Oracle 12.2 Setup

WEEK 02

Database Administration

FEAA | SDBIS-SIA MASTER

Contents

Week 2 – Installing Oracle

This week we're going to install an Oracle database server on a VirtualBox virtual machine running a Linux operating system. Let's get started!

Getting Ready for the Playground Oracle Virtual Machine

Note: *If you're in the class, just skip this section since the environment is already prepared.*

First things first! You need a computer with a 64bit CPU architecture. Obviously, it must be running a 64bit operating system. It doesn't matter if it's Windows, OS X, Linux or Solaris. Then you need to download and install VirtualBox. Any version higher than 5.2.8 will cut it through.

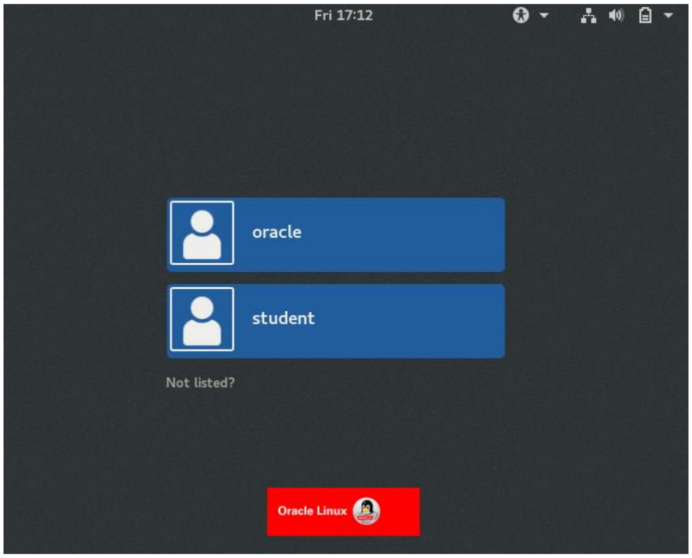
Next, download the virtual machine image from: https://www.dropbox.com/s/iy88t092bf95o18/abd-20190214.ova?dl=1. It's a ~14GB file, so you need a good Internet connection and a lot of patience. Also check if you have enough local disk space to accommodate this big file.

Import and Start Oracle VM

Now that we have the virtual machine downloaded, we need to import it. Open VirtualBox and go to "*File > Import Appliance…*". Choose the previously downloaded "*abd-20190214.ova*" file and press "*Import*". You'll get an estimation about the time needed to finish this operation in a progress bar. It depends on the computer performance you're working with.

Note: *If you're in the classroom, the file to be imported is already available on a location which will be provided during the class.*

After VM has finished importing, simply start it and wait for the login prompt.

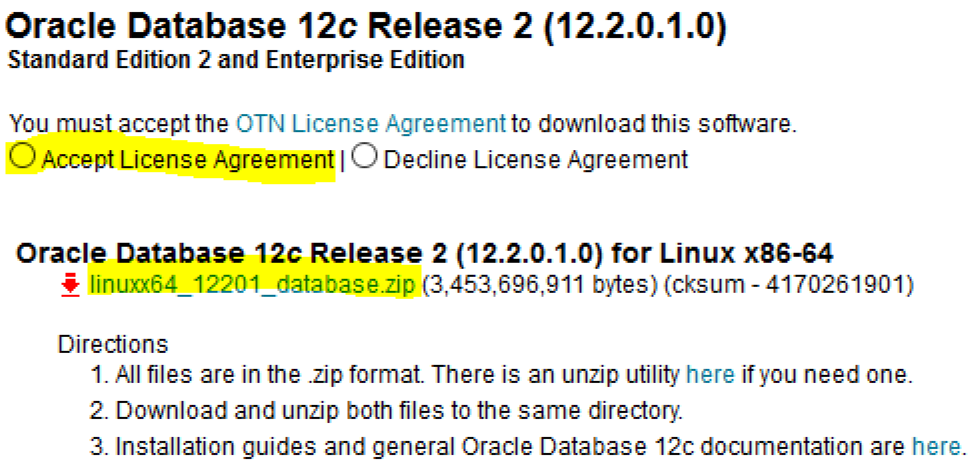


The accounts we'll use during classes are:

Account	Password
root	root!
oracle	oracle
student	student

Downloading the Oracle Software

We'll install an Oracle 12.2 database server and we need to download the corresponding software package from the Oracle official download area.



Oracle Database 12*c* Release 2 (12.2.0.1.0)
Standard Edition 2 and Enterprise Edition

You must accept the OTN License Agreement to download this software.
○ Accept License Agreement | ○ Decline License Agreement

Oracle Database 12*c* Release 2 (12.2.0.1.0) for Linux x86-64
linuxx64_12201_database.zip (3,453,696,911 bytes) (cksum - 4170261901)

Directions
1. All files are in the .zip format. There is an unzip utility here if you need one.
2. Download and unzip both files to the same directory.
3. Installation guides and general Oracle Database 12c documentation are here.

Tick the "*Accept License Agreement*" and then download "*linuxx64_12201_database.zip*" file. An Oracle free account is required to download the software, so you must create one.

Note: *Because downloading such a big file is time consuming this file was already provisioned inside the VM itself. You may find it in "/opt/kits" folder. So, if you're using the Oracle VM we provided, there's no need to download anything.*

Preparing the OS Environment

Login with "*oracle*" user and open a terminal. You should see the "Terminal" icon on the desktop. Go to *"/opt/kits"* folder and unzip the oracle software package.

```
cd /u01/kits
unzip -q linuxx64_12201_database.zip
```

You should see now a "*database*" folder under "*/opt/kits*".

