

Se puede observar que los trabajos realizados son verdaderos retos para los participantes o miembros del grupo que se integró para este propósito.

Iniciaremos con la documentación para instalar lo siguiente:

1. Gestor de Base de Datos DB2 de IBM.
2. Gestor de Base de Datos MySQL.
3. Seguidamente Oracle XE.

Y por último todos los pasos a seguir para conectar vía ODBC el gestor Oracle XE con DB2, MySQL, de SQL Developer a DB2 y MySQL vía JDBC.

Entremos en materia, a continuación, la documentación descrita anteriormente.

Proceso de instalación de la base de datos DB2 En Debian 9

El primer paso se define creando un directorio en la raíz con el nombre de Downloads y nos ubicamos en la misma con el siguiente comando:

```
mkdir /Downloads
```

```
cd /Downloads
```

Seguidamente iniciamos la descarga del archivo de instalación de db2 con el siguiente comando:

```
wget https://www.dropbox.com/s/icsm9k7qciyztha/v11.1\_linuxx64\_expc.tar.gz
```

Descripciones del archivo

Nombre del archivo: v11.1_linuxx64_expc.tar.gz

Tamaño: 740M

Extensión: .tar.gz

Versión de db2: 11.1

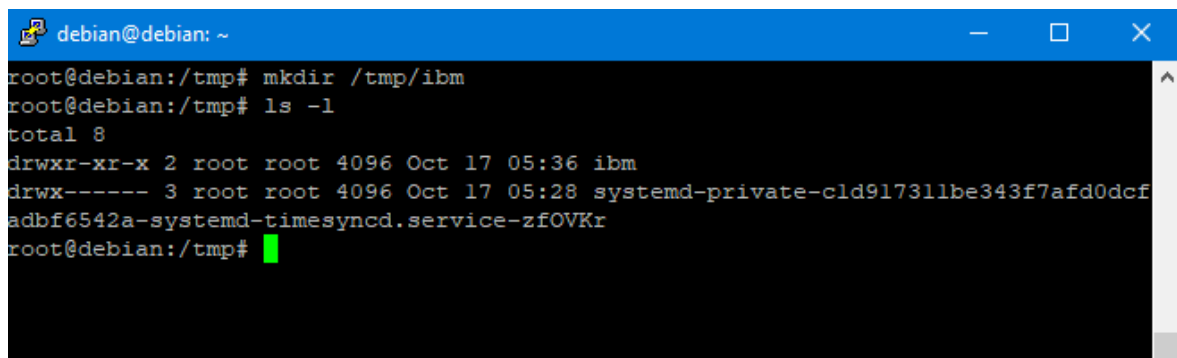
La descarga de este archivo se realizó de la página oficial de IBM en la siguiente dirección con los pasos correspondientes:

<https://www.ibm.com/developerworks/ssa/downloads/im/udbexp/index.html>

Luego fue subido a un servidor de almacenamiento para generar un link de descarga directa.

Creamos un directorio temporal para nuestra instalación con el nombre de “ibm” en el cual extraeremos el contenido del archivo descargado.

```
mkdir /tmp/ibm
```

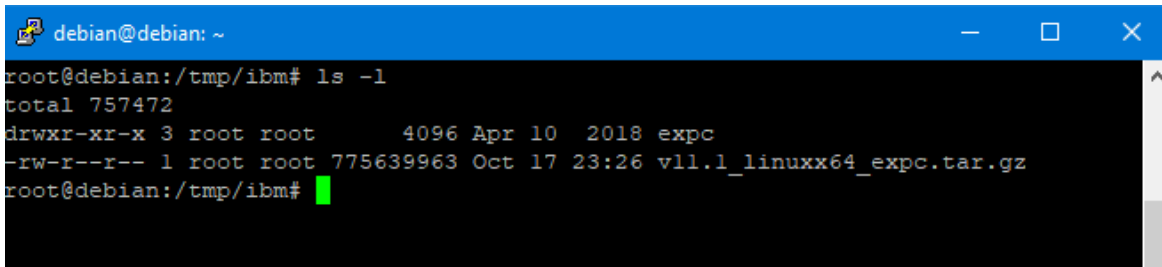


```
debian@debian: ~
root@debian:/tmp# mkdir /tmp/ibm
root@debian:/tmp# ls -l
total 8
drwxr-xr-x 2 root root 4096 Oct 17 05:36 ibm
drwx----- 3 root root 4096 Oct 17 05:28 systemd-private-cld917311be343f7afd0dcf
adbf6542a-systemd-timesyncd.service-zfOVKr
root@debian:/tmp#
```

Nos ubicamos en la carpeta donde se encuentra el archivo descargado y copiamos el archivo a la carpeta temporal “ibm”.

cd /Downloads

cp v11.1_linuxx64_expc.tar.gz /tmp/ibm

A terminal window titled 'debian@debian: ~' with standard window controls. The prompt is 'root@debian:/tmp/ibm#'. The command 'ls -l' has been executed, showing the following output:

```
total 757472
drwxr-xr-x 3 root root    4096 Apr 10  2018 expc
-rw-r--r-- 1 root root 775639963 Oct 17 23:26 v11.1_linuxx64_expc.tar.gz
```

The cursor is on the line following the file listing.

Ya copiado el archivo nos ubicamos en el directorio “ibm” y descomprimos el archivo

cd /tmp/ibm

tar -xvf v11.1_linuxx64_expc.tar.gz

Seguidamente se genera una carpeta con el nombre de **expc** y nos ubicamos en la misma

cd expc.

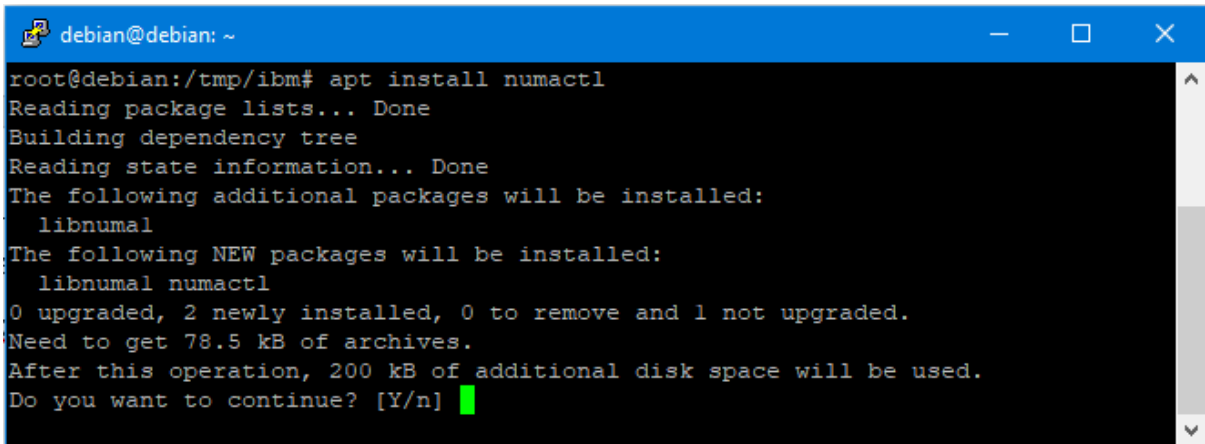
Una vez que corramos el archivo db2prereqcheck, este nos dirá si nuestro sistema cumple con las condiciones y también podremos ver un log, en el cual nos explica cuales funciones no se encontraron o hacen falta revisar para que funcione bien nuestra instalación.

Se hace la observación que en la versión 11.1 el archivo db2prereqcheck no puede determinar si nuestro sistema cumple con las condiciones. Pero este paso puede ser omitido ya que no afecta la instalación ni uso del software. En versiones anteriores el archivo funciona correctamente

Antes de ejecutar el comando de instalación, es necesario que instalemos unas dependencias que nos harán falta:

apt-get install numactl -y

(-y será útil para dar “yes” a toda la instalación)

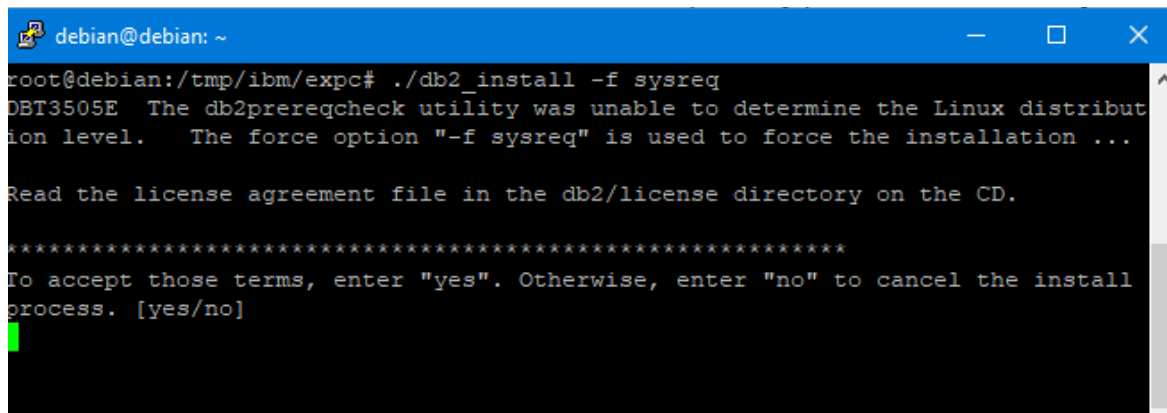
A terminal window titled 'debian@debian: ~' with standard window controls. The prompt is 'root@debian:/tmp/ibm#'. The user enters 'apt install numactl'. The terminal shows the following output: 'Reading package lists... Done', 'Building dependency tree', 'Reading state information... Done', 'The following additional packages will be installed: libnumal', 'The following NEW packages will be installed: libnumal numactl', '0 upgraded, 2 newly installed, 0 to remove and 1 not upgraded.', 'Need to get 78.5 kB of archives.', 'After this operation, 200 kB of additional disk space will be used.', and 'Do you want to continue? [Y/n]' with a green cursor on 'n'.

```
root@debian:/tmp/ibm# apt install numactl
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
 libnumal
The following NEW packages will be installed:
 libnumal numactl
0 upgraded, 2 newly installed, 0 to remove and 1 not upgraded.
Need to get 78.5 kB of archives.
After this operation, 200 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

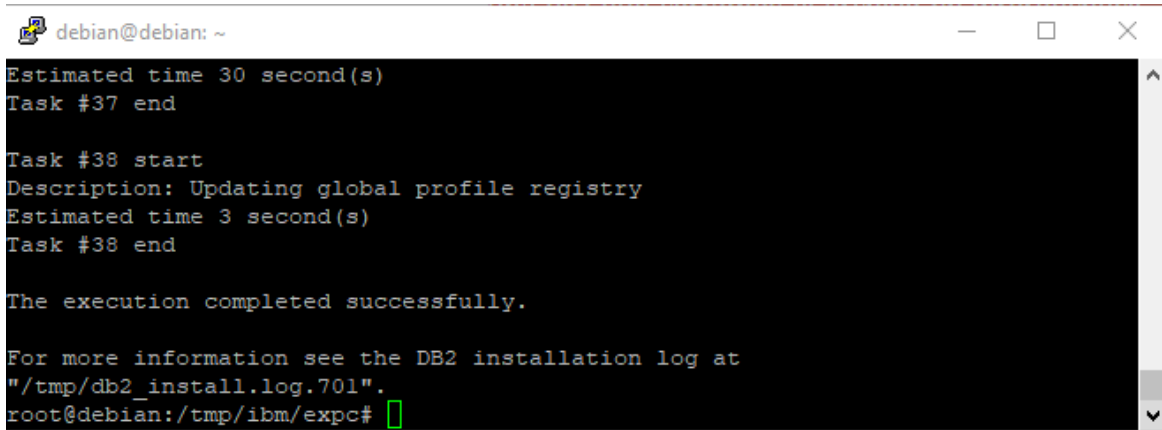
Una vez instaladas nuestras dependencias siempre ubicamos en el directorio expc para procedemos con la instalación, se recomienda utilizar el archivo db2setup para la instalación, pero debido a que estamos instalando en SO Debian utilizaremos el archivo **db2_install** y agregaremos el texto **-f sysreq** para omitir algunos errores de verificación con nuestro sistema, así que ejecutaremos el comando:

./db2_install -f sysreq

Escribimos en minúsculas "yes" y presionamos enter, para continuar con la instalación, tomara un tiempo aproximado de 3 minutos la instalación, todo dependerá de la velocidad del procesador del equipo donde estamos instalando, al finalizar nos mostrara algo como esto:

A terminal window titled 'debian@debian: ~' with standard window controls. The prompt is 'root@debian:/tmp/ibm/expc#'. The user enters './db2_install -f sysreq'. The terminal shows the following output: 'DBT3505E The db2prereqcheck utility was unable to determine the Linux distribution level. The force option "-f sysreq" is used to force the installation ...', 'Read the license agreement file in the db2/license directory on the CD.', a separator line of asterisks, and 'To accept those terms, enter "yes". Otherwise, enter "no" to cancel the install process. [yes/no]' with a green cursor on 'y'.

