

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

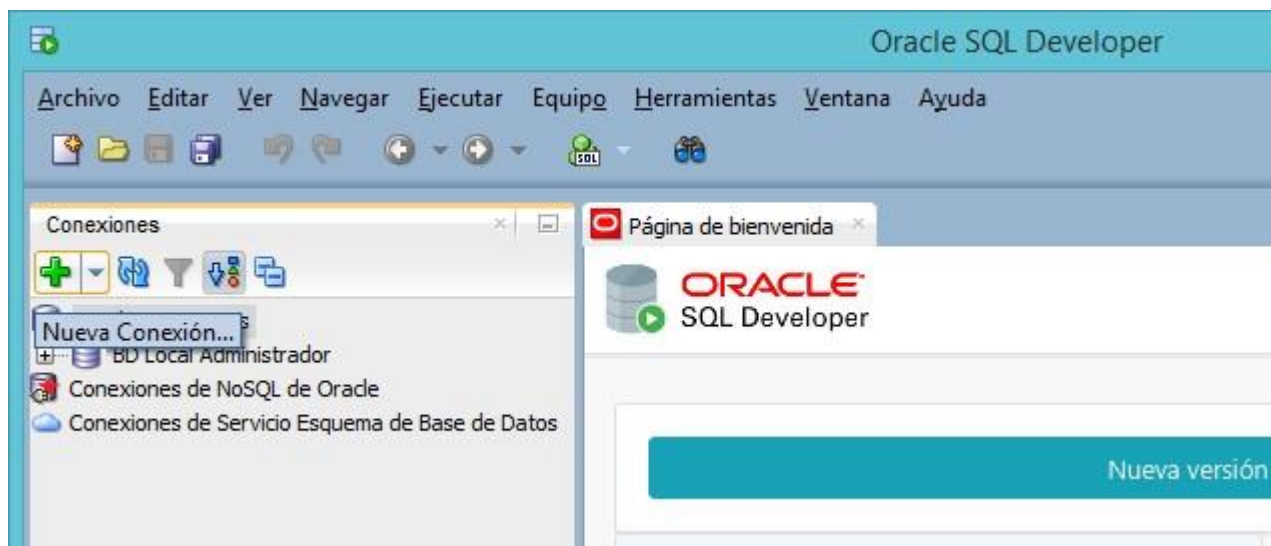
Bases de Datos

2da. Práctica Dirigida
(Segundo Semestre 2020)

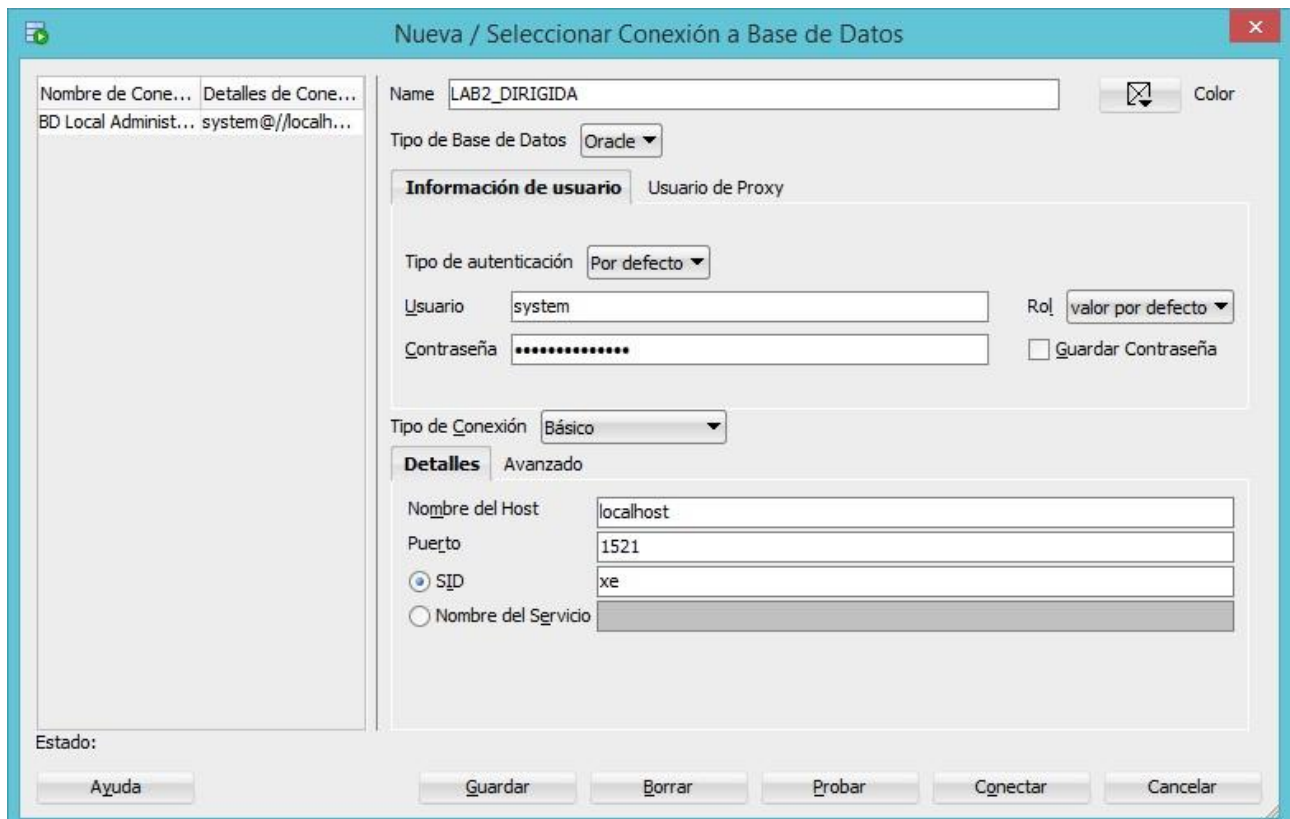
GUIA – SQL DDL – SQL DML

Conectarse a una base de datos

Primero, ejecute **Oracle SQL Developer**, cierre la pestaña **Página de bienvenida**, y en el panel de **Conexiones**, haga clic en el ícono **+** para crear una nueva conexión.



Se abrirá la siguiente ventana:

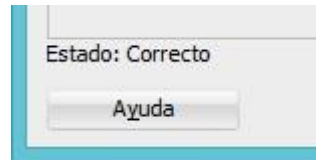


Ingrese los siguientes datos para la conexión:

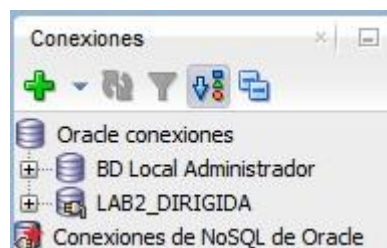
- Nombre de conexión (Name): **LAB2_DIRIGIDA**
- Usuario: **system**
- Contraseña: **Debe escribir la contraseña que ingresó al instalar Oracle Database Express Edition.**

Puede dejar el resto de los parámetros tal y como aparecen por defecto.