Usually, Oracle is quite picky on fulfilling all OS prerequisites. Fortunately, the Oracle VM playground was already prepared, and all dependent packages were installed, as well as other related settings like "*sysctl*" or "*shell limits*". For those of you who want to install Oracle on your own Linux, it is mandatory to follow all the installation steps. You may find detailed instructions here. In fact, this link is already available as a bookmark in the playground VM.

Now, let's prepare the folders where Oracle software is going to be installed. As "root" user, run the following commands (to switch to "root", use the "su –" command):

```
mkdir -p /u01/app/oracle/product/12.2.0.1/db_1
chown -R oracle:oinstall /u01
chmod -R 775 /u01
```

What does it mean "775"? Why the "-p" argument of the "mkdir" command? If you don't know, how can you find more information about "mkdir -p" command?

The next step is to define some environment variables which must be defined first. On a RedHat OS they usually go to "*/home/oracle/.bash_profile*" file. However, to keep things clean and simple, just go to "/opt/oradba/dbs" and launch the "*setup_oracle_profile.sh*" script.

```
cd /opt/oradba/dbs
./setup_oracle_profile.sh
```

Now, we need to refresh the shell environment. Either close the terminal window and start a new one, or run the following command:

```
source ~/.bash_profile
```

Either way, using the following command you should see the ORACLE_HOME environment variable set:

```
env | grep ORACLE_HOME
ORACLE_HOME=/u01/app/oracle/product/12.2.0.1/db_1
```

If you don't see the value for the *ORACLE_HOME* variable, stop now! Review the previous steps and find out what went wrong.

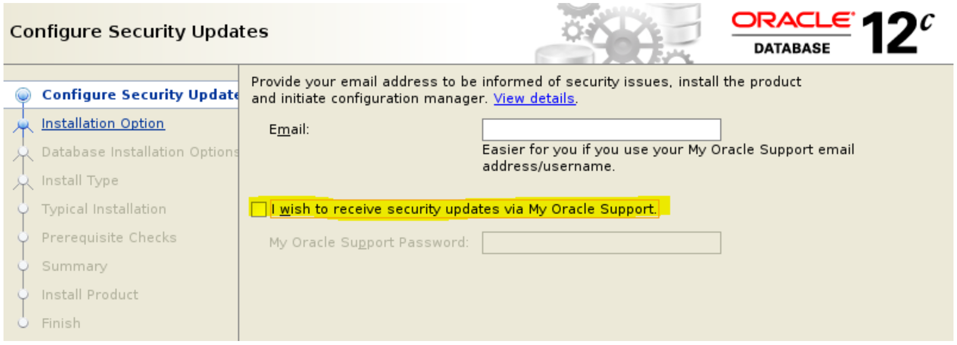
What is that "|" character between "env" and "grep"? And what's the idea of using "grep" anyway?

Install Oracle Database Server Software

Go to "/opt/kits/database" folder and launch the so-called Oracle Universal Installer. Don't be afraid, it's just a fancy name for their "uncool" installer tool:

```
cd /opt/kits/database
./runInstaller
```

You'll be welcomed by the first wizard page:



Provide your email address to be informed of security issues, install the product and initiate configuration manager. View details.

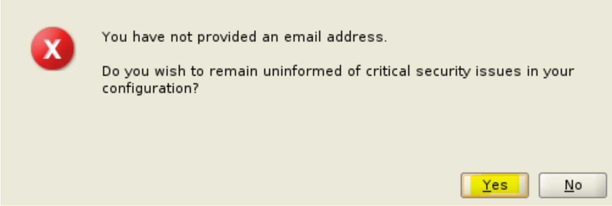
Email:

Easier for you if you use your My Oracle Support email address/username.

□ I wish to receive security updates via My Oracle Support.

My Oracle Support Password:

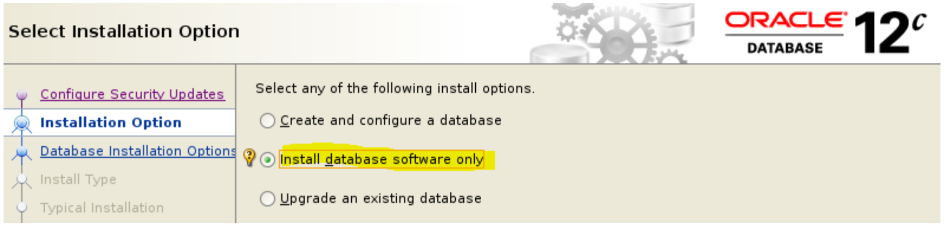
Untick the highlighted option above and choose next. The following warning can be safely ignored. It asks for our confirmation that indeed we are fool enough to remain uninformed of Oracle critical issues which may come up in the future.



You have not provided an email address.

Do you wish to remain uninformed of critical security issues in your configuration?

It's a test/playground environment so we can take this risk. Press "Yes" and the next wizard page is displayed:



Select Installation Option

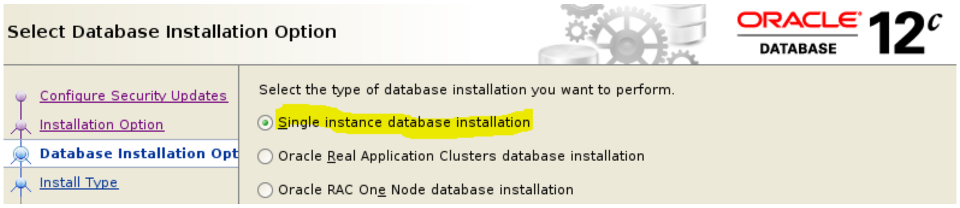
Select any of the following install options.

