**Definition of a Pluggable Database (PDB)**

**A Pluggable Database (PDB)** is a fully functional, self-contained database that resides within a Container Database (CDB) in Oracle's multitenant architecture. The PDB shares the common resources of the CDB, such as the memory and background processes, but operates as a separate and independent database.

**Key Characteristics of a PDB:**

**Self-contained:** Each PDB has its own set of data files, tablespaces, and system metadata (like users and privileges).

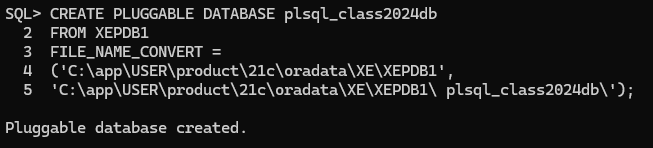
**Multitenancy:** Multiple PDBs can coexist within a single CDB, offering a multitenant architecture where you can manage many databases within a unified framework.

**Separation of Applications:** PDBs allow for isolated database environments, which is useful for separating applications or tenants.

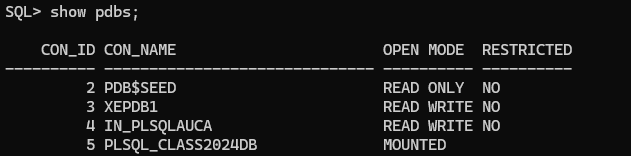
**Task number 1:**

**1.2. Create the Pluggable Database**

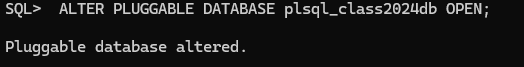
Run the following command to create a new pluggable database named new\_pdb:



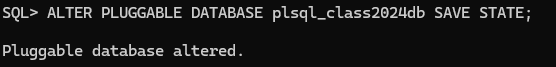
**Verify if successfully created and visible:**



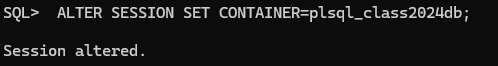
**Open a PDB:**



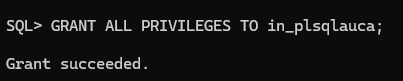
**Saving the state :**



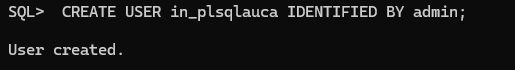
**Altering session:**



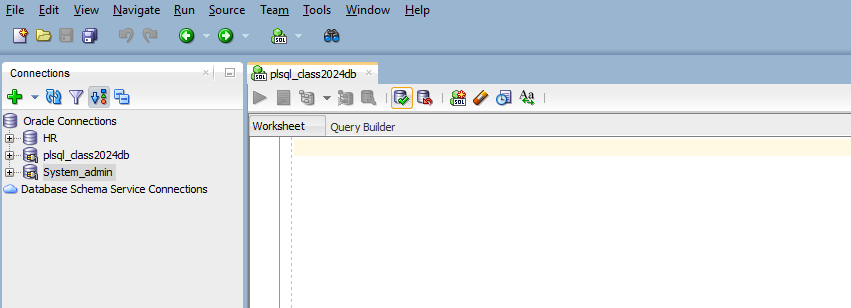
**Granting privileges:**



**Create a User in a PDB**:



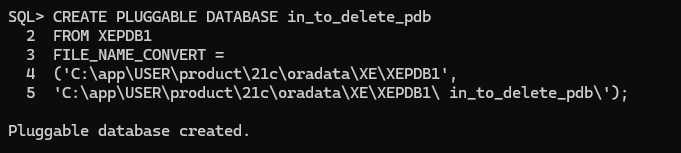
**Making a connection in sql developer using the user we created in sql plus and the credentials created :**



**Task number 2:**

**Create the Pluggable Database from CDB:**

Now, you can issue the command to create the pluggable database.



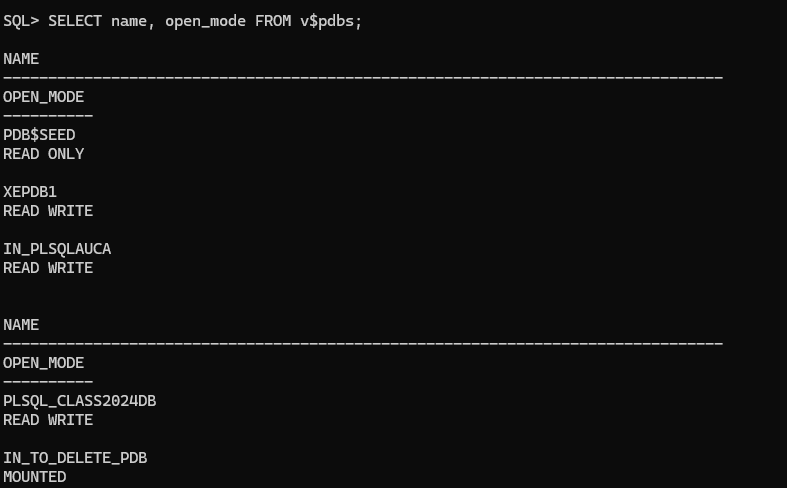
**Open the New Pluggable Database**

After creating the PDB, you need to open it:

**ALTER PLUGGABLE DATABASE in\_to\_delete\_pdb OPEN;**

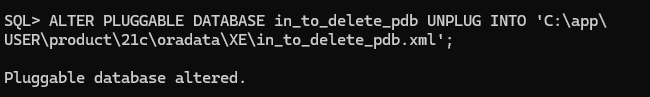
**Verify the New PDB**

You can verify that the PDB has been created by running**:**



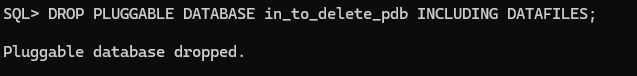
**Unplug the Pluggable Database**

Next, unplug the PDB. This operation will create an XML file that stores metadata about the PDB.



**Drop the Pluggable Database**

Now that the PDB is unplugged, you can drop it. By default, this command will only drop the PDB's metadata. To remove all the associated data files, use the **INCLUDING DATAFILES** option.



**Verify the Drop**

You can verify that the PDB has been successfully deleted by running:

