

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

Bases de Datos
1er. Laboratorio Dirigido
(Ciclo Verano 2023)

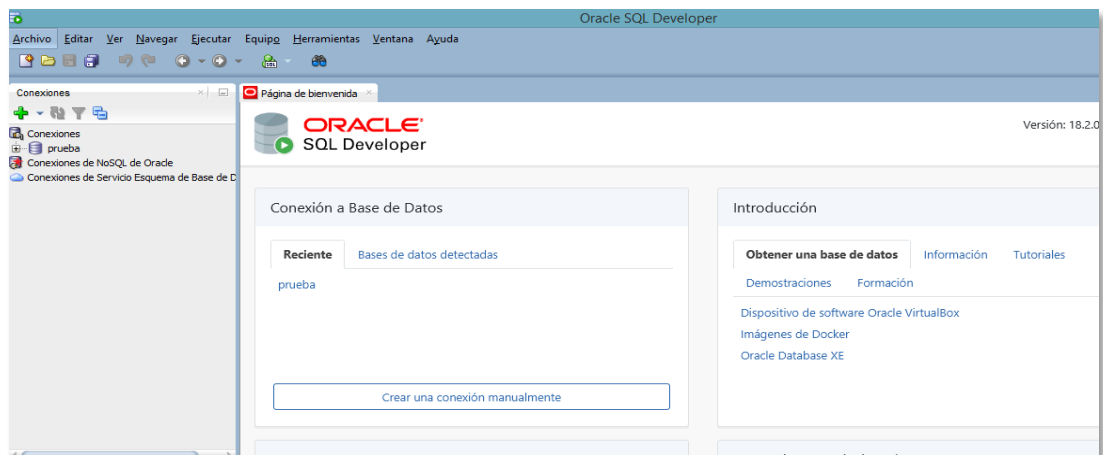
Indicaciones generales:

- Duración: **50 minutos.**

Guía: SQL DDL – SQL DML

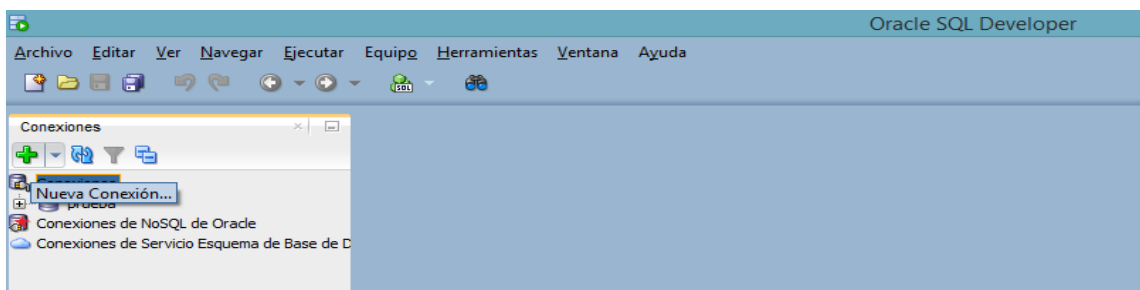
Conectarse a una base de datos:

1.- Ejecutar el programa Oracle SQL Developer.

