLE Database

PDB storage options

Selected Container database is on OMF. PDB datafiles will be placed in the following location.

Storage type:

File System

Database location:

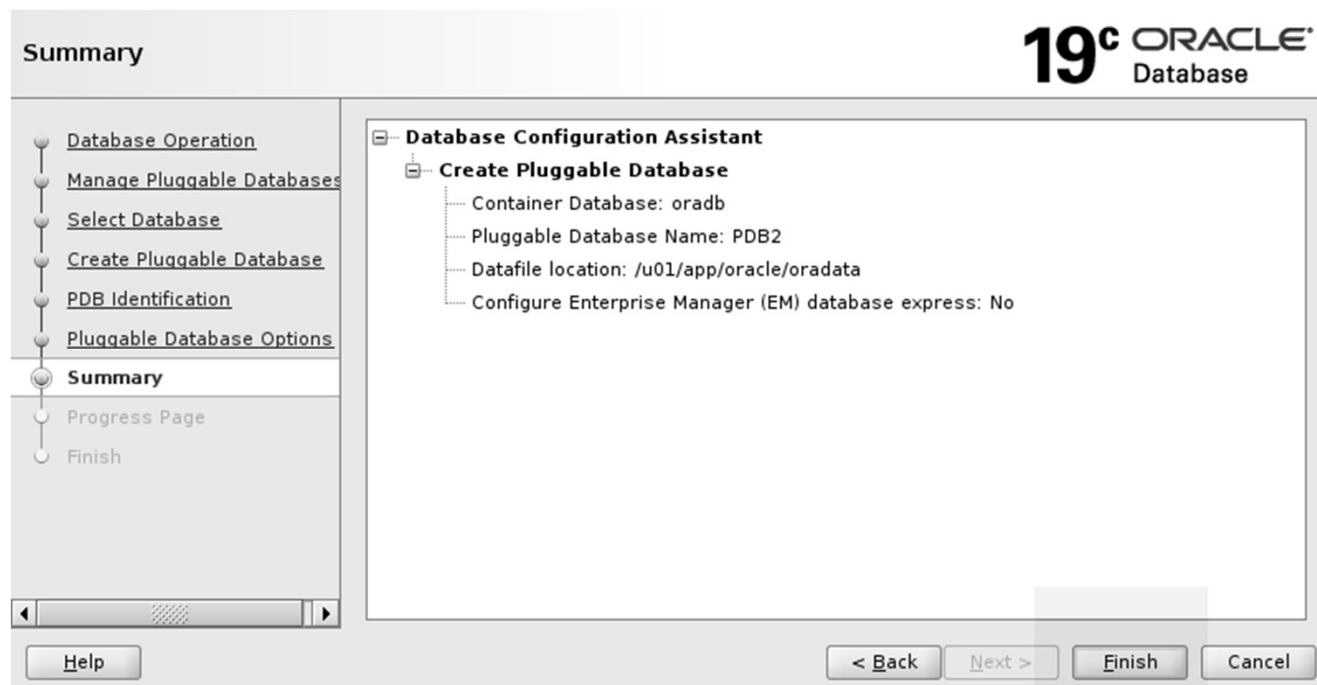
/u01/app/oracle/oradata

Browse...

