

# Administrar usuarios y bases de datos

<b>Administrar usuarios y bases de datos</b>	<b>1</b>
¿Qué aprenderás?	2
Introducción	2
Algunas características de Oracle Database	3
Administración de usuarios en Oracle Database	4
Conexión a la base desde SqlPlus	4
Crear Usuario	5
Otorgar permisos a un usuario	6
Probando la conexión	7
Más instrucciones de administración	13



**¡Comencemos!**

## ¿Qué aprenderás?

- Crear usuarios y asignar permisos para administrar una base de datos.
- Crear bases de datos para almacenar información de forma organizada.

## Introducción

Si piensas en el funcionamiento de una biblioteca, para que una persona acceda a un libro, debe contar con algún tipo de credencial de socio. Para poder reservar ese libro y tenerlo en préstamo, no sólo requerirá de la credencial, sino también ser del grupo de usuarios que pueden pedir libros y además no contar con ninguna deuda.

Ese método de control es útil para resguardar la seguridad de los libros de la biblioteca y tener un control de quién accede a qué servicios.

De la misma manera, una de las primeras acciones que debemos realizar sobre nuestra base de datos es definir los permisos, es decir, qué usuario puede acceder a ella y qué acciones puede realizar.

A continuación veremos cómo administrar usuarios dentro de una base de datos, viendo qué permisos pueden tener al momento de enfrentarse a una base de datos.

**¡Vamos con todo!**



## Algunas características de Oracle Database

Todo aquello que escribamos de ahora en adelante en la consola o en el cliente, forma parte de las instrucciones que el intérprete de Oracle ejecutará, por lo tanto, debemos dejar claro **algunos** puntos con respecto a la sintaxis de Oracle:

- Marcamos el **término de una instrucción** con el carácter ";" (punto y coma). Así el intérprete sabrá que terminamos la instrucción, el no hacerlo implica que al ejecutar, se quede esperando a que terminemos la consulta o instrucción. Algunos clientes como el caso de DBeaver, lo completa internamente, pero de todas maneras siempre es recomendable incluir este carácter, a menos que lo ejecutemos más adelante desde un String de Java - jdbc, en ese caso no incluimos el punto y coma dentro del String.
- El intérprete de Oracle, no diferencia entre si escribimos con mayúsculas o minúsculas, incluso si las mezclamos, ya que no hace diferencia entre ellas, por lo tanto Oracle es **case insensitive**.
- Las instrucciones realizan infinidad de tareas, entre las que están algunas en las que debemos **asignar valores** a un nombre que inventemos, por ejemplo, al crear una tabla, ponemos su nombre o incluso un alias. Dada esta necesidad (entre otras), debemos saber que para estos fines, **no podemos utilizar palabras reservadas por el sistema**, entre las que están: **WHERE, JOIN, CREATE, etc.** Para poder consultar la lista completa de las palabras reservadas por Oracle, ir al [siguiente enlace](#).

Nos adelantamos un momento para un ejemplo de creación de una tabla. (Luego veremos en detalle cómo trabaja esta instrucción). Vemos que no podemos asignar el nombre de una tabla a una palabra reservada, en este caso es join.

```
-- No cumple
CREATE TABLE JOIN();

-- Cumple
CREATE TABLE PERSONA();
```

## Administración de usuarios en Oracle Database

A la hora de trabajar, más de una persona ocupa la información de una misma base de datos. Oracle Database da la posibilidad de crear usuarios mediante la definición del rol administrador de base de datos, quien tenga este rol será el encargado de asignar roles, los que van a definir el rango de acciones posibles para usuarios normales; así como la creación/modificación/eliminación de tablas, revocación de roles/usuarios por nombrar algunas de las funcionalidades, así sólo los usuarios registrados van a poder acceder a la base de datos.

### Conexión a la base desde SqlPlus

Ingresamos a SqlPlus:

```
SQL*Plus: Release 11.2.0.1.0 Production on Jue Sep 10 13:31:08 2020  
  
Copyright (c) 1982, 2010, Oracle. All rights reserved.  
  
Enter user-name:
```

Ingresamos las credenciales que configuramos, puede que en los equipos que se estén utilizando, posean sus propias credenciales, en caso que no conecte se deberá solicitar asistencia al personal de soporte. Para nuestro caso las credenciales son:

```
username: system  
password: admin
```

Lo que debe entregar el siguiente resultado:

```
SQL*Plus: Release 11.2.0.1.0 Production on Jue Sep 10 13:31:08 2020  
  
Copyright (c) 1982, 2010, Oracle. All rights reserved.  
  
Enter user-name: system  
Enter password:  
  
Connected to:  
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit  
Production  
With the Partitioning, OLAP, Data Mining and Real Application Testing
```

```
options
```

```
SQL>
```

## Crear Usuario

En esta versión de Oracle, al momento de crear un usuario, estaremos creando un esquema. Esto significa que **este usuario tendrá una base de datos con el mismo nombre asignada** para luego poder conectarnos y comenzar a gestionar datos por medio del lenguaje SQL.