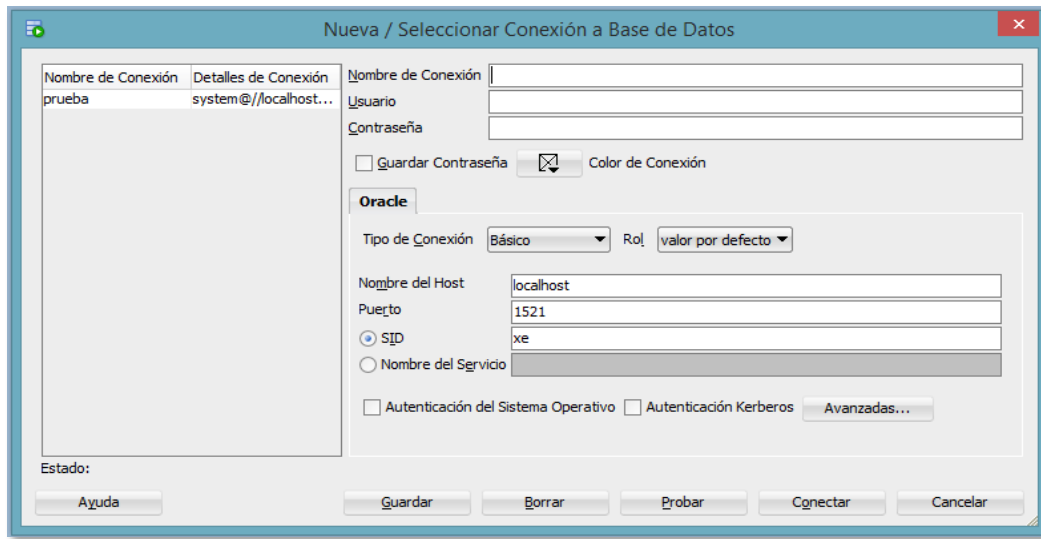


2.- Opcionalmente cerrar la página de bienvenida.

3.- En el panel de **Conexiones, hacer clic en el ícono **+** para crear una nueva conexión.**



Se abrirá la siguiente ventana:

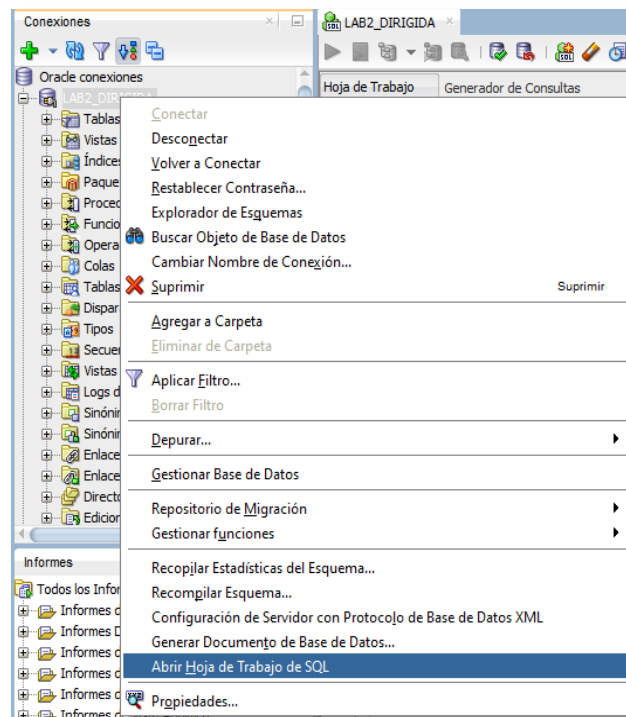


4.- Ingresar un **nombre de conexión** deseado, por ejemplo, su código de alumno, el usuario **system** (o el usuario proporcionado para la conexión), **contraseña** y el nombre del **host**.

5.- Hacer clic en el botón **Probar**. Si todo está bien, aparecerá el mensaje **Correcto** al costado del Estado.

6.- Luego de hacer clic en los botones **Guardar** y **Conectar**. El nombre de la conexión creada aparecerá en el panel de conexiones.

7.- Hacer clic derecho sobre el nombre de la conexión y aparecerá el siguiente menú contextual:



8.- Seleccione la opción **Abrir Hoja de Trabajo de SQL**. Al costado del panel de conexiones aparecerá una ventana que es la hoja de trabajo donde se podrán escribir y ejecutar los comandos SQL.

CASO: Un centro de estudios de idiomas desea modelar la base de datos para soportar la información de sus alumnos, docentes, cursos y horarios.

