Tutorial:

Installing and Creating an Oracle Database 19c on Linux 7

By Ahmed Baraka

Tutorial Overview

In this tutorial, you will install and create an Oracle database 19c on Linux 7 on a virtual machine. It will be a CDB database, without Oracle Restart services.

In high level, you will perform the following:

- Change the Settings of the Appliance srv1

- Make the machine IP address static

- Configure Putty to connect to srv1

- Set up the environment variables for the OS account `oracle`

- Change the kernel parameter values to the recommended values

- Install more packages

- Install Oracle database software and create the sample database

Required Resources

- A PC with a **free 8GB** in its memory. This means the RAM memory in your PC should be **at least 12GB**.

- At least 30GB free disk space.

- The PC is connected to the Internet

Database Specifications

The tutorial aims at creating a database with the following high level specifications:

SID	oradb
Home	/u01/app/oracle/product/19.0.0/db_1
Release	19c
Version	19.3
CDB/non-CDB	CDB
OS	Linux 7.8
Hostname	srv1
Storage Option	file system

Required Software and Packages

To implement this tutorial, you must have the following:

- **Oracle VirtualBox**, version 6. This tutorial was implemented on VirtualBox 6.0.22 for Windows. It can be obtained from the following link.

- **Oracle Virtualbox appliance** with a fresh installation of Oracle Linux 7.x. You can download a pre-built one with Oracle Linux 7.8 from here.
  Alternatively, you can create one from scratch. The procedure to create an VM machine with Linux 7.x is explained in many articles in the Internet. Just Google it!

- **Oracle Database 19c** installation files for Linux x86-64. This can be downloaded from Oracle site. Search the Internet for "Oracle Database 19c installation files for Linux x86-64". At the time of this writing, its link is here. This tutorial was implemented using Oracle Database 19c (version 19.3) for Linux x86-64.

  Note: download the zip file, not the rpm file.

- **Putty**: which is a utility that provides a command line prompt to connect to a Linux server from Windows.

Tutorial Steps

### A. Making Modifications on the Settings of the Appliance `srv1`

In the following steps, you will prepare the appliance srv1 for the tutorial. The tutorial steps assume that you have the appliance opened in the VirtualBox window.

1. In VirtualBox Manager, open the "**Settings**" of `srv1`, click on "**Shared Folders**" link in the right-hand pane. Add shared folder by pressing "**plus**" icon. Then select path to the location of the oracle software installation folder, and mark the checkbox "**Auto-mount**". You can change the "**Folder Name**", if you want to.