El usuario **system**, es el usuario administrador del sistema, es el que viene por defecto, entonces no deberíamos usarlo para almacenar datos productivos, más bien esta es la cuenta encargada de la administración, es decir, posee tareas como la administración de usuario o asignación de permisos entre otras muchas tareas que cumple un DBA. Nosotros no profundizaremos en tareas administrativas, ya que nuestra meta es poder comprender el lenguaje de consultas SQL aplicado al desarrollo web, utilizando una de las tantas herramientas, como es el caso de Oracle Database u Oracle Sql; pero SQL se aplica a muchos otros motores con pequeñas variaciones en su sintaxis. Veremos entonces las instrucciones necesarias para cumplir con nuestra tarea y no nos complicamos con configuraciones profundas que tengan que ver con la administración.

A continuación crearemos nuestro usuario el cual podremos utilizar y para eso debemos saber lo siguiente.

Sintaxis de creación de usuario. `${text}` = se refiere al texto que se reemplaza.

```
CREATE USER ${nombre_usuario} IDENTIFIED BY ${contrasena};
```

Para el usuario alumno y el password 1234:

```
SQL> CREATE USER alumno IDENTIFIED BY 1234;
```

```
User created.
```

## Otorgar permisos a un usuario

Podemos otorgar permisos específicos a un usuario, en nuestro caso, nos queman las manos para comenzar la administración de datos, por lo tanto procedemos a asignar todos los permisos a nuestro usuario alumno. Para profundizar en todos los privilegios que posee el motor Oracle, pude visitar el siguiente [enlace](#).

Para esta tarea utilizamos la palabra `GRANT`.

Sintaxis:

```
GRANT ALL PRIVILEGES TO ${usuario};
```

```
SQL> GRANT ALL PRIVILEGES TO alumno;
```

```
Grant succeeded.
```

Detengámonos un poco en esta parte, acabamos de crear un usuario, además le asignamos todos los permisos, llegó la hora de probarlo, conectándonos desde SqlPlus y los dos clientes que descargamos, luego, cuál utilizaremos para trabajar será opcional, y en cualquiera de las tres herramientas, podemos hacer lo mismo. Según mi opinión y basado en mis años de programación, la más usable para nuestro caso es DBeaver, ya que al ser una herramienta más nueva, posee novedosas características basadas en la experiencia de usuario, existen otras mejores como es el caso por ejemplo de [navicat](#); pero su precio desmotiva las ganas de utilizarlo, no es el caso de DBeaver, ya que no solo nos sirve para Oracle, es compatible con muchos otros, pero todo lo anterior es completamente a gusto personal.

## Probando la conexión

En primer lugar probamos la conexión con nuestro nuevo usuario en SqlPlus:

Con el comando `disconnect`, nos desconectamos, para luego conectarnos con `connect alumno`, es decir `connect ${nombre usuario}`. Ingresamos la contraseña, entonces podremos ver que la conexión tuvo éxito.

```
SQL> disconnect;
Disconnected from Oracle Database 11g Enterprise Edition Release
11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing
options
SQL> connect alumno
Enter password:
Connected.
```

Ahora probemos si es que tendremos el mismo resultado desde Sql Developer.

