

BASES DE DATOS

P
R
Á
C
T
I
C
A
S

Recursos para BBDD



Práctica 1: Instalación SQL Developer

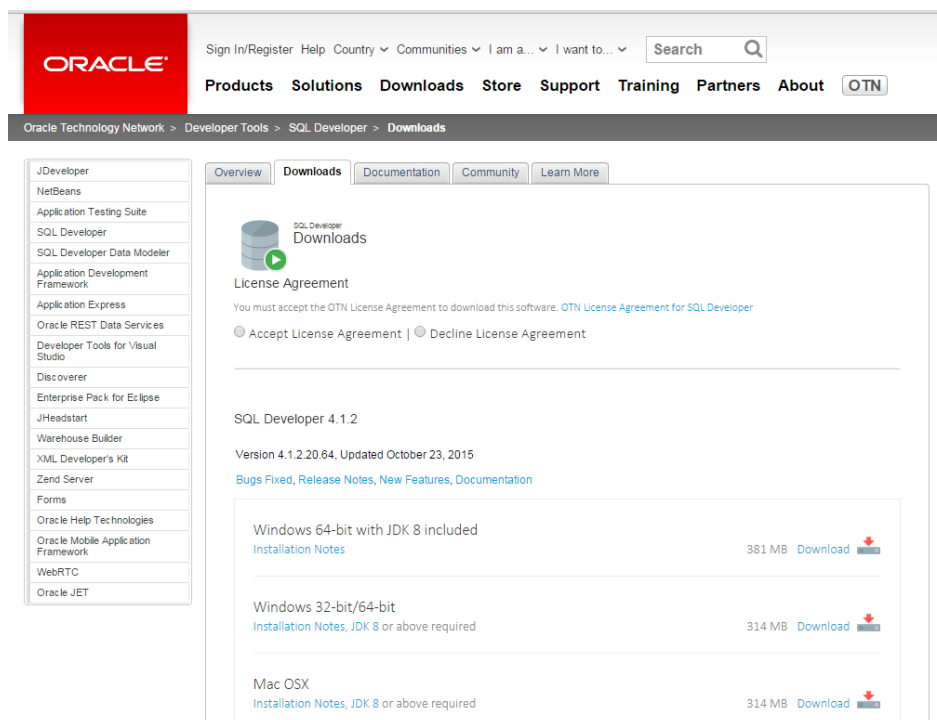
SQL Developer es una herramienta gráfica de Oracle que simplifica las tareas de desarrollo para bases de datos. Con ella podemos navegar, editar y crear objetos de base de datos Oracle, ejecutar sentencias SQL, editar y depurar PL SQL, ejecutar informes, etc. Además también permite crear modelos de datos, generar los DDL correspondientes, realizar modelos relacionales e ingeniería entre el modelo lógico y el relacional.

Aunque está pensada para trabajar con Oracle, también podemos crear conexiones y trabajar con otras bases de datos como MySQL.

INSTALACIÓN

Descargamos el archivo comprimido desde la página de descargas de Oracle:

<http://www.oracle.com/technetwork/developer-tools/sql-developer/downloads/index.html>

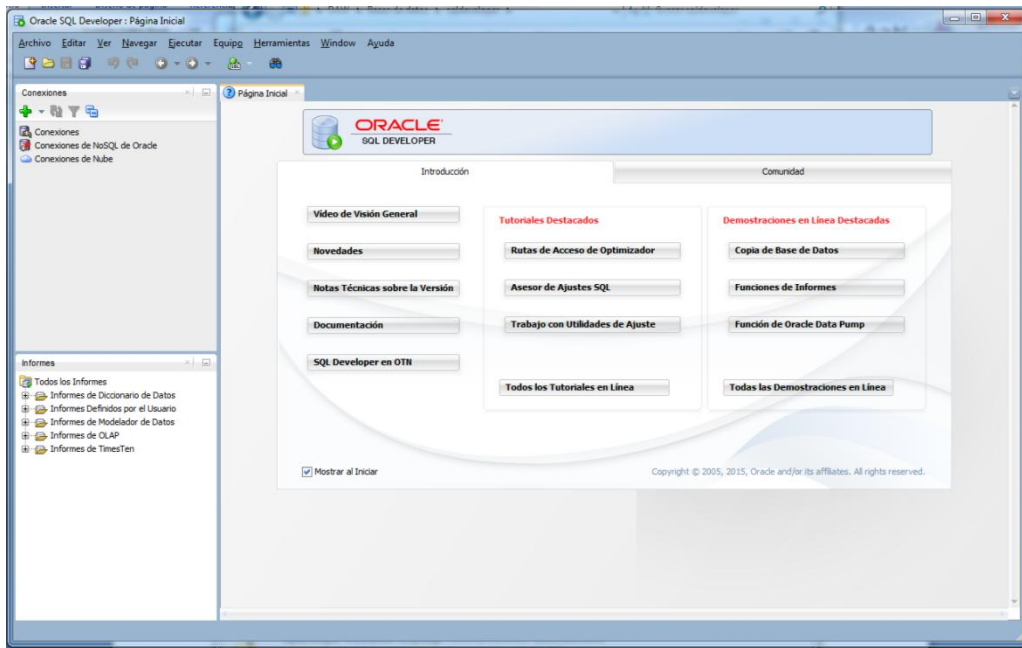


Aceptamos los términos de la licencia y elegimos la versión adecuada a nuestro sistema operativo. Introducimos el mail y la contraseña con la que nos hemos registrado en Oracle y después ya podremos realizar la descarga.

Se descarga un archivo comprimido **sqldeveloper-xxxx.zip**. No se instala, si no que al descomprimir ya tenemos en la carpeta **sqldeveloper** un archivo ejecutable denominado **sqldeveloper.exe**.

Para poder utilizarlo debemos tener el **jdk** de Java. Si ya lo tenemos instalado nos pedirá la ruta en la que se encuentra y si no lo tenemos, podemos elegir una de las versiones que lo traen incluido.


Una vez finalizado todo el proceso podemos ver su pantalla principal:



Práctica2: Crear conexiones a la BD oracle desde SQL Developer

Vamos a conectar a la base de datos PRUEBAS que hemos creado en oracle 11g a través del usuario alumno.

Abrimos SQL Developer y vemos el cuadro **Conexiones** a la izquierda.

Para crear una conexión hacemos clic en **Nueva conexión...** en el botón  del panel de conexiones o en **Archivo/Nuevo/Conexión a Base de Datos**.

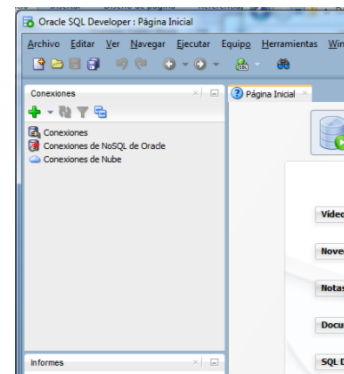
Se nos muestra un cuadro de datos de la conexión que hemos de completar (puede tardar en abrirse).

Nombre de la conexión: será el nombre que se verá en el panel conexiones (podemos tener varias).

Usuario: el usuario que va a utilizar esta conexión (ALUMNO01 en nuestro caso)

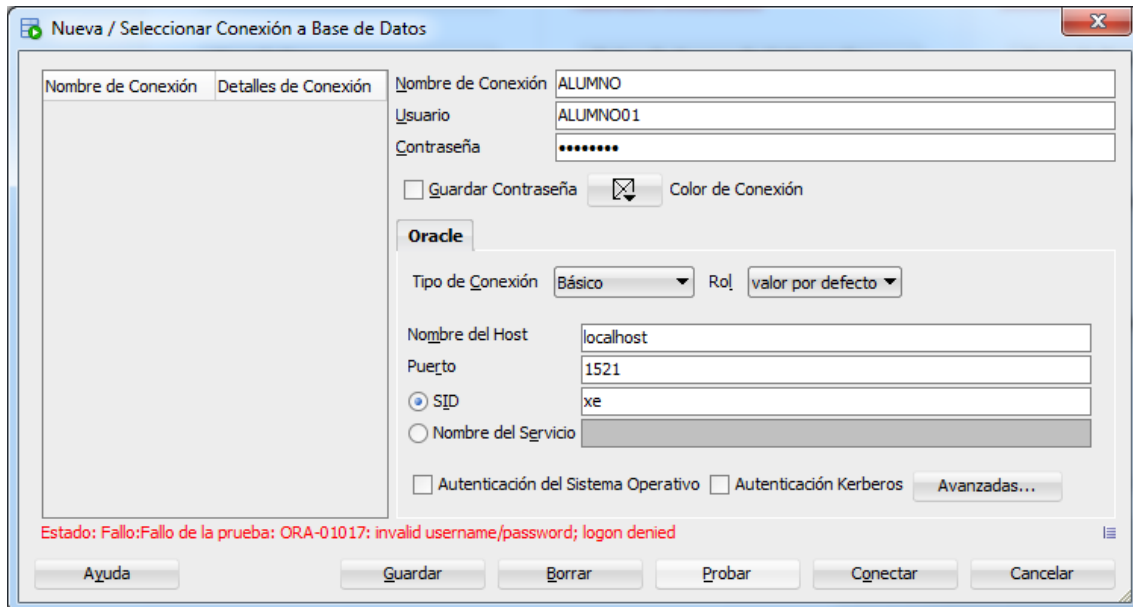
Contraseña: la contraseña del usuario (también alumno01 en nuestro caso)

Tipo de conexión: dejamos todas las opciones por defecto.



Si fuésemos a crear una conexión con el administrador SYS, en **Rol** hay que seleccionar SYSDBA. Además hay que conocer el **puerto** y el **SID** (nombre de la instancia). Las versiones exprés de oracle crean una única instancia a la que llaman XE. Todos estos datos los hemos visto al final de la instalación de la BD.

Los datos quedarían como en la imagen siguiente. Después hacemos clic en **Probar** para ver si la conexión funciona, y vemos que da error:



Esto sucede porque primero hemos de activar el usuario y darle permisos de acceso. Para ello:

- Vamos a menú **Inicio/Oracle Database 11g Express Edition**, y abrimos el elemento **Run SQL Command Line**.
- Conectamos (si no estamos ya) como **system** e introducimos nuestra contraseña (**oracle**)
- Escribimos:

```
SQL> CREATE USER ALUMNO01  
2 IDENTIFIED BY alumno01  
3 DEFAULT TABLESPACE USERS;
```

Se especifica el espacio de tabla USERS, si se omitiera, los objetos creados por el usuario ALUMNO01, se almacenarán por defecto en el tablespace System que es el lugar donde se encuentra almacenado el diccionario de datos.

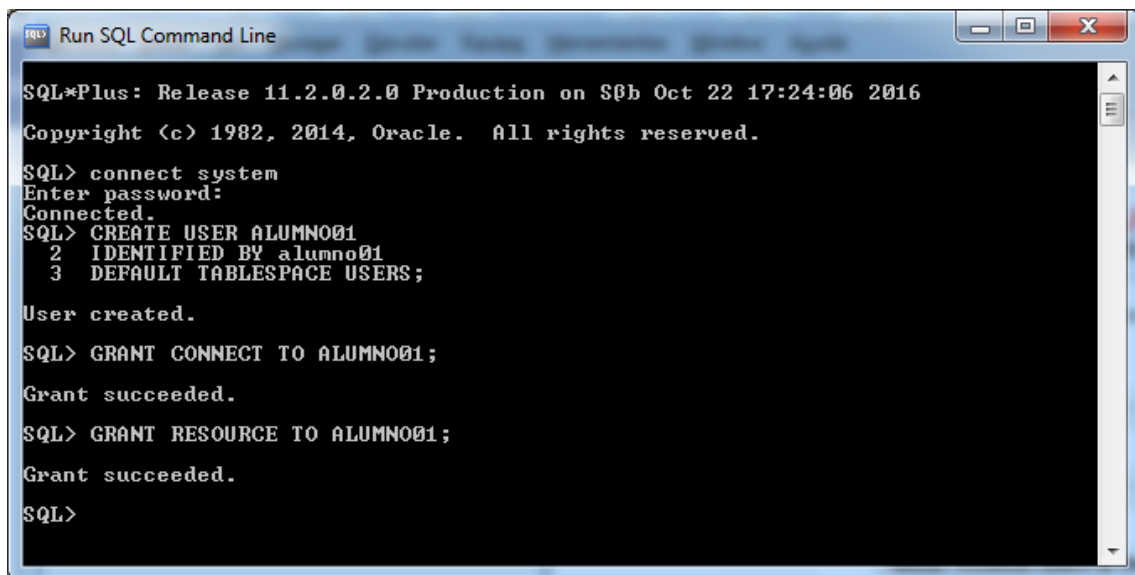
Ahora le daremos privilegios que le permitan conectarse al servidor Oracle. Para ello, introducimos la sentencia:

```
SQL> GRANT CONNECT TO ALUMNO01;
```

Ahora, ALUMNO01 puede conectarse a la instancia Oracle, pero no podrá crear ningún tipo de objetos como tablas, índices, vistas, unidades almacenadas o procedimientos almacenados. Para ello, introducimos la sentencia:

```
SQL> GRANT RESOURCE TO ALUMNO01;
```

Podemos ver el proceso completo en la siguiente imagen:



```
SQL*Plus: Release 11.2.0.2.0 Production on 08b Oct 22 17:24:06 2016
Copyright (c) 1982, 2014, Oracle. All rights reserved.

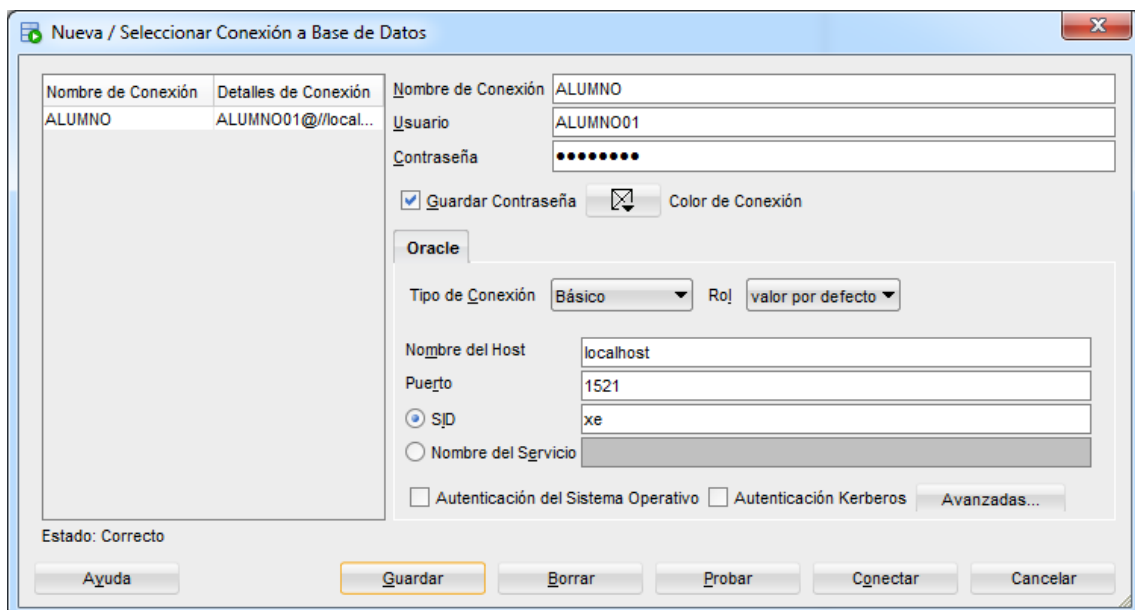
SQL> connect system
Enter password:
Connected.
SQL> CREATE USER ALUMNO01
2 IDENTIFIED BY alumno01
3 DEFAULT TABLESPACE USERS;

User created.
SQL> GRANT CONNECT TO ALUMNO01;

Grant succeeded.
SQL> GRANT RESOURCE TO ALUMNO01;

Grant succeeded.
SQL>
```

Ahora sí podremos crear la conexión en SQL Developer. Al hacer clic en **Probar** vemos que aparece en la parte inferior del cuadro **Estado: Correcto**



Nombre de Conexión	Detalles de Conexión
ALUMNO	ALUMNO01@//local...

Nombre de Conexión: ALUMNO
Usuario: ALUMNO01
Contraseña:
☒ Guardar Contraseña ☐ Color de Conexión

Oracle

Tipo de Conexión: Básico Rol: valor por defecto

Nombre del Host: localhost
Puerto: 1521
☒ SID: xe
☐ Nombre del Servicio

☐ Autenticación del Sistema Operativo ☐ Autenticación Kerberos Avanzadas...

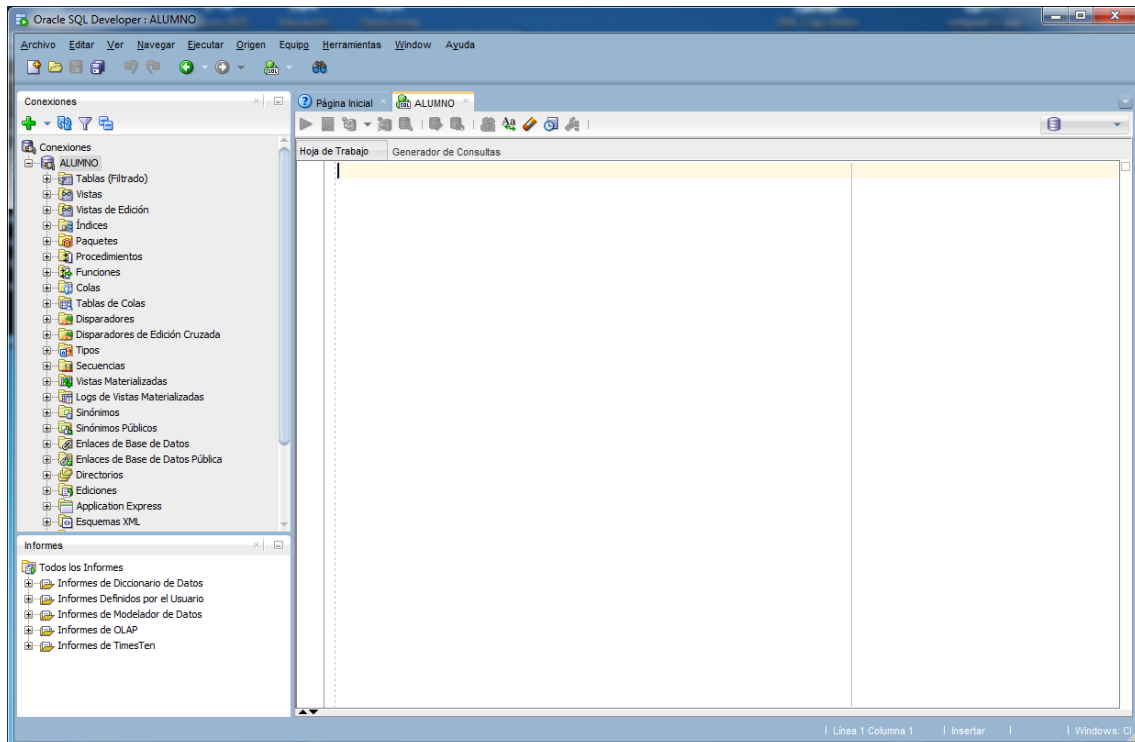
Estado: Correcto

Ayuda Guardar Borrar Probar Conectar Cancelar

-Comprobado que no hay error, hacemos clic en **Guardar contraseña** y después en **Guardar** y en **Conectar**.

-Ahora ya podemos ver nuestra conexión en el panel **Conexiones** y comenzar a trabajar con nuestra base de datos.

Una vez creada la conexión hacemos clic en ella y se abre la ventana de trabajo SQL en donde podremos escribir y ejecutar las sentencias. También podemos ver todos los objetos que tiene el usuario: tablas, vistas, etc.



Práctica 3 Habilitar usuario HR

HR (Human Resources) es un esquema con diversos objetos proporcionado por Oracle y que podemos utilizar como ejemplos. Vamos a habilitarnos para poder utilizarlo.

Para ello hemos de conectarnos a la base de datos, utilizaremos la línea de comando de SQL.

Vamos a menú **Inicio/Oracle Database 11g Express Edition**, y abrimos el elemento **Run SQL Command Line**.

Nos conectamos con el usuario **SYS**, para ello escribimos:

connect sys as sysdba

nos pedirá la clave, y escribimos ***oracle***.

Una vez conectado como sys desbloqueamos la cuenta de HR escribiendo:

alter user hr account unlock;

Aparecerá un mensaje indicando que el usuario se ha desbloqueado.