```
root@debian:/tmp/ibm/expc# ./db2_install -f sysreq
DBT3505E The db2prereqcheck utility was unable to determine the Linux distribution level. The force option "-f sysreq" is used to force the installation ...
Read the license agreement file in the db2/license directory on the CD.
*****
To accept those terms, enter "yes". Otherwise, enter "no" to cancel the install process. [yes/no]
```

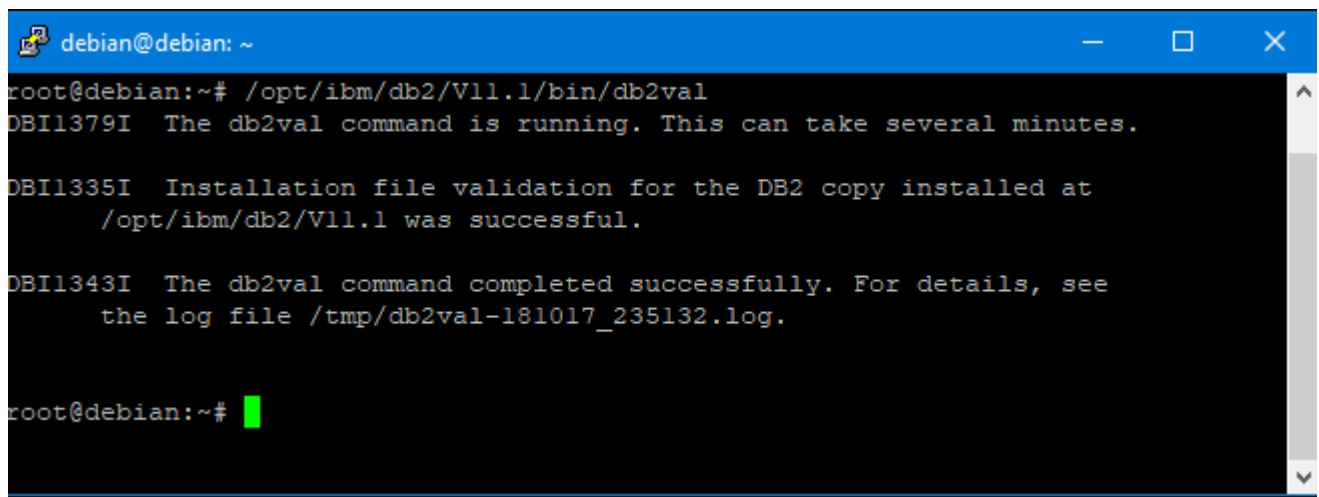


```
debian@debian: ~  
Estimated time 30 second(s)  
Task #37 end  
  
Task #38 start  
Description: Updating global profile registry  
Estimated time 3 second(s)  
Task #38 end  
  
The execution completed successfully.  
  
For more information see the DB2 installation log at  
"/tmp/db2_install.log.701".  
root@debian:/tmp/ibm/expc#
```

En donde claramente podremos ver que la instalación ha completado satisfactoriamente y por último tendremos un archivo con el log (bitácora) de la instalación el cual podremos consultar con el comando "nano /tmp/db2_install.log.701" <-- este fue en nuestro caso, en el de usted puede cambiar el nombre.

Ya hemos terminado nuestra instalación debemos verificar que la instalación haya concluido satisfactoriamente con el comando:

/opt/ibm/db2/V10.5/bin/db2val



```
debian@debian: ~  
root@debian:~# /opt/ibm/db2/V11.1/bin/db2val  
DBI1379I The db2val command is running. This can take several minutes.  
  
DBI1335I Installation file validation for the DB2 copy installed at  
/opt/ibm/db2/V11.1 was successful.  
  
DBI1343I The db2val command completed successfully. For details, see  
the log file /tmp/db2val-181017_235132.log.  
  
root@debian:~#
```

Debido a que en la página de IBM nos indica los grupos y usuarios iniciales que debemos crear, entonces lo hacemos tal como se nos instruye en la página oficial.

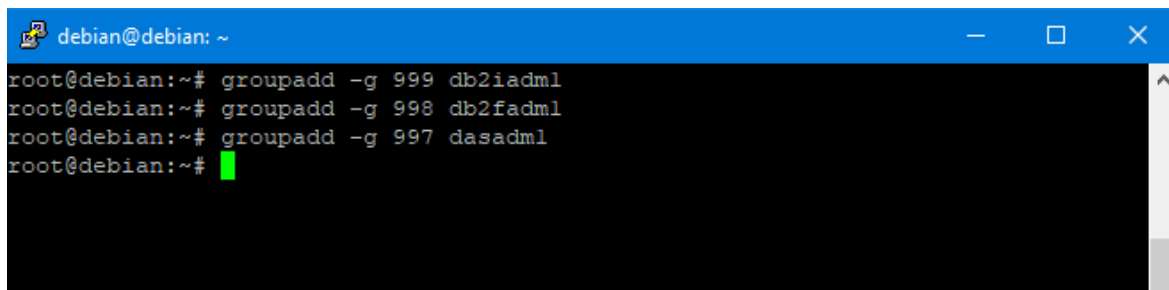
Usuario	Nombre de usuario de ejemplo	Nombre de grupo de ejemplo
Propietario de la instancia	db2inst1	db2iadm1
Usuario delimitado	db2fenc1	db2fadm1
Usuario del Servidor de administración de DB2	dasusr1	dasadm1

Empezaremos creando los grupos con los siguientes comandos:

```
# groupadd -g 999 db2iadm1
```

```
# groupadd -g 998 db2fadm1
```

```
# groupadd -g 997 dasadm1
```

A terminal window titled 'debian@debian: ~' with standard window controls. It shows three commands being executed in root mode: 'groupadd -g 999 db2iadm1', 'groupadd -g 998 db2fadm1', and 'groupadd -g 997 dasadm1'. Each command is followed by a confirmation message: 'groupadd: group db2iadm1 (999) already exists', 'groupadd: group db2fadm1 (998) already exists', and 'groupadd: group dasadm1 (997) already exists'. The prompt returns to root@debian:~# after each command. A green cursor is visible at the end of the third command line.

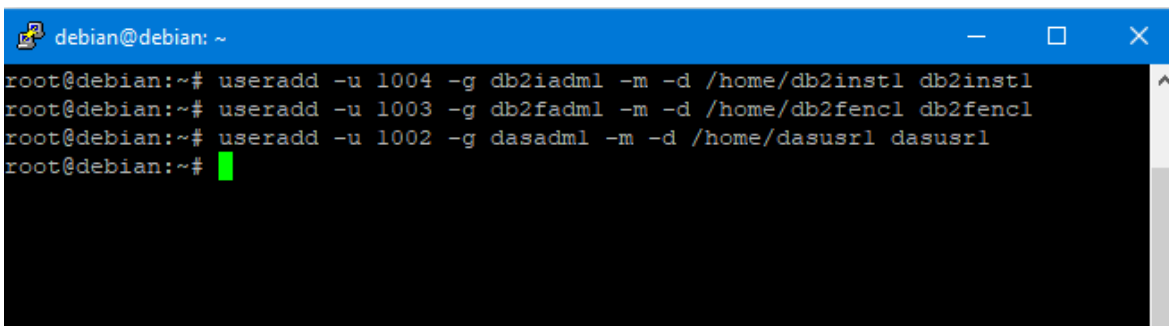
```
root@debian:~# groupadd -g 999 db2iadm1
groupadd: group db2iadm1 (999) already exists
root@debian:~# groupadd -g 998 db2fadm1
groupadd: group db2fadm1 (998) already exists
root@debian:~# groupadd -g 997 dasadm1
groupadd: group dasadm1 (997) already exists
root@debian:~#
```

Luego crearemos los usuarios:

```
# useradd -u 1004 -g db2iadm1 -m -d /home/db2inst1 db2inst1
```

```
# useradd -u 1003 -g db2fadm1 -m -d /home/db2fenc1 db2fenc1
```

```
# useradd -u 1002 -g dasadm1 -m -d /home/dasusr1 dasusr1
```

A terminal window titled 'debian@debian: ~' with standard window controls. It shows three commands being executed in root mode: 'useradd -u 1004 -g db2iadm1 -m -d /home/db2inst1 db2inst1', 'useradd -u 1003 -g db2fadm1 -m -d /home/db2fenc1 db2fenc1', and 'useradd -u 1002 -g dasadm1 -m -d /home/dasusr1 dasusr1'. Each command is followed by a confirmation message: 'useradd: user db2inst1 (1004) already exists', 'useradd: user db2fenc1 (1003) already exists', and 'useradd: user dasusr1 (1002) already exists'. The prompt returns to root@debian:~# after each command. A green cursor is visible at the end of the third command line.

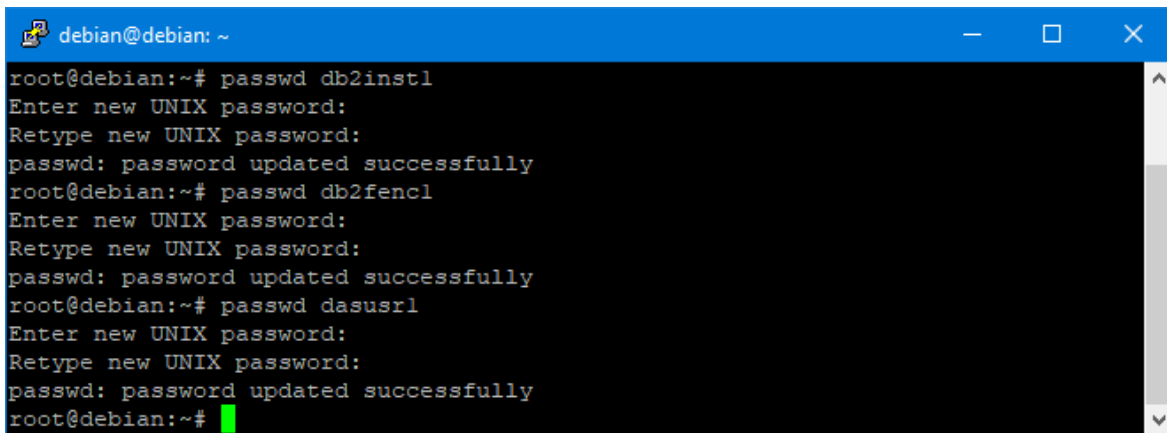
```
root@debian:~# useradd -u 1004 -g db2iadm1 -m -d /home/db2inst1 db2inst1
useradd: user db2inst1 (1004) already exists
root@debian:~# useradd -u 1003 -g db2fadm1 -m -d /home/db2fenc1 db2fenc1
useradd: user db2fenc1 (1003) already exists
root@debian:~# useradd -u 1002 -g dasadm1 -m -d /home/dasusr1 dasusr1
useradd: user dasusr1 (1002) already exists
root@debian:~#
```

Y por ultimo le asignaremos una contraseña, es necesario escribir 2 veces la contraseña de cada usuario para que tome efecto la asignación:

passwd db2inst1

passwd db2fenc1

passwd dasusr1

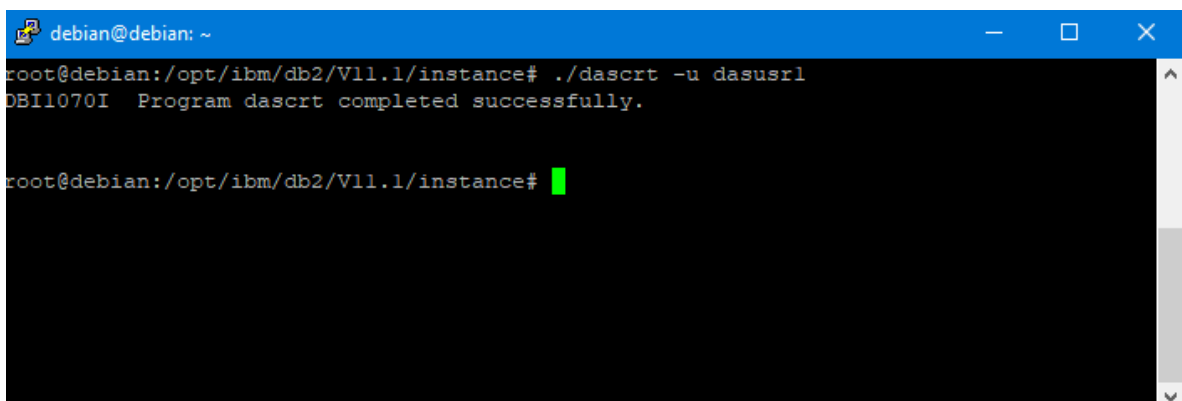
A terminal window titled 'debian@debian: ~' with standard window controls. It shows three password update commands being executed as root. Each command prompts for a new password and its retype, followed by a confirmation message.