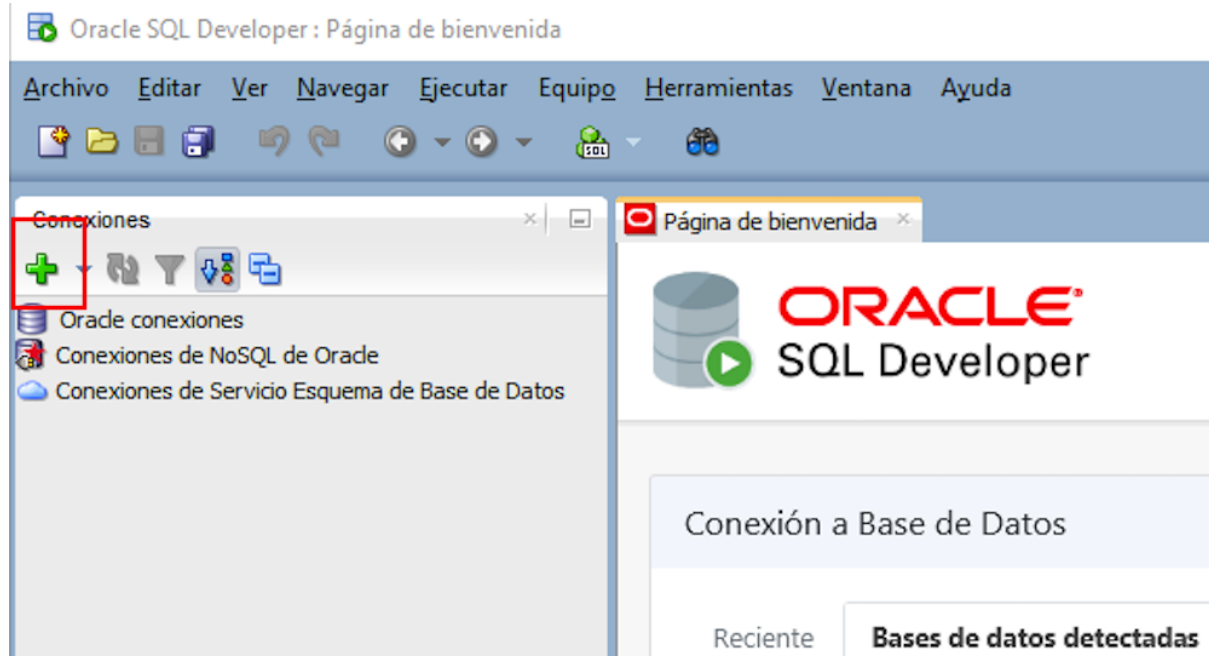


Imagen 1. Agregar nueva conexión Sql Developer.

Esto nos abre una nueva ventana con un formulario que utilizaremos para conectarnos. Los parámetros que utilizaremos serán los siguientes:

- **Name:** Base de datos Alumnos
  - Es el nombre que le daremos a nuestra base de datos para identificarla con un lenguaje más usable para una persona.
- **Usuario:** alumno
  - Es el usuario que creamos por sqlplus.
- **Contraseña:** 1234
  - Es la contraseña que indicamos cuando creamos al usuario alumno en sqlplus.
- **Nombre del host:** localhost
  - Debido a que el servidor de base de datos, que es lo mismo que decir Oracle Database, decir localhost, se refiere a que se encuentra en nuestra computadora o servidor.
- **Puerto:** 1521
  - Tenemos aproximadamente 65 mil puertos disponibles para establecer distintas conexiones a distintos servidores, por ejemplo el puerto 21 que hace referencia al protocolo ftp, para la transferencia de archivos o el 80 que es para http. Oracle Database, se reserva este puerto para exponer su conexión, obviamente esto es configurable, pero el puerto que tenemos por defecto es el 1521.
- **Sid:** ORCLE
  - Dependiendo del producto se indica este nombre, esta versión tiene el sid igual a ORCLE, como se puede ver en el resumen de la instalación. También hay otros productos que tienen el sid como xe; pero este no es el caso.

Es posible además indicar el color, esto es porque en nuestra vida profesional, tendremos bases de datos productivas, las cuales son muy sensibles a cambios, y esas bases podrían llevar un color rojo, y las de desarrollo, un color verde, pero esto es completamente opcional.