Cambiamos el password, antes de que el server nos diga que ha expirado. Para ello escribiremos

alter user hr identified by hr;

```
SQL> Run SQL Command Line

SQL*Plus: Release 11.2.0.2.0 Production on
Copyright (c) 1982, 2014, Oracle. All rights reserved.

SQL> connect sys as sysdba
Enter password:
Connected.
SQL> alter user hr account unlock;

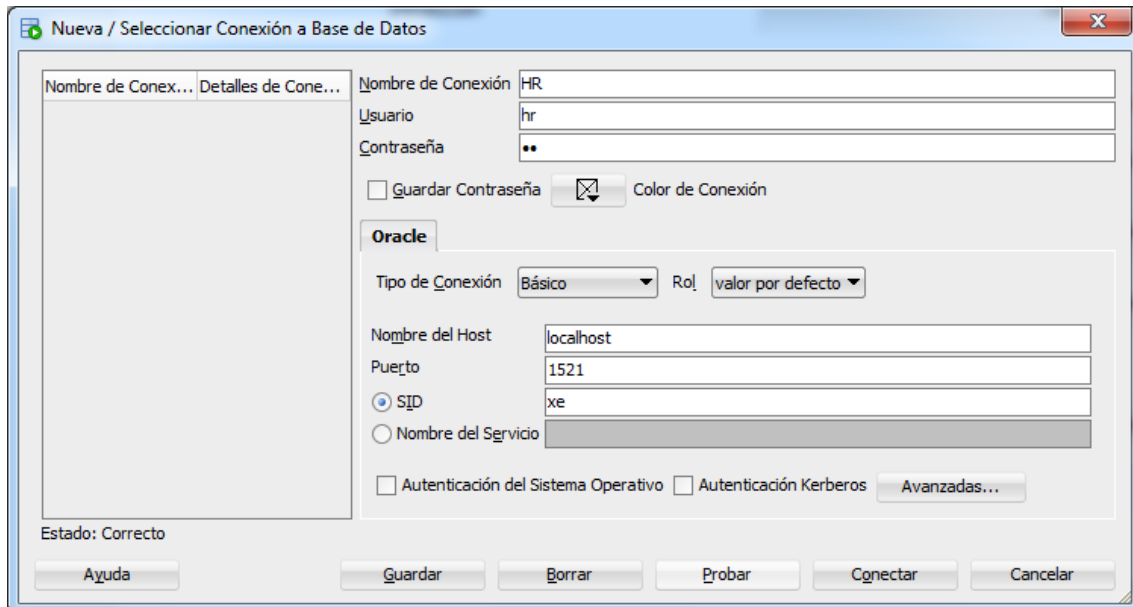
User altered.

SQL> alter user hr identified by hr;

User altered.

SQL> _
```


Ahora el usuario **hr** tiene la clave **hr** y ya podemos conectarnos desde SQL Developer y trabajar con el usuario **HR**.



NOTA: si al conectarnos a nuestro usuario HR aparece un aviso de que la contraseña va a caducar, vamos a la hoja de trabajo y escribimos:

alter user hr identified by xxxx

donde xxxx será la nueva clave que vamos a utilizar.

Ejecutamos la sentencia haciendo clic en el botón . Después hacemos clic con el botón derecho del ratón en la conexión en el panel de Conexiones, elegimos Propiedades y allí hemos de poner la nueva contraseña.

Vemos que ahora tenemos dos conexiones en SQL Developer. Utilizaremos una u otra para según qué prácticas.

Práctica 4 Data Modeler de Oracle

Es una herramienta de modelado de base de datos desarrollada por Oracle y que en las últimas versiones de SQL Developer ya viene incorporada.

Características:

- Los modelos se almacenan en el sistema de ficheros con estructura de directorios.
- Puede trabajar con cualquier base de datos.
- Permite realizar ingeniería inversa (crear un modelo de base de datos a partir de una base de datos existente).
- Dispone de diseño lógico, relacional y físico.
- Permite definir dominios de tipos de datos.
- Tiene diseño visual y muchas herramientas.
- Dispone de múltiples opciones para generar DDL

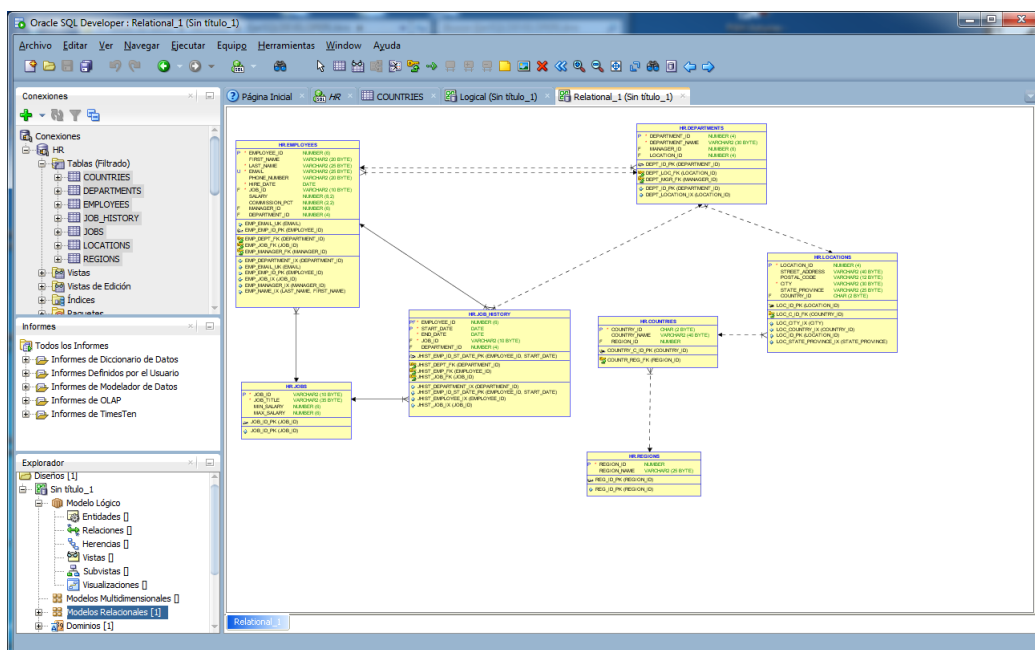
DDL → Data Definition Language, (o Data Description Language según autores), es la parte del SQL dedicada a la definición de la base de datos, consta de sentencias para definir la estructura de la base de datos, permite definir gran parte del nivel interno de la base de datos. Por este motivo estas sentencias serán utilizadas normalmente por el administrador de la base de datos.

INICIAR DATA MODELER

En SQL Developer vamos al menú **Ver/Data Modeler/Explorador** y se visualiza el módulo en la parte inferior izquierda. Si lo desplegamos podremos ver Modelo lógico, Modelo relacional, etc.

Vamos a ver cómo sería el modelo relacional de las tablas de ejemplo de HR.