○ Create and configure a database

● Install database software only

○ Upgrade an existing database

We choose the second option. The first one would be also a good fit, but we'll go with the second option because we'd like to emphasize the difference between having the software installed and the database on top of it. You may have just the software installed and no database, but not the other way around. Also, you may have more than one database sharing the same Oracle software. The next screen is:



Select Database Installation Option

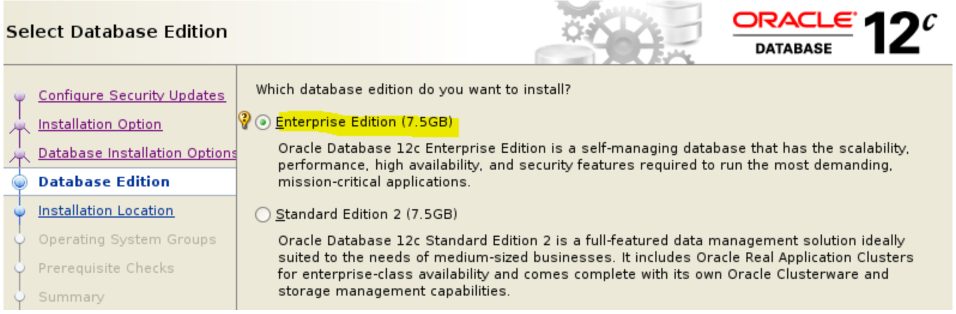
Select the type of database installation you want to perform.

● Single instance database installation

○ Oracle Real Application Clusters database installation

○ Oracle RAC One Node database installation

We let the default first option selected and choose next.



Select Database Edition

Which database edition do you want to install?

● Enterprise Edition (7.5GB)

Oracle Database 12c Enterprise Edition is a self-managing database that has the scalability, performance, high availability, and security features required to run the most demanding, mission-critical applications.

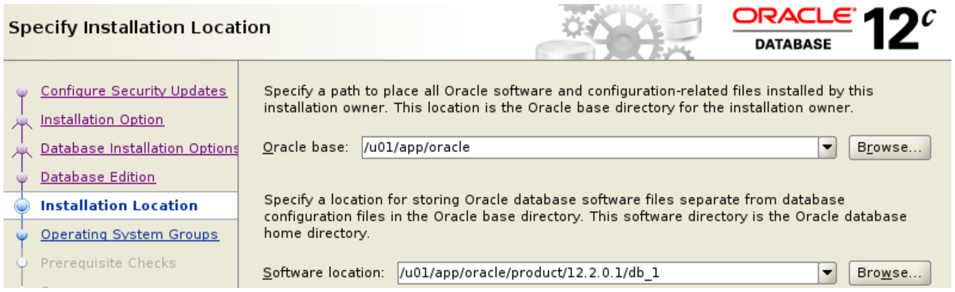
○ Standard Edition 2 (7.5GB)

Oracle Database 12c Standard Edition 2 is a full-featured data management solution ideally suited to the needs of medium-sized businesses. It includes Oracle Real Application Clusters for enterprise-class availability and comes complete with its own Oracle Clusterware and storage management capabilities.

Again, we let the default option which is "Enterprise Edition" and we move forward.

What is the difference between Oracle Enterprise Edition and Standard Edition? What would you choose for your company?

The next setup page is where you specify the ORACLE_BASE and the ORACLE_HOME locations. ORACLE_BASE is the common location for all Oracle related software (not necessarily Oracle database server), whereas ORACLE_HOME points to the folder underneath ORACLE_BASE where, in this case, the database server software is going to be installed.



Specify Installation Location

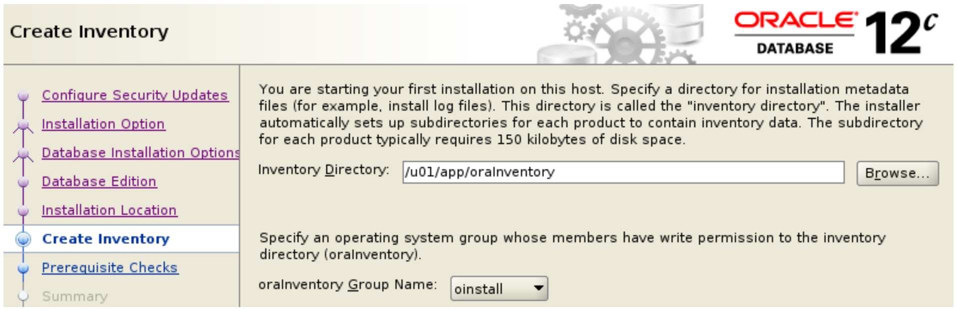
Specify a path to place all Oracle software and configuration-related files installed by this installation owner. This location is the Oracle base directory for the installation owner.

Oracle base: /u01/app/oracle

Specify a location for storing Oracle database software files separate from database configuration files in the Oracle base directory. This software directory is the Oracle database home directory.

Software location: /u01/app/oracle/product/12.2.0.1/db_1

The next screen is the following:



Create Inventory

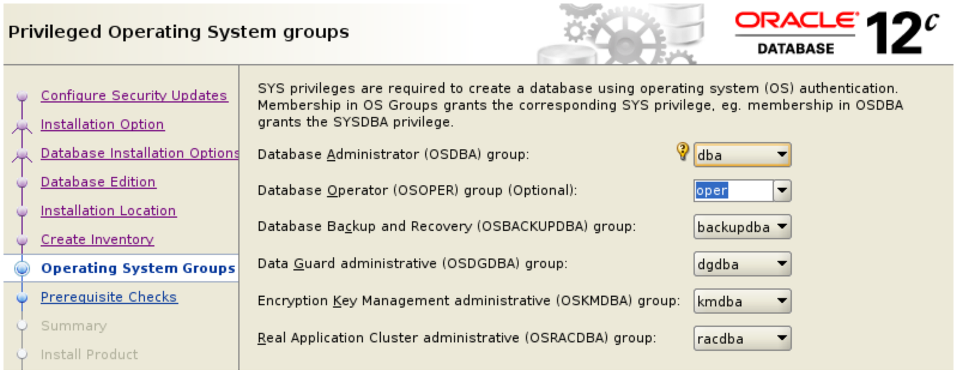
You are starting your first installation on this host. Specify a directory for installation metadata files (for example, install log files). This directory is called the "inventory directory". The installer automatically sets up subdirectories for each product to contain inventory data. The subdirectory for each product typically requires 150 kilobytes of disk space.