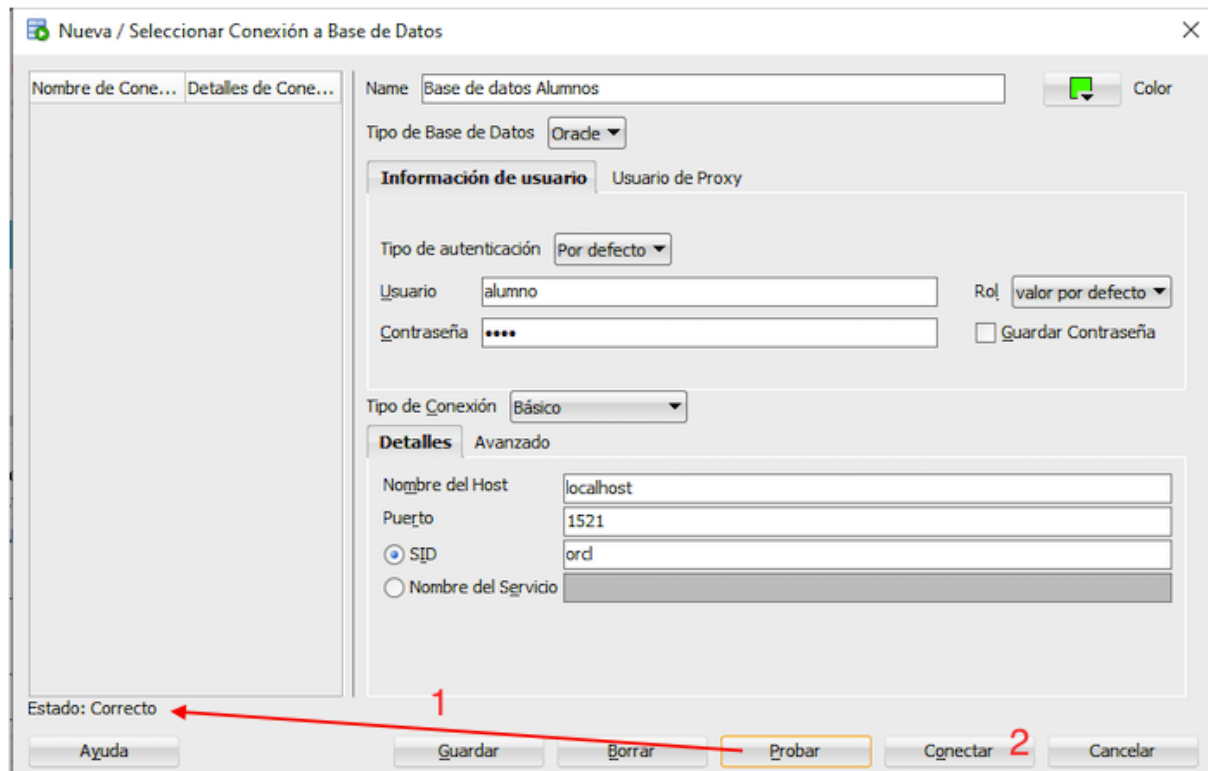


Imagen 2. Conectar desde Sql Developer.

Podemos ver que luego de pedirnos la contraseña una vez más, ya tenemos la conexión lista, además de un editor en blanco esperando nuestras instrucciones. Se puede ver también que el verde parece que no ha sido una gran elección para esta base.

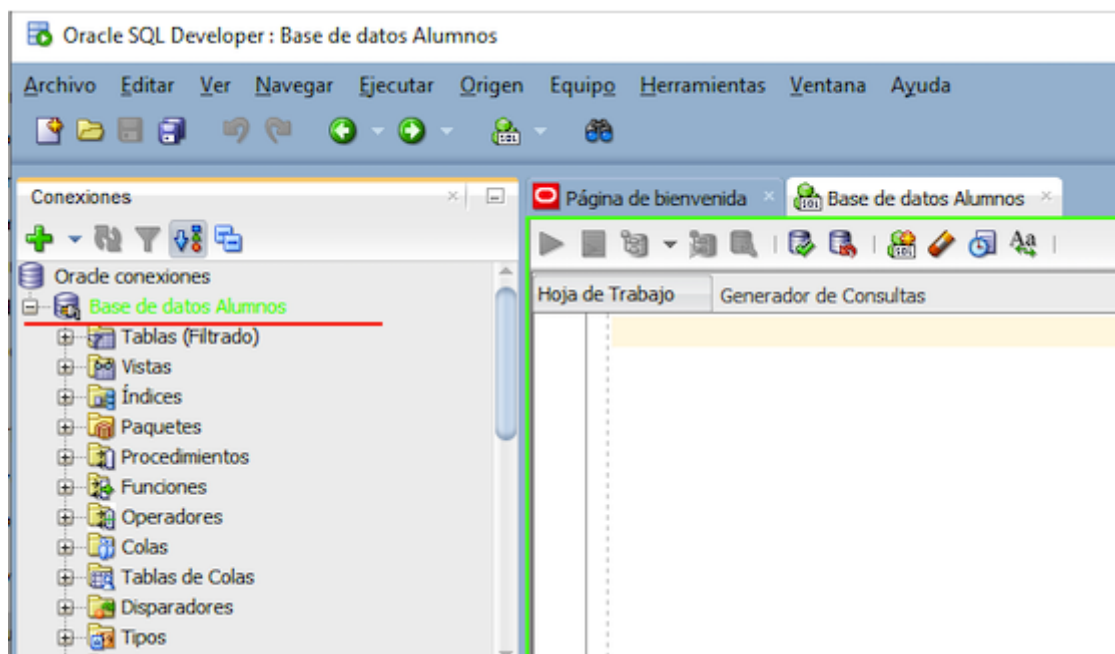


Imagen 3. base de datos conectada con Sql Developer.

Ya hemos tenido éxito conectándonos en las dos últimas herramientas mencionadas, nos queda probar finalmente la conexión desde DBeaver.