Para ello hacemos clic con el botón derecho del ratón sobre Modelos relacionales y en el menú contextual elegimos **Nuevo Modelo Relacional**. Vemos una hoja de trabajo para crear el modelo; lo que hacemos es seleccionar todas las tablas que aparecen en HR y arrastrarlas a esta hoja de trabajo. Aparece entonces una imagen como la siguiente:



Con la barra de botones podremos agregar tablas, vistas, claves ajenas, etc.



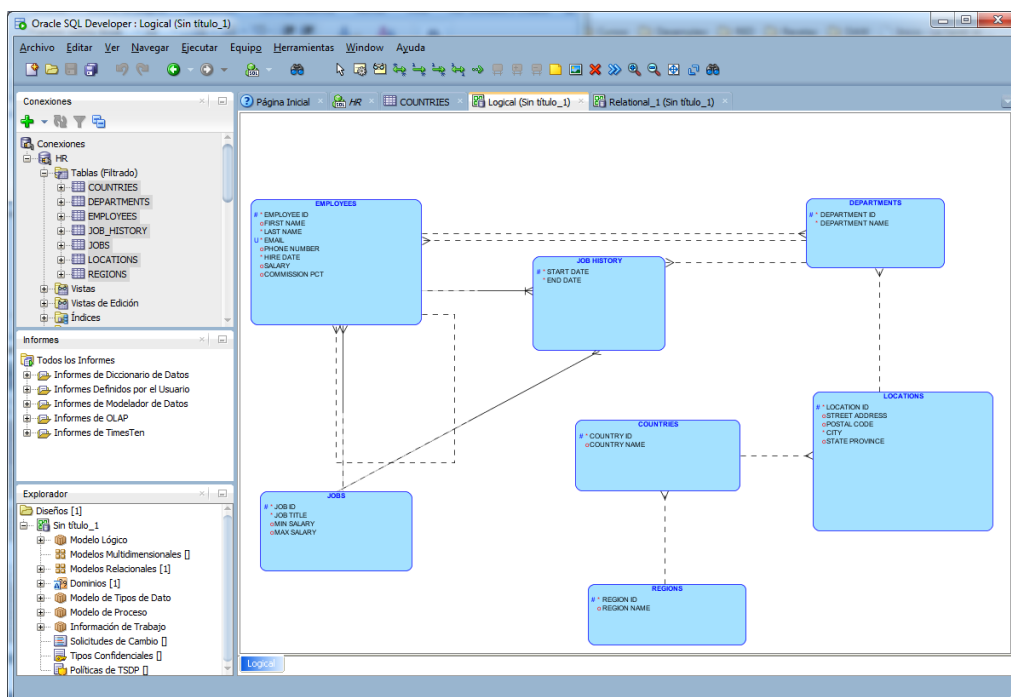
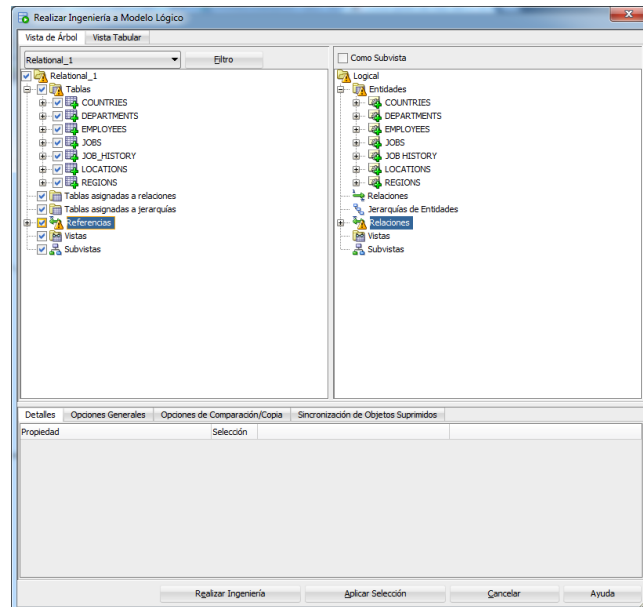
Si hacemos clic en el botón



Realizar ingeniería al modelo lógico

se crearán las entidades y las relaciones correspondientes al modelo que estamos viendo. También se pueden seleccionar qué elementos queremos convertir.

Hacemos clic en el botón **Realizar ingeniería** y obtenemos el Modelo Lógico Entidad/Relación que vemos en la imagen siguiente:



NOTA: Data Modeler utiliza la notación Barker para el modelo E/R. Ver notación Barker en **Anexo Notación Baker en modelo E/R.pdf**

Desde el menú **Archivo/Data Modeler** podremos abrir, guardar, imprimir,... los trabajos realizados con esta herramienta.