```
root@debian:~# passwd db2inst1
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@debian:~# passwd db2fenc1
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@debian:~# passwd dasusr1
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@debian:~#
```

Ya que tenemos todo lo descrito anteriormente funcionando satisfactoriamente, debemos crear el DAS y para eso debemos movernos a la ruta `"/opt/ibm/db2/V11.1/instance"`, para lo cual corremos el siguiente comando:

cd /opt/ibm/db2/V11.1/instance

./dasprt -u dasusr1

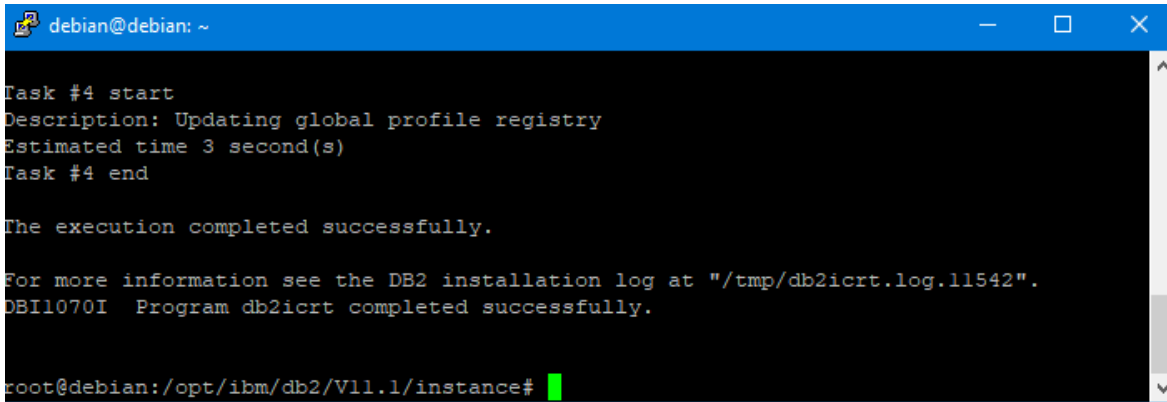
A terminal window titled 'debian@debian: ~' with standard window controls. It shows the user navigating to the directory /opt/ibm/db2/V11.1/instance and running the ./dasprt -u dasusr1 command. The output indicates the program completed successfully.

```
root@debian:/opt/ibm/db2/V11.1/instance# ./dasprt -u dasusr1
DBI1070I  Program dasprt completed successfully.

root@debian:/opt/ibm/db2/V11.1/instance#
```

Después de crear nuestro DAS, es necesario crear nuestra instancia, debemos estar siempre sobre la dirección: "/opt/ibm/db2/V11.1/instance"

./db2icrt -a server -u db2fenc1 db2inst1

A terminal window titled 'debian@debian: ~' with a blue header bar. The terminal output shows the completion of Task #4, which was updating the global profile registry. It states 'The execution completed successfully.' and provides a log file path: '/tmp/db2icrt.log.11542'. The prompt is now 'root@debian:/opt/ibm/db2/V11.1/instance#'.

```
Task #4 start
Description: Updating global profile registry
Estimated time 3 second(s)
Task #4 end

The execution completed successfully.

For more information see the DB2 installation log at "/tmp/db2icrt.log.11542".
DBI1070I  Program db2icrt completed successfully.

root@debian:/opt/ibm/db2/V11.1/instance#
```

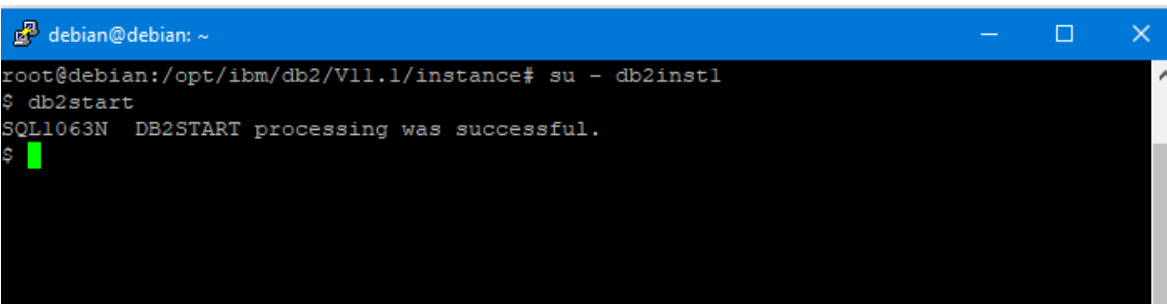
Ya hemos terminado con la instalación, ahora utilizaremos el data base manager para crear nuestra primera Base de Datos.

Para eso tendremos que cambiarnos de usuario a: **db2inst1** con el comando:

su - db2inst1

Al mostrarnos la línea con el signo \$ entonces escribimos el comando para iniciar nuestra instancia:

\$ db2start

A terminal window titled 'debian@debian: ~' with a blue header bar. The terminal shows the user switching to 'db2inst1' and then running 'db2start'. The output is 'SQL1063N DB2START processing was successful.' followed by a new prompt '\$'.

```
root@debian:/opt/ibm/db2/V11.1/instance# su - db2inst1
$ db2start
SQL1063N  DB2START processing was successful.
$
```

De esta forma concluimos la instalación de nuestra base de datos, DB2 Express-C en Debian Jessie 8.8. Para poder entrar al administrador pueden escribir el comando "**db2**"

Y mostrará lo siguiente:

db2 =>


```
debian@debian: ~  
For general help, type: ?.  
For command help, type: ? command, where command can be  
the first few keywords of a database manager command. For example:  
  ? CATALOG DATABASE for help on the CATALOG DATABASE command  
  ? CATALOG           for help on all of the CATALOG commands.  
  
To exit db2 interactive mode, type QUIT at the command prompt. Outside  
interactive mode, all commands must be prefixed with 'db2'.  
To list the current command option settings, type LIST COMMAND OPTIONS.  
  
For more detailed help, refer to the Online Reference Manual.  
  
db2 =>
```

Crearemos la primera base de datos de ejemplo con el comando que siguiente:

CREATE DATABASE EJEMPLO

Este proceso puede tardar un aproximado de 3 a 5 Minutos.

```
debian@debian: ~  
For command help, type: ? command, where command can be  
the first few keywords of a database manager command. For example:  
  ? CATALOG DATABASE for help on the CATALOG DATABASE command  
  ? CATALOG           for help on all of the CATALOG commands.  
  
To exit db2 interactive mode, type QUIT at the command prompt. Outside  
interactive mode, all commands must be prefixed with 'db2'.  
To list the current command option settings, type LIST COMMAND OPTIONS.  
  
For more detailed help, refer to the Online Reference Manual.  
  
db2 => CREATE DATABASE EJEMPLO
```

```
debian@debian: ~  
the first few keywords of a database manager command. For example:  
  ? CATALOG DATABASE for help on the CATALOG DATABASE command  
  ? CATALOG           for help on all of the CATALOG commands.  
  
To exit db2 interactive mode, type QUIT at the command prompt. Outside  
interactive mode, all commands must be prefixed with 'db2'.  
To list the current command option settings, type LIST COMMAND OPTIONS.  
  
For more detailed help, refer to the Online Reference Manual.  
  
db2 => CREATE DATABASE EJEMPLO  
DB20000I  The CREATE DATABASE command completed successfully.  
db2 =>
```

Manual de Instalación En Debian 9 MySQL

Instalación a través de repositorio

- En caso no se tenga instalado, procedemos a instalar el siguiente repositorio:

Comando: **apt install gdebi-core** (se encarga de la instalación del repositorio)

- Nos dirigimos a <https://dev.mysql.com/downloads/repo/apt/> para verificar cual es la última versión disponible:



- Ahora lo descargamos dentro del directorio Downloads, por medio de un wget que contenga al final el nombre del archivo:

```
root@srvdebian:~/Downloads# wget http://dev.mysql.com/get/mysql-apt-config_0.8.10-1_all.deb
```

```
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://dev.mysql.com/get/mysql-apt-config_0.8.10-1_all.deb [following]
--2018-10-22 19:46:23-- https://dev.mysql.com/get/mysql-apt-config_0.8.10-1_all.deb
Connecting to dev.mysql.com (dev.mysql.com)|137.254.60.11|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://repo.mysql.com//mysql-apt-config_0.8.10-1_all.deb [following]
--2018-10-22 19:46:24-- https://repo.mysql.com//mysql-apt-config_0.8.10-1_all.deb
Resolving repo.mysql.com (repo.mysql.com)... 23.54.21.233
Connecting to repo.mysql.com (repo.mysql.com)|23.54.21.233|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 35970 (35K) [application/x-debian-package]
Saving to: 'mysql-apt-config_0.8.10-1_all.deb'

mysql-apt-config_0. 100%[=====>] 35.13K --.-KB/s in 0.07s

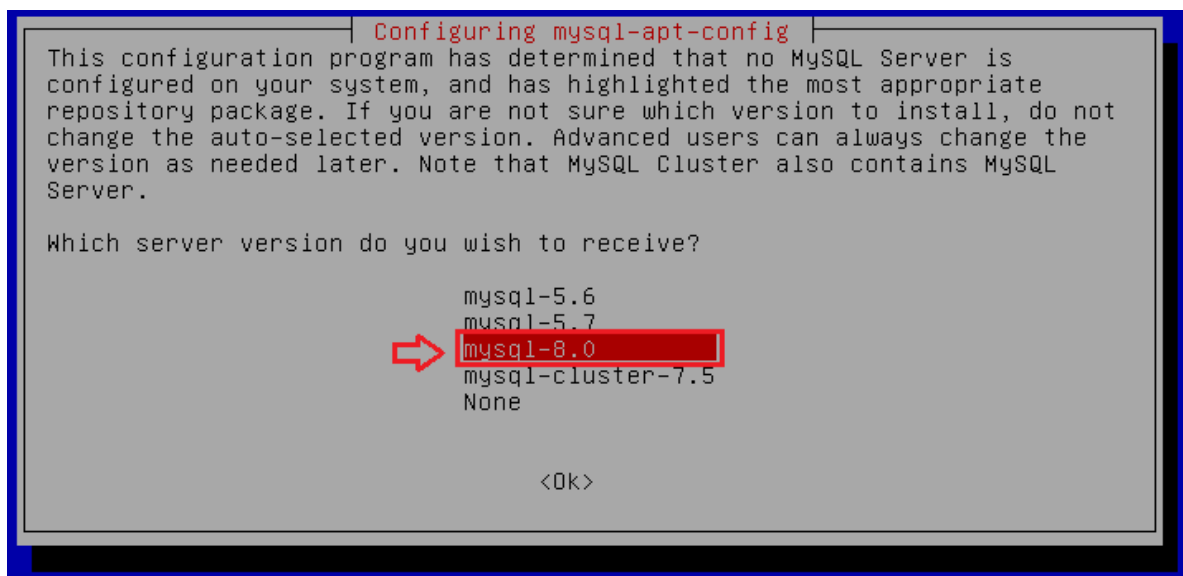
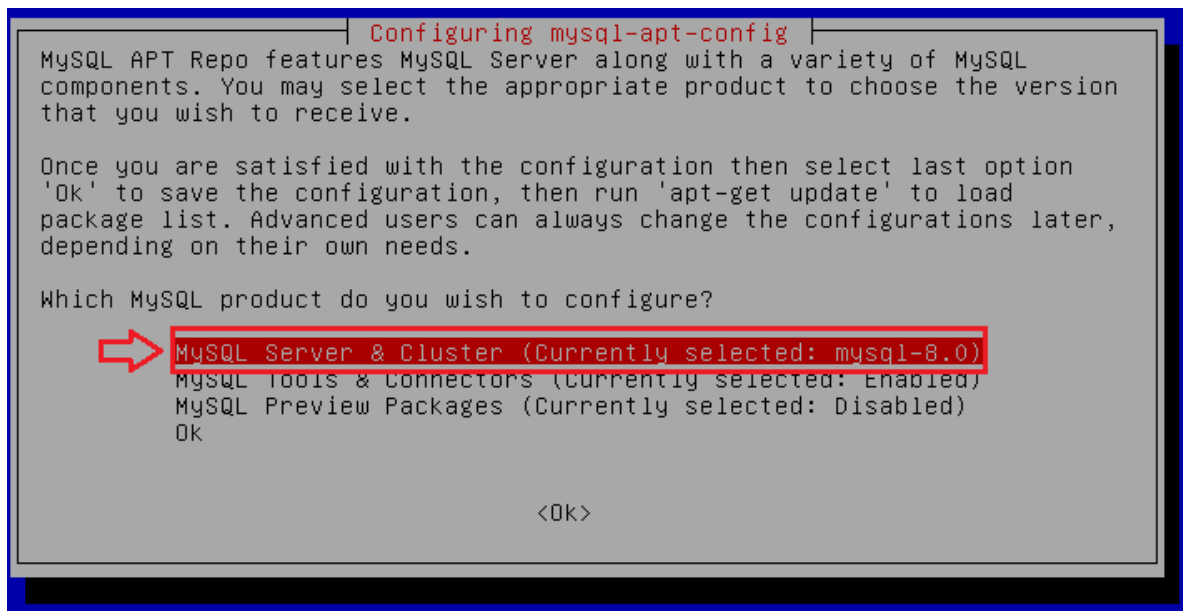
2018-10-22 19:46:24 (524 KB/s) - 'mysql-apt-config_0.8.10-1_all.deb' saved [35970/35970]

root@srvdebian:~/Downloads# ls
Disk1 oracle-xe-11.2.0-1.0.x86_64.rpm.zip?dl=1
mysql-apt-config_0.8.10-1_all.deb
root@srvdebian:~/Downloads#
```

- Instalamos el archivo descargado con el comando gdebi y aceptamos su instalación:

Comando: **gdebi mysql-apt-config_0.8.10-1_all.deb**

- Ahora emergerá una ventana preguntándonos que servicios queremos descargar, a lo que indicamos con la tecla Enter los siguientes y aceptamos al finalizar con “OK”:



```
Which MySQL product do you wish to configure?