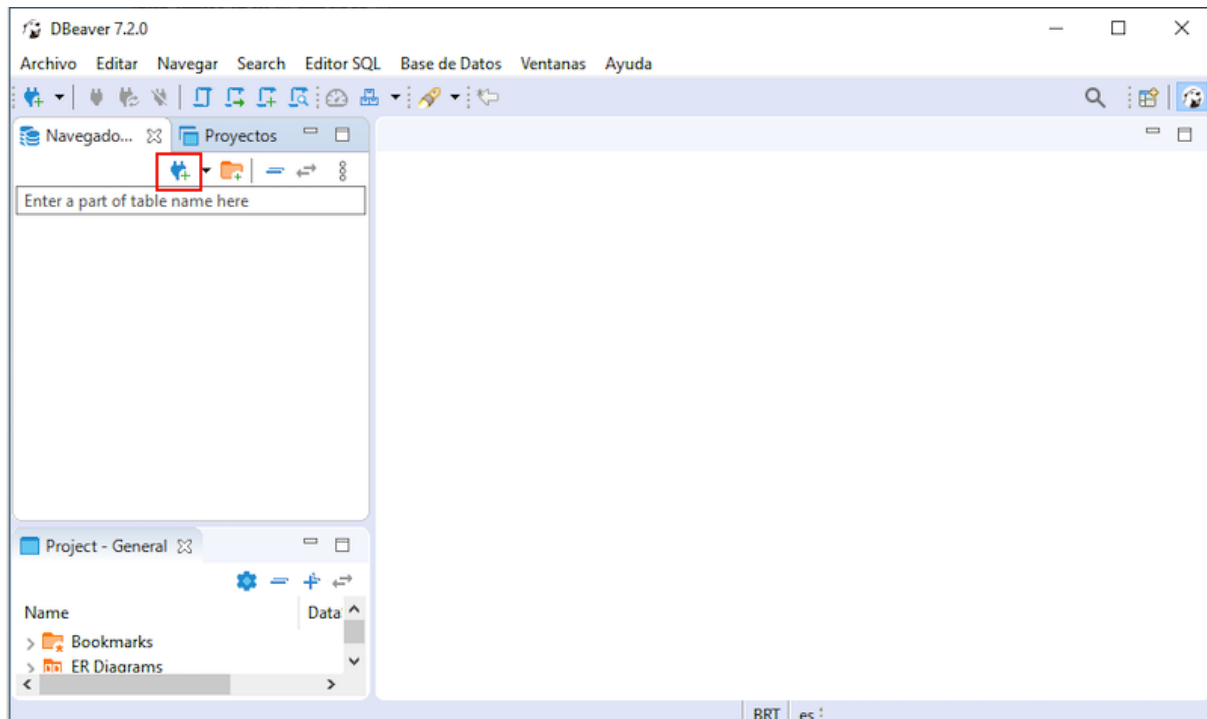


Imagen 4. Nueva conexión con DBeaver.

Esto nos abre una nueva ventana con opciones de conexión a distintos motores. Adivinaron, seleccionamos la que dice Oracle y presionamos siguiente.

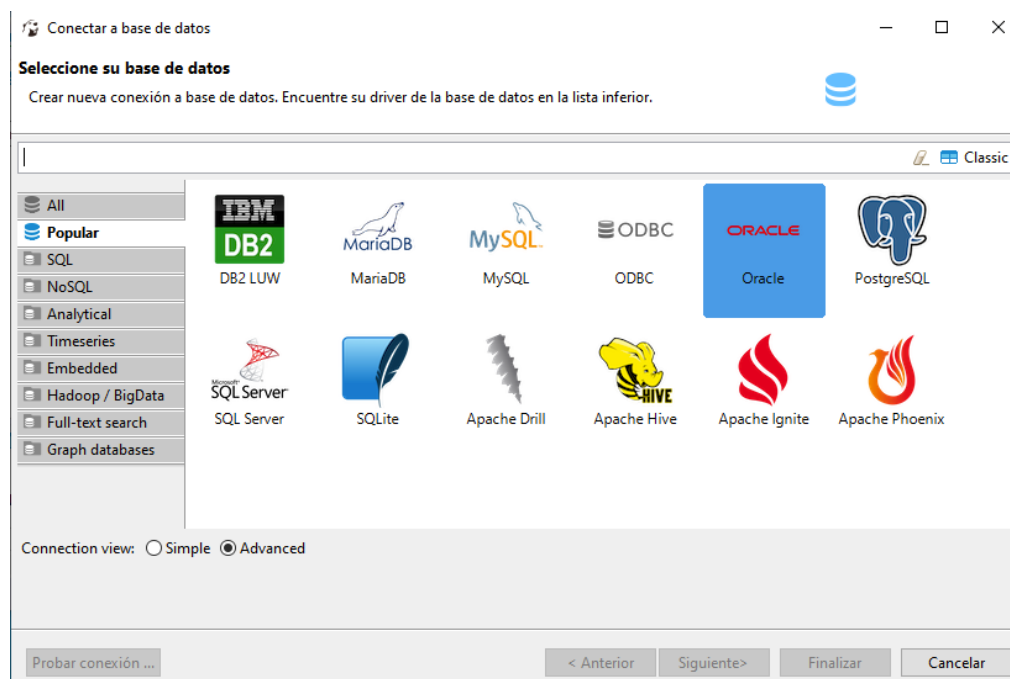


Imagen 5. Opciones de conexión.