SQL DDL – Creación y modificación de tablas de bases de datos:

1.- Crear la tabla **CE_PERSONA** con los siguientes campos:

Campo	Tipo	Requerido	Clave Primaria
DNI	VARCHAR2(8 BYTE)	SÍ	SÍ
NOMBRES	VARCHAR2(100 BYTE)	SÍ	
APELLIDO_PATERNO	VARCHAR2(20 BYTE)	SÍ	
APELLIDO_MATERNO	VARCHAR2(20 BYTE)	SÍ	
CORREO_ELECTRONICO	VARCHAR2(100 BYTE)	SÍ	
DIRECCION	VARCHAR2(200 BYTE)	NO	
TELEFONO	NUMBER	NO	

```
CREATE TABLE CE_PERSONA (
  DNI          VARCHAR2(8 BYTE) NOT NULL,
  NOMBRES      VARCHAR2(100 BYTE) NOT NULL,
  APELLIDO_PATERNO  VARCHAR2(20 BYTE) NOT NULL,
  APELLIDO_MATERNO  VARCHAR2(20 BYTE) NOT NULL,
  CORREO_ELECTRONICO VARCHAR2(100 BYTE) NOT NULL,
  DIRECCION      VARCHAR2(200 BYTE),
  TELEFONO      NUMBER,
  CONSTRAINT PERSONA_PK PRIMARY KEY (DNI)
);
```

Otra forma:

```
CREATE TABLE CE_PERSONA (
  DNI          VARCHAR2(8 BYTE) NOT NULL,
  NOMBRES      VARCHAR2(100 BYTE) NOT NULL,
  APELLIDO_PATERNO  VARCHAR2(20 BYTE) NOT NULL,
  APELLIDO_MATERNO  VARCHAR2(20 BYTE) NOT NULL,
  CORREO_ELECTRONICO VARCHAR2(100 BYTE) NOT NULL,
  DIRECCION      VARCHAR2(200 BYTE),
  TELEFONO      NUMBER
);

ALTER TABLE CE_PERSONA ADD CONSTRAINT PERSONA_PK PRIMARY KEY (DNI);
```

Comentarios:

```
COMMENT ON COLUMN CE_PERSONA.DNI IS 'DNI de la persona';
COMMENT ON COLUMN CE_PERSONA.NOMBRES IS 'Nombres de la persona';
COMMENT ON COLUMN CE_PERSONA.APELLIDO_PATERNO IS 'Apellido paterno de la persona';
COMMENT ON COLUMN CE_PERSONA.APELLIDO_MATERNO IS 'Apellido materno de la persona';
COMMENT ON COLUMN CE_PERSONA.CORREO_ELECTRONICO IS 'Correo electrónico de la persona';
COMMENT ON COLUMN CE_PERSONA.DIRECCION IS 'Dirección de la persona';
COMMENT ON COLUMN CE_PERSONA.TELEFONO IS 'Teléfono de la persona';
```

2.- Crear la tabla **CE_UBIGEO** con los siguientes campos:

Campo	Tipo	Requerido	Clave Primaria
ID_DISTRITO	VARCHAR2(6 BYTE)	SÍ	SÍ
NOMBRE_DISTRITO	VARCHAR2(50 BYTE)	SÍ	

```
CREATE TABLE CE_UBIGEO (
  ID_DISTRITO VARCHAR2(8 BYTE) NOT NULL,
  NOMBRE_DISTRITO VARCHAR2(50 BYTE) NOT NULL,
  CONSTRAINT UBIGEO_PK PRIMARY KEY (ID_DISTRITO)
);
```

3.- Renombrar el campo **ID_DISTRITO** de la tabla **CE_UBIGEO** por **UBIGEO**:

```
ALTER TABLE CE_UBIGEO RENAME COLUMN ID_DISTRITO TO UBIGEO;
```

4.- Modificar el tamaño de las columnas **APELLIDO_PATERNO** y **APELLIDO_MATERNO** de la tabla **CE_PERSONA** de 20 a 50:

```
ALTER TABLE CE_PERSONA MODIFY APELLIDO_PATERNO VARCHAR2(50 BYTE);
ALTER TABLE CE_PERSONA MODIFY APELLIDO_MATERNO VARCHAR2(50 BYTE);
```

Modificar el tipo de la columna **TELEFONO** de **NUMBER** a **VARCHAR2(20 BYTE)**:

```
ALTER TABLE CE_PERSONA MODIFY TELEFONO VARCHAR2(20 BYTE);
```