MySQL Server & Cluster (Currently selected: mysql-8.0)
MySQL Tools & Connectors (Currently selected: Enabled)
MySQL Preview Packages (Currently selected: Disabled)
Ok
<Ok>

Warning: apt-key should not be used in scripts (called from postinst maintainers
script of the package mysql-apt-config)
OK
root@srvdebian:~/Downloads# _
```

- Procedemos a actualizar y ejecutar la instalación del servidor:

Comando: **apt update**

Comando: **apt install mysql-server**

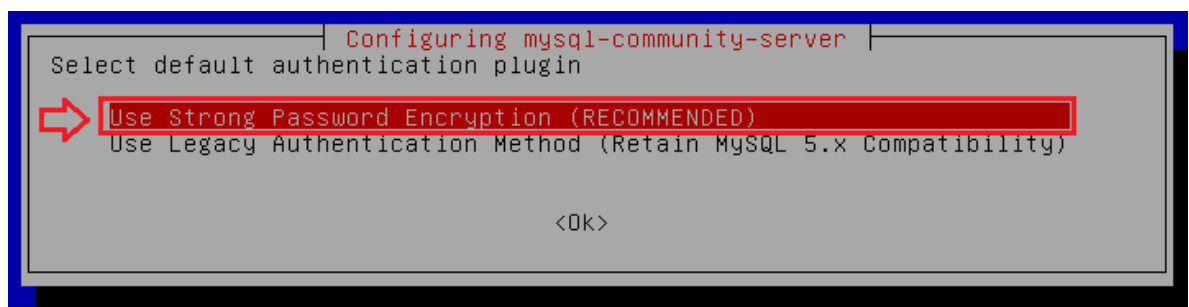
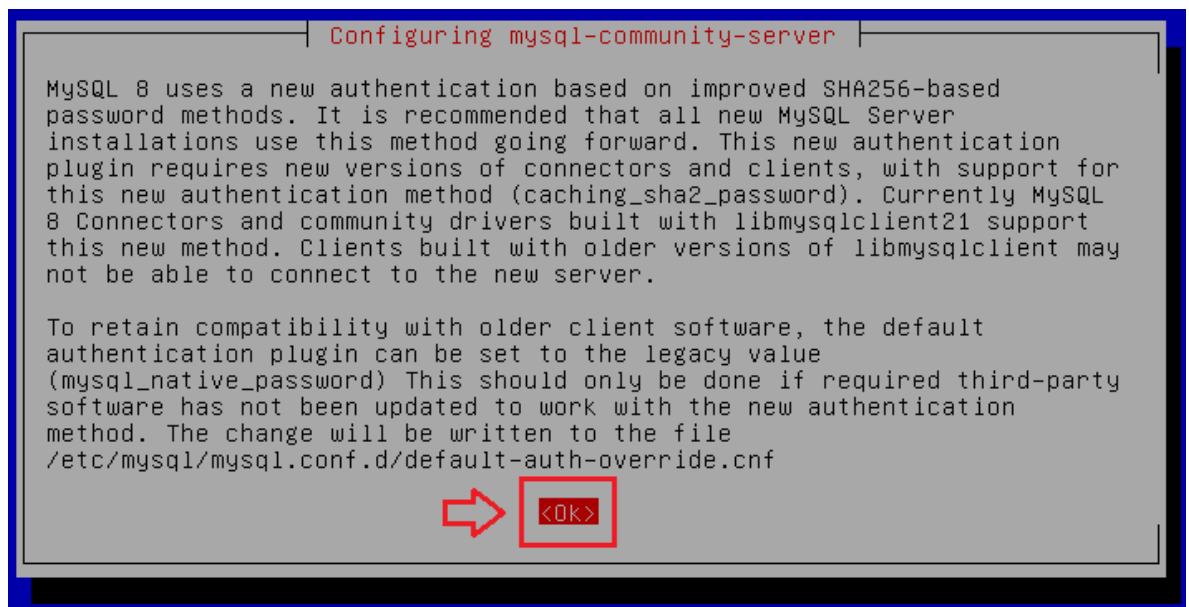
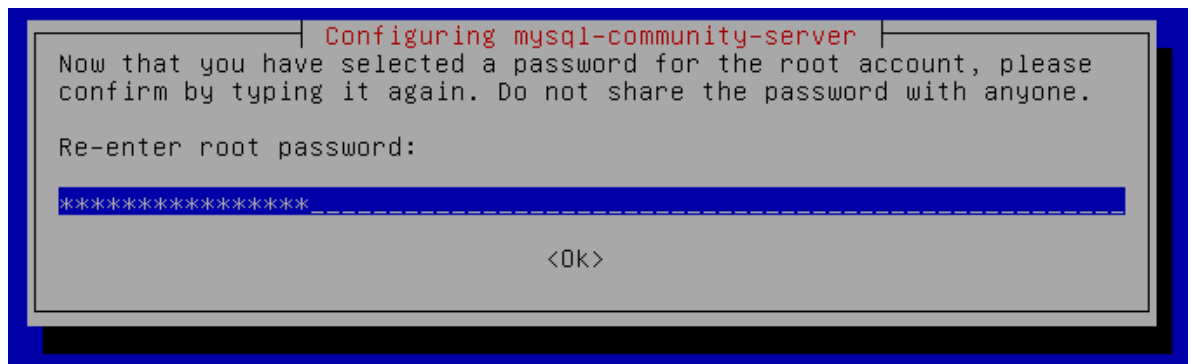
```
root@srvdebian:~/Downloads# apt install mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libmecab2 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client
  mysql-common mysql-community-client mysql-community-client-core
  mysql-community-server mysql-community-server-core psmisc
The following NEW packages will be installed:
  libmecab2 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client
  mysql-common mysql-community-client mysql-community-client-core
  mysql-community-server mysql-community-server-core mysql-server psmisc
0 upgraded, 12 newly installed, 0 to remove and 24 not upgraded.
Need to get 56.9 MB of archives.
After this operation, 422 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

- Ingresamos nuestra contraseña y configuración:

```
Configuring mysql-community-server
Please provide a strong password that will be set for the root account
of your MySQL database. Leave it blank to enable password less login
using UNIX socket based authentication.

Enter root password:
*****_

<Ok>
```



- Si todo fue ejecutado correctamente, MySQL se encuentra instalado en su versión Server 8:

Manual de Instalación En Debian 9 a Oracle XE

- Procedemos a instalar los repositorios necesarios:

Comando: **apt install vim** (Editor de textos).

Comando: **apt install alien** (Conversor de Formato a .deb).

Comando: **apt install net-tools -y** (Conforman la base de trabajo en red de Linux).

Comando: **apt-get install unzip** (Descomprime los archivos).

Comando: **apt-get install bc** (calculadora de líneas de comando).

Comando: **apt-get update, apt-get install ksh** (Interpreta comandos, autocomplete nombres de archivo y gestiona tareas).

Comando: **apt-get update, apt-get install gawk** (Modifica. Busca y transforma archivos).

Descarga y Conversión de Oracle:

- Tras instalar los repositorios, creamos en la raíz un directorio llamado Downloads:

Comando: **mkdir Downloads**

- Luego procedemos a ingresar a dicho directorio:

Comando: **cd Downloads/**

- Ahora iniciamos la descarga de Oracle 11g (por medio de un servidor y su enlace usando wget):

Comando: **wget https://www.dropbox.com/s/fi3xhghjtqcaq7e/oracle-xe-11.2.0-1.0.x86_64.rpm.zip?dl=1**

- Procedemos a verificar que el archivo se encuentre descargado en el directorio:

Comando: **ls**

```
root@srvdebian:~/Downloads# ls
Disk1 oracle-xe-11.2.0-1.0.x86_64.rpm.zip?dl=1
root@srvdebian:~/Downloads#
```

- Ya con el archivo descargado, procedemos a descomprimirlo:

Comando: **unzip oracle-xe-11.2.0-1.0.x86_64.rpm.zip?dl=1**

- Ahora entramos al directorio que se generó tras descomprimir el archivo:

Comando: `cd Disk1/`

- Por ultimo comprobamos los archivos existentes en dicho directorio (Disk1):

Comando: `ls`

- Y empezamos a convertir el `archiv.rpm` a un `.deb` (Este proceso puede durar bastante, finalizara la conversión hasta que mande un mensaje de que el archivo se ha generado.....):

```
oracle-xe_11.2.0-2_amd64.deb generated
```

Comando: `alien --scripts oracle-xe-11.2.0-1.0.x86_64.rpm`

Instalación Oracle XE

- Tras finalizar la conversión del `archivo.rpm` a `.deb`, procedemos a instalar Oracle XE y prepararlo para su configuración:

Comando: `dpkg -i oracle-xe*.deb`

```
root@srvdebian:~/Downloads/Disk1# ls
oracle-xe-11.2.0-1.0.x86_64.rpm  response
oracle-xe_11.2.0-2_amd64.deb    upgrade
root@srvdebian:~/Downloads/Disk1# dpkg -i oracle-xe*.deb
(Reading database ... 50845 files and directories currently installed.)
Preparing to unpack oracle-xe_11.2.0-2_amd64.deb ...
Unpacking oracle-xe (11.2.0-2) over (11.2.0-2) ...
Setting up oracle-xe (11.2.0-2) ...
Executing post-install steps...
You must run '/etc/init.d/oracle-xe configure' as the root user to configure the
database.

Processing triggers for libc-bin (2.24-11+deb9u3) ...
Processing triggers for systemd (232-25+deb9u4) ...
Processing triggers for mime-support (3.60) ...
root@srvdebian:~/Downloads/Disk1# _
```

- Creamos el link simbólico (Symbolic Link):

Comando: `ln -s /usr/bin/awk /bin/awk`

```
root@srvdebian:~/Downloads/Disk1# ln -s /usr/bin/awk /bin/awk
```

- Por ultimo verificamos el id de oracle:

Comando: `id oracle`

```
root@srvdebian:~/Downloads/Disk1# id oracle
uid=1004(oracle) gid=3001(dba) groups=3001(dba)
root@srvdebian:~/Downloads/Disk1# _
```

Configuración Oracle XE

- Ahora empezaremos con la correspondiente configuración de Oracle XE en donde colocaremos los puertos predeterminados, contraseña de usuario sys y configuraremos la iniciación de la base de datos de forma manual:

Comando: `/etc/init.d/oracle-xe configure`

```
-----
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]:

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.
root@debianserver:/#
```

- Por ultimo procedemos a activar y configurar el comando a sqlplus, para lo que necesitaremos ingresar como el usuario definido por Oracle:

Comando: `su - oracle`

```
root@srvdebian:~# su - oracle
```

```
oracle@srvdebian:~$
```

- Tras ingresar como usuario Oracle utilizamos el editor de archivos VIM he ingresamos las siguientes líneas al documento creado:

Comando: `vim .profile`

```
oracle@srvdebian:~$ vim .profile
```


Anexo

Si por cualquier motivo existe el error de petición de disco al querer instalar Oracle seguir la siguiente instrucción:

- Ingresar a la sources.list y comentar el apartado correspondiente:

Comando: **vim /etc/apt/sources.list**

Proceso de configuración para leer tablas desde Oracle(Debian 9) a DB2(Debian 9)

Ingresamos como usuario Oracle con el comando

su - oracle

```
root@debian:~# su - oracle
oracle@debian:~$ _
```

Cambiamos de directorio con el siguiente comando:

cd /u01/app/Oracle/product/11.2.0/xe/hs/admin

y creamos un archivo con el nombre init [nombre de mi DSN].ora

```
oracle@debian:~/product/11.2.0/xe/hs/admin$ ls -l
total 20
-rwxr-xr-x 1 oracle dba 460 Nov  5 17:42 initDB2GERBER.ora
-rwxr-xr-x 1 oracle dba 489 Aug 29 2011 initdg4odbc.ora
-rw-r--r-- 1 oracle dba 259 Nov  5 13:59 initMYSQLGERBER.ora
-rwxr-xr-x 1 oracle dba 391 Aug 29 2011 listener.ora.sample
-rwxr-xr-x 1 oracle dba 250 Aug 29 2011 tnsnames.ora.sample
oracle@debian:~/product/11.2.0/xe/hs/admin$ _
```

E ingresamos los siguientes parámetros:

En este archivo haremos referencia al nombre de nuestro DSN, nuestras librerías, nuestros archivos odbc y odbcinst y el tipo de lenguaje que maneja db2 para la conversión de tipo de dato.

```
HS_FDS_CONNECT_INFO=DB2GERBER
HS_FDS_TRACE_LEVEL=off
HS_FDS_SHAREABLE_NAME=/opt/ibm/odbc_cli/clidriver/lib/libdb2.so
HS_FDS_SUPPORT_STATISTICS=FALSE
set ODBCINSTINI=/etc/odbcinst.ini
set ODBCINI=/etc/odbc.ini
set DB2INSTANCE=db2inst1
HS_LANGUAGE=AMERICAN_AMERICA.US7ASCII
```

En nuestro caso se está utilizando nano como editor de texto guardamos con control + o, + Enter y cerramos con control + x, + enter.