Inventory Directory: /u01/app/oraInventory

Specify an operating system group whose members have write permission to the inventory directory (oraInventory).

oraInventory Group Name: oinstall

Here, Oracle is asking for the inventory location. This is just a folder where Oracle keeps track of all software packages it installs. On Windows this information is usually kept in Windows registry, but on Linux (and on other OS platforms) this facility can’t be used. Just leave the default and move forward.



Privileged Operating System groups

SYS privileges are required to create a database using operating system (OS) authentication. Membership in OS Groups grants the corresponding SYS privilege, eg. membership in OSDBA grants the SYSDBA privilege.

Database Administrator (OSDBA) group: dba

Database Operator (OSOPER) group (Optional): oper

Database Backup and Recovery (OSBACKUPDBA) group: backupdba

Data Guard administrative (OSDGDBA) group: dgdba

Encryption Key Management administrative (OSKMDBA) group: kmdba

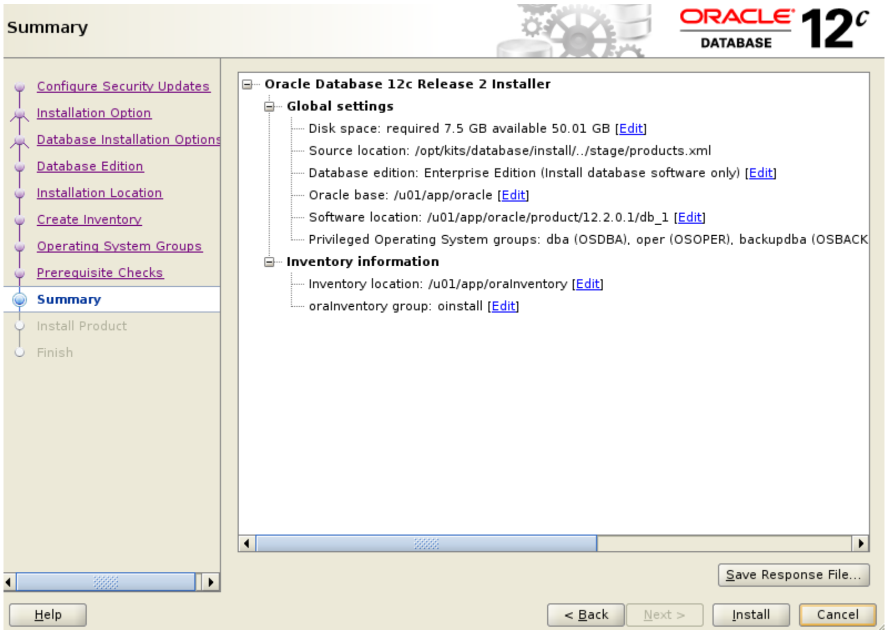
Real Application Cluster administrative (OSRACDBA) group: racdba

The screen above asks for a mapping between an OS group and the administrative tasks allowed to the users belonging to those groups. For example, if an OS user is part of the “*dba*” OS group, then it will implicitly be a DBA database administrator with all the corresponding rights. We can have a look at the groups “oracle” user belongs to:

```
id oracle
uid=54321(oracle) gid=54321(oinstall)
groups=54321(oinstall),54322(dba),54323(oper),54324(backupdba),54325(dgdb
a),54326(kmdba),54330(racdba)
```

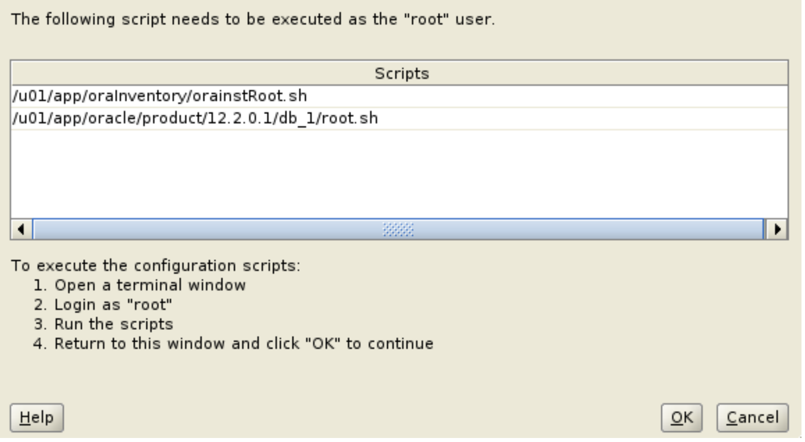
We can see that "oracle" user is part of all the groups shown in the setup screen, which means it can do all possible Oracle administrative tasks.

