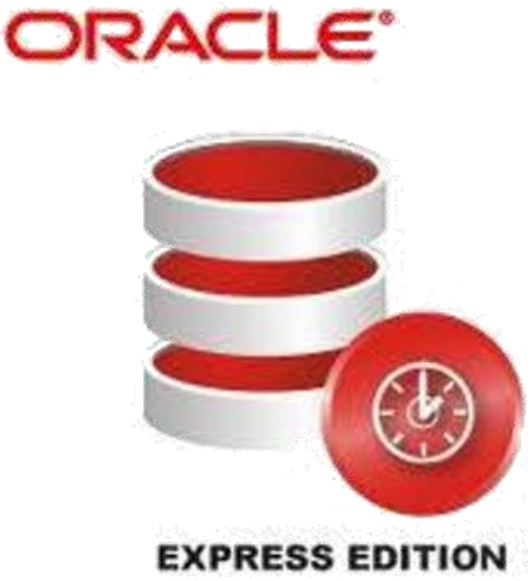
ORACLE Academy

academy.oracle.com

Configuring Oracle Database Express Edition.

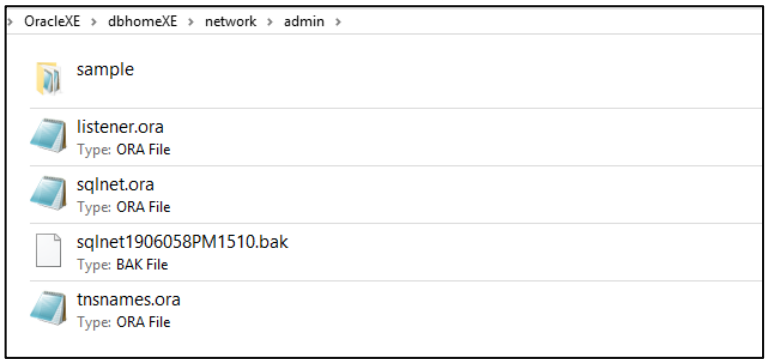


Configuring the Oracle Database Express Edition (XE).

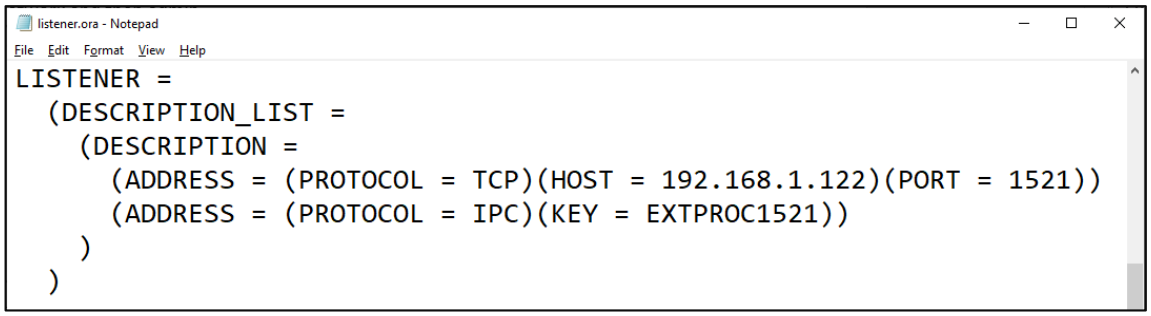
In this tutorial you will learn how to configure the Oracle Database Express Edition (XE) so that it can be used as the data source for a Java application using JDBC.

Configuring the Oracle Database Express Edition (XE) connection settings.