OSCAR ANTONIO LOPEZ ACAJABON

Luego retrocedemos dos directorios para ubicarnos en network/admin y poder configurar los archivos listener.ora y tnsnames.ora escribimos los siguientes comandos:

```
cd ..  
cd ..  
cd network  
cd admin  
ls -l
```

```
oracle@debian:~/product/11.2.0/xe/network/admin$ ls -l  
total 12  
-rwxr-xr-x 1 oracle dba 1010 Nov  5 14:17 listener.ora  
drwxr-xr-x 2 oracle dba 4096 Oct 21 15:18 samples  
-rwxr-xr-x 1 oracle dba  780 Nov  5 14:18 tnsnames.ora  
oracle@debian:~/product/11.2.0/xe/network/admin$ _
```

Editamos el archivo listener.ora con el editor de texto que deseen en nuestro caso utilizaremos nano

Agregamos tanto los parámetros del DSN para db2 de la siguiente manera:

SID_NAME= Nombres de nuestros DSN

En ORACLE_HOME=Dirección donde se encuentran todos los archivos de configuración de Oracle

En PROGRAMA=Nombre de nuestro archivo dg4odbc

En EVS=LD_LIBRARY_PATH= Direcciones de donde se encuentran nuestras librerías

DB2

```
(SID_DESC =  
  (SID_NAME = DB2GERBER)  
  (ORACLE_HOME=/u01/app/oracle/product/11.2.0/xe)  
  (PROGRAM=dg4odbc)  
  (EVS=LD_LIBRARY_PATH=/opt/ibm/odbc_cli/clidriver/lib:/u01/app/oracle/pro$  
)
```

Guardamos con control + o, + Enter y cerramos con control + x, + enter.

Bien ahora debemos configurar nuestro archivo tnsnames.ora y ejecutamos el siguiente comando:

nano tnsnames.ora

y agregamos los siguientes parámetros:

En SID agregaremos el nombre de nuestro DSN y en HS le indicamos al tnsnames que estamos configurando servicios heterogéneos.

OSCAR ANTONIO LOPEZ ACAJABON

```
DB2GERBER =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)(HOST = debian)(PORT = 1521))  
    (CONNECT_DATA = (SID = DB2GERBER))  
      (HS=OK)  
  )  
)
```

Guardamos con control + o, + Enter y cerramos con control + x, + Enter.

Reiniciamos los servicios tnsnames con los siguientes comandos:

lsnrctl stop

lsnrctl start

Verificamos si el listener está escuchando las configuraciones realizadas en el archivo.

```
Service "DB2GERBER" has 1 instance(s).  
  Instance "DB2GERBER", status UNKNOWN, has 1 handler(s) for this service...
```

Tecleamos el siguiente comando para terminar con la verificación de las configuraciones tnsping [nombre de DSN]

```
Used TNSNAMES adapter to resolve the alias  
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = debian)(  
PORT = 1521)) (CONNECT_DATA = (SID = DB2GERBER)) (HS=OK))  
OK (30 msec)  
oracle@debian:~/product/11.2.0/xe/network/admin$ _
```

Y vemos que la configuración ha sido realizada con éxito. Luego entramos el sqlplus e ingresamos el usuarios **sys as sys dba** y nuestra respectiva contraseña.

```
oracle@debian:~/product/11.2.0/xe/network/admin$ sqlplus  
SQL*Plus: Release 11.2.0.2.0 Production on Thu Nov 8 06:01:19 2018  
Copyright (c) 1982, 2011, Oracle. All rights reserved.  
Enter user-name: sys as sysdba  
Enter password:  
Connected to an idle instance.  
SQL> _
```

Levantamos el servicio de nuestro Oracle con el siguiente comando:

startup

```
SQL> startup
ORACLE instance started.

Total System Global Area  417546240 bytes
Fixed Size                  2227080 bytes
Variable Size              285213816 bytes
Database Buffers           125829120 bytes
Redo Buffers                4276224 bytes
Database mounted.
```

Y creamos nuestro DBLINK con el siguiente script para tener acceso a las tablas creadas en DB2:

```
CREATE PUBLIC DATABASE LINK [nombre de mi dblink] CONNECT TO "db2inst1"
IDENTIFIED BY "[contraseña del usuario db2inst1]" USING '[nombre de mi DSN]';
```

Ahora nos vamos a nuestro servidor DB2 e iniciamos el servicio de db2 con los siguientes comandos:

su – db2inst1

db2start

```
sroot@debian:~# su - db2inst1
$ db2start
SQL1063N  DB2START processing was successful.
$ _
```

Realizamos una consulta desde oracle con el siguiente script:

```
SELECT * FROM [nombre de mi tabla]@[nombre de mi dblink]
```

En nuestro caso estamos conectado a una base de datos EJEMPLO con una tabla alumno y el nombre de nuestro dblink es DB2 y quedaría se la siguiente manera:

```
SELECT * FROM ALUMNO@DB2;
```

```
SQL> SELECT * FROM ALUMNO@DB2;

  ID NOMBRE
---
  1 PRIMER DATO
  2 SEGUNDO DATO
  3 TERCER DATO

SQL>
```

Proceso de configuración para leer tablas desde Oracle(Debian9) a MYSQL(Debian9)

Nos ubicamos en el directorio etc para configurar nuestro archivo odbc.ini e ingresamos los siguientes parámetros:

[nombre de nuestro DSN]

Driver= ruta donde se encuentran nuestras librerías

Server= ip de nuestro servidor

PORT= puerto a utilizar

USER= nombre de nuestra usuario

Password= contraseña de nuestro usuario

Database= nombre de nuestra base de datos

nano /etc/odbc.ini

```
[MYSQLGERBER]
Driver      = /opt/mysql/mysql-connector-odbc-8.0.13-linux-debian9-x86-64bit/lib
Description = Connector/ODBC 8.0 ANSI Driver DSN
SERVER      = 192.168.43.118
PORT        = 3306
USER        = PROYECTO
Password    = 123
Database    = PROYECTO
OPTION      = 3
SOCKET      =
~
```

Ingresamos como usuario Oracle con el comando:

su – oracle

```
root@debian:~# su - oracle
oracle@debian:~$ _
```

Cambiamos de directorio con el siguiente comando:

cd /u01/app/Oracle/product/11.2.0/xe/hs/admin

y creamos un archivo con el nombre init [nombre de mi DSN].ora

```
ls -oracle@debian:~/product/11.2.0/xe/hs/admin$ ls -l
total 20
-rwxr-xr-x 1 oracle dba 460 Nov  5 17:42 initDB2GERBER.ora
-rwxr-xr-x 1 oracle dba 489 Aug 29 2011 initdg4odbc.ora
-rw-r--r-- 1 oracle dba 259 Nov  5 13:59 initMYSQLGERBER.ora ←
-rwxr-xr-x 1 oracle dba 391 Aug 29 2011 listener.ora.sample
-rwxr-xr-x 1 oracle dba 250 Aug 29 2011 tnsnames.ora.sample
oracle@debian:~/product/11.2.0/xe/hs/admin$ _
```

Ingresamos los siguientes parámetros:

En este archivo haremos referencia al nombre de nuestro DSN, nuestras librerías, nuestros archivos odbc y odbcinst y el tipo de lenguaje que maneja MYSQL para la conversión de tipo de dato.

```
HS_FDS_CONNECT_INFO=MYSQLGERBER
HS_FDS_SHAREABLE_NAME=/opt/mysql/mysql-connector-odbc-8.0.13-linux-debian9-x86-$
HS_FDS_FETCH_ROWS=1
set ODBCINSTINI=/etc/odbcinst.ini
set ODBCINI=/etc/odbc.ini
HS_LANGUAGE=AMERICAN_AMERICA.WE8ISO8859P15
```

En nuestro caso se está utilizando nano como editor de texto guardamos con control + o, + Enter y cerramos con control + x, + enter.

Luego retrocedemos dos directorios para ubicarnos en network/admin y poder configurar los archivos listener.ora y tnsnames.ora escribimos los siguientes comandos:

```
cd ..
cd ..
cd network
cd admin
ls -l
```

```
oracle@debian:~/product/11.2.0/xe/network/admin$ ls -l
total 12
-rwxr-xr-x 1 oracle dba 1010 Nov  5 14:17 listener.ora
drwxr-xr-x 2 oracle dba 4096 Oct 21 15:18 samples
-rwxr-xr-x 1 oracle dba  780 Nov  5 14:18 tnsnames.ora
oracle@debian:~/product/11.2.0/xe/network/admin$ _
```

Editamos el archivo listener.ora con el editor de texto que deseen en mi caso utilizaremos nano

Agregamos tanto los parámetros del DSN para MySQL de la siguiente manera:

SID_NAME= Nombres de nuestros DSN

En ORACLE_HOME=Dirección donde se encuentran todos los archivos de configuración de Oracle

En PROGRAMA=Nombre de nuestro archivo dg4odbc

En EVS=LD_LIBRARY_PATH= Direcciones de donde se encuentran nuestras librerías

```
(SID_DESC =
  (SID_NAME=MYSQLGERBER)
  (ORACLE_HOME=/u01/app/oracle/product/11.2.0/xe)
  (PROGRAM=dg4odbc)
  (ENVS=LD_LIBRARY_PATH=/opt/mysql/mysql-connector-odbc-8.0.13-linux-debian$
)
)
```


guardamos con control + o, + Enter y cerramos con control + x, + enter.

Bien ahora debemos configurar nuestro archivo tnsnames.ora y ejecutamos el siguiente comando:

nano tnsnames.ora

y agregamos los siguientes parámetros:

En SID agregaremos el nombre de nuestro DSN y en HS le indicamos al tnsnames que estamos configurando servicios heterogéneos.

```
MYSQLGERBER =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = debian)(PORT = 1521))
    (CONNECT_DATA = (SID=MYSQLGERBER))
    (HS=OK)
  )
```

guardamos con control + o, + Enter y cerramos con control + x, + enter.