The last screen before starting the setup process is the summary below:



Oracle Database 12c Release 2 Installer
- Global settings
  - Disk space: required 7.5 GB available 50.01 GB [Edit]
  - Source location: /opt/kits/database/install/../stage/products.xml
  - Database edition: Enterprise Edition (Install database software only) [Edit]
  - Oracle base: /u01/app/oracle [Edit]
  - Software location: /u01/app/oracle/product/12.2.0.1/db_1 [Edit]
  - Privileged Operating System groups: dba (OSDBA), oper (OSOPER), backupdba (OSBACK
- Inventory information
  - Inventory location: /u01/app/oraInventory [Edit]
  - oraInventory group: oinstall [Edit]

"*Save Response File…*" is a feature we're not going to use, but for your information it allows to save all previous options so that to allow you to reuse them on subsequent installations. It is very helpful for automating the setup. For now, we'll just go with the "Install" button. You should see now some progress information. At some point you'll be prompted to run some scripts:



The following script needs to be executed as the "root" user.

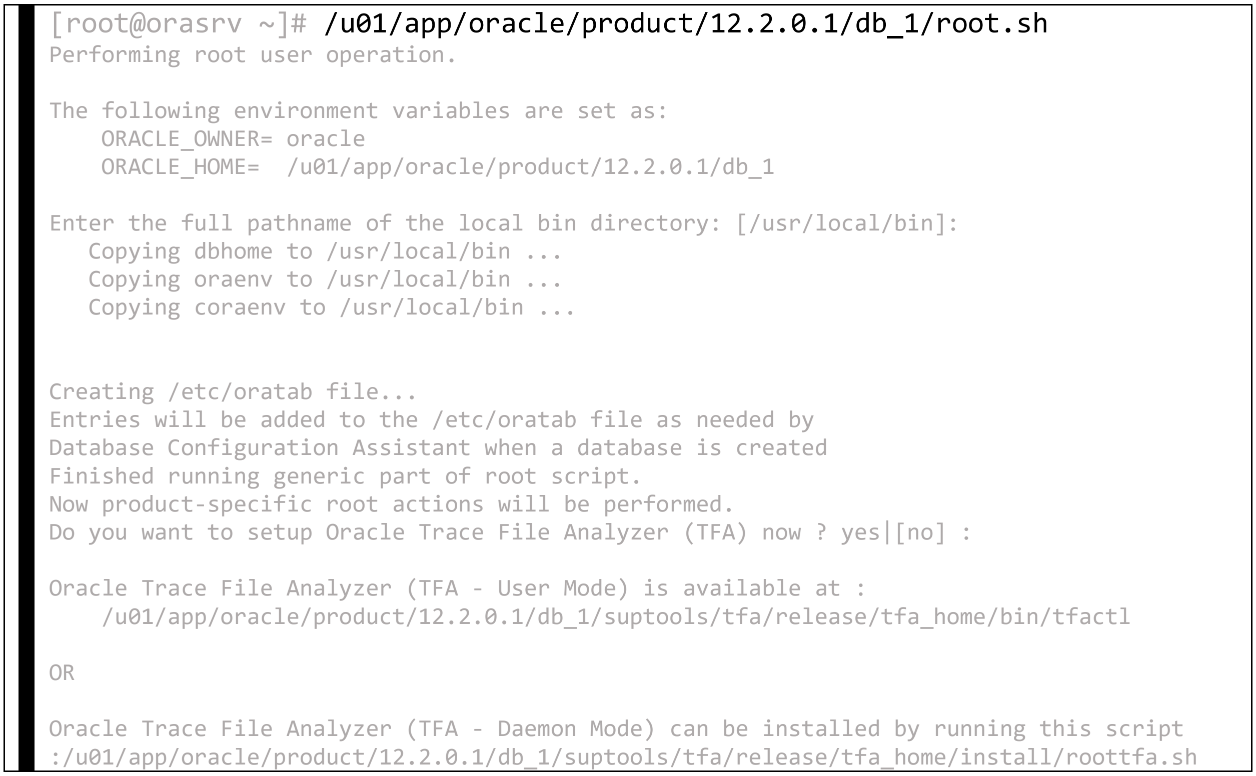
Scripts
/u01/app/oraInventory/orainstRoot.sh
/u01/app/oracle/product/12.2.0.1/db_1/root.sh

To execute the configuration scripts:
1. Open a terminal window
2. Login as "root"
3. Run the scripts
4. Return to this window and click "OK" to continue

Leave the above window open and run the specified scripts using "root" user. The first script would be:

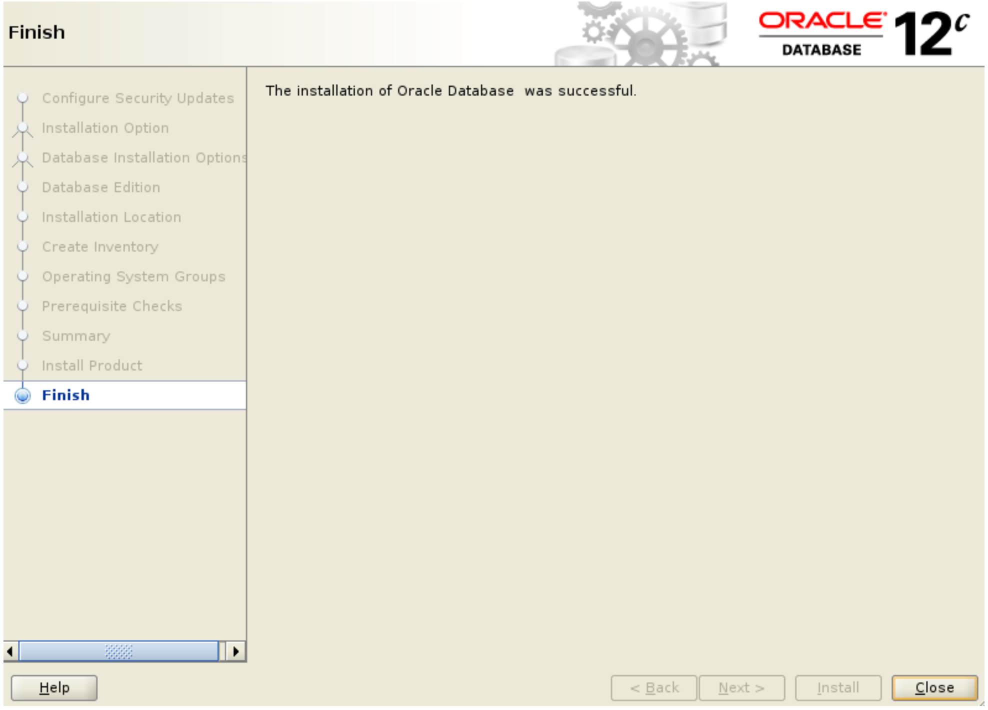
```
[root@orasrv ~]# /u01/app/oraInventory/orainstRoot.sh
Changing permissions of /u01/app/oraInventory.
Adding read,write permissions for group.
Removing read,write,execute permissions for world.

Changing groupname of /u01/app/oraInventory to oinstall.
The execution of the script is complete.
```