    This folder will be used to easily exchange files between the hosting PC and Linux in the VM machine. In the rest of this tutorial document, this folder will be referred to as the **staging folder**.
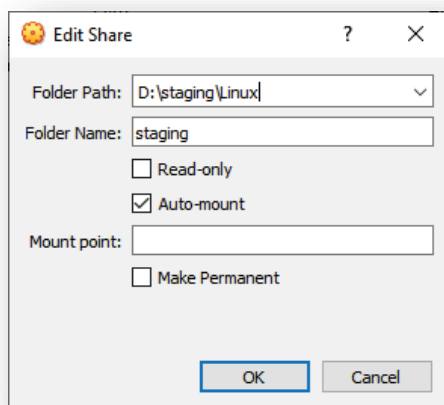
2. Make sure the Network adapter type **Bridged Adapter** and its name is the same as the network card of your PC. This makes your VM appliance appears in your network as a separate host and will be assigned an IP address based on your network configuration.
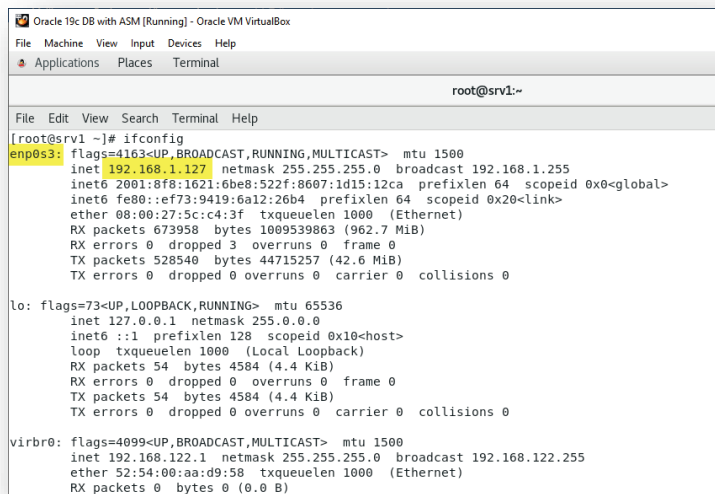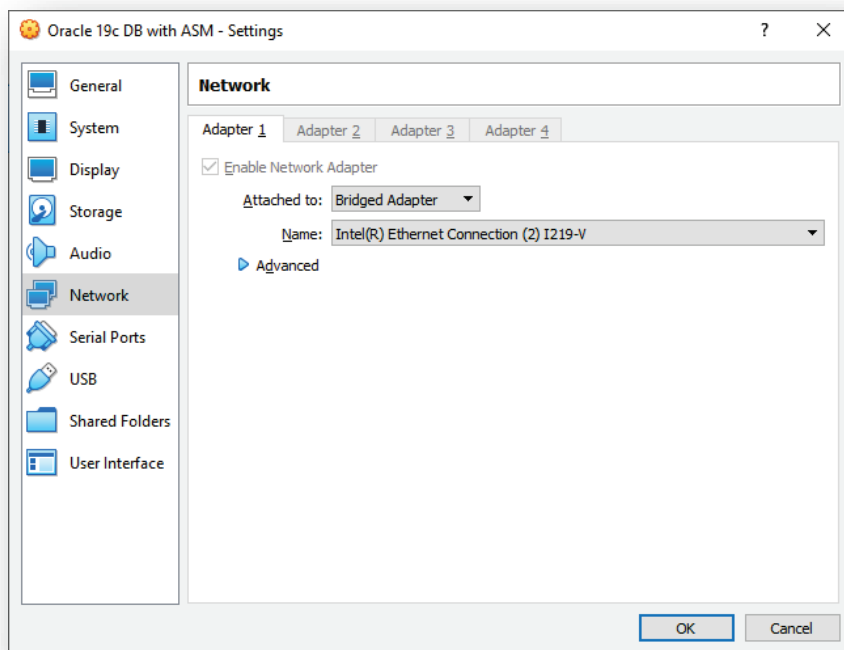
3. Optionally, set a description for the appliance and change its name to "**Oracle 19c DB**"
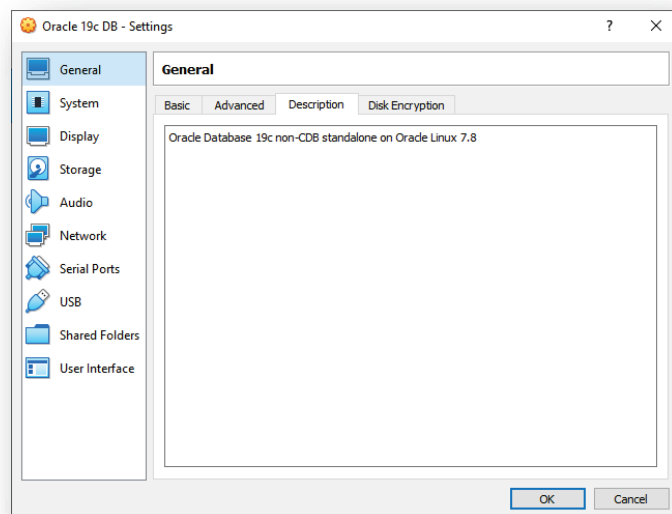
B. Making the IP Address Static

In the following steps, you will make the IP address assigned to `srv1` static. We need to make this step because we want to make sure that the machine will always have the same IP address when it is rebooted.

