

Instalación de los gestores de bases de datos

Bases de datos

1 Oracle.....3

2 Postgres.....9

1 Oracle

Primero verificamos si tenemos instalado Java en nuestro sistema operativo, en este caso, Debian.

```
alexrr@pc-alex:~$ java -version
openjdk version "1.8.0_181"
OpenJDK Runtime Environment (build 1.8.0_181-8u181-b13-2-deb9u1-b13)
OpenJDK 64-Bit Server VM (build 25.181-b13, mixed mode)
alexrr@pc-alex:~$
```

Si no sale nada, instalar JavaJDK con los siguientes comandos:

sudo add-apt-repository ppa:webupd8team/java

sudo apt-get update

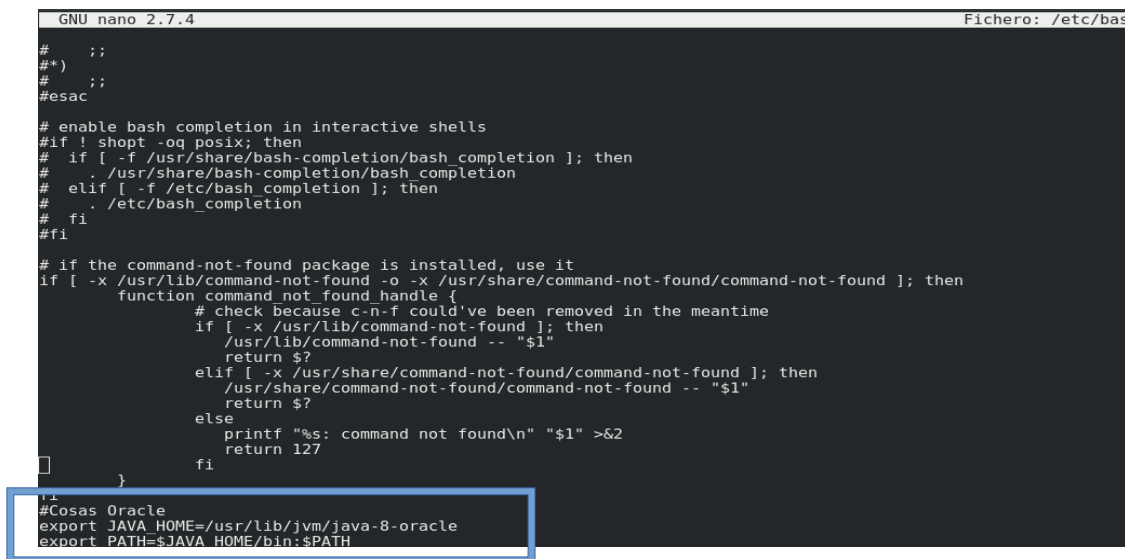
sudo apt-get install oracle-java8-installer

Si los ejecutamos en ese orden, deberá denjarnos ejecutar el comando `java -version`.

Una vez que tengamos JavaJDK tendremos que copiar el siguiente mensaje en nuestro `.bashrc`:

export JAVA_HOME=/usr/lib/jvm/java-8-oracle

export PATH=\$JAVA_HOME/bin:\$PATH



```
GNU nano 2.7.4 Fichero: /etc/bas
# ;;
#*)
# ;;
#esac

# enable bash completion in interactive shells
#if ! shopt -oq posix; then
# if [ -f /usr/share/bash-completion/bash_completion ]; then
# . /usr/share/bash-completion/bash_completion
# elif [ -f /etc/bash_completion ]; then
# . /etc/bash_completion
# fi
#fi

# if the command-not-found package is installed, use it
if [ -x /usr/lib/command-not-found -o -x /usr/share/command-not-found/command-not-found ]; then
function command_not_found_handle {
    # check because c-n-f could've been removed in the meantime
    if [ -x /usr/lib/command-not-found ]; then
        /usr/lib/command-not-found -- "$1"
        return $?
    elif [ -x /usr/share/command-not-found/command-not-found ]; then
        /usr/share/command-not-found/command-not-found -- "$1"
        return $?
    else
        printf "%s: command not found\n" "$1" >&2
        return 127
    fi
}

##
#Cosas Oracle
export JAVA_HOME=/usr/lib/jvm/java-8-oracle
export PATH=$JAVA_HOME/bin:$PATH
```