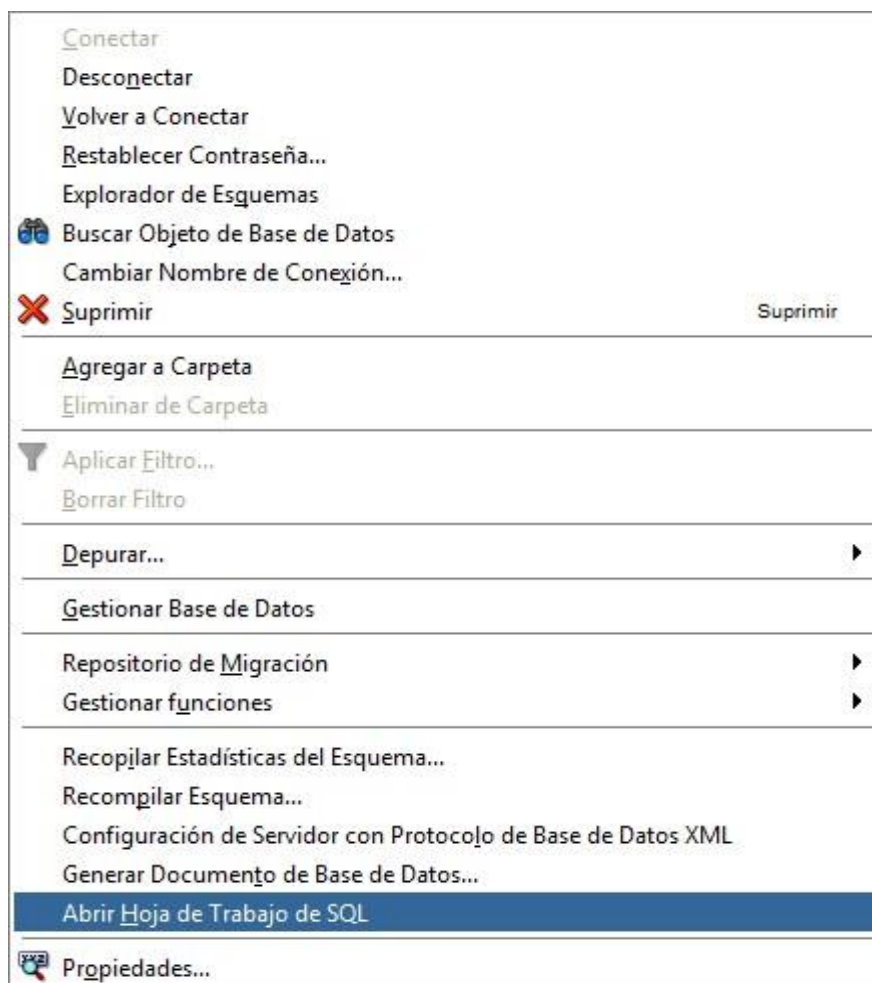
Haga clic en el botón **Probar**. Si todo está bien configurado, aparecerá el mensaje **Estado: Correcto** en la parte inferior izquierda.



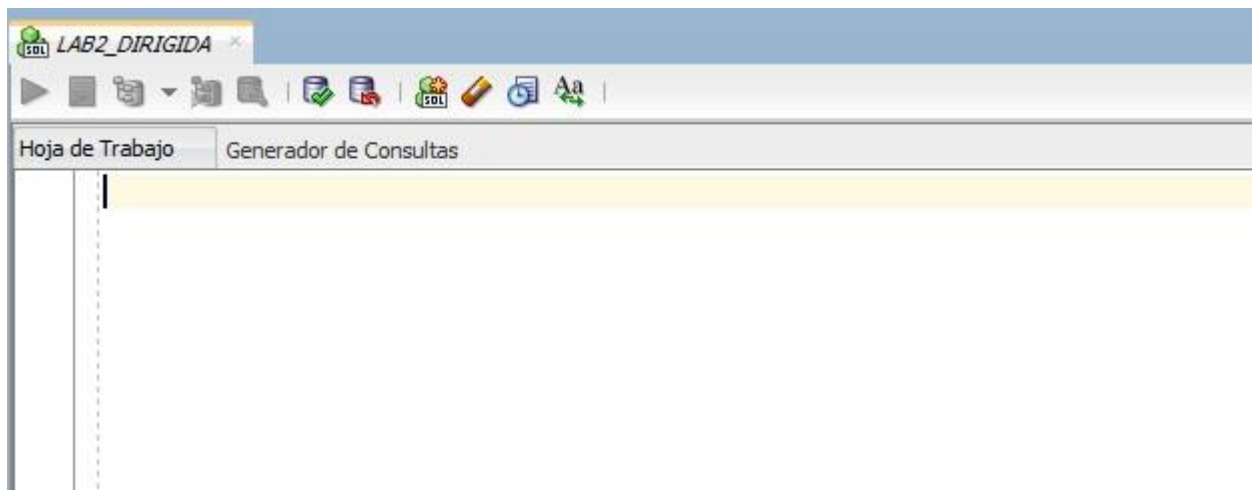
Luego, haga clic en el botón **Guardar**, y luego clic en el botón **Conectar**. El nombre de la conexión creada (**LAB2_DIRIGIDA**) aparecerá en el panel de Conexiones.