4. Start `srv1`

5. Login to the VirtualBox window of `srv1` as `root`

6. Open a terminal window, issue `ifconfig` command, and obtain the current IP address assigned to the machine. It is the IP address assigned to the NIC `enp0s3`

```
Oracle 19c DB with ASM [Running] - Oracle VM VirtualBox
File   Machine   View   Input   Devices   Help
 Applications    Places     Terminal

                                   root@srv1:~

File   Edit   View   Search   Terminal   Help
[root@srv1 ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.127  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 2001:8f8:1621:6be8:522f:8607:1d15:12ca  prefixlen 64  scopeid 0x0<global>
        inet6 fe80::ef73:9419:6a12:26b4  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:5c:c4:3f  txqueuelen 1000  (Ethernet)
        RX packets 673958  bytes 1009539863 (962.7 MiB)
        RX errors 0  dropped 3  overruns 0  frame 0
        TX packets 528540  bytes 44715257 (42.6 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 54  bytes 4584 (4.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 54  bytes 4584 (4.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

virbr0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 192.168.122.1  netmask 255.255.255.0  broadcast 192.168.122.255
        ether 52:54:00:aa:d9:58  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
```
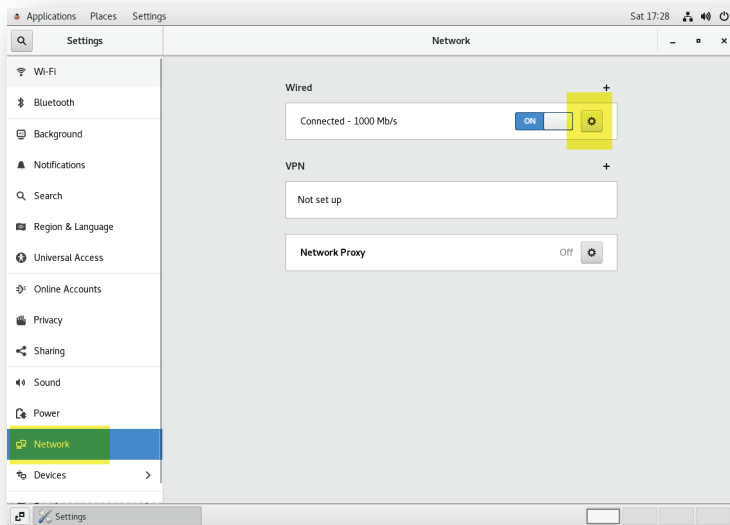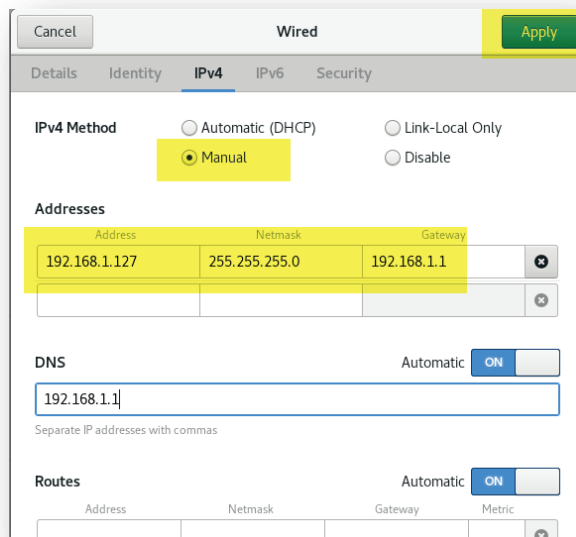
A tutorial edited by Ahmed Baraka

7. Open Settings window: **Applications** -> **System Tools** -> **Settings**
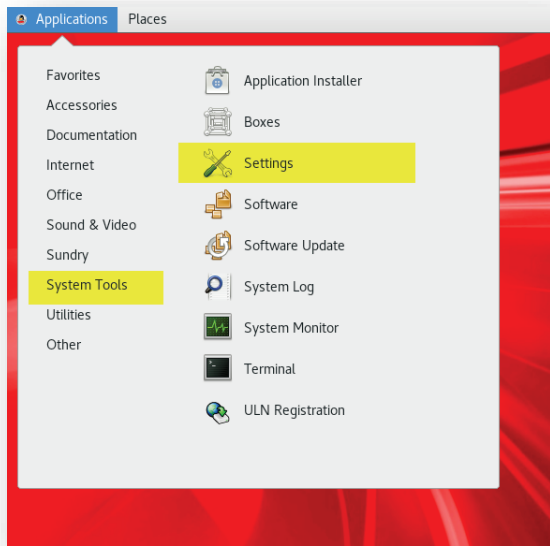
8. Open Network settings

9. Click on **IPv4** tab, select the **Manual** option, then enter the IP address and DNS information. Then click on **Apply** button.