Esto nos conduce a una nueva ventana. Lo que sucede es que DBeaver es un software libre y no forma parte de su responsabilidad, dar soporte a los drivers de conexión que otorgan las distintas compañías como es el caso de Oracle, es por ello, que se nos presenta esta interfaz que hará todo esto por nosotros. Solo los limitamos a confirmar la descarga.

Lo primero es ver que tenemos los drivers listos para la descarga.

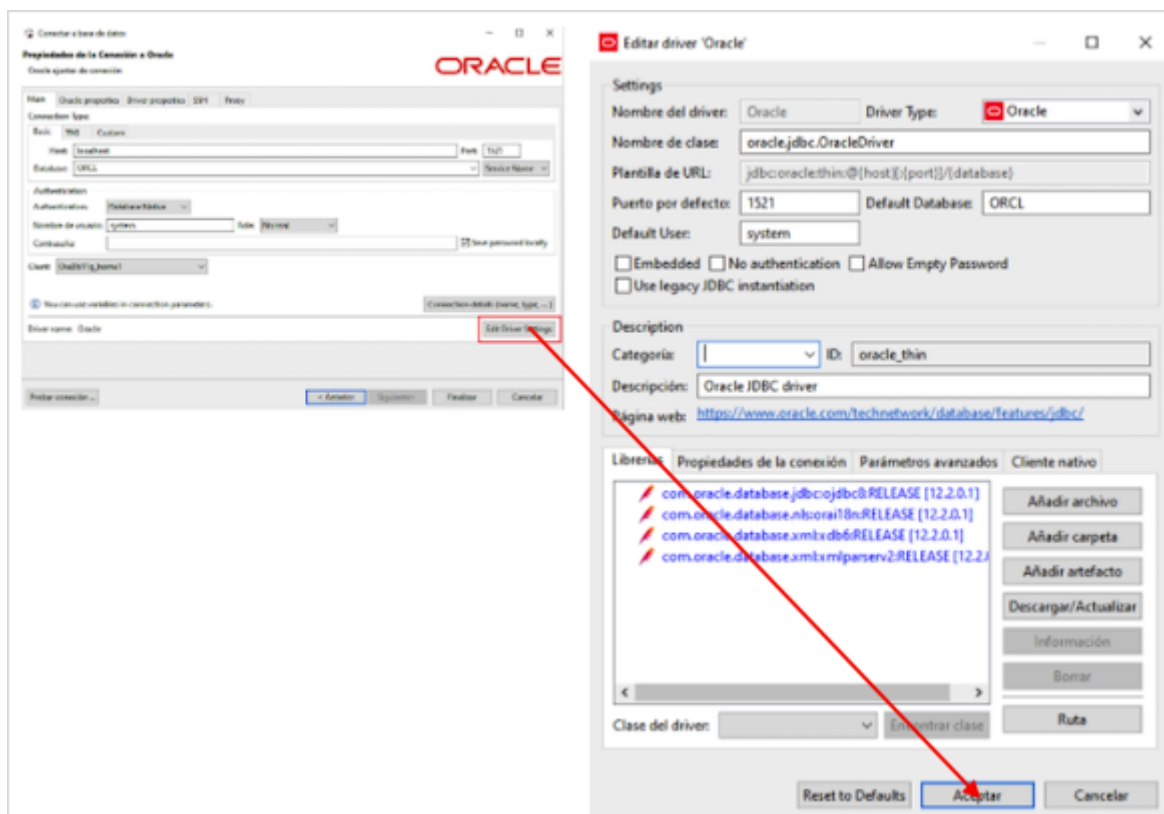


Imagen 6. Revisión de drivers.

Luego llenamos correctamente los datos para la conexión, podemos ver que está casi todo listo, solamente ingresamos el usuario y la contraseña, **además de cambiar el service name por SID** luego presionamos probar conexión.

