

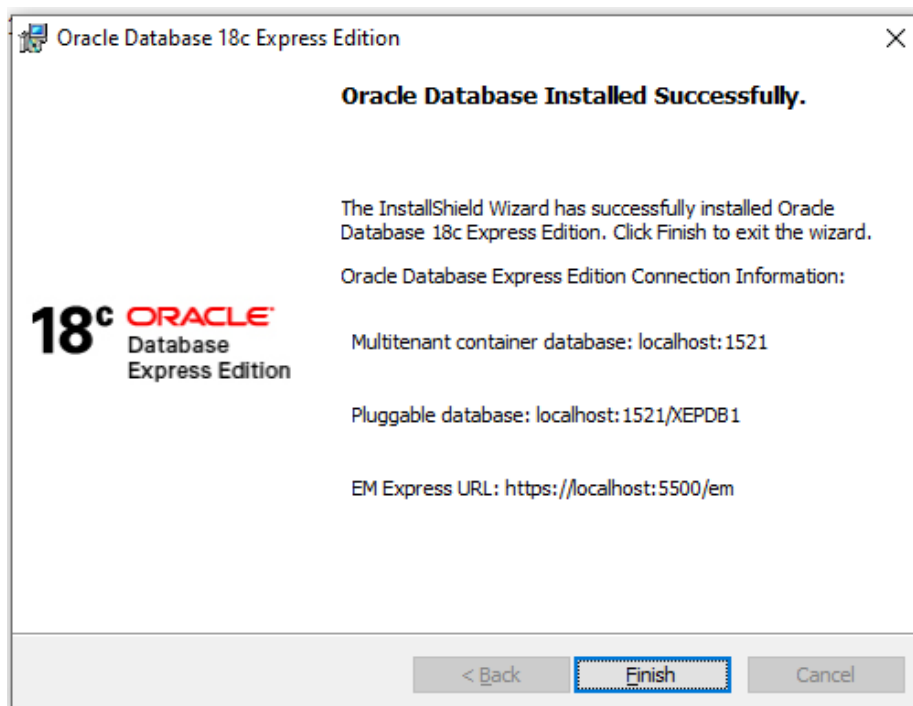
DJANGO ORACLE DATABASE INTEGRATION

Downloading Instructions (prerequisites):

1. Download Oracle Database 18c or 19c [64 bit] (Django 3.2 only supports version 12c and above, 11g will not work). If older version or 32bit is already installed, check the uninstall section below to completely remove that version of oracle database.
2. Download SQL developer with JDK included 64-bit.
3. Install cx_Oracle package (pip install cx_Oracle)

Installation:

1. Install Oracle 18c (for this installation purpose, I'm using **system** as password for **system/sys/pdbadmin**)



2. Register the Pluggable database in tnsnames.ora file. The file will be located inside admin in network. [C:\app\<username>\product\18.0.0\dbhomeXE\network\admin\tnsnames.ora]
3. Add the below mentioned details to the end of the tnsnames.ora

```
XEPDB1 =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = LOCALHOST)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = XEPDB1)
  )
)
```

```
tnsnames.ora
File Edit Format View Help
# tnsnames.ora Network Configuration File: C:\app\Rakesh\product\18.0.0\dbhomeXE\NETWORK\ADMIN\tnsnames.ora
# Generated by Oracle configuration tools.

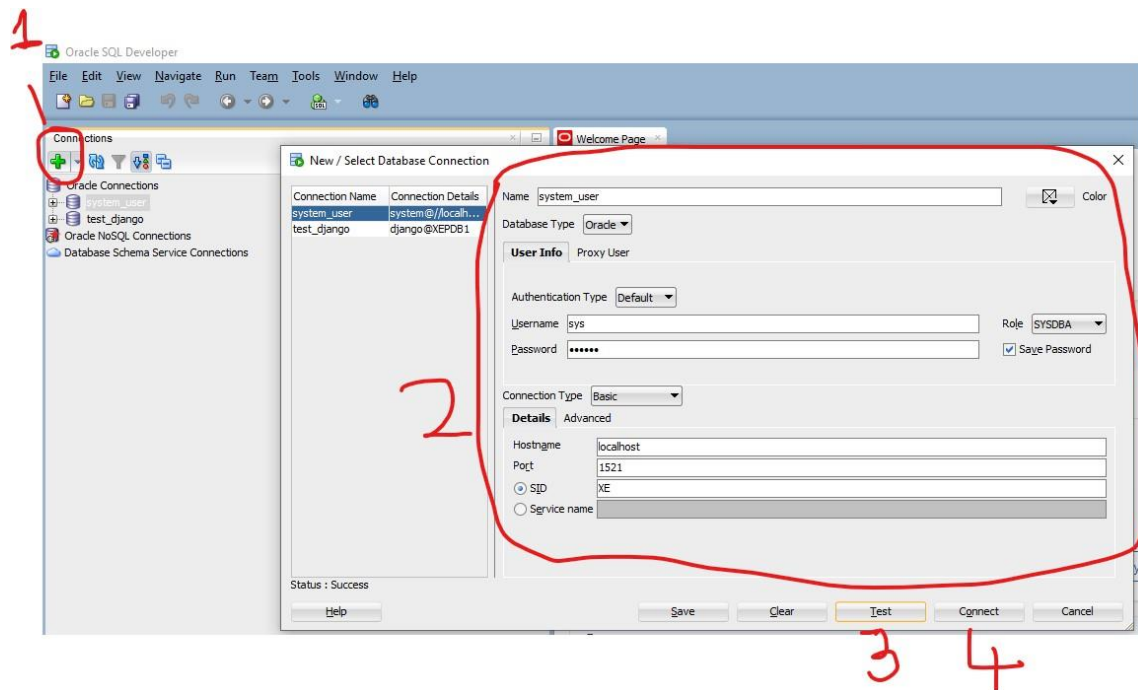
XE =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = DESKTOP-OTAIIRI8)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = XE)
    )
  )

LISTENER_XE =
  (ADDRESS = (PROTOCOL = TCP)(HOST = DESKTOP-OTAIIRI8)(PORT = 1521))

ORACL_R_CONNECTION_DATA =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
    )
    (CONNECT_DATA =
      (SID = CLRExtProc)
      (PRESENTATION = RO)
    )
  )

XEPOB1 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = LOCALHOST)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = XEPDB1)
    )
  )
```

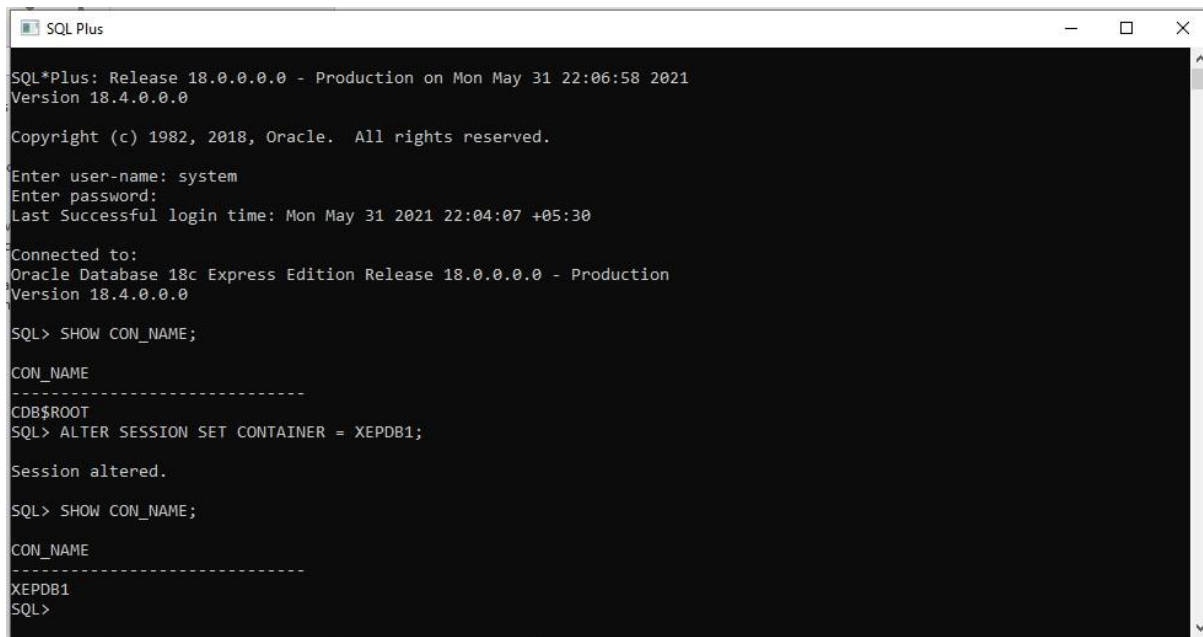
4. Extract/Install SQL developer from the zip file and open SQL Developer.
5. Create a Connection for **sys** as **SYSDBA** [password = **system**] then click test and connect.