10. In the Terminal window, ping the IP address to make sure that the changes are successful.

11. Edit the `/etc/hosts` file and add the hostname and the IP address to it.

    ```
    vi /etc/hosts
    ```
    ```
    192.168.1.127 srv1.localdomain srv1
    ```

12. Verify that the changes are registered in the NIC configuration file.

    ```
    cat /etc/sysconfig/network-scripts/ifcfg-enp0s3
    ```

13. Ping `srv1` to make sure the changes were successful.

    ```
    ping srv1
    ```

14. In the hosting PC, open a command line window and make sure you can ping the IP address of `srv1`.

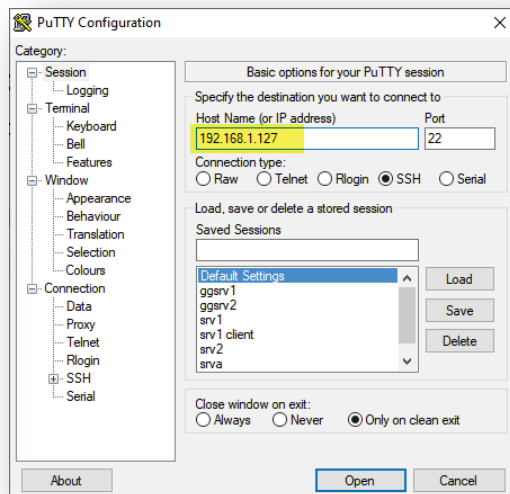   **Note**: Make sure the firewall in your PC allows communication with Oracle VirtualBox.

    ```
    C:\> ping 192.168.1.127
    ```
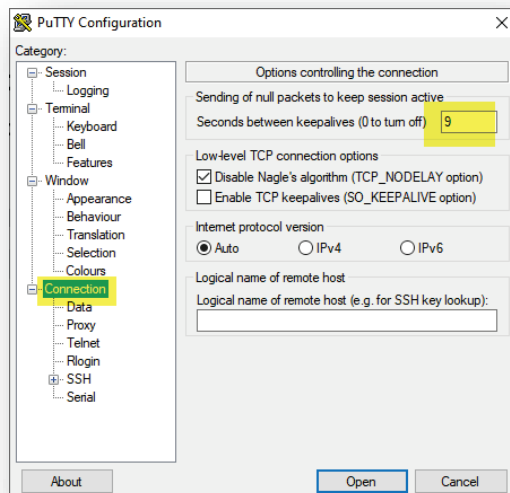
C. Configuring Putty

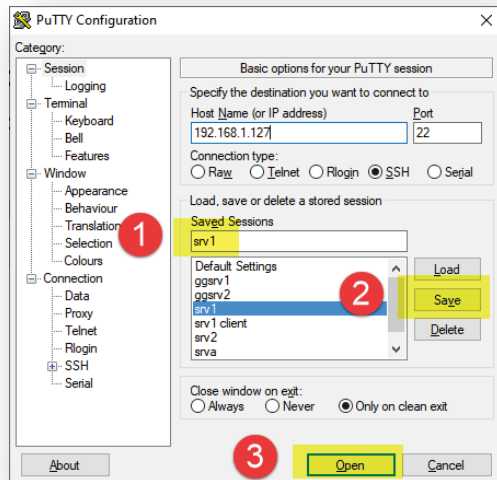In the following steps, you will configure Putty to connect to `srv1`

15. Open Putty then enter the IP address of srv1 in the **Host Name** field.

16. Click on Connection then set the "**Seconds between keepalives**" to 9.

17. Save the configuration as `srv1` then open the session.

18. Login as `root` in the Putty session.

D. Setting up Environment Variables for the Software Owner Account

In the following steps, you will configure the OS variables for the software owner account (`oracle`).