1. Open the directory where you installed the OracleXE database and go to dbhomeXE, network and then admin directory.

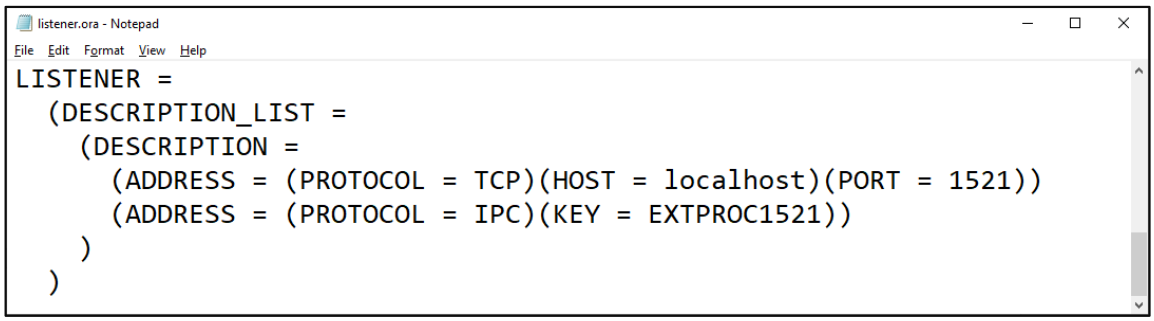


2. Open the listener.ora file in a text editor application.



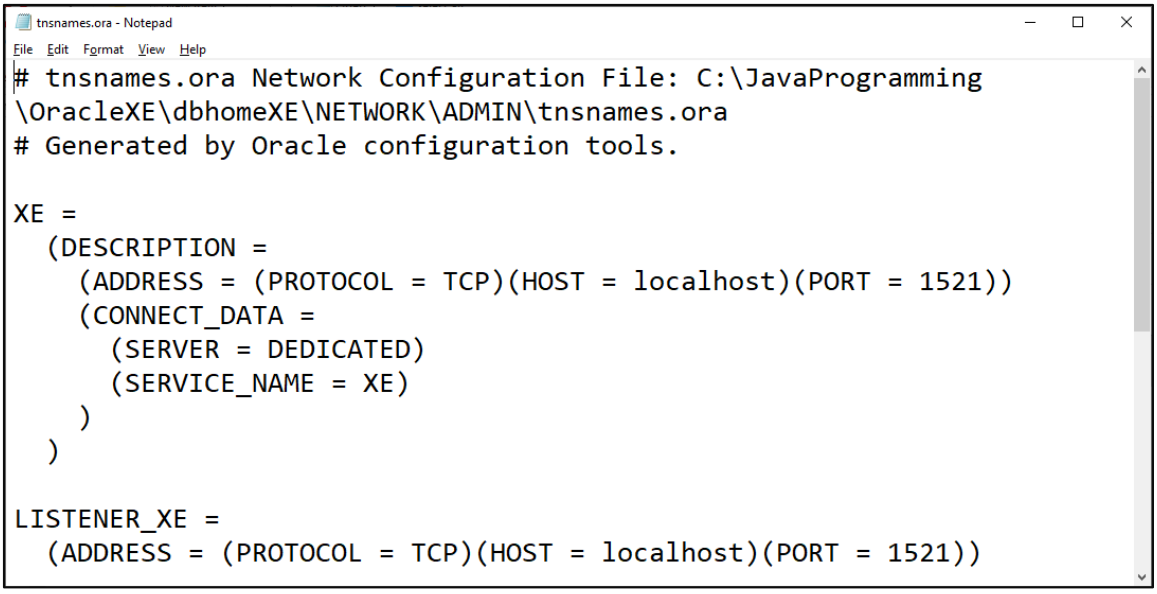
In this example the HOST has an IP address of 192.168.1.122, change the value to localhost, this will make it easier to connect to the database. If it is already localhost then do not make any changes.

The default PORT number is 1521 for Oracle databases, take note of the actual value.



Save and close the file.

3. Open the tnsnames.ora file and ensure that the HOST values are changed to localhost if required. There are two of them in this file that may need to be changed.



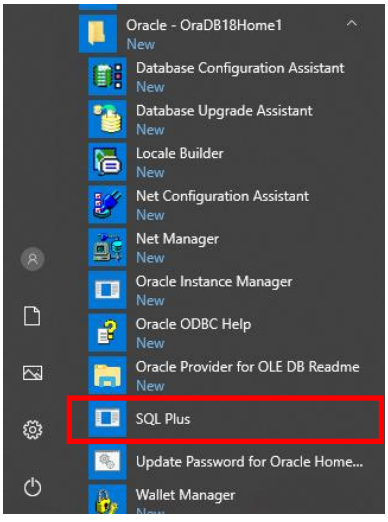
Save and close the file.

4. Restart your machine so that the database will implement the new configuration settings!

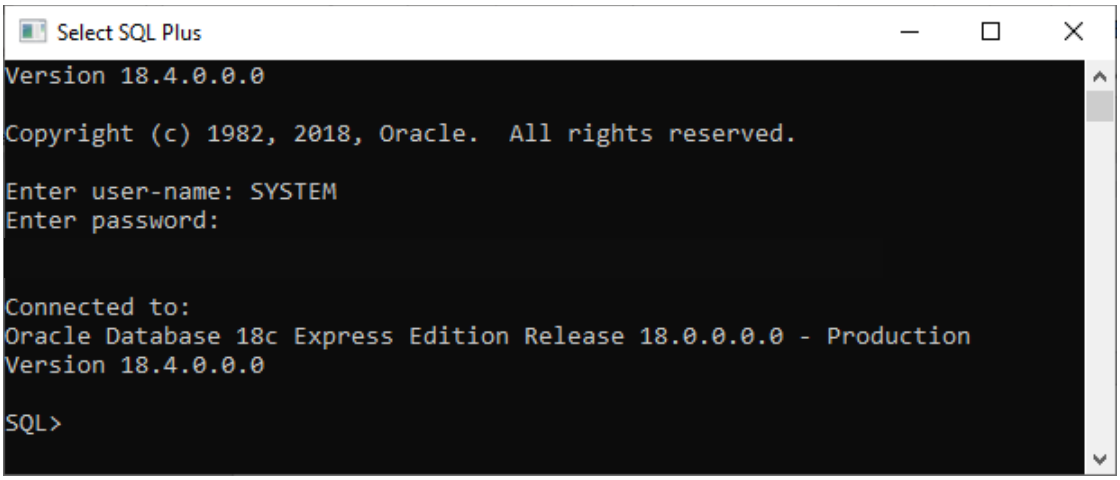
Accessing the Oracle Database Express Edition (XE).