CREACIÓN DE MODELOS DE DATOS

**Botones de la barra de
herramientas Modelo
Lógico:**



Seleccionar



Nueva Entidad



Nueva Vista



Nueva Relación N:M



Nueva Relación 1:N



Nueva Identificación de Relación 1:N



Nueva Relación 1:1

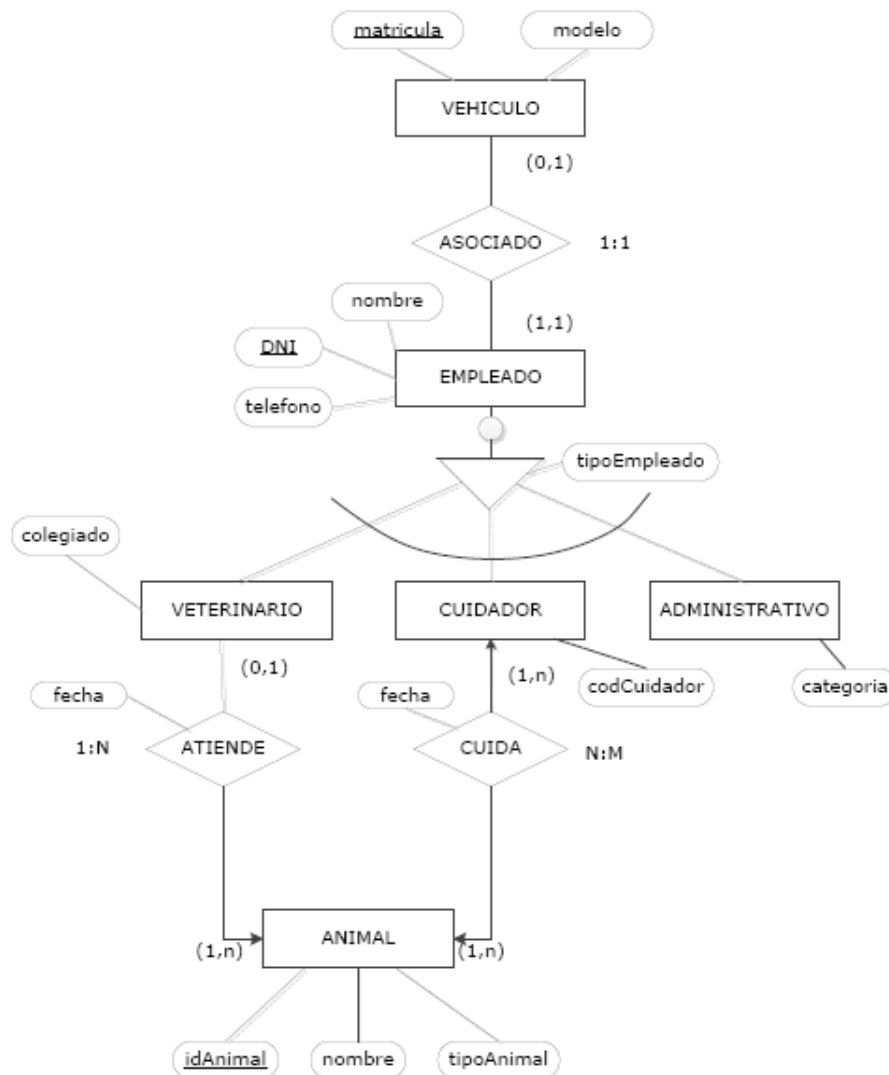


Nueva Sustitución de tipo



Realizar Ingeniería a Modelo Relacional

Supongamos el esquema de un modelo E/R de la siguiente figura:



En el **Explorador** de Data Modeler hacemos clic con el botón derecho en **Diseños** y elegimos **Nuevo Diseño**. Vemos un nuevo diseño **Sin Título**; sobre él hacemos clic con el botón derecho del ratón y elegimos **Guardar diseño**. Una vez guardado, vemos su nombre en el Explorador.

Después en la barra de botones hacemos clic en el botón **Nueva entidad** y se nos abre la ventana de propiedades de la entidad que iremos completando.

En **General** completamos los datos de la entidad como el nombre, la abreviatura o el supertipo si lo tuviera.

En **Atributos** agregamos los atributos a la entidad. Cada atributo incluye su nombre y tipo de datos. También marcamos si es identificador principal (UID) o identificador de relación (UID de Relación) o atributo obligatorio. Además podemos seleccionar su dominio.

Para el tipo de datos elegimos **Lógico** y se muestra en **Tipo de Origen** los tipos predefinidos de dónde seleccionamos el tipo de nuestro atributo. Para elegir un **Dominio** hemos de tenerlo creado previamente.

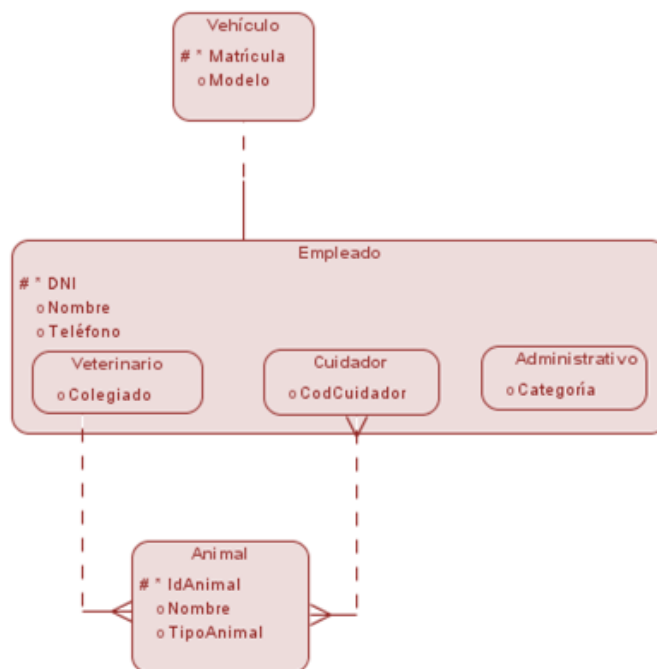
Del resto de propiedades las más importantes son:

- **Identificadores únicos** en donde se visualizan éstos y las claves primarias (PUID).
- **Relaciones** donde se muestran las relaciones asociadas a la entidad.
- **Propiedades de volumen** que indica el volumen de datos de la entidad, la forma normal requerida,
- **Realizar ingeniería** para especificar el modelo relacional en el que esta entidad deberá ser propagada.
- **Comentarios**

Así vamos creando las entidades una a una.

A continuación creamos las relaciones. Para ello utilizamos los botones de la barra de herramientas teniendo en cuenta que es **IMPORTANTE** la dirección en que se traza la relación. Si es una relación 1:N irá de la entidad del 1 a la de N. Si es una relación 1:1 habrá que ver las cardinalidades de las entidades. En nuestro caso irá de Empleado (0,1) a Vehículo (1,1). En las propiedades de la relación indicamos el nombre, revisamos origen-destino y especificamos los atributos de la relación.

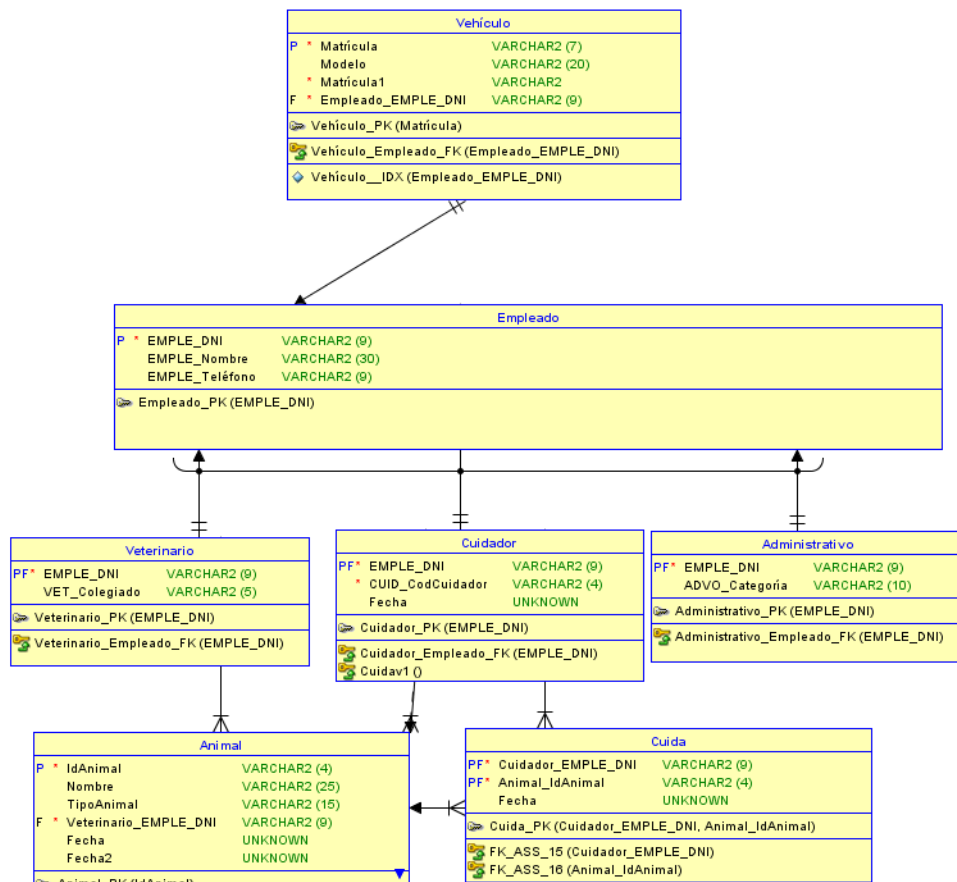
El modelo final quedará más o menos así:



Finalmente creamos las jerarquías. Para ello en las propiedades de las entidades subtipo vamos a **Subtipos** y en **Generación de Árbol Secundario** seleccionamos el tipo de transformación que se hará:

- **Tabla única:** cuando los subtipos se diferencian en muy pocos atributos y las relaciones que tienen con otras entidades del esquema son las mismas para todos o casi todos.
- **Tabla para cada entidad:** cuando existen muchos atributos distintos entre los subtipos y se quiere mantener los atributos comunes a todos ellos en una tabla.
- **Tabla por secundario:** cuando hay muchos atributos distintos y los accesos realizados sobre los datos de los distintos subtipos siempre afectan a atributos comunes.

Lo siguiente es crear el Modelo Relacional. Para ello vamos a la barra de botones y hacemos clic en el botón **Realizar Ingeniería a Modelo Relacional**. Se abre una nueva pestaña con el modelo creado. Ponemos un poco de orden si es necesario y vemos algo como la imagen siguiente:



Falta crear la definición de las tablas (DDL) para lo que pulsamos el botón **Generar DDL** de la barra de botones del modelo relacional.

En la ventana que se abre se elige la base de datos y el modelo relacional y pulsamos **Generar**. Se abre una ventana para añadir opciones en la generación de DDL. Al hacer clic en **Aceptar** se muestra el script para la creación de las tablas. Hacemos clic en **Guardar**.

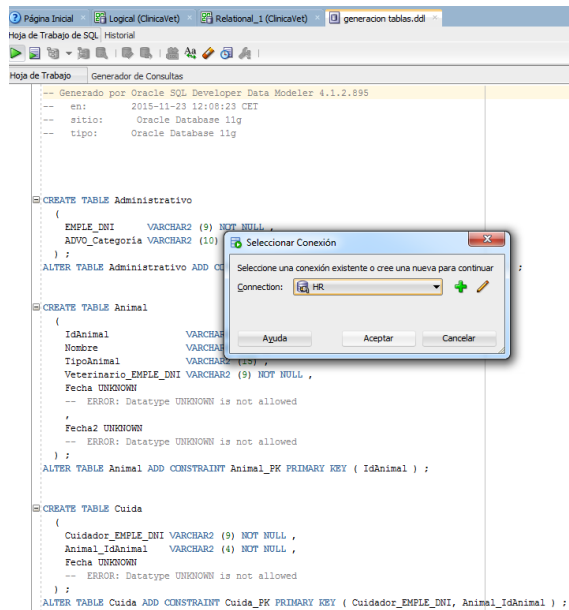
```

Oracle Database 11g | Relational_1 | Generar | Borrar
---
3 -- sitio: Oracle Database 11g
4 -- tipo: Oracle Database 11g
5
6
7
8
9 CREATE TABLE Administrativo
10 (
11     EMPLE_DNI VARCHAR2 (9) NOT NULL ,
12     ADV_O_Categoría VARCHAR2 (10)
13 );
14 ALTER TABLE Administrativo ADD CONSTRAINT Administrativo_PK PRIMARY KEY ( EMPLE_DNI );
15
16
17 CREATE TABLE Animal
18 (
19     IdAnimal VARCHAR2 (4) NOT NULL ,
20     Nombre VARCHAR2 (25) ,
21     TipoAnimal VARCHAR2 (15) ,
22     Veterinario_EMPLE_DNI VARCHAR2 (9) NOT NULL ,
23     Fecha UNKNOWN
24     -- ERROR: Datatype UNKNOWN is not allowed
25     ,
26     Fecha2 UNKNOWN
27     -- ERROR: Datatype UNKNOWN is not allowed
28 );
29 ALTER TABLE Animal ADD CONSTRAINT Animal_PK PRIMARY KEY ( IdAnimal );
30
  
```

Botones: Guardar, Buscar, Cerrar, Ayuda

Para crear las tablas en el esquema de un usuario en SQL Developer vamos a **Archivo/Abrir**. Pulsamos el botón **Ejecutar Script** de la barra de botones en la hoja de trabajo SQL. Nos pedirá una conexión y elegimos HR.

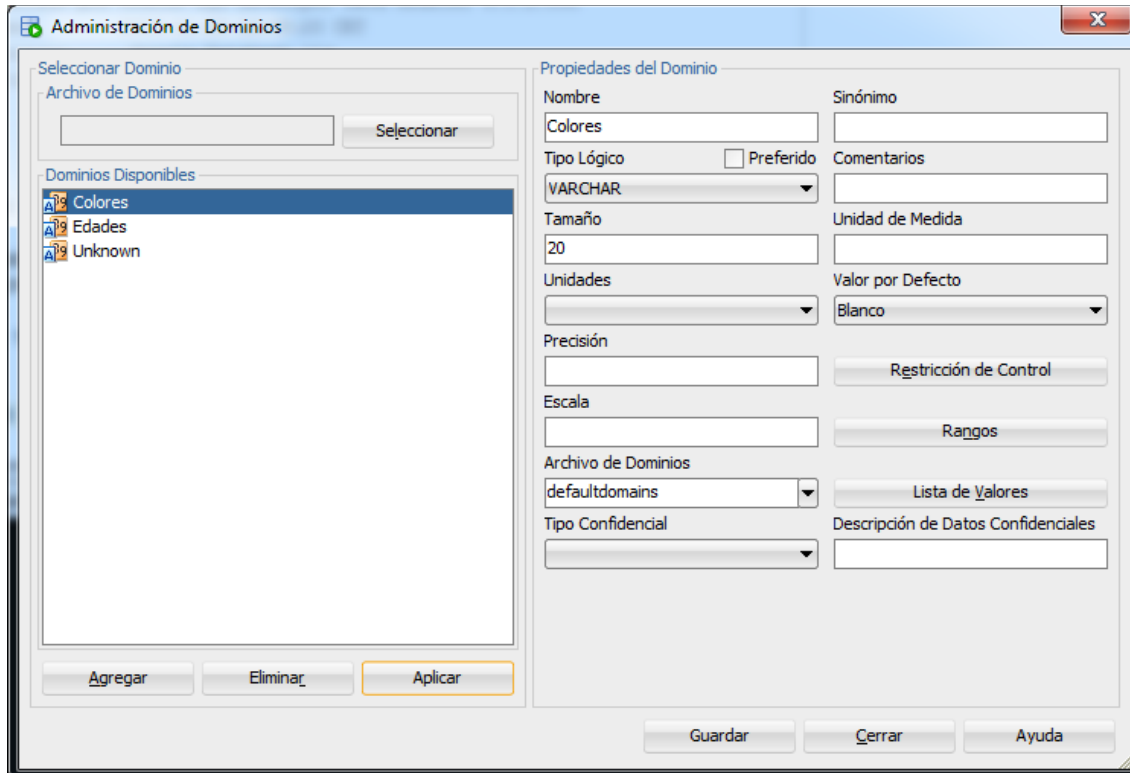
Las nuevas tablas aparecen en el explorador **Conexiones**. Actualizamos la conexión haciendo clic en el botón **Actualizar** si no vemos las nuevas tablas creadas.