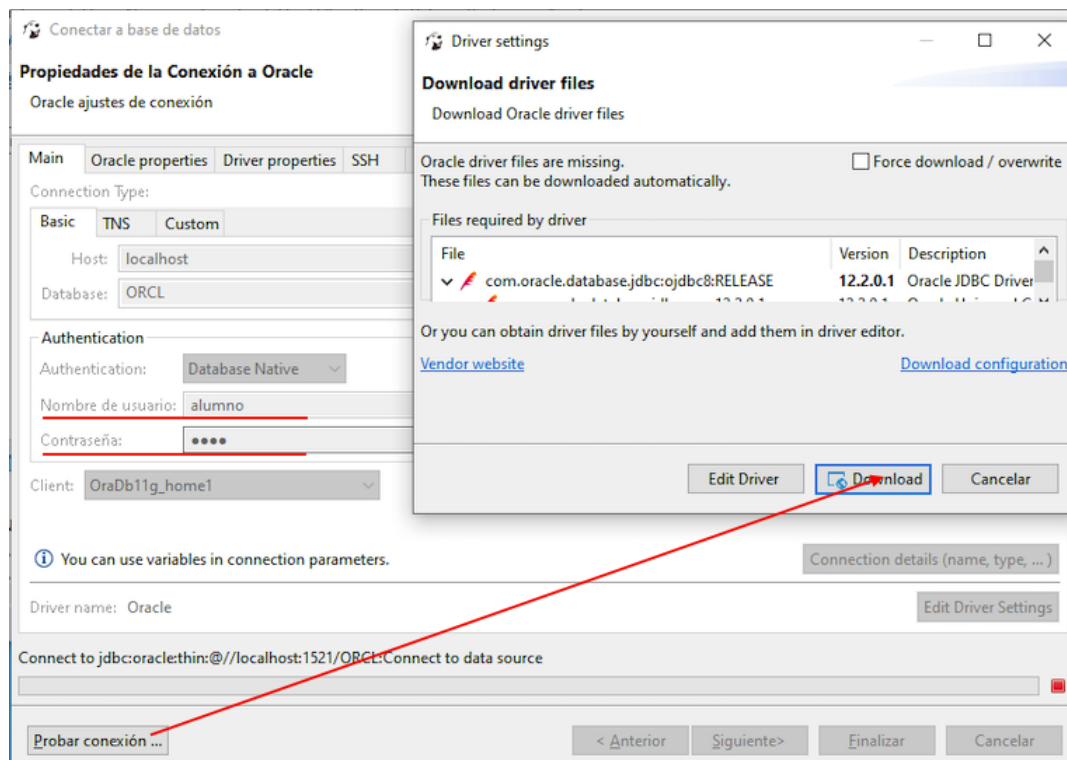


Imagen 7. Confirmación de la descarga (se realiza solamente una vez).

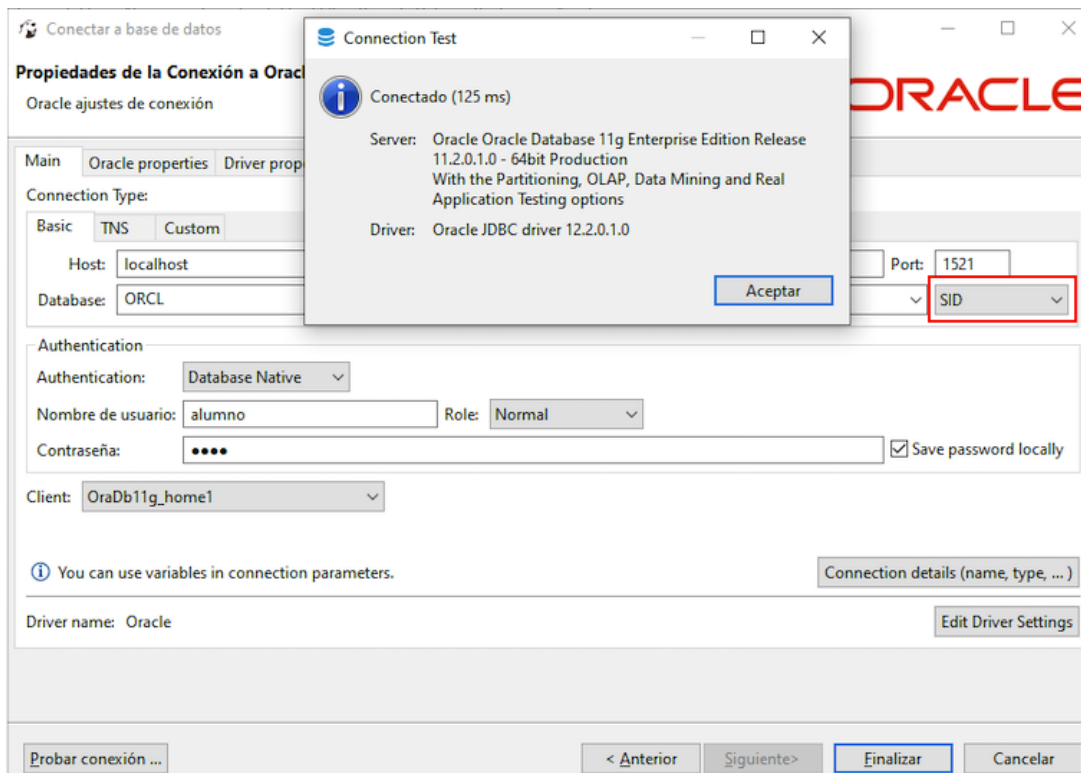


Imagen 8. Finaliza la descarga y prueba exitosa.

Luego, presionamos finalizar y podremos ver nuestra base de datos conectada y lista para poder ser utilizada. Desde ahora, todos los ejemplos que requieran ejecutarse en un cliente, serán realizados con Dbeaver dada su versatilidad. Hay que tener en cuenta, que las instrucciones Sql, ejecutan las mismas órdenes en cualquiera de los clientes.