19. In the Putty session, make sure the current user is `root` then add `oracle` account to `vboxsf` group.

The `vboxsf` group was created by VirtualBox Guest Additions and it allows its members to access the shared folder (staging folder) in the hosting machine. In a production real life machine, you may not need to implement this step.

```
usermod -a -G vboxsf oracle
```

20. Switch the current user to `oracle` and make a backup copy of its bash profile file:

```
su - oracle
mv ~/.bash_profile ~/.bash_profile_bkp
```

21. Open the `.bash_profile` file with the vi editor

```
vi ~/.bash_profile
```

22. Add the following to it.

```
# .bash_profile

if [ -f ~/.bashrc ]; then
. ~/.bashrc
fi

ORACLE_BASE=/u01/app/oracle; export ORACLE_BASE
ORACLE_SID=oradb; export ORACLE_SID
ORACLE_HOME=$ORACLE_BASE/product/19.0.0/db_1; export ORACLE_HOME

NLS_DATE_FORMAT="DD-MON-YYYY HH24:MI:SS"; export NLS_DATE_FORMAT
TNS_ADMIN=$ORACLE_HOME/network/admin; export TNS_ADMIN

PATH=$PATH:$HOME/.local/bin:$HOME/bin
PATH=${PATH}:/usr/bin:/bin:/usr/local/bin
PATH=.:${PATH}:$ORACLE_HOME/bin
export PATH

LD_LIBRARY_PATH=$ORACLE_HOME/lib
LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:$ORACLE_HOME/oracm/lib
LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/lib:/usr/lib:/usr/local/lib
export LD_LIBRARY_PATH
CLASSPATH=$ORACLE_HOME/JRE
CLASSPATH=${CLASSPATH}:$ORACLE_HOME/jlib
CLASSPATH=${CLASSPATH}:$ORACLE_HOME/rdbms/jlib
CLASSPATH=${CLASSPATH}:$ORACLE_HOME/network/jlib
export CLASSPATH
export TEMP=/tmp
export TMPDIR=/tmp
umask 022
```

23. As `root`, run the following code to create the groups and directories required by Oracle software.

```
su –

groupadd oinstall
usermod -g oinstall oracle
mkdir -p /u01/app/oracle/product/19.0.0/db_1
mkdir -p /u01/app/oraInventory
chown -R oracle:oinstall /u01/app/oracle
chown -R oracle:oinstall /u01/app/oraInventory
```

E. Changing Kernel Parameter Values

In the following step, you will change the kernel parameter values to the values recommended by Oracle.

24. Make sure the current user is `root`

25. Create the following file then add the code that follows to it.

```
vi /etc/sysctl.d/97-oracle-database-sysctl.conf
```

```
fs.aio-max-nr = 1048576
fs.file-max = 6815744
kernel.shmall = 2097152
kernel.shmmax = 4294967295
kernel.shmmni = 4096
kernel.sem = 250 32000 100 128
net.ipv4.ip_local_port_range = 9000 65500
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
```

26. Change the current values of the kernel parameters:

```
/sbin/sysctl --system
```

27. Reboot `srv1`

F. Install More Packages

In the following steps, you will install further packages in `srv1` that are required by Oracle database software.

28. Open Putty and login to `srv1` as `root`

29. Run the following code to install further packages required by Oracle software.
```
yum install ksh
yum install libaio-devel.x86_64
```

G. Installing Oracle Database Software and Creating the Database

In the following steps, you will install Oracle database software in `srv1` and create the database.

30. Copy the Oracle database software installation file to the staging folder.

At the time of this writing, the installation file name downloaded from Oracle site is `LINUX.X64_193000_db_home.zip`

31. In the Putty session, change the current user to `oracle` then extract the installation file into the Oracle database software home directory

```
su - oracle
unzip /media/sf_staging/LINUX.X64_193000_db_home.zip -d $ORACLE_HOME >/dev/null
```

32. In the VirtualBox window of `srv1`, and login as `oracle`

33. Open a terminal window, change the current directory to the Oracle database home directory and run the `gridSetup.sh` script.

```
cd $ORACLE_HOME
./runInstaller
```