The second script asks for some input data from the user. Just hit ENTER to accept all defaults.



```
[root@orasrv ~]# /u01/app/oracle/product/12.2.0.1/db_1/root.sh
Performing root user operation.

The following environment variables are set as:
    ORACLE_OWNER= oracle
    ORACLE_HOME=  /u01/app/oracle/product/12.2.0.1/db_1

Enter the full pathname of the local bin directory: [/usr/local/bin]:
   Copying dbhome to /usr/local/bin ...
   Copying oraenv to /usr/local/bin ...
   Copying coraenv to /usr/local/bin ...


Creating /etc/oratab file...
Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.
Now product-specific root actions will be performed.
Do you want to setup Oracle Trace File Analyzer (TFA) now ? yes|[no] :

Oracle Trace File Analyzer (TFA - User Mode) is available at :
    /u01/app/oracle/product/12.2.0.1/db_1/suptools/tfa/release/tfa_home/bin/tfactl

OR

Oracle Trace File Analyzer (TFA - Daemon Mode) can be installed by running this script
:/u01/app/oracle/product/12.2.0.1/db_1/suptools/tfa/release/tfa_home/install/roottfa.sh
```

After running those two scripts, come back to the "Execute Configuration Scripts" window and press "OK". The final screen should be:



The installation of Oracle Database was successful.

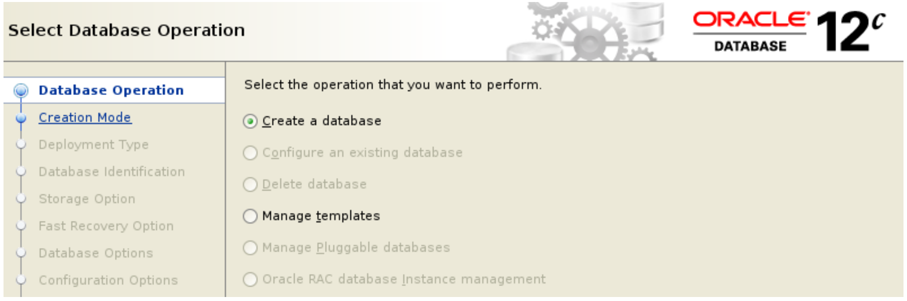
Hit “Close”! Congratulations! You have successfully installed the Oracle database server software.

Creating an Oracle Database

Ok, we have the Oracle software in place, but we haven’t installed a database yet. To get one up and running we can use a few tools, but the most common one is to use the “*Database Configuration Assistant*”. The short name of the command launching this tool is “*dbca*”. So, as “oracle” OS user, fire up a new terminal and type “dbca”, as shown below (of course, with ENTER at the end):

```
dbca
```

The GUI (Graphical User Interface) of this tool is very similar with the Oracle Universal Installer in the sense that a sequence of wizard pages is shown. The first page is the one below:

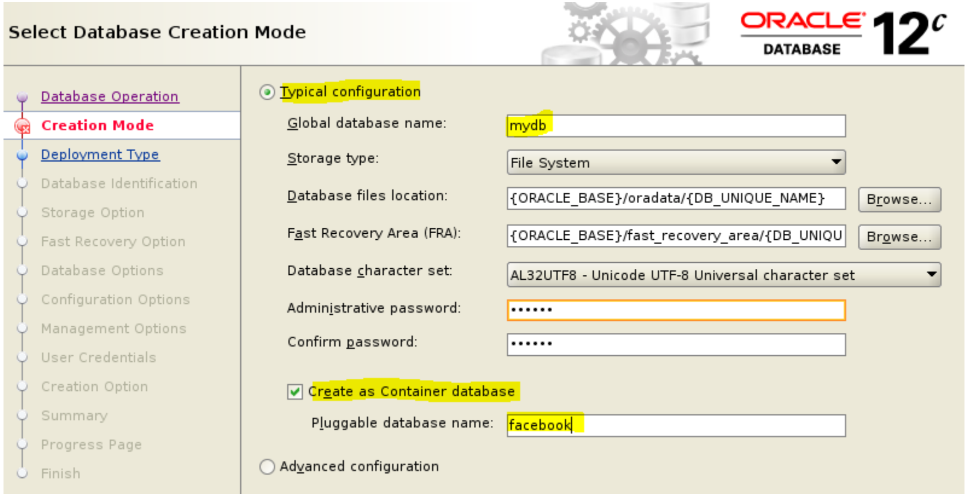


Select Database Operation

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

By looking at this page we can figure out that this tool is not only for creating new databases but also for configuring and delete them. However, we’re now interested in creating a new database therefore we let the first option checked and move forward to the next page.