5.- Agregar el campo **TIPO_PERSONA** de tipo **NUMBER** a la tabla **CE_PERSONA**, el campo es obligatorio:

```
ALTER TABLE CE_PERSONA ADD TIPO_PERSONA NUMBER NOT NULL;
```

Indicar con un comentario que el **TIPO_PERSONA** puede ser **1** para **alumno**, y **2** para **docente**.

```
COMMENT ON COLUMN CE_PERSONA.TIPO_PERSONA IS 'Tipo de persona. 1: alumno, 2: docente';
```

6.- Agregar el campo **UBIGEO** de tipo **VARCHAR(6 BYTE)** a la tabla **CE_PERSONA**, el campo no es obligatorio:

```
ALTER TABLE CE_PERSONA ADD UBIGEO VARCHAR2(6 BYTE) NULL;
```

```
ALTER TABLE CE_PERSONA ADD UBIGEO VARCHAR2(6 BYTE);
```

7.- Modificar el campo **UBIGEO** de la tabla **CE_PERSONA** para que sea una clave foránea referenciando a la tabla **CE_UBIGEO** mediante su llave primaria **UBIGEO**:

```
ALTER TABLE CE_PERSONA ADD CONSTRAINT PER_UBI_FK FOREIGN KEY (UBIGEO) REFERENCES CE_UBIGEO (UBIGEO);
```

SQL DML – Sentencias para manipulación de datos (INSERT, UPDATE, SELECT):

1.- Insertar los siguientes datos en la tabla **CE_UBIGEO**:

UBIGEO	NOMBRE_DISTrito
140108	Chorrillos
140109	La Victoria
140110	La Molina
140111	Lince
140117	Pueblo Libre

140124	San Isidro
140125	Barranco
140142	Los Olivos

```

INSERT INTO CE_UBIGEO (UBIGEO, NOMBRE_DISTRITO)
VALUES ('140108', 'Chorrillos');

INSERT INTO CE_UBIGEO (UBIGEO, NOMBRE_DISTRITO)
VALUES ('140109', 'La Victoria');

INSERT INTO CE_UBIGEO (UBIGEO, NOMBRE_DISTRITO)
VALUES ('140110', 'La Molina');

INSERT INTO CE_UBIGEO (UBIGEO, NOMBRE_DISTRITO)
VALUES ('140111', 'Lince');

INSERT INTO CE_UBIGEO (UBIGEO, NOMBRE_DISTRITO)
VALUES ('140117', 'Pueblo Libre');

INSERT INTO CE_UBIGEO (UBIGEO, NOMBRE_DISTRITO)
VALUES ('140124', 'San Isidro');

INSERT INTO CE_UBIGEO (UBIGEO, NOMBRE_DISTRITO)
VALUES ('140125', 'Barranco');

INSERT INTO CE_UBIGEO (UBIGEO, NOMBRE_DISTRITO)
VALUES ('140142', 'Los Olivos');

```

2.- Ejecutar el **script de tablas** proporcionado y el de inserción de datos, en este orden:

- **INF246_2023-0_Lab1_Dirigida_Tablas.sql**
- **INF246_2023-0_Lab1_Dirigida_Inserts.sql**

3.- Realice una consulta que muestre las personas cuyo **ubigeo** sea **140125**:

```

SELECT * FROM CE_PERSONA WHERE UBIGEO = '140125';

```

4.- Realice una consulta que muestre los **nombres y apellidos** de las personas cuyo **nombre** empieza con la letra **D**:

```

SELECT NOMBRES, APELLIDO_PATERNO, APELLIDO_MATERNO FROM CE_PERSONA
WHERE NOMBRES LIKE 'D%';