34. Respond to the Installer windows as follows:

Window	Action
Configuration Option	Select the following option: "**Create and Configure a single instance database.**"
System Class	Select the following option: "**Server Class**"
Database Edition	Select the following option: "**Enterprise Edition**"
Installation Location	Keep the default values
Create Inventory	oraInventory Group Name: **oracle**
Configuration Type	Select the following option: **General Purpose**
Database Identifiers	Global Database Name: **oradb.localdomain** Oracle SID: **oradb** Pluggable Database Name: **pdb1**

Configuration Options	Do **not** mark the AMM checkbox **Memory**: 5120 MB **Character set**: Use Unicode (AL32UTF8) **Sample Schemas**: (optional) **Mark the checkbox** "Install sample schema in the database"
Database Storage	Make sure "File System" option is selected
Management Options	Make sure the checkbox is **not** marked.
Recovery Option	Mark the checkbox **Enable Recovery** Make sure "File system" option is selected
Schema Password	Set passwords for the accounts
Operating System Groups	Select the "**oinstall**" group for all the options, except the OSOPER keep it blank.
Root Script Execution	**Mark the checkbox** "Automatically run configuration scripts" and enter the root password
Prerequisite Checks	All the Prerequisite Checks should pass.
Summary	Click on **Install** button
Install Product	When the installation reaches to nearly 12%, if will display a confirmation message. Click on **Yes** button.
Finish	click on **Close** button

35. After the installation and database creation are finished, verify the database is up and running by logging to it as sysdba

```
sqlplus / as sysdba
```

36. Check if a connection entry to `oradb` is added to the `tnsnames.ora` file

The `tnsnames.ora` file was created.

```
cat $TNS_ADMIN/tnsnames.ora
```

37. In the VirtualBox window of `srv1`, start the Firefox browser and open the EM Express using the following URL. Accept the warning displayed by the browser. Enter the `sys` username, its password, leave the container name blank then click on Login button.

https://srv1:5500/em

A tutorial edited by Ahmed Baraka

Note: You may consider creating and deploying a script that automatically starts up the database when the machine is rebooted.

The steps to perform this task are as follows:

1. Open Putty and login as root to srv1

2. Edit the `oratab` file

```
vi /etc/oratab
```

3. Change the last field for the database line to **Y**

```
oradb:/u01/app/oracle/product/19.0.0/db_1:Y
```

4. Create the file `/etc/init.d/dbora` and add the following code in it:

```
vi /etc/init.d/dbora
```

```
#! /bin/sh
# description: Oracle auto start-stop script.
ORA_HOME=/u01/app/oracle/product/19.0.0/db_1
ORA_OWNER=oracle

case "$1" in
'start')
    # Start the Oracle databases:
    # Remove "&" if you don't want startup as a background process.
    su - $ORA_OWNER -c "$ORA_HOME/bin/dbstart $ORA_HOME" &
    touch /var/lock/subsys/dbora
    ;;

'stop')
    # Stop the Oracle databases:
    su - $ORA_OWNER -c "$ORA_HOME/bin/dbshut $ORA_HOME" &
    rm -f /var/lock/subsys/dbora
    ;;
esac
```

5. Change the group of the `dbora` file to `oinstall`, and set its permissions to 750

```
chgrp oinstall /etc/init.d/dbora
chmod 750 /etc/init.d/dbora
```

6. Create symbolic links to the `dbora` script in the appropriate run-level script directories

```
ln -s /etc/init.d/dbora /etc/rc.d/rc0.d/K01dbora
ln -s /etc/init.d/dbora /etc/rc.d/rc3.d/S99dbora
ln -s /etc/init.d/dbora /etc/rc.d/rc5.d/S99dbora
```

7.   Restart `srv1` and wait for a few minutes to allow the database to automatically start up.

8.   Login as `oracle` to `srv1` and verify that the database has automatically started.

```
ps -ef | grep pmon
sqlplus / as sysdba
```

Summary

- In high level, the procedure to install and create an Oracle database 19c database (without Oracle Restart) goes through the following stages:

  o   Set up the environment variables for the `oracle` OS account

  o   Change the kernel parameter values to the recommended values

  o   Install the missing packages required by Oracle software

  o   Install Oracle database software and create the database

Note: The appliance created by me in this tutorial can be downloaded from the following link:
http://www.ahmedbaraka.com/public/download/