Reiniciamos los servicios tnsnames con los siguientes comandos:

lsnrctl stop

lsnrctl start

y verificamos si el listener está escuchando las configuraciones realizadas en el archivo

```
Service "MYSQLGERBER" has 1 instance(s).
  Instance "MYSQLGERBER", status UNKNOWN, has 1 handler(s) for this service...
```

Tecleamos el siguiente comando para terminar con la verificación de las configuraciones tnsping [nombre de DSN]

```
Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = debian)(
PORT = 1521)) (CONNECT_DATA = (SID=MYSQLGERBER)) (HS=OK))
OK (30 msec)
oracle@debian:~/product/11.2.0/xe/network/admin$
```

Y vemos que la configuración ha sido realizada con éxito. Luego entramos el sqlplus e ingresamos el usuario sys as sys dba y nuestra respectiva contraseña.

```
oracle@debian:~/product/11.2.0/xe/network/admin$ sqlplus
SQL*Plus: Release 11.2.0.2.0 Production on Thu Nov 8 06:01:19 2018
Copyright (c) 1982, 2011, Oracle. All rights reserved.
Enter user-name: sys as sysdba
Enter password:
Connected to an idle instance.
SQL> _
```

Levantamos el servicio de nuestro Oracle con el siguiente comando:

startup

```
SQL> startup
ORACLE instance started.

Total System Global Area  417546240 bytes
Fixed Size                  2227080 bytes
Variable Size              285213816 bytes
Database Buffers           125829120 bytes
Redo Buffers                4276224 bytes
Database mounted.
```

Y creamos nuestro DBLINK con el siguiente script para tener acceso a las tablas creadas en MYSQL:

```
CREATE PUBLIC DATABASE LINK [nombre de mi dblink] CONNECT TO "db2inst1"
IDENTIFIED BY "[contraseña del usuario db2inst1]" USING '[nombre de mi DSN]';
```

Ahora nos vamos a nuestro servidor MYSQL con el usuario root e ingresamos nuestra contraseña con el siguiente comando:

mysql -p

```
root@debian:~# mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.13 MySQL Community Server - GPL

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> _
```

Creamos nuestra base de datos y una tabla como ejemplo

```
CREATE DATABASE PROYECTO;
```

```
CREATE TABLE ALUMNO (CODIGO INT, NOMBRE VARCHAR (100));
```

```
USE PROYECTO;
```

```
DESCRIBE ALUMNO;
```

```
mysql> DESCRIBE ALUMNO
-> ;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CODIGO | int(11)       | YES  |     | NULL    |       |
| NOMBRE | varchar(100)  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.01 sec)

mysql>
```

Creamos un usuario con su respectiva contraseña y damos permisos para gestionar la base de datos anteriormente creada.

```
CREATE USER PROYECTO@[ip de nuestro cliente] IDENTIFIED BY '123';
```

```
GRANT SELECT, INSERT, UPDATE, DELETE ON PROYECTO.* PROYECTO@[ip de nuestro cliente];
```

```
SELECT USER FROM mysql.user;
```

Y verificamos que el usuario ha sido creado con éxito.

```
mysql> SELECT USER FROM mysql.user;
+-----+
| USER |
+-----+
| oracleconn |
| PROYECTO |
| mysql.infoschema |
| mysql.session |
| mysql.sys |
| root |
+-----+
5 rows in set (0.00 sec)

mysql> _
```

Ahora volvemos nuestro servidor oracle y creamos nuestro DBLINK que tendrá acceso a mysql

```
CREATE PUBLIC DATABASE LINK [nombre de nuestro dblink] CONNECT TO "PROYECTO"
IDENTIFIED BY "123" USING "[nombre de nuestro DSN]";
```

Ahora ejecutamos un script de consulta para leer los datos que están en nuestro servidor mysql

```
SELECT * FROM ALUMNO@MYSQL;
```

```
SQL> SELECT * FROM ALUMNO@MYSQLD;  
  
CODIGO NOMBRE  
-----  
      1 ALUMNO UNO  
      2 ALUMNO DOS  
      3 ALUMNO TRES  
  
SQL>
```

Proceso de configuración de unixodbc y ODBC de DB2 de cliente a servidor Debian.

Iniciamos sesión como root.

Instalamos unixodbc con el siguiente comando:

Sudo apt-get install unixodbc

Verificamos que el controlador se haya instalado correctamente con el comando

sudo dpkg -l | grep -i unixodbc

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@debian:~# sudo dpkg -l | grep -i unixodbc
ii unixodbc 2.3.4-1 amd64
Basic ODBC tools
ii unixodbc-dev 2.3.4-1 amd64
ODBC libraries for UNIX (development files)
root@debian:~#
```

Adicional descargamos un comprimido con el nombre de unixODBC-2.3.7.tar.gz de la página <http://www.unixodbc.org>. Podemos compartir el archivo por medio de un dispositivo usb, o bien subirlo un servidor de almacenamiento en la nube para luego para generar un link de descarga directa.

Pegamos el archivo en la carpeta de Downloads y extraemos el contenido del archivo. Con el siguiente comando:

tar -xvf unixODBC-2.2.12.tar

Ejecute el siguiente comando para cambiar al subdirectorio creado cuando se extrajo el archivo unixODBC-2.2.12.tar: **cd unixODBC-2.2.12.**

```
root@debian:~/Downloads/unixODBC-2.3.7# pwd
/root/Downloads/unixODBC-2.3.7
root@debian:~/Downloads/unixODBC-2.3.7#
```

Para configurar sin la GUI, ejecute el siguiente comando: **./configure --prefix=/usr/local/unixODBC --enable-gui=no**

Ejecute el siguiente comando para compilar el paquete (este paso puede tardar unos minutos en completarse): **make**

Ejecute el siguiente comando para instalar programas, archivos de datos y documentación: **make install**

Este paso coloca los subdirectorios bin, etc, lib y include en el directorio /usr/local/unixODBC. Estos subdirectorios contienen los programas, archivos de datos y documentación para el paquete del administrador de controladores unixODBC.

Verificamos que la ejecución del archivo se haya realizado correctamente y nos ubicamos en la siguiente dirección:

cd /usr/local

```
root@debian:~/Downloads/unixODBC-2.3.7# cd /usr
root@debian:/usr# cd local
root@debian:/usr/local# ls -l
total 36
drwxrwsr-x 2 root staff 4096 Oct 17 23:48 bin
drwxrwsr-x 2 root staff 4096 Oct 2 22:15 etc
drwxrwsr-x 2 root staff 4096 Oct 2 22:15 games
drwxrwsr-x 2 root staff 4096 Oct 2 22:15 include
drwxrwsr-x 4 root staff 4096 Oct 2 22:50 lib
lrwxrwxrwx 1 root staff 9 Oct 2 22:15 man -> share/man
drwxrwsr-x 2 root staff 4096 Oct 2 22:15 sbin
drwxrwsr-x 8 root staff 4096 Oct 17 15:38 share
drwxrwsr-x 2 root staff 4096 Oct 2 22:15 src
drwxr-sr-x 7 root staff 4096 Oct 29 15:59 unixODBC
root@debian:/usr/local# _
```

Luego entramos al subdirectorio etc con siguiente comando etc y copiamos los archivos odbc.ini y dbcinst.ini al directorio etc principal con el siguiente comando:

cp odbc.ini /etc/

cp dbcinst.ini /etc/

verificamos que los archivos hayan sido copiados correctamente con el siguiente comando:

cd /etc/

ls -l | grep -i odbc

```
root@debian:/etc# ls -l | grep -i odbc
drwxr-xr-x 2 root root 4096 Dec 12 2016 ODBCDataSources
-rw-r--r-- 1 root root 147 Nov 1 03:01 odbc.ini
-rw-r--r-- 1 root root 174 Nov 1 03:01 dbcinst.ini
root@debian:/etc#
```

Procedemos a configurar e archivo odbc.ini con el siguiente comando y con los apropiados parámetros:

nano odbc.ini

```
[ODBC Data Source]
DB2GERBER = IBM DB2 ODBC DRIVER

[DB2GERBER]
DRIVER = /opt/ibm/odbc_cli/clidriver/lib/libdb2.so
Description = DB2 ODBC Database
```

Guardamos cambios presionando “Control + o” + Enter

Salimos del editor con “Control + x”

Luego configuramos el archivo odbc.ini con el siguiente comando y parámetros:

nano odbcinst.ini

```
[NOMBRE_DE_DSN]
DBAlias = nombre_de_nuestra_BD
HostName = nombre_de_nuestra_BD
Protocol = TCPIP
Hostname = ip_de_nuestro_servidor
Port = 50000
```

```
[DB2GERBER]
dbalias=EJEMPLO
HostName=EJEMPLO
Protocol=TCPIP
HostName=192.168.1.5
PortNumber=50000

[mysql]
Driver=/usr/lib/x86_64-linux-gnu/odbc/libmyodbc8w.so
UsageCount=1
```

Guardamos cambios presionando “Control + o” + Enter

Salimos del editor con “Control + x”

Seguidamente descargamos el ODBC propio de DB2 de la página oficial de IBM en la cual se requiere estar registrado para tener acceso a la descarga. Podemos compartir el archivo por medio de un dispositivo usb, o bien subirlo un servidor de almacenamiento en la nube para luego para generar un link de descarga directa.

Nombre del archivo: v11.13fp3b_linuxx64_odbc_cli.tar.gz

Creamos el directorio “ibm” en la siguiente dirección o si ya está creado podríamos omitir este paso:

mkdir /opt/ibm

```
root@debian:/opt/ibm# pwd
/opt/ibm
root@debian:/opt/ibm#
```

Pegamos el archivo en el directorio anteriormente creado y los descomprimos con el siguiente comando:

tar -xvf v11.13fp3b_linuxx64_odbc_cli.tar.gz

Debemos editar un archivo con el nombre de db2cli.ini con el siguiente comando:

nano /opt/ibm/odbc_cli/clidriver/cfg/db2cli.ini

y en dentro del archive agregamos los siguientes parámetros:

[NOMBRE_DE_DSN]

Database = nombre_de_nuestra_BD

HostName = ip_de_nuestro_servidor

DBAlias = nombre_de_nuestra_BD

Port = 50000

Protocol = TCPIP

UID = db2inst1

PWD = db2inst1

```
[DB2GERBER]
Database = EJEMPLO
HostName = 192.168.1.7
DBAlias = EJEMPLO
Port = 50000
Protocol = TCPIP
UID = db2inst1
PWD = db2inst1
```

Ya realizadas las respectivas configuraciones procedemos a realizar pruebas de conexión con la herramienta isql

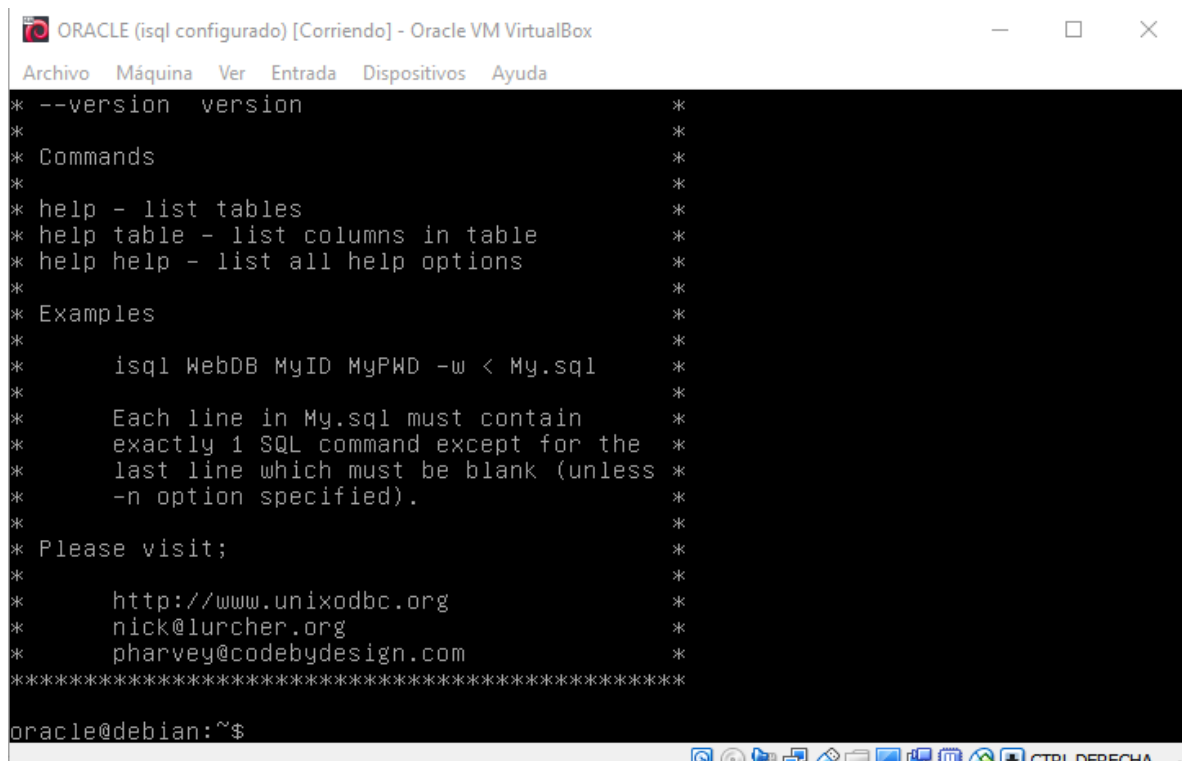
Entramos como usuario Oracle:

su - oracle

```
root@debian:~# su - oracle
oracle@debian:~$ _
```

Verificar que la herramienta isql este correctamente instalada con el comando:

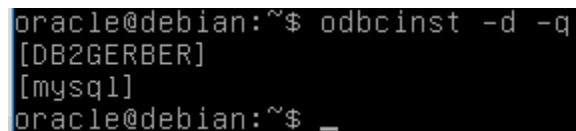
Isql