Haga clic derecho sobre el nombre de la conexión **LAB2_DIRIGIDA** y aparecerá el siguiente menú contextual:



Hacemos clic en **Abrir Hoja de Trabajo de SQL**. Nos saldrá una pantalla similar a la siguiente:



Esta es la **hoja de trabajo**, donde se podrán escribir y ejecutar los scripts SQL.

Creación de tablas de base de datos (CREATE TABLE)

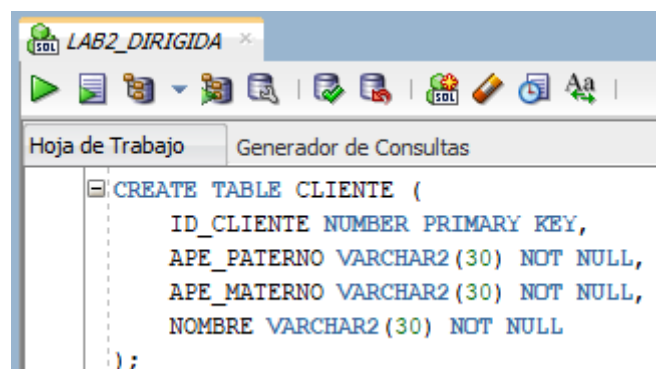
1. Cree la tabla **CLIENTE** con los siguientes campos:

Campo	Tipo	Tamaño	¿Clave primaria?	¿Admite nulos?
ID_CLIENTE	NUMBER	-	Sí	No
APE_PATERNO	VARCHAR2	30	No	No
APE_MATERNO	VARCHAR2	30	No	No
NOMBRE	VARCHAR2	30	No	No

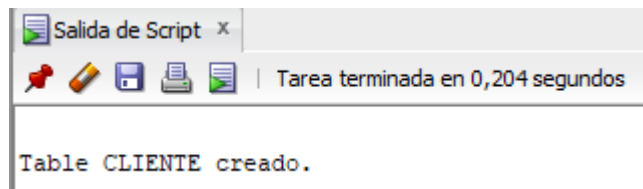
Escriba el siguiente script SQL en la hoja de trabajo:

```
CREATE TABLE CLIENTE (
    ID_CLIENTE NUMBER PRIMARY KEY,
    APE_PATERNO VARCHAR2(30) NOT NULL,
    APE_MATERNO VARCHAR2(30) NOT NULL,
    NOMBRE VARCHAR2(30) NOT NULL
);
```

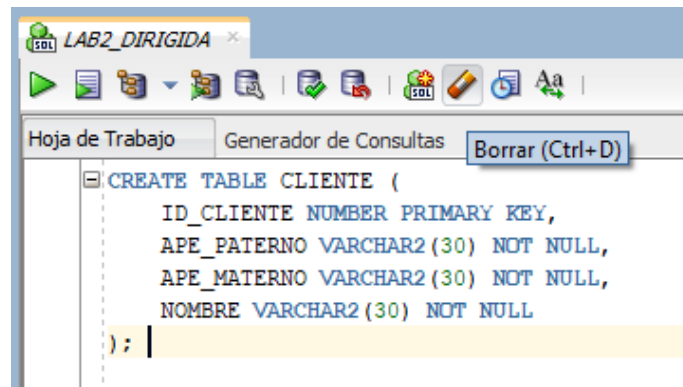
Para ejecutar el script SQL en la hoja de trabajo, haga clic en el botón **Ejecutar Script**, o presione la tecla **F5**.