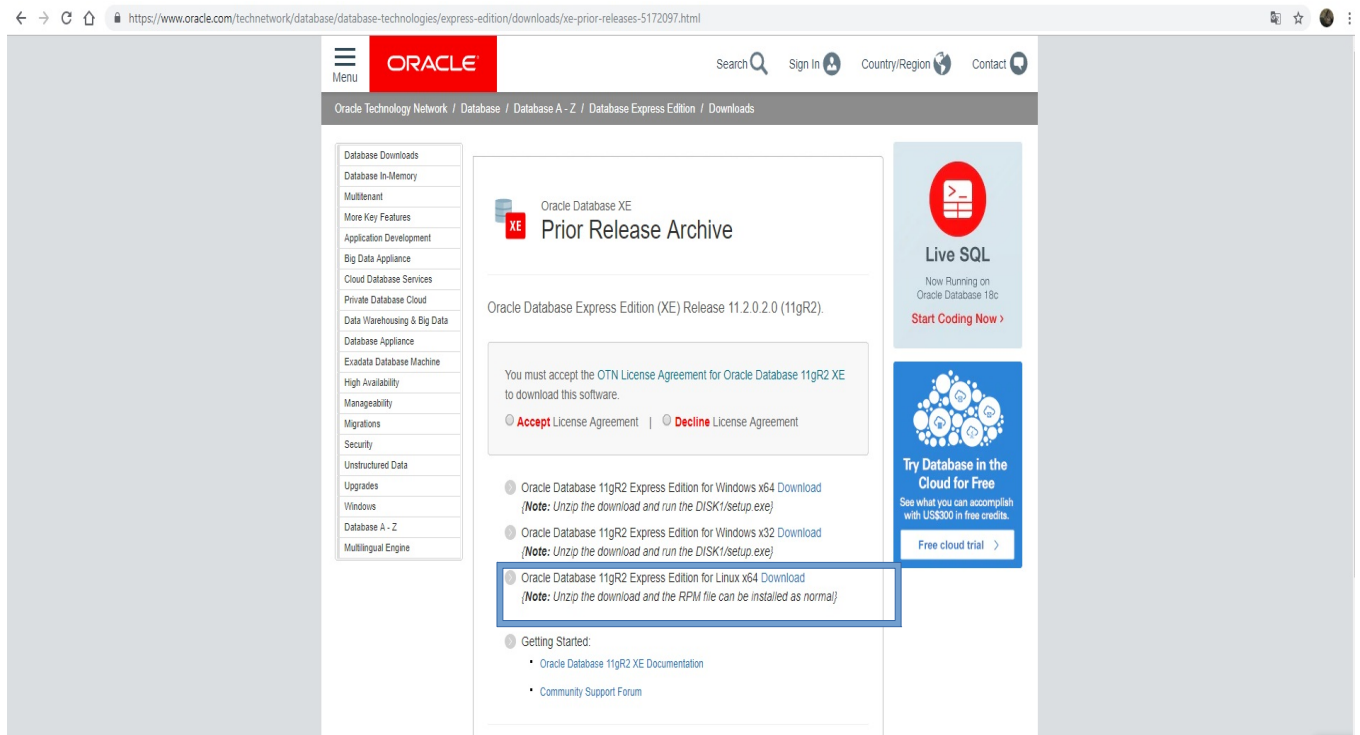
Recargamos la `bashrc` con el comando **source /etc/bash.bashrc**

Para comprobar que esta bien añadido utilizamos el comando:

echo \$JAVA_HOME

```
alexrr@pc-alex:~$ echo $JAVA_HOME
/usr/lib/jvm/java-8-oracle
alexrr@pc-alex:~$
```

Acto seguido vamos a la página de descarga de Oracle para descargarnos la version que mas nos convenga, en este caso descargaremos la versión 11g Express Edition.