```

5.- Realice una consulta que muestre los **apellidos concatenados** por un **espacio en blanco**, seguido de una **coma** y luego los **nombres de la persona**; además del **ubigeo** y del **nombre del distrito** donde vive, los datos deben estar **ordenados** por **apellido paterno**.

```
SELECT P.APELLIDO_PATERNO || ' ' || P.APELLIDO_MATERNO || ', ' ||  
P.NOMBRES AS NOMBRE_COMPLETO, U.UBIGEO, U.NOMBRE_DISTRITO  
FROM CE_PERSONA P, CE_UBIGEO U  
WHERE P.UBIGEO = U.UBIGEO  
ORDER BY 1;
```

6.- Ejecutar una consulta que muestre los **horarios** en los que se encuentra el **docente con DNI 73061678**, el resultado debe mostrar los **apellidos concatenados** por un **espacio en blanco**, seguido de una **coma** y luego los **nombres de la persona**; además del **ID de la clase**, el **nombre de la clase**, la **fecha inicio** y **fin**, y por último la **hora inicio** y **fin** para cada **horario**.

```
SELECT P.APELLIDO_PATERNO || ' ' || P.APELLIDO_MATERNO || ', ' ||  
P.NOMBRES AS NOMBRE_DOCENTE, C.ID_CLASE, C.NOMBRE, C.FECHA_INICIO,  
C.FECHA_FIN, H.HORA_INICIO, H.HORA_FIN  
FROM CE_PERSONA P, CE_HORARIO H, CE_CLASE C  
WHERE P.DNI = H.DNI_DOCENTE  
AND H.ID_CLASE = C.ID_CLASE  
AND P.TIPO_PERSONA = 2  
AND P.DNI = '73061678';
```

7.- Actualizar el **costo** de todas las clases del centro de estudios cuya **fecha de fin sea mayor al 1ro de febrero del 2023** en **5%**:

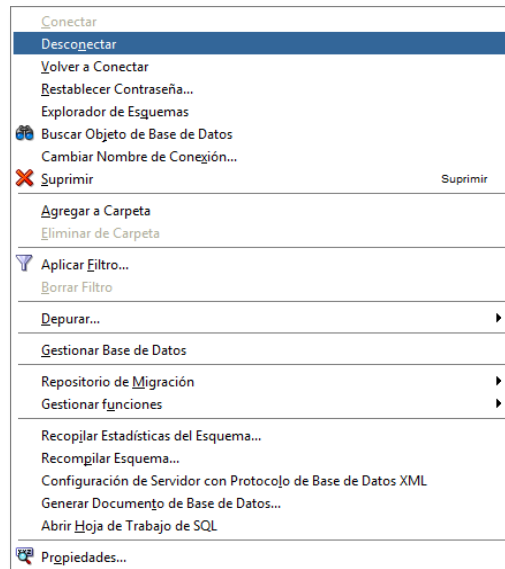
```
UPDATE CE_CLASE SET COSTO = COSTO * 1.05 WHERE FECHA_FIN >  
TO_DATE('01/02/2023', 'DD/MM/YYYY');
```

8.- Actualizar el **costo** de la clase con **código BAS01** en un **10%** y además corregir el **nombre** de **Basico 01** a **Basic 01**:

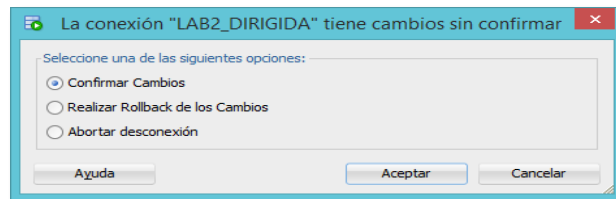
```
UPDATE CE_CLASE SET COSTO = COSTO * 1.1, NOMBRE = 'Basic 01' WHERE  
CODIGO_CLASE = 'BAS01';
```

Desconectarse de la base de datos:

1.- Para desconectarse de la base de datos, hacer **clic derecho** sobre el nombre de la conexión y aparecerá el siguiente menú contextual:



2.- Haga clic en la opción **Desconectar**. Si sale un cuadro de diálogo como el siguiente, seleccione la opción **Confirmar cambios**, y clic en **Aceptar**.



Pando, 27 de enero del 2023
Elaborado y revisado por OTM.