En el panel Salida de Script, deberá salir el siguiente mensaje:



Puede hacer clic en el botón **Borrar**, o presionar las teclas **Ctrl+D**, para limpiar la hoja de trabajo.



2. Cree la tabla **PRODUCTO** con los siguientes campos:

Campo	Tipo	Tamaño	¿Clave primaria?	¿Admite nulos?
ID_PRODUCTO	NUMBER	-	Sí	No
NOMBRE	VARCHAR2	30	No	No
PRECIO	NUMBER		No	No

Escriba el siguiente script SQL en la hoja de trabajo y ejecútelo:

```
CREATE TABLE PRODUCTO (
    ID_PRODUCTO NUMBER NOT NULL,
    NOMBRE VARCHAR2(30) NOT NULL,
    PRECIO NUMBER NOT NULL,
    PRIMARY KEY (ID_PRODUCTO)
);
```

3. Cree la tabla **DETALLE_VENTA** con los siguientes campos:

Campo	Tipo	Tamaño	¿Clave primaria?	¿Admite nulos?
ID_DETALLE_VENTA	NUMBER	-	Sí	No
ID_CLIENTE	NUMBER	-	No	No
ID_PRODUCTO	NUMBER	-	No	No
CANTIDAD	NUMBER	-	No	No
TOTAL	NUMBER	-	No	No

```
CREATE TABLE DETALLE_VENTA (
    ID_DETALLE_VENTA NUMBER NOT NULL,
    ID_PRODUCTO NUMBER NOT NULL,
    CANTIDAD NUMBER NOT NULL,
    TOTAL NUMBER NOT NULL,
    PRIMARY KEY (ID_DETALLE_VENTA)
);
```

4. Cree la tabla **VENTA** con los siguientes campos:

Campo	Tipo	Tamaño	¿Clave primaria?	¿Admite nulos?
ID_VENTA	NUMBER	-	Sí	No
ID_CLIENTE	NUMBER	-	No	No
FECHA_VENTA	DATE		No	No

```
CREATE TABLE VENTA (  
    ID_VENTA NUMBER NOT NULL,  
    ID_CLIENTE NUMBER NOT NULL,  
    FECHA_VENTA DATE NOT NULL,  
    PRIMARY KEY (ID_VENTA)  
);
```

Modificación de tablas de base de datos (ALTER TABLE)

1. Modifique el tamaño del campo **NOMBRE** de la tabla PRODUCTO de tal manera que su **tamaño** sea **50**.

```
ALTER TABLE PRODUCTO MODIFY NOMBRE VARCHAR2(50);
```

2. Agregue la columna FECHA_NACIMIENTO tipo DATE a la tabla **CLIENTE**.

```
ALTER TABLE CLIENTE ADD FECHA_NACIMIENTO DATE;
```

3. Agregar en la tabla **DETALLE_VENTA** el campo **ID_VENTA** de tipo NUMBER.

```
ALTER TABLE DETALLE_VENTA ADD ID_VENTA NUMBER;
```

4. Agregar en la tabla **PRODUCTO** el campo **STOCK** de tipo NUMBER.

```
ALTER TABLE PRODUCTO ADD STOCK NUMBER;
```

5. Hacer que el campo **ID_VENTA** recién añadido a la tabla **DETALLE_VENTA** sea una **clave foránea** que haga referencia a la tabla **VENTA**.

```
ALTER TABLE DETALLE_VENTA ADD FOREIGN KEY (ID_VENTA) REFERENCES VENTA (ID_VENTA);
```