6. Open SQL worksheet of system user and create a user named Django and grant privileges.

Note:

- You can only create users from system user/account with CREATE USER privilege.
- As 18c is a cloud-based database, you need to change your connection from CDB\$ROOT to pluggable database [XEPDB1] to create users.

A screenshot of an SQL*Plus terminal window. The window title is "SQL Plus". The text inside shows the following sequence of events: 1. SQL*Plus Release 18.0.0.0.0 - Production on Mon May 31 22:06:58 2021. Version 18.4.0.0.0. 2. Copyright (c) 1982, 2018, Oracle. All rights reserved. 3. Enter user-name: system. 4. Enter password: (prompted). 5. Last Successful login time: Mon May 31 2021 22:04:07 +05:30. 6. Connected to: Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production. Version 18.4.0.0.0. 7. SQL> SHOW CON_NAME; 8. Output: CON_NAME, CDB\$ROOT. 9. SQL> ALTER SESSION SET CONTAINER = XEPDB1; 10. Output: Session altered. 11. SQL> SHOW CON_NAME; 12. Output: CON_NAME, XEPDB1. 13. SQL> (prompted).

```
SQL*Plus: Release 18.0.0.0.0 - Production on Mon May 31 22:06:58 2021
Version 18.4.0.0.0

Copyright (c) 1982, 2018, Oracle. All rights reserved.

Enter user-name: system
Enter password:
Last Successful login time: Mon May 31 2021 22:04:07 +05:30

Connected to:
Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production
Version 18.4.0.0.0

SQL> SHOW CON_NAME;

CON_NAME
-----
CDB$ROOT
SQL> ALTER SESSION SET CONTAINER = XEPDB1;

Session altered.

SQL> SHOW CON_NAME;

CON_NAME
-----
XEPDB1
SQL>
```

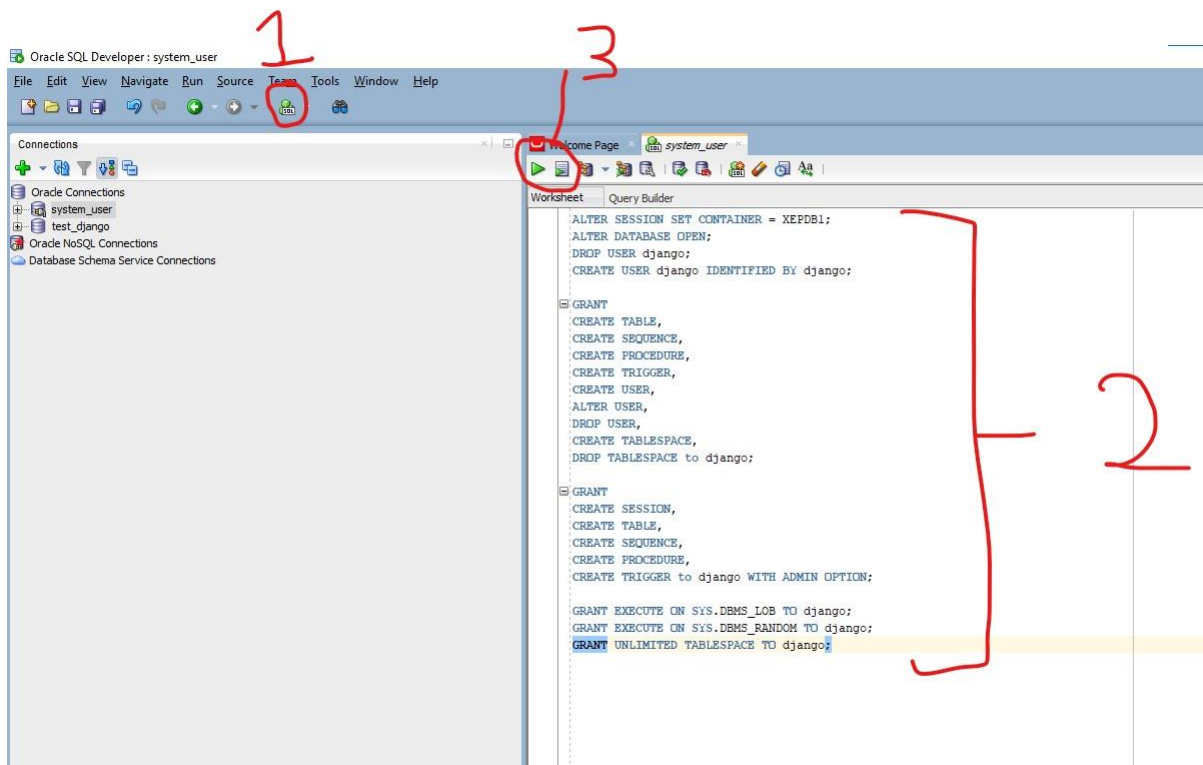
7. To create a new user name Django and grant privileges. Add the following SQL commands to the system user SQL worksheet, then run the scripts (F5).

```
ALTER SESSION SET CONTAINER = XEPDB1;
ALTER DATABASE OPEN;
DROP USER django;
CREATE USER django IDENTIFIED BY django;
```