```
ORACLE (isql configurado) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
* --version version *
* *
* Commands *
* *
* help - list tables *
* help table - list columns in table *
* help help - list all help options *
* *
* Examples *
* *
* isql WebDB MyID MyPWD -w < My.sql *
* *
* Each line in My.sql must contain *
* exactly 1 SQL command except for the *
* last line which must be blank (unless *
* -n option specified). *
* *
* Please visit; *
* *
* http://www.unixodbc.org *
* nick@lurcher.org *
* pharvey@codebydesign.com *
*****
oracle@debian:~$
```

Verificamos nuestros DataSources disponibles con el comando

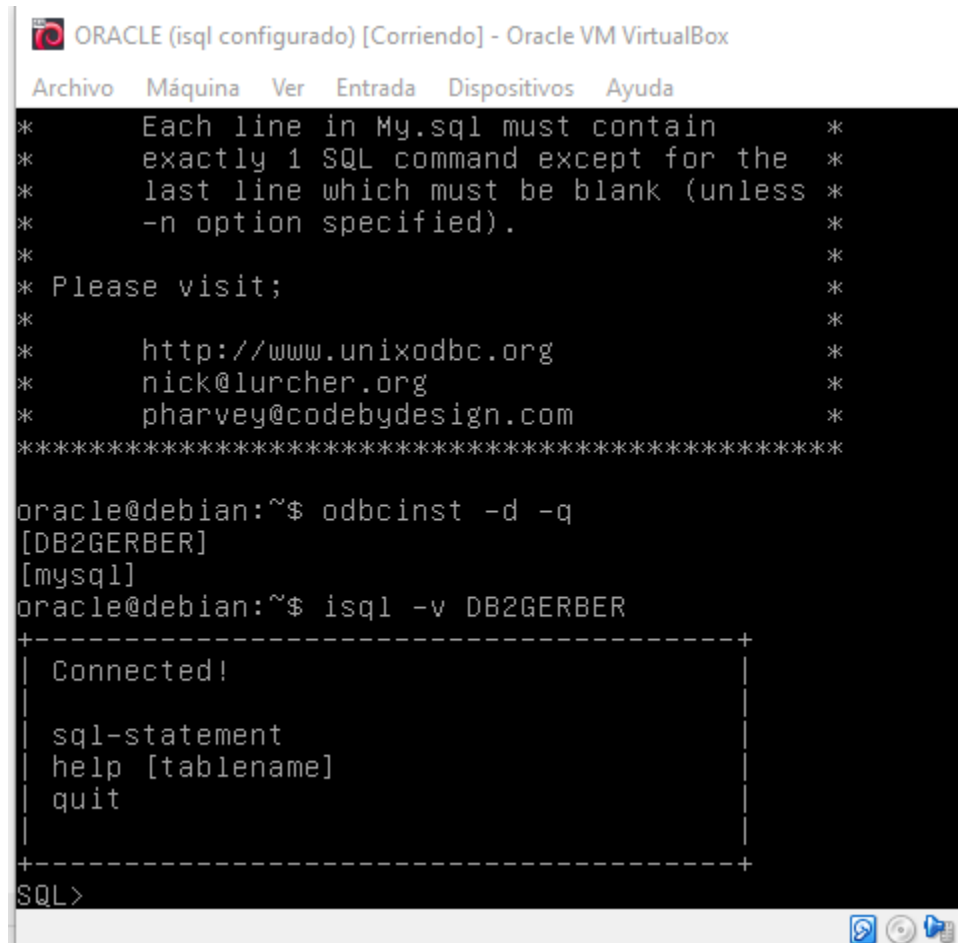
odbcinst -d -q



```
oracle@debian:~$ odbcinst -d -q
[DB2GERBER]
[mysql]
oracle@debian:~$ _
```

Nos conectamos a DBGERBER que en nuestro caso ya hemos configurado dicho DataSource.

Isql -v DB2GERBER



```
ORACLE (isql configurado) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
*      Each line in My.sql must contain      *
*      exactly 1 SQL command except for the  *
*      last line which must be blank (unless *
*      -n option specified).                  *
*
* Please visit;                              *
*
*      http://www.unixodbc.org                *
*      nick@lurcher.org                      *
*      pharvey@codebydesign.com               *
*****

oracle@debian:~$ odbcinst -d -q
[DB2GERBER]
[mysql]
oracle@debian:~$ isql -v DB2GERBER
+-----+
| Connected!
|
| sql-statement
| help [tablename]
| quit
|
+-----+
SQL>
```

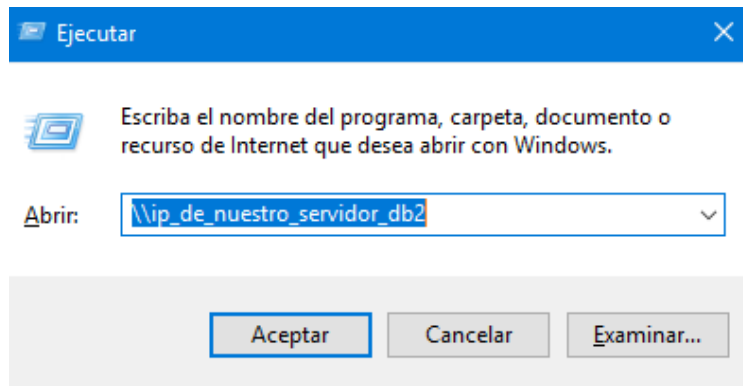
Verificamos que la conexión ha sido exitosa.

Proceso de configuración para leer tablas desde SQL Developer(Windows) a DB2

Debemos copiar el archivo db2jcc.jar a nuestro cliente Windows en este caso estaremos usando un servidor samba para enviar archivos de Linux hacia Windows y trabajaremos con el siguiente comando:

```
cp /opt/ibm/db2/V11.1/java/db2jcc.jar /var/samba/shares/public
```

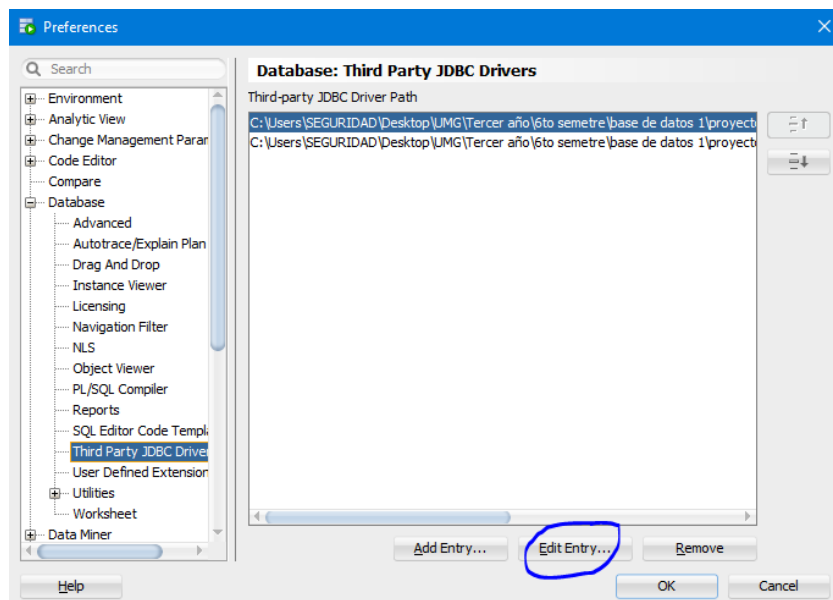
luego de ejecutar este comando ya se encuentra el archivo en la carpeta compartida la podremos acceder desde Windows presionando Windows + r y teclado la ip de nuestro servidor donde se encuentra db2



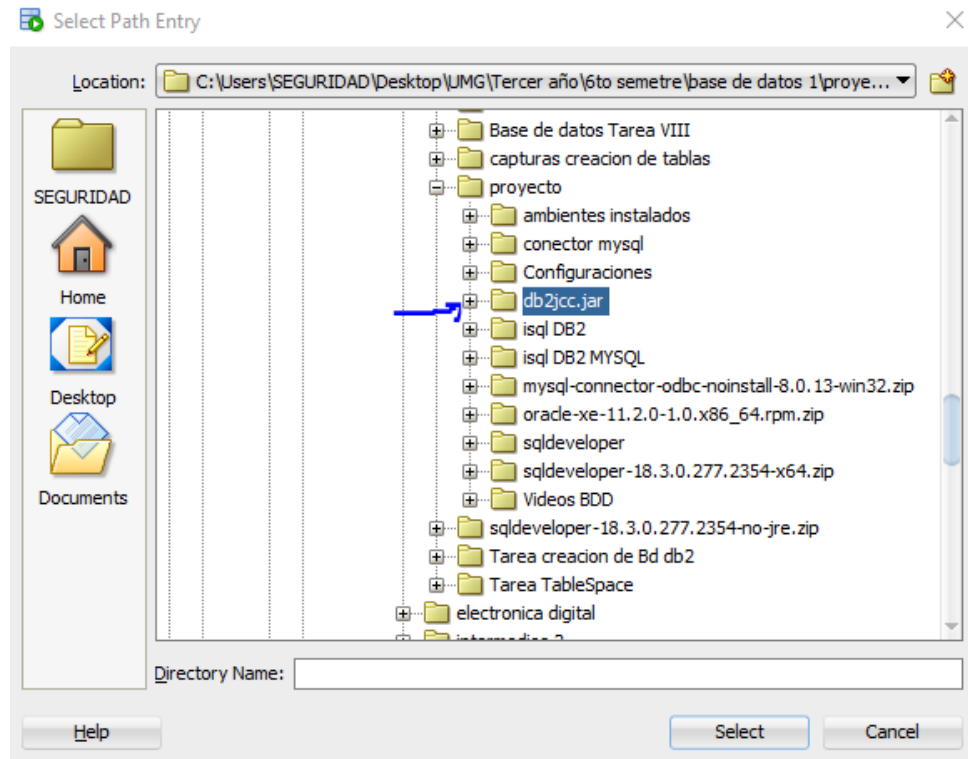
Podremos copiar nuestro archivo a cualquier directorio en nuestro caso lo copiaremos al escritorio. Ejecutamos nuestra aplicación de sql developer

Seleccionamos la opción

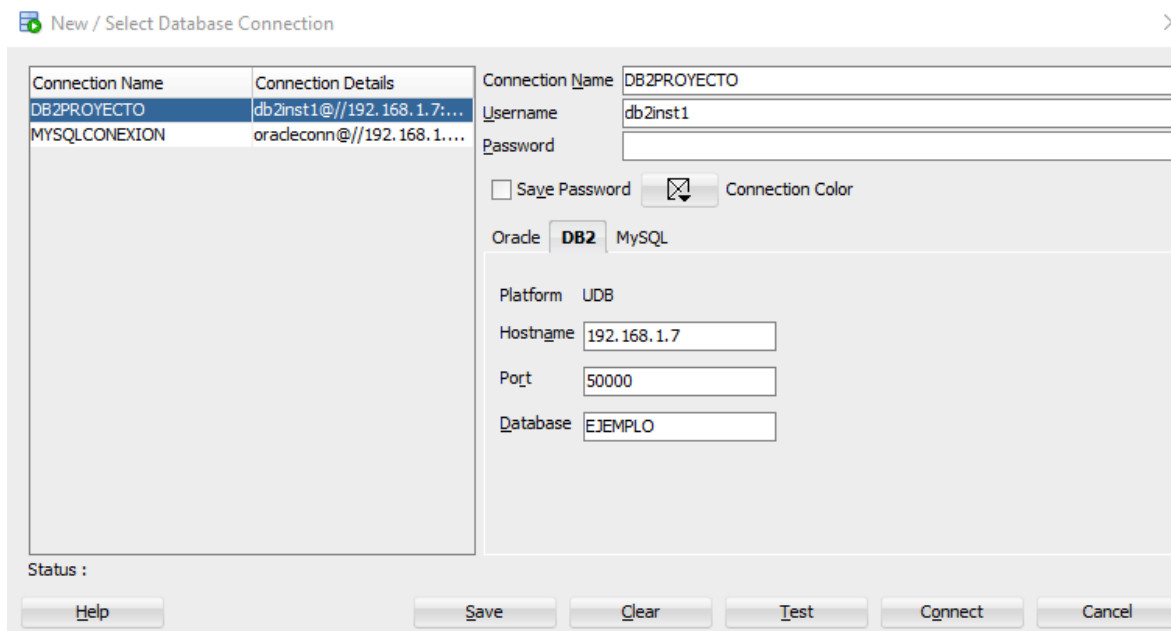
-tools -preferences -Database -Third Party JDBC Drivers -Add entry



Buscamos el directorio donde fue almacenado nuestro archivo db2jcc.jar lo seleccionamos y damos ok



Luego nos dirigimos a la opción de conexiones y agregamos una nueva conexión. Ya aparecerá DB2 y agregamos los respectivos parámetros para conectarnos a nuestras tablas de db2.



En nuestro caso la conexión y los parámetros serán los siguientes:

Connection name: DB2PROYECTO

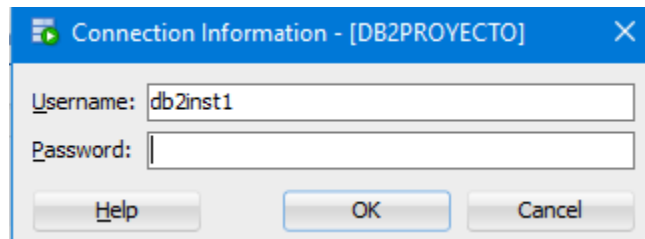
Username: db2inst1

Password: db2inst1

Hostname: ip_del_servidor

Port: 50000

Guardamos los cambios y vemos que hemos realizado la conexión con éxito.



Y verificamos que ya nos solicita password para ingresar a nuestras tablas.

Proceso de configuracion para conectar SQL developer con MySql

Descargamos el jdbc de mysql desde la página oficial

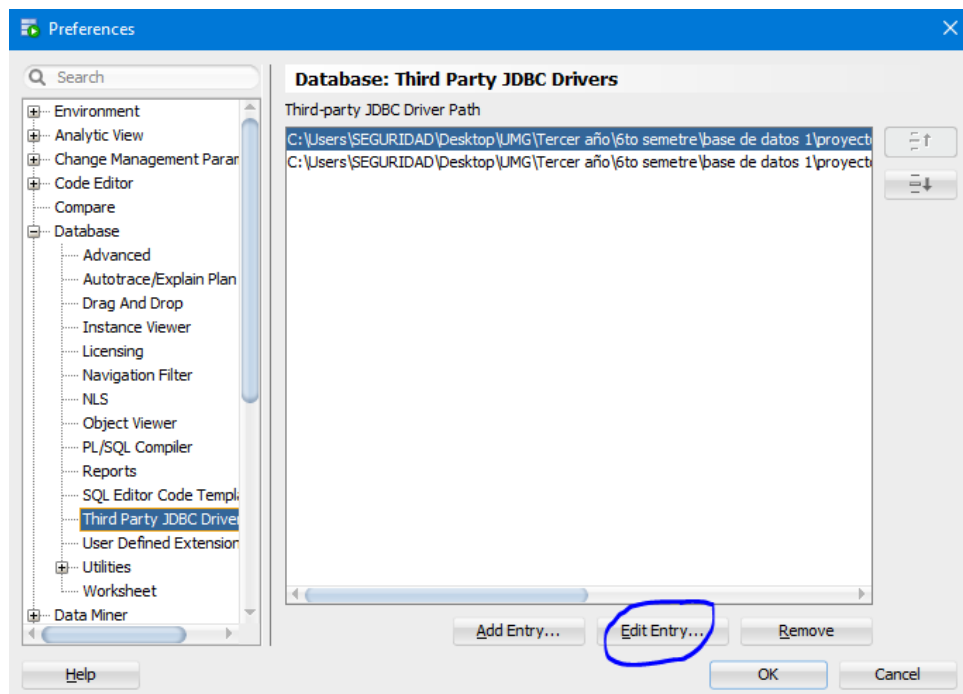
<https://www.mysql.com/products/connector/>

Developed by MySQL	
ADO.NET Driver for MySQL (Connector/.NET)	Download
ODBC Driver for MySQL (Connector/ODBC)	Download
JDBC Driver for MySQL (Connector/J)	Download
Node.js Driver for MySQL (Connector/Node.js)	Download
Python Driver for MySQL (Connector/Python)	Download
C++ Driver for MySQL (Connector/C++)	Download
C Driver for MySQL (Connector/C)	Download
C API for MySQL (mysqlclient)	Download

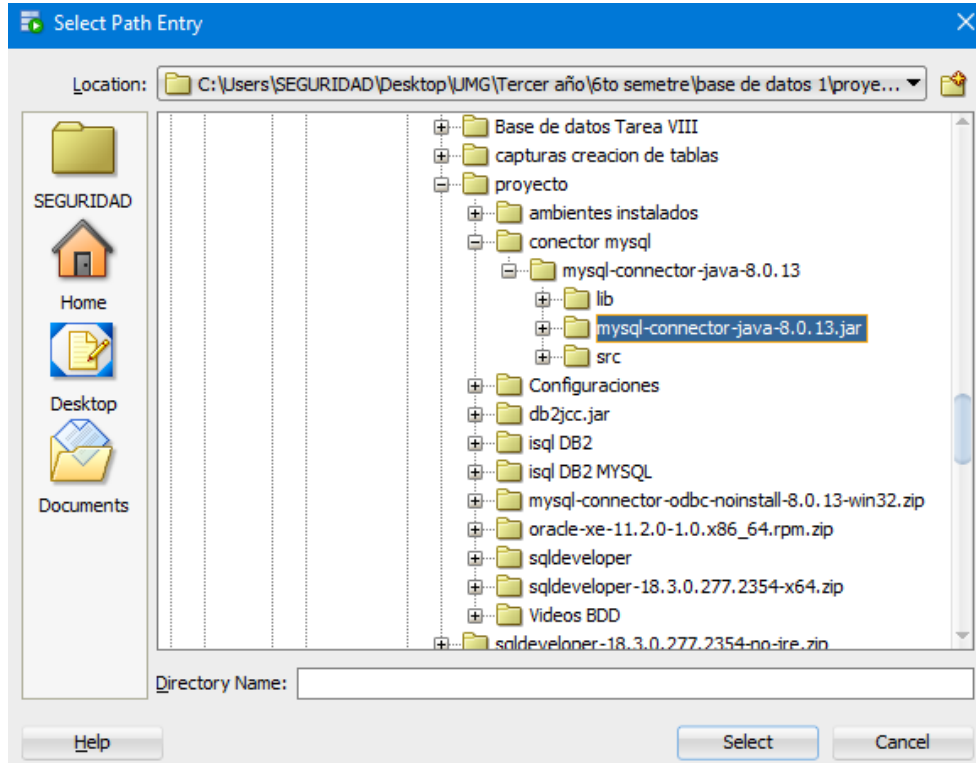
Podremos copiar nuestro archivo a cualquier directorio en nuestro caso lo copiaremos al escritorio. Ejecutamos nuestra aplicación de sql developer

Seleccionamos la opción

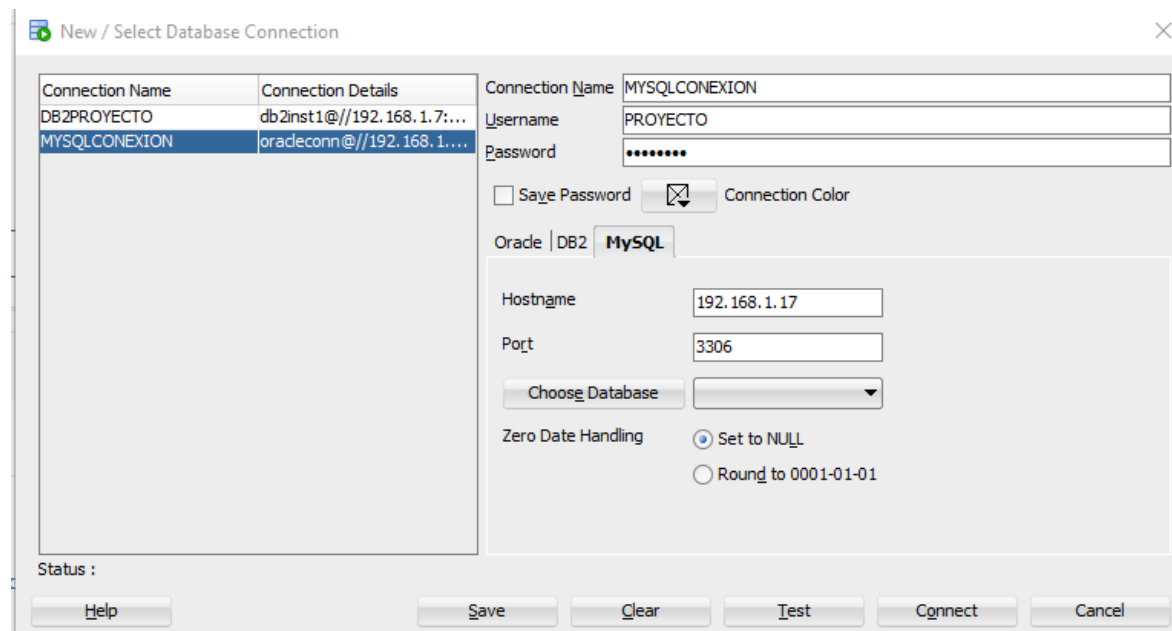
-tools -preferences -Database -Third Party JDBC Drivers -Add entry



Buscamos el directorio donde fue almacenado nuestro archivo db2jcc.jar lo seleccionamos y damos ok



Luego nos dirigimos a la opción de conexiones y agregamos una nueva conexión Ya aparecerá DB2 y agregamos los respectivos parámetros para conectarnos a nuestras tablas de db2.



En nuestro caso la conexión y los parámetros serán los siguientes:

Connection name: MYSQLCONEXION

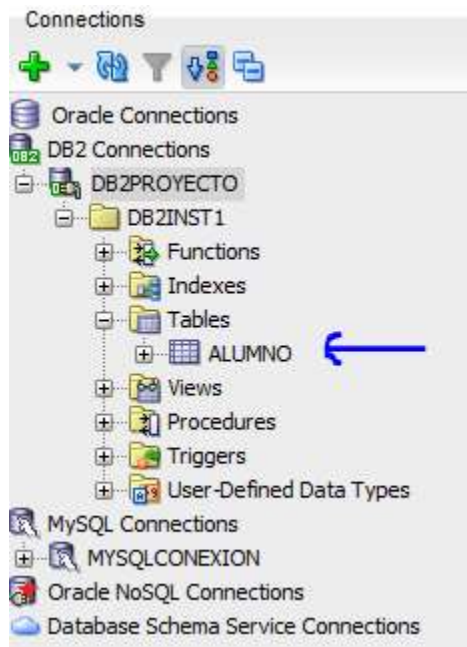
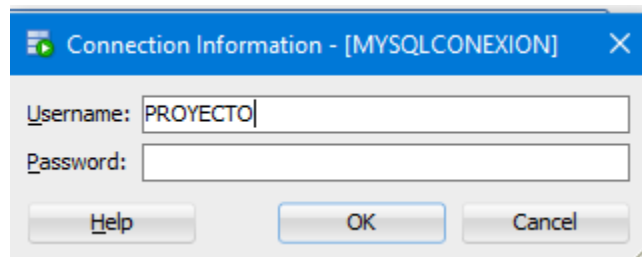
Username: PROYECTO

Password: PROYECTO

Hostname: ip_del_servidor

Port: 50000

Guardamos los cambios y vemos que hemos realizado la conexión con éxito ingresamos nuestra password y ya es posible visualizar nuestras tablas.



Conclusiones

1. Haber realizado este proyecto nos dio conocimientos fundamentales sobre la conectividad que se puede hacer entre diferentes gestores de base de datos utilizando sistemas operativos en modo consola, en este caso del proyecto el SO utilizado fue Debian.
2. Se puede destacar que los comandos para realizar el CRUD son bastante similares entre los gestores de Base de Datos por lo cual el tiempo de aprendizaje se reduce en el caso ya se tengan conocimientos generales.
3. Realmente fue una experiencia nueva para todos nosotros el realizar el proyecto debido que adicional de conocer el SO Linux, nos enseñó a perseverar, a ser extrovertidos al tener que preguntar en nuestro entorno a personas que conocen sobre el tema. Estas consultas también se realizaron en internet para poder tener suficiente información para no fallar.

Recomendaciones

1. Es necesario seguir practicando lo aprendido para reforzar el conocimiento general adquirido sobre como comunicar diferentes bases de datos usando por lo menos 2 sistemas operativos Linux y Windows.
2. Es importante reforzar y aprender los diferentes comandos adicionales que no se utilizaron en el desarrollo del proyecto y poder así dar un mejor soporte (si es nuestro trabajo).
3. Es vital seguir investigando sobre los diferentes comandos y gestores de bases de datos para reforzar nuestra experiencia de ingenieros.

Bibliografía

Se utilizaron los siguientes links (ligas) de apoyo:

1. <https://www.ibm.com/developerworks/ssa/downloads/im/udbexp/index.html>
2. <https://www.mysql.com/products/connector/>
3. El valioso aporte de el Ing. Edgar Civil con notas publicadas en su blog, así como respuestas brindadas por correo electrónico y clases magistrales.