It is not important that you fully understand the following commands, but you must enter them exactly as shown to avoid errors.

1. From the start menu select **Oracle – OraDB18Home1** and then select **SQL Plus** from the drop-down menu.



2. For the username type **SYSTEM** and provide **root** as the password.



This logs you in as a system user to the container database. The container database holds pluggable databases where you can create tables and store data. You cannot create database users at this level.

3. As you are working in the command line it can be useful to format the output. Enter the following three commands to format the width of the columns, the size of each line and how each page will be displayed in your output.

```
SQL> column column_name format a20;
SQL> set linesize 300;
SQL> set pagesize 0;
SQL>
```

4. To view the contents of the container database, enter the following command:

```
SQL> SELECT name, con_id
  2  FROM v$pdbs;
```

5. You will see that your container database is called PDB$SEED and the pluggable database that was created during installation is called XEPDB1. The values 2 and 3 are the id values of each database.

```
PDB$SEED                       2
XEPDB1                         3
```

6. As well as the database name you need to know the name of the service so that you can access the pluggable database. This is normally the lowercase version of the uppercase database name, but it is worth checking. You can use the id value to help you here!

```
SQL> SELECT name AS "Service Name"
  2  FROM v$active_services
  3  WHERE con_id = 3;
```

The name of the service will be displayed in the results of the database query.

```
xepdb1

SQL>
```

In this example the service name is the default, if you have anything else take a note of it.

7. To access the pluggable database, you have to enter the following statement to alter the current session.

```
SQL> ALTER SESSION SET container = xepdb1;

Session altered.

SQL>
```

Creating a user in Oracle Database Express Edition (XE).

1. To create a database user that can be used to access the data held in the tables run the following statement that allows the creation of a username and uses the IDENTIFIED BY statement to set a password.

```
SQL> create user orcluser identified by jdbcuser;

User created.

SQL>
```

When you create a user in databases the account has no access privileges, they have to be defined by the database administrator. You can be very specific about assigning database privileges ensuring that users do not have any more access than they should have. For the benefit of this exercise this user will be granted all privileges.

2. Use the following statement to grant the orcluser account all privileges on the XEPDB1 pluggable database.

```
SQL> GRANT ALL PRIVILEGES TO orcluser;

Grant succeeded.

SQL>
```

3. To test the new account, you need to disconnect (log out) of the SYSTEM account.

```
SQL> DISCONNECT
Disconnected from Oracle Database 18c Express Edition Release 18.0.0.0.0
Version 18.4.0.0.0
SQL>
```

4. Use the conn statement to connect (log in) to the pluggable database.

```
SQL> conn orcluser/jdbcuser@//localhost:1521/xepdb1
Connected.
SQL>
```

conn – connect

orcluser/jdbcuser – username and password to be used to log in.

@ - identifies the following information is the connection values required.

//localhost – connect to the local machine.

:1521 – port number to use for the connection.

/xepdb1 – service name to use to identify the pluggable database.

Creating tables and data by running a script in Oracle Database Express Edition (XE).

1. Before you continue in SQL Plus download the **Oracle_Schema_HR.sql** file and store it locally on your machine. For this exercise the file was stored in a directory called **sql** on the **C drive**.

2. To run a SQL script (a file that contains a series of SQL statements) in SQL PLUS use the START statement followed by the path to the file needed.

```
SQL> START C:\sql\Oracle_Schema_HR.sql;
```

Don't worry if you see any errors the script is written in a way that it may cause some.

3. You have created a Human Resource (HR) database that contains the following tables:

   - job_grades;
   - job_history;
   - employees;
   - jobs;
   - departments;
   - locations;
   - countries;
   - regions;

   You can view the contents of the tables by running a SELECT query using the following format:

   ```
   SELECT *
   FROM tablename;
   ```

```
SQL> SELECT *
  2  FROM job_grades;
A          1000         2999
B          3000         5999
C          6000         9999
D         10000        14999
E         15000        24999
F         25000        40000

6 rows selected.

SQL>
```

   TASK: Run select queries for each of the tables in the pluggable database.

4. To close down SQL Plus use the EXIT command.

```
SQL> EXIT;
```