Select Database Creation Mode

● Typical configuration

Global database name: mydb

Storage type: File System

Database files location: {ORACLE_BASE}/oradata/{DB_UNIQUE_NAME}

Fast Recovery Area (FRA): {ORACLE_BASE}/fast_recovery_area/{DB_UNIQU

Database character set: AL32UTF8 - Unicode UTF-8 Universal character set

Administrative password: ••••••

Confirm password: ••••••

☑ Create as Container database

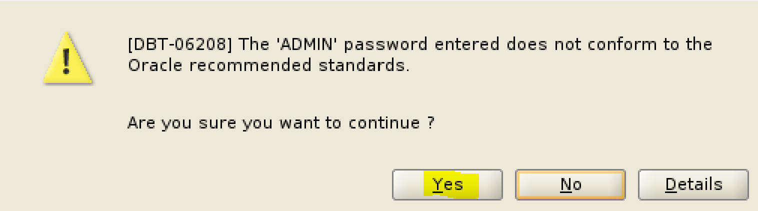
Pluggable database name: facebook

○ Advanced configuration

We go with a “*Typical configuration*” where we let the system to configure our database with the recommended options. We provide the “*Global database name*” as “*mydb*”, and “*oracle*” as the “*Administrative password*”. We also choose to create the database as a container. This is a new feature in Oracle 12c version. Basically, we can tell Oracle that under the same server backend we want to manage a special kind of databases called “pluggable databases”. In the screen above we want to create a pluggable database called “*facebook*”. Later on, we can easily add more such databases. Also, it is important to decide now the “*Database character set*” as it is difficult to change it after the database is created. The default “UTF-8” character set is the recommended one because it can accommodate any symbol (a.k.a glyph). For now, just leave the “*Fast Recovery Area (FRA)*” field as it is. We’ll cover this concept when we reach the “Backup & Recovery” topic.

> *Look at the “Database files location” field and try to guess where the database files are going to be created.*

Press the “Next” button and a warning is displayed.

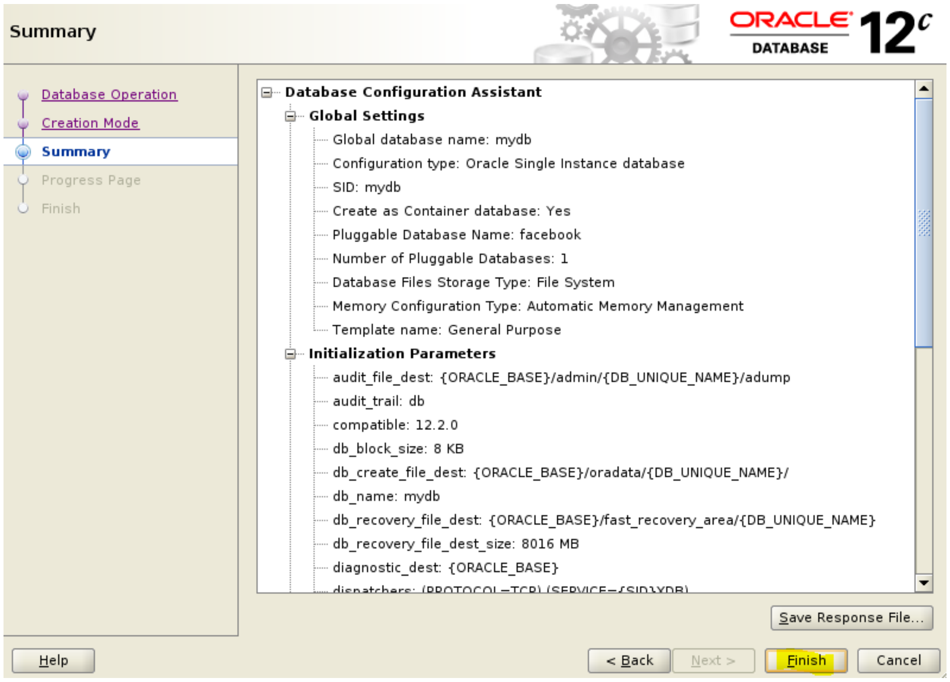


[DBT-06208] The 'ADMIN' password entered does not conform to the Oracle recommended standards.

Are you sure you want to continue ?

The reason is that we provided a very simple and easy to guess password and Oracle wants us to confirm that we are aware of the risk. Of course, we’re going to use more complex passwords for productive environments, but for this playground database we won’t bother. So, just confirm with “Yes”.