☒ Create default user tablespace

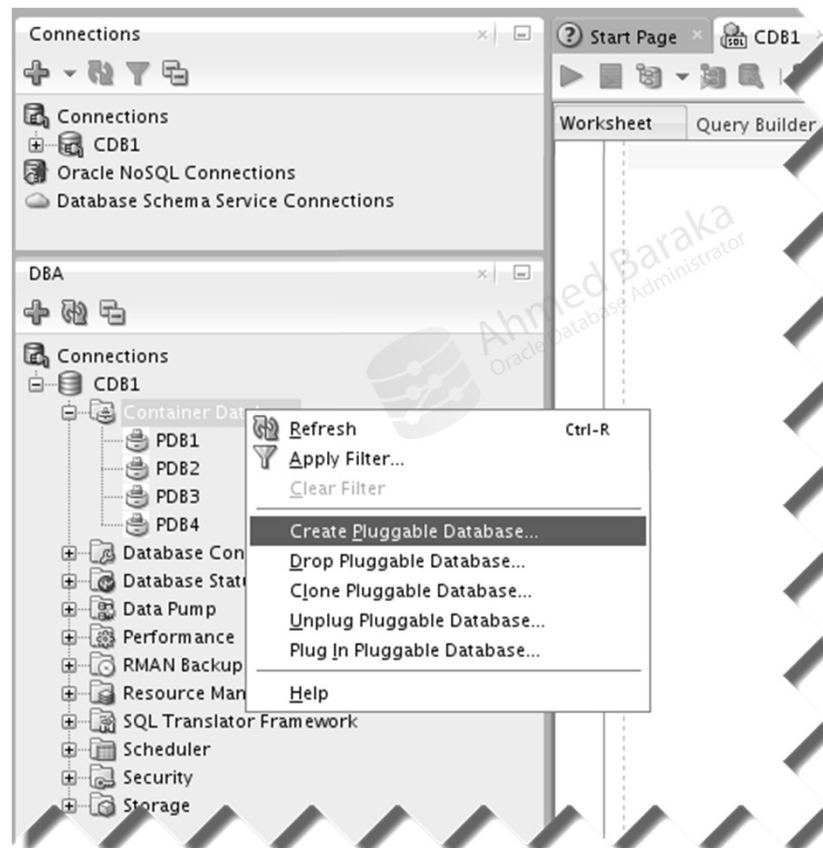
# Duplicating an Active PDB using DBCA

---



# Creating a PDB using SQL Developer

---



# Options for Creating a PDB using CREATE PLUGGABLE DATABASE command

---

Category	Option	Description
Copying	<b>From the seed</b>	By using the files of the CDB seed or application seed.
	<b>Clone a local PDB</b>	By cloning a source local PDB.
	<b>Clone a remote PDB</b>	By cloning a remote PDB.
	<b>Clone a non-CDB</b>	By cloning a non-CDB.
Plugging In	<b>Plugging in an Unplugged PDB</b>	By plugging in an unplugged PDB
	<b>Adopting a Non-CDB as a PDB (Using the DBMS_PDB)</b>	By creating an unplugged PDB from a non-CDB and then plug the unplugged PDB into the CDB.
Relocating	<b>Relocating a PDB</b>	By moving the files associated with the PDB to a new location.
Proxy PDB	<b>Creating a PDB as a proxy PDB</b>	By referencing a different PDB with a database link

# File Location of the New PDB

---

Clause or Initialization Parameter	Precedence Order	Description
<b>FILE_NAME_CONVERT</b> clause	1	Filenames and destination of the data files are defined by converting specific string in the source filenames to another defined string.
<b>CREATE_FILE_DEST</b> clause	2	Sets the OMF destination in the new PDB and creates the new datafiles in it.
<b>DB_CREATE_FILE_DEST</b> initialization parameter	3	When defined in the root, it specifies the default location for Oracle Managed Files (OMF) for the CDB.
<b>PDB_FILE_NAME_CONVERT</b> initialization parameter	4	This initialization parameter maps names of existing files to new file names when processing a CREATE PLUGGABLE DATABASE statement.



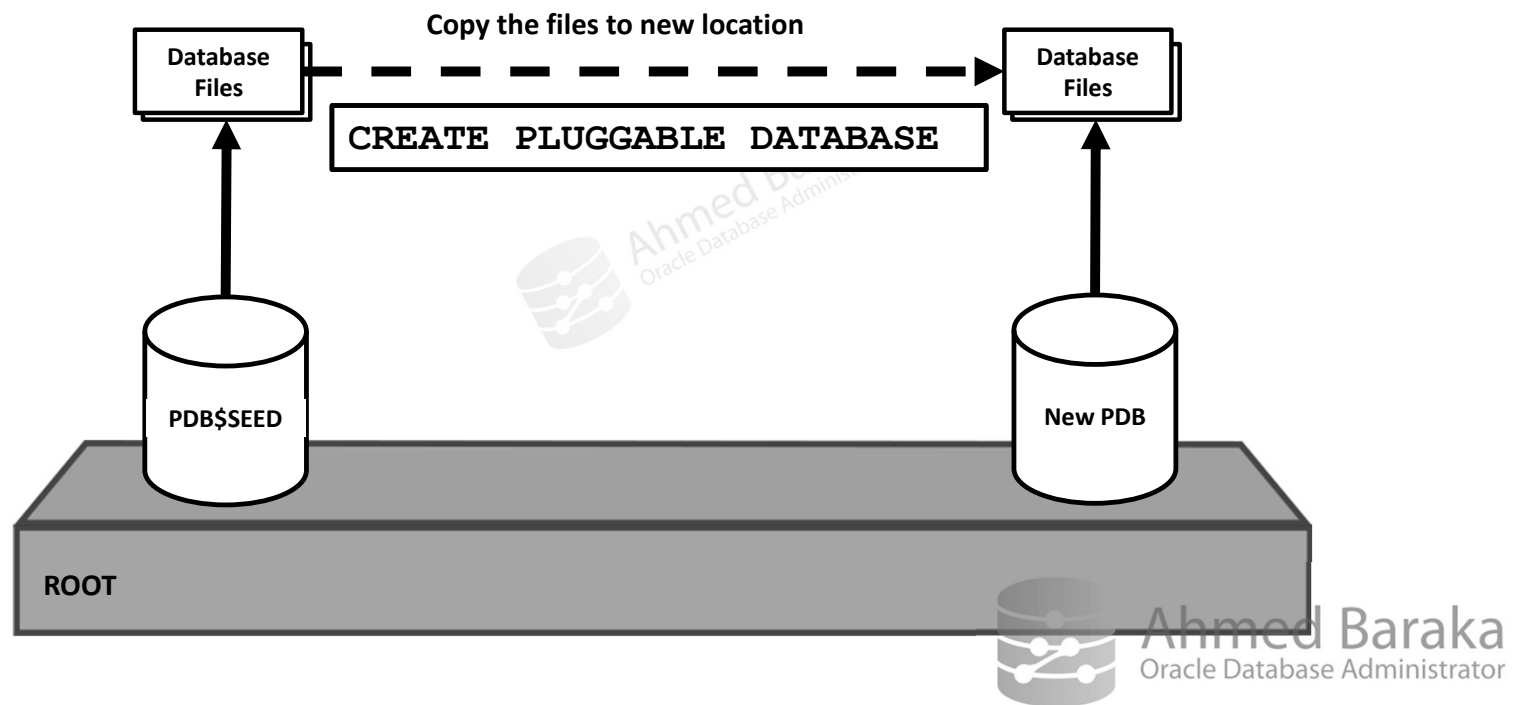
# More CREATE PLUGGABLE DATABASE Optional Clauses