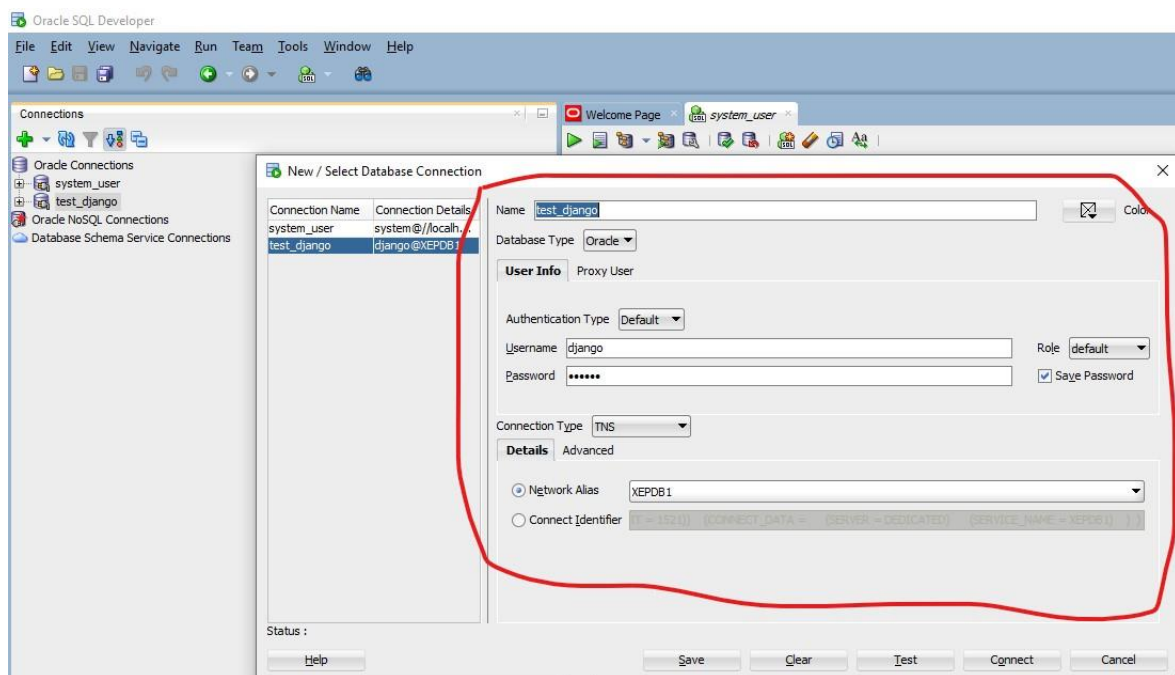
```
GRANT
CREATE TABLE,
CREATE SEQUENCE,
CREATE PROCEDURE,
CREATE TRIGGER,
CREATE USER,
ALTER USER,
DROP USER,
CREATE TABLESPACE,
DROP TABLESPACE TO django;
```

```
GRANT
CREATE SESSION,
CREATE TABLE,
CREATE SEQUENCE,
CREATE PROCEDURE,
CREATE TRIGGER TO django WITH ADMIN OPTION;
```

```
GRANT EXECUTE ON SYS.DBMS_LOB TO django;
GRANT EXECUTE ON SYS.DBMS_RANDOM TO django;
GRANT UNLIMITED TABLESPACE TO django;
```