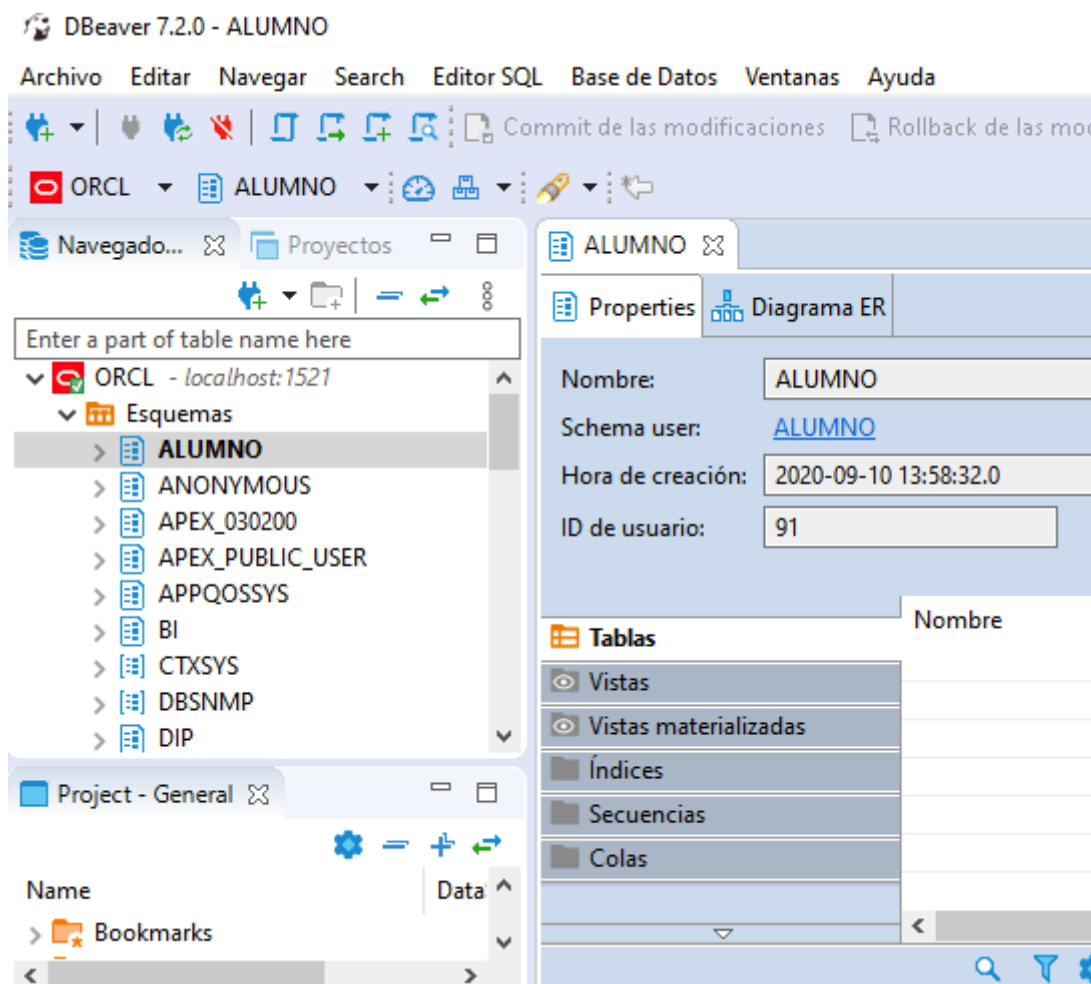


Imagen 9. Conexión de base de datos desde DBeaver.

## Más instrucciones de administración

Antes de partir de lleno con la creación de la estructura de nuestra base de datos, veamos qué otras cosas interesantes podemos gestionar con nuestro usuario administrador.

Identificar el usuario que está conectado.

```
SQL> select user from dual;
```

```
USER
-----
SYSTEM
```

Eliminar un usuario, para efectos de ejemplificar, creamos una base y la eliminamos.

```
SQL> create user nueva_base identified by foo;

User created.

SQL> drop user nueva_base cascade;

User dropped.
```

Seleccionar los usuarios del sistema, podemos notar que hay varios que no hemos creado nosotros, pero Oracle sí. Notamos también que entre ellos se encuentra la que creamos, me refiero a ALUMNO o alumno, que como ya sabemos es lo mismo.

```
SQL> select username from dba_users;

USERNAME
-----
MGMT_VIEW
SYS
SYSTEM
DBSNMP
SYSMAN
ALUMNO
.
.
.
```

Por ahora esto es suficiente respecto a lo que necesitamos de la gestión de las bases, para profundizar sobre la gestión de usuarios y otras funcionalidades de Oracle, recomiendo el siguiente [enlace](#).