---

Clause	Task/Description
<b>DEFAULT TABLESPACE</b>	Create a new tablespace and make it the default tablespace for the new PDB.
<b>STORAGE</b>	Limit the amount of storage that the PDB can use. If omitted, no limit is set.
<b>ROLES</b>	Define the roles that you want to grant to the local role PDB_DBA. PDB_DBA role is in turn granted to the local administrator.
<b>NO DATA</b>	Clone the data model definition of the source PDB without cloning its data

# About Creating a PDB from the Seed Container

---



# Creating a PDB from the Seed Procedure

---

1. Login to the root container as a common user (like **SYSTEM**)
2. Run the **CREATE PLUGGABLE DATABASE** statement. You must specify a local administrator for the PDB.
3. Open the new PDB in read/write mode.

```
ALTER PLUGGABLE DATABASE pdb2 OPEN;
```

4. Back up the PDB.

# Creating a PDB from the Seed Example 1

---

```
CREATE PLUGGABLE DATABASE pdb1  
  ADMIN USER pdb1admin IDENTIFIED BY mypassword  
  ROLES= (CONNECT,DBA) ;
```

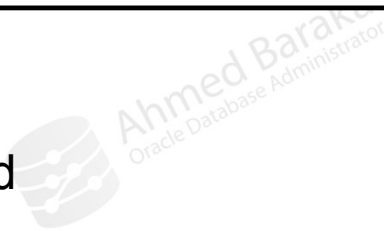
- Assumptions:
  - Either `DB_CREATE_FILE_DEST` for the CDB, or the `PDB_FILE_NAME_CONVERT` initialization parameter are set.
  - No storage limit is required

# Creating a PDB from the Seed Example 2

---

```
CREATE PLUGGABLE DATABASE pdb1  
  ADMIN USER pdb1admin IDENTIFIED BY mypassword  
  ROLES=(DBA)  
  CREATE_FILE_DEST='/u01/oradata/cdb1/pdb1';
```