Nos tendremos que registrar para poder descargar el archivo, acto seguido la descarga comenzará.

```
alexrr@pc-alex:~$ cd Descargas/  
alexrr@pc-alex:~/Descargas$ ls ora*  
oracle-xe-11.2.0-1.0.x86_64.rpm.zip  
alexrr@pc-alex:~/Descargas$
```

Usamos el siguiente comando para descomprimir el archivo:

unzip oracle-xe-11.2.0-1.0.x86_64.rpm.zip

Nos creara una carpeta llamada Disk1.

```
alexrr@pc-alex:~/Descargas$ cd Disk1/  
alexrr@pc-alex:~/Descargas/Disk1$ ls  
oracle-xe-11.2.0-1.0.x86_64.rpm oracle-xe_11.2.0-2_amd64.deb response upgrade  
alexrr@pc-alex:~/Descargas/Disk1$
```

El archivo coloreado en rojo estará en .rpm(En este caso yo ya lo cambié a .deb), por lo que habrá que cambiarlo, para ello precisamos de instalar los siguientes programas:

sudo apt-get install alien libaio1 unixodbc

Acto seguido usamos el comando:

sudo alien --scripts -d oracle-xe-11.2.0-1.0.x86_64.rpm

Este comando durará varios minutos, por lo que podemos aprovechar para añadir comandos a diferentes ficheros.

Añadimos el fichero chkconfig en /sbin/chkconfig y ponemos las siguientes lineas:

```
#!/bin/bash

#Oracle 11gR2 XE installer chkconfig hack for Debian

file=/etc/init.d/oracle-xe

if [[ ! `tail -n1 $file | grep INIT` ]]; then

echo >> $file

echo '### BEGIN INIT INFO' >> $file

echo '# Provides: OracleXE' >> $file

echo '#Required-Start: $remote_fs $syslog' >> $file

echo '#Required-Stop: $remote_fs $syslog' >> $file

echo '#Default-Start: 2 3 4 5' >> $file

echo '#Default-Stop: 0 1 6' >> $file

echo '# Short-Description: Oracle 11g Express Edition' >> $file

echo '### END INIT INFO' >> $file

fi

update-rc.d oracle-xe defaults 80 81

#EOF
```

Y damos los permisos al archivo:

sudo chmod 755 /sbin/chkconfig

También crearemos un archivo con las configuraciones del kernel /etc/sysctl.d/60-oracle.conf

```
#Oracle 11g XE kernel parameters

fs.file-max=6815744

net.ipv4.ip_local_port_range=9000 65000

kernel.sem=250 32000 100 128

kernel.shmmax=536870912
```

Y cargamos los parametros con este comando:

sudo service procps start

Y añadimos este comando:

sudo touch /var/lock/subsys/listener

Después de todo esto instalamos Oracle ubicado en la carpeta Disk1:

sudo dpkg --install oracle-xe_11.2.0-2_amd64.deb

Tardará varios minutos en instalarlo.

Luego de instalarlo ejecutamos el siguiente comando:

sudo mount -t tmpfs shmfs -o size={Tu cantidad máxima de Ram} /dev/shm

Creamos el fichero /etc/rc2.d/S01shm_load y copiamos los siguientes comandos:

```
#!/bin/sh
case "S1" in
start) mkdir /var/lock/subsys 2>/dev/null
touch /var/lock/subsys/listener
rm /dev/shm 2>/dev/null
mkdir /dev/shm 2>/dev/null
mount -t tmpfs shmfs -o size=4096m /dev/shm ;;
*) echo error
exit 1 ;;
esac
```

sudo chmod 755 /etc/rc2.d/S01shm_load

Y ya podremos crear nuestra base de datos:

sudo /etc/init.d/oracle-xe configure

```

Oracle Database 11g Express Edition Configuration
-----
This will configure on-boot properties of Oracle Database 11g Express
Edition. The following questions will determine whether the database should
be starting upon system boot, the ports it will use, and the passwords that
will be used for database accounts. Press <Enter> to accept the defaults.
Ctrl-C will abort.

Specify the HTTP port that will be used for Oracle Application Express [8080]:10080
Specify a port that will be used for the database listener [1521]:

Specify a password to be used for database accounts. Note that the same
password will be used for SYS and SYSTEM. Oracle recommends the use of
different passwords for each database account. This can be done after
initial configuration:
Confirm the password:

Do you want Oracle Database 11g Express Edition to be started on boot (y/n) [y]:n

Starting Oracle Net Listener...Done
Configuring database...Done
Starting Oracle Database 11g Express Edition instance...Done
Installation completed successfully.