8. Create a new connection for **django** [password = **django**], Click test and connect.



Note:

To connect to Django user [which is inside the pluggable database] in SQL Plus
You need to enter the username as <username>@<pluggable database name>

Ex:

Username = django@xepdb1

Password = django

9. Install cx_Oracle using **pip install cx_Oracle** inside your virtual environment (if any present)

For further information refer cx_oracle documentation page.

https://cx-oracle.readthedocs.io/en/latest/user_guide/installation.html

Django integration and migration:

1. To integrate the oracle database to your Django project, change the database section in the settings.py as the details furnished below

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.oracle',
        'NAME': 'localhost:1521/xepdb1',
        'USER': 'django',
        'PASSWORD': 'django',
        'HOST': '',
        'PORT': ''
    }
}
```

2. Run **migrate** command to integrate oracle database to your Django project.

Note:

- If you find an error saying table already exists in database, use
Python manage.py migrate --fake-initial

Creating models:

When creating a model, you need to add a table name using **CLASS META**.

```
from django.db import models

# Create your models here.

class product(models.Model):
    product_code = models.CharField(max_length= 5 , unique=True)
    name = models.CharField(max_length=200)
    quantity = models.IntegerField(default=1, null= True, blank= True)

    class Meta:
        db_table = 'products'
```

For more information refer Django documentation.

- <https://docs.djangoproject.com/en/3.2/ref/databases/#oracle-notes>

Commands to know:

- Python manage.py **makemigrations**
- Python manage.py **migrate**
- Python manage.py **migrate --fake-initial**

- To view the SQL command for the model created (after **makemigrations**):
Python manage.py **sqlmigrate <app name> <migration file number>**
eg: python manage.py sqlmigrate myapp 0001

- To view the existing table inside the database in **Model format**:
Python manage.py **inspectdb**

Uninstalling oracle:

Windows

In the past I've had many problems uninstalling all Oracle products from Windows systems. Here's my last resort method:

- Uninstall all Oracle components using the Oracle Universal Installer (OUI).
- Run regedit.exe and delete the HKEY_LOCAL_MACHINE/SOFTWARE/Oracle key. This contains registry entries for all Oracle products.
- If you are running 64-bit Windows, you should also delete the HKEY_LOCAL_MACHINE/SOFTWARE/Wow6432Node/Oracle key if it exists.
- Delete any references to Oracle services left behind in the following part of the registry (HKEY_LOCAL_MACHINE/SYSTEM/CurrentControlSet/Services/Ora*). It should be pretty obvious which ones relate to Oracle.
- Reboot your machine.
- Delete the "C:\Oracle" directory, or whatever directory is your ORACLE_BASE.
- Delete the "C:\Program Files\Oracle" directory.
- If you are running 64-bit Windows, you should also delete the "C:\Program Files (x86)\Oracle" directory.
- Remove any Oracle-related subdirectories from the "C:\ProgramData\Microsoft\Windows\Start Menu\Programs\" directory.
- Empty the contents of your "C:\temp" directory.
- Empty your recycle bin.

At this point your machine will be as clean of Oracle components as it can be without a complete OS reinstall.

Remember, manually editing your registry can be very destructive and force an OS reinstall so only do it as a last resort.

If some DLLs can't be deleted, try renaming them, then after a reboot delete them.

References:

- https://cx-oracle.readthedocs.io/en/latest/user_guide/installation.html
- <https://docs.djangoproject.com/en/3.2/ref/databases/#oracle-notes>
- <https://www.oracletutorial.com/getting-started/install-oracle/>
- http://www.srikanthtechnologies.com/blog/python/django_oracle_orm.aspx
- <https://www.youtube.com/watch?v=cci3PifT54U>
- <https://www.youtube.com/watch?v=s0Qmc9WwoEk>
- <https://developer.oracle.com/dsl/vasiliev-django.html>
- <https://oracle-base.com/articles/misc/manual-oracle-uninstall>