- Assumptions:
  - No storage limit is required



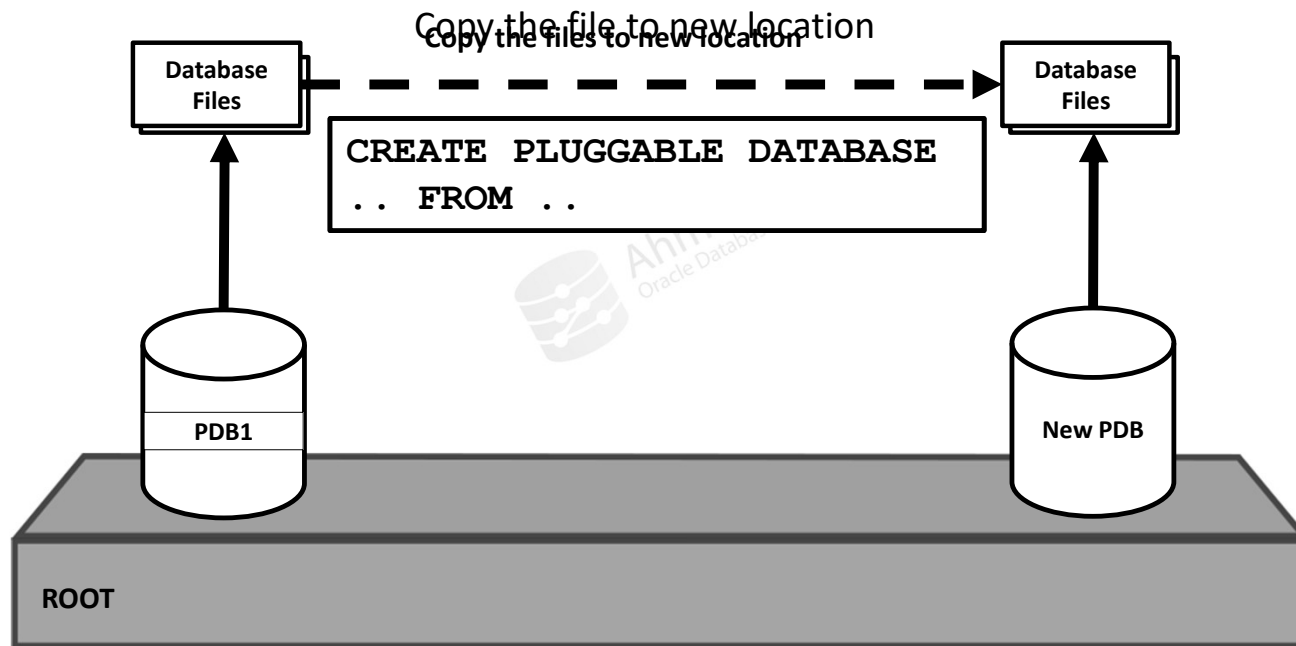
# Creating a PDB from the Seed Example 3

---

```
CREATE PLUGGABLE DATABASE pdb2
  ADMIN USER pdb2admin IDENTIFIED BY mypassword
  STORAGE (MAXSIZE 2G)
  DEFAULT TABLESPACE HR
  DATAFILE '/u01/oracle/dbs/pdb2/hr01.dbf' SIZE 250M AUTOEXTEND ON
  FILE_NAME_CONVERT = ('/u01/oracle/dbs/pdbseed/',
    '/u01/oradata/cdb1/pdb2/');
```

# About Cloning from a Local PDB

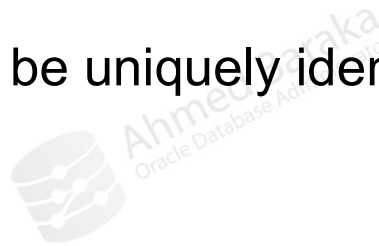
---



# Cloning from a PDB Considerations

---

- In 12.1: the source PDB must be in a READ-ONLY state.
- In 12.2: the source PDB can be open, providing that the CDB is in **ARCHIVELOG** mode with local UNDO enabled.
- Each PDB in a CDB must be uniquely identifiable.





# Cloning from a Local PDB Procedure

---

1. Connect to the root as a common user
2. Close the source PDB then start it in READ ONLY (if needed)

```
ALTER PLUGGABLE DATABASE pdb1 CLOSE;  
ALTER PLUGGABLE DATABASE pdb1 OPEN READ ONLY;
```

3. Clone the PDB

```
CREATE PLUGGABLE DATABASE pdb2 FROM pdb1;
```

```
CREATE PLUGGABLE DATABASE pdb2 FROM pdb1  
FILE_NAME_CONVERT=('/u01/pdb1', '/u01/pdb2');
```

4. Open the PDB in read/write mode

# Dropping a PDB

---

- The PDB must be closed
- Must be done from the root
- Use the following statement:

```
DROP PLUGGABLE DATABASE PDB1 [ [ KEEP | INCLUDING ] DATAFILES];
```

- **KEEP DATAFILES** requires the PDB to be unplugged
  - **INCLUDING DATAFILES** requires the PDB to be in mounted mode (closed) or unplugged
- The dbca and SQL Developer can also be used

# Summary

---

In this lecture, you should have learnt how to perform the following:

- Describe PDB creation process and its prerequisites
- Create a PDB using dbca and SQL Developer
- Create a PDB using **CREATE PLUGGABLE DATABASE** statement
- Create a PDB from the Seed Container
- Clone a Local PDB
- Drop a PDB