```

*En el cuadrado señalado nos pedirá la contraseña, en este caso la contraseña será dios

Por último añadimos a nuestro .bashrc la siguiente línea:

```

export ORACLE_HOME=/u01/app/oracle/product/11.2.0/xe
export ORACLE_SID=XE
export NLS_LANG=`$ORACLE_HOME/bin/nls_lang.sh`
export ORACLE_BASE=/u01/app/oracle
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
export PATH=$ORACLE_HOME/bin:$PATH

```

source /etc/bash.bashrc

```

alexrr@pc-alex:~/Descargas/Disk1$ echo $ORACLE_HOME
/u01/app/oracle/product/11.2.0/xe
alexrr@pc-alex:~/Descargas/Disk1$ 

```

Los comandos para arrancar, apagar y ver el estado de oracle son los siguientes:

sudo service oracle-xe start ARRANCAR

sudo service oracle-xe stop APAGAR

sudo service oracle-xe status ESTADO

*Si queremos tener histórico en nuestra base de datos, instalar rlwrap

Para empezar a usar la base de datos debemos poner sqlplus en la terminal

```
alexrr@pc-alex:~/Descargas/Disk1$ sqlplus
SQL*Plus: Release 11.2.0.2.0 Production on Mar Dic 25 13:59:55 2018
Copyright (c) 1982, 2011, Oracle. All rights reserved.
Enter user-name: 
```

En user-name ponemos system y en password ponemos dios.

Ejecutamos los siguientes comandos para la creación de un usuario propio:

create user alexrr identified by alex;

Concedemos los permisos al usuario:

grant connect to alexrr;

grant resource to alexrr;

grant unlimited tablespace to alexrr;

Y salimos:

disconnect

Y ahora podremos entrar con el usuario creado.

2 Postgres

Para instalar postgres solo debemos poner en la terminal el siguiente comando:

sudo apt-get install postgresql

Añadimos al sistema el usuario postgres:

useradd -m postgres

Y nos cambiamos a ese usuario:

su – postgres

Allí creamos nuestra base de datos:

createdb nombreBD

Ejecutamos el interprete de comandos:

psql

Mostramos las bases de datos existentes:

\l

Para salir:

q

Para conectarse a la base de datos

\c nombreBD

Para mostrar las tablas:

\d

Para mostrar los campos de una tabla:

\d nombreTabla

3 MariaDB

Instalamos los siguientes programas:

```
sudo apt-get install software-properties-common dirmngr
```

Añadimos la clave de repositorios de mariaDB:

```
sudo apt-key adv --recv-keys --keyserver keyserver.ubuntu.com 0xF1656F24C74CD1D8
```

Añadimos el repositorio de mariaDB a sources.list

```
sudo add-apt-repository 'deb [arch=amd64,i386,ppc64el]  
http://mirror.upb.edu.co/mariadb/repo/10.3/debian stretch main'
```

Actualizamos los repositorios:

```
sudo apt-get update
```

Y instalamos MariaDB:

```
sudo apt-get install mariadb-server
```

Nos conectamos como root a MariaDB:

```
mysql -u root -p
```

Creamos un usuario:

```
GRANT ALL PRIVILEGES ON *.* TO 'alexrr'@'localhost'  
IDENTIFIED BY 'alex' WITH GRANT OPTION;
```

Salimos:

```
exit
```

Entramos con el usuario que creamos:

```
mysql -u alexrr -p
```

Para mostrar las bases de datos:

```
show databases;
```

Para crear una base de datos nueva:

```
create database {nombre}
```

Para ponernos en la base de datos:

```
use {nombre}
```

Para ver las tablas que tiene:

```
show tables;
```

Para conectarnos en una base de datos concreta:

```
mysql -u alexrr -p {nombre}
```