CREAR DOMINIOS

El dominio es el conjunto de valores que puede tomar un atributo. Por lo general son del tipo “cadena de 30” caracteres o “entero 2 dígitos”, pero pueden ser más precisos.

Para ello vamos a **Herramientas/Data Modeler/Administrador de Dominios** y se abre el siguiente cuadro:



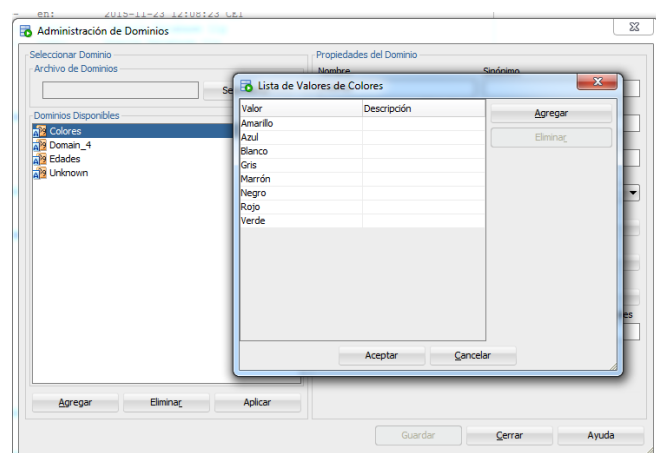
Los dominios se guardan en un archivo que seleccionaremos en el cuadro **Seleccionar Dominio**. El archivo por defecto que se carga es **defaultdomains.xml** guardado en la carpeta de **sqldeveloper**.

Desde este cuadro podemos **Agregar** o **Eliminar** dominios. En **Propiedades del Dominio** definimos todas las características:

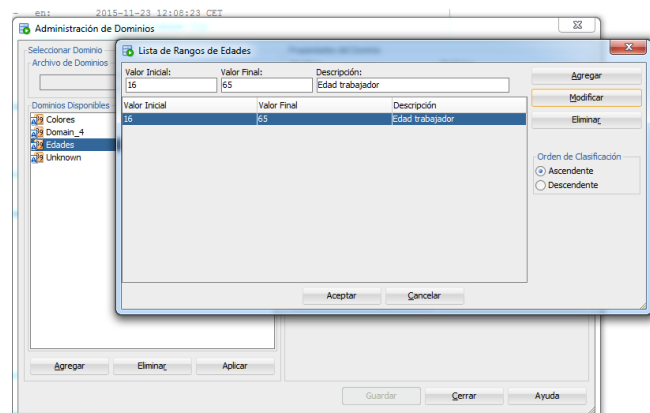
- Nombre del dominio
- Tipo lógico (numérico, carácter, fecha, etc.)
- Tamaño máximo (número máximo de caracteres)
- Precisión (número máximo de dígitos)
- Escala (dígitos decimales)
- Valor por defecto
- Restricción de control, para ver y editar las restricciones CHECK general y específica para el dominio.
- Rangos de valores para acotar el dominio
- Lista de valores para que el dominio sólo pueda tomar un valor definido en esta lista

Las restricciones **CHECK** son restricciones de control de la tabla que afectan a una o más columnas y que evitan que se guarde “basura” (datos no válidos). Por ejemplo, si los precios no pueden ser negativos podríamos tener un check que evite guardar precios negativos. Si las notas de los alumnos deben estar entre 0 y 100 podemos tener un check que evite introducir notas fuera de ese rango. Si las fechas no pueden ser anteriores al día 1 de enero de 2016 podemos tener un check que evite introducir fechas anteriores.

Ejemplo de **Lista de valores:**
Colores



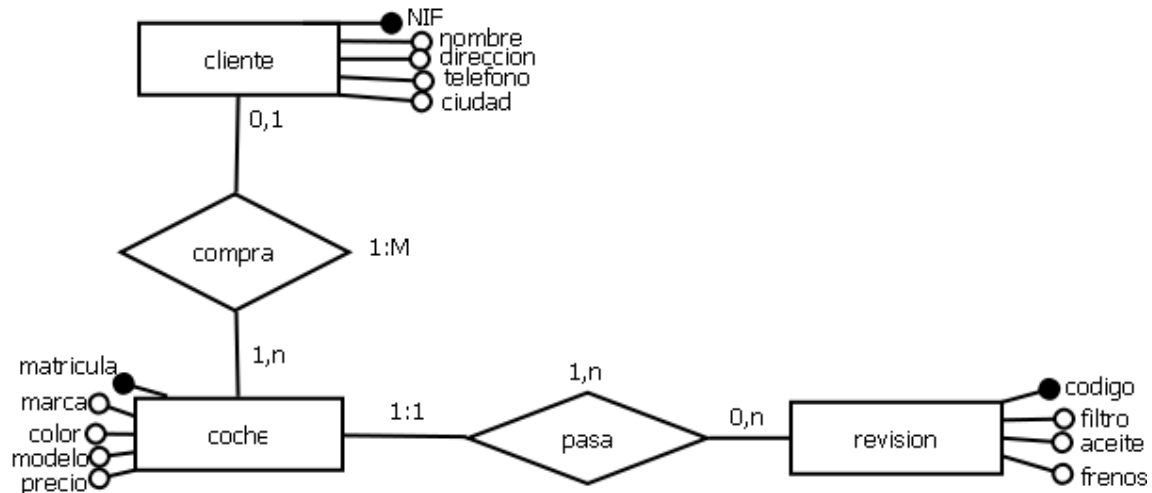
Ejemplo de **Rango de valores:**
Edad Trabajador



Cuando se crea la DDL se añaden estas restricciones a la definición de la tabla, todas con tipo CHECK.

Práctica 5

Dado el siguiente esquema E/R de la figura:



Pasarlo a un modelo lógico con Data Modeler y realizar la ingeniería al modelo relacionar. Finalmente crear las DDL de la definición de tablas.

- Se pide además:
- Agregar un atributo **fecha** a las relaciones **compra** y **pasa**.
- Los tipos de datos serán VARCHAR (tamaño) para los datos de texto, NUMERIC para los numéricos, DATE para las fechas y BOOLEAN para los atributos de **revisión filtro, aceite y frenos**.
- Previamente se creará un Dominio, con un Rango para el atributo **precio** y una Lista de Valores para el atributo **color**, que se aplicará al definir los atributos en las entidades.

NOTA: en SQL Developer utilizar el usuario hr.