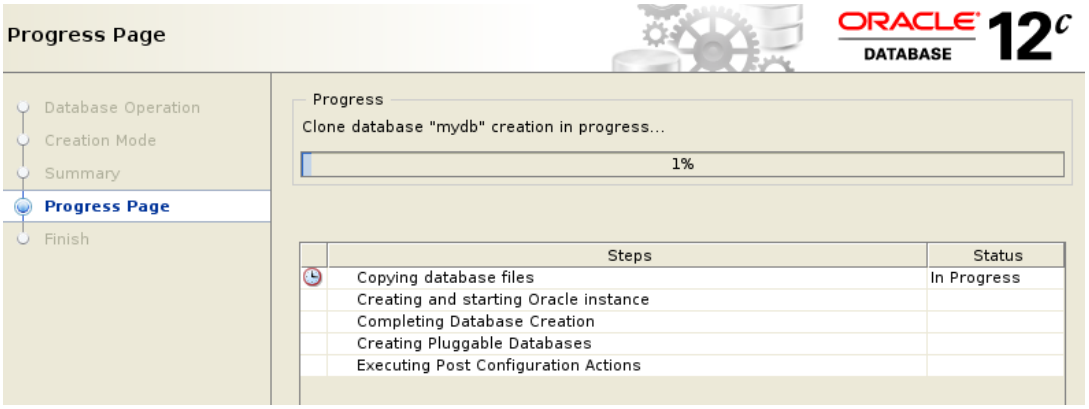
The next screen is a summary for the database which is about to be created:



```
Database Configuration Assistant
  Global Settings
    Global database name: mydb
    Configuration type: Oracle Single Instance database
    SID: mydb
    Create as Container database: Yes
    Pluggable Database Name: facebook
    Number of Pluggable Databases: 1
    Database Files Storage Type: File System
    Memory Configuration Type: Automatic Memory Management
    Template name: General Purpose
  Initialization Parameters
    audit_file_dest: {ORACLE_BASE}/admin/{DB_UNIQUE_NAME}/adump
    audit_trail: db
    compatible: 12.2.0
    db_block_size: 8 KB
    db_create_file_dest: {ORACLE_BASE}/oradata/{DB_UNIQUE_NAME}/
    db_name: mydb
    db_recovery_file_dest: {ORACLE_BASE}/fast_recovery_area/{DB_UNIQUE_NAME}
    db_recovery_file_dest_size: 8016 MB
    diagnostic_dest: {ORACLE_BASE}
    dispatchers: (PROTOCOL=TCP) (SERVICE={SID}XDB)
```

What do you think "Save Response File..." is for? When do you think such a feature is useful?

Press "Finish" and wait for the database setup to finish. You get progress information in a screen like the one shown below:

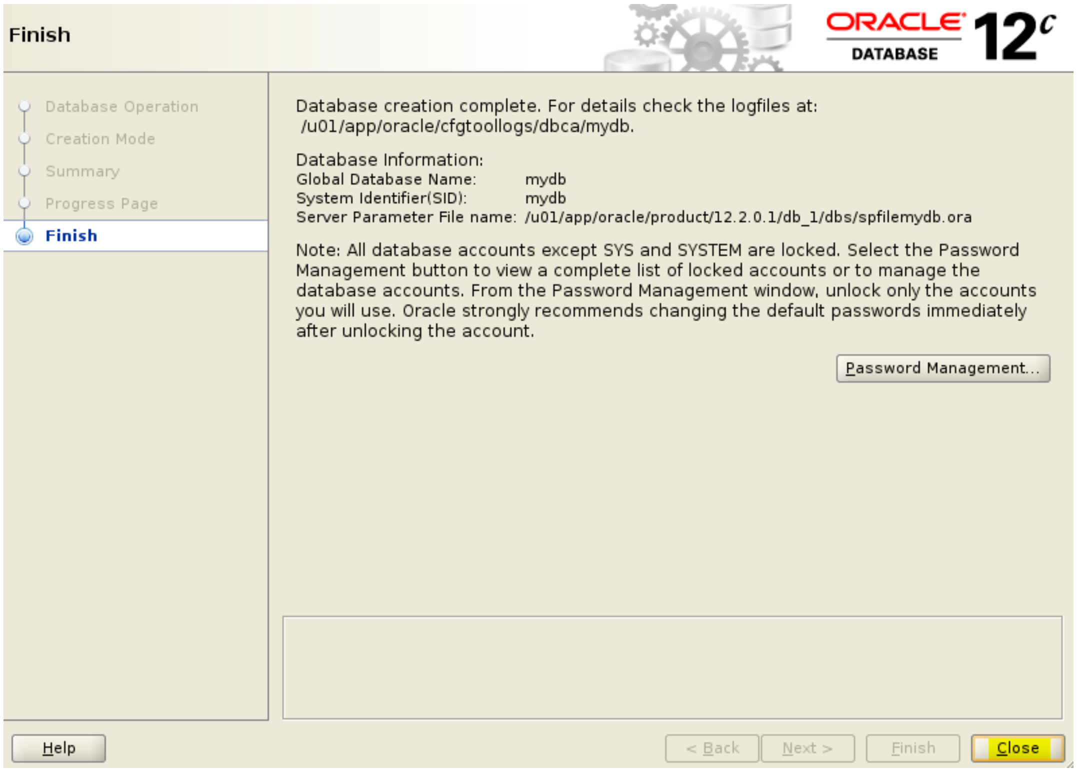


Clone database "mydb" creation in progress...

1%

Steps	Status
Copying database files	In Progress
Creating and starting Oracle instance	
Completing Database Creation	
Creating Pluggable Databases	
Executing Post Configuration Actions	

In Oracle, creating a new container or a non-container database is a time-consuming process, so be patient. If all goes well, you should see the final screen of the database setup:



Database creation complete. For details check the logfiles at:
/u01/app/oracle/cfgtoollogs/dbca/mydb.

Database Information:
Global Database Name: mydb
System Identifier(SID): mydb
Server Parameter File name: /u01/app/oracle/product/12.2.0.1/db_1/dbs/spfilemydb.ora

Note: All database accounts except SYS and SYSTEM are locked. Select the Password Management button to view a complete list of locked accounts or to manage the database accounts. From the Password Management window, unlock only the accounts you will use. Oracle strongly recommends changing the default passwords immediately after unlocking the account.

Hit "Close" and congratulate yourself. You've successfully created a new Oracle database.

Tips & Tricks

If you want to start over again with the setup, you can easily achieve that by reverting all changes in the playground VM. Just connect with *root*, and run the following command:

```
[root@orasrv ~]# week1.sh
```

Useful Links

Installation instructions: https://oracle-base.com/articles/12c/oracle-db-12cr2-installation-on-oracle-linux-6-and-7

Oracle official documentation: https://docs.oracle.com/en/database/oracle/oracle-database/12.2/index.html