6. Hacer que el campo **ID_CLIENTE** en la tabla **VENTA** sea una **clave foránea** que haga referencia a la tabla **CLIENTE**.

```
ALTER TABLE VENTA ADD FOREIGN KEY (ID_CLIENTE) REFERENCES CLIENTE (ID_CLIENTE);
```

Sentencias SQL para manipulación de datos (SELECT, INSERT, UPDATE, DELETE)

1. Insertar los siguientes datos en la tabla **PRODUCTO**.

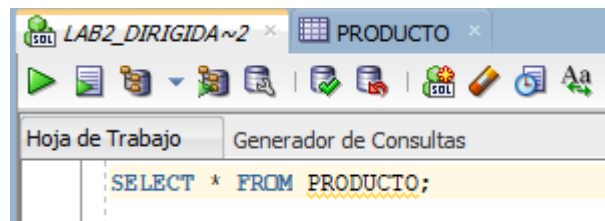
```
INSERT INTO PRODUCTO (ID_PRODUCTO, NOMBRE, PRECIO, STOCK) VALUES (1, 'Arroz  
1kg. Costeño', 3, 10);  
INSERT INTO PRODUCTO (ID_PRODUCTO, NOMBRE, PRECIO, STOCK) VALUES (2, 'Azúcar  
Rubia 1kg', 2, 8);  
INSERT INTO PRODUCTO (ID_PRODUCTO, NOMBRE, PRECIO, STOCK) VALUES (3,  
'Galleta soda', 1, 20);  
INSERT INTO PRODUCTO (ID_PRODUCTO, NOMBRE, PRECIO, STOCK) VALUES (4,  
'Chocolate Sublime', 1, 3);
```

```
INSERT INTO PRODUCTO (ID_PRODUCTO, NOMBRE, PRECIO, STOCK) VALUES (5,
'Shampoo', 16,2);
```

2. Mostrar el contenido de la tabla **PRODUCTO**.

```
SELECT * FROM PRODUCTO;
```

Para visualizar consultas (SELECT), puede seleccionar todo el script, y hacer clic en el botón **Sentencia de Ejecución**, o presionar las teclas **Ctrl+Enter**.



La consulta debería mostrarse de la siguiente manera:

Resultado de la Consulta x				
Todas las Filas Recuperadas: 5 en 0,041 segundos				
	ID_PRODUCTO	NOMBRE	PRECIO	STOCK
1	1	Arroz 1kg. Costeño	3	10
2	2	Azúcar Rubia 1kg	2	8
3	3	Galleta soda	1	20
4	4	Chocolate Sublime	1	3
5	5	Shampoo	16	2

3. Copie el contenido del archivo **INF246_2020-2_LAB2_DIRIGIDA_inserts.sql** en la hoja de trabajo y ejecute el script.
4. Muestre el contenido de la tabla **CLIENTE**.

```
SELECT * FROM CLIENTE;
```

5. Muestre los **apellidos y nombres** de las personas, ordenadas por **apellido paterno**.

```
SELECT APE_PATERNO, APE_MATERNO, NOMBRE
FROM CLIENTE
ORDER BY APE_PATERNO;
```

	APE_PATERNO	APE_MATERNO	NOMBRE
1	Perez	Garcia	Rosa
2	Prado	Garcia	Maria
3	Zupe	Alvarado	Anna

6. Muestre los productos con stock mayor a 1 y precio mayor a 2 ordenados por el precio y stock de forma descendiente.

```
SELECT NOMBRE, PRECIO, STOCK
FROM PRODUCTO
WHERE PRECIO > 2 AND STOCK > 1
ORDER BY PRECIO ASC, STOCK ASC;
```

	NOMBRE	PRECIO	STOCK
1	Arroz 1kg. Costeño	3	10
2	Shampoo	16	2

7. Muestre los apellidos y nombres concatenados con espacios y comas de los clientes que realizaron compras entre Agosto y Setiembre del 2019

```
SELECT C.APE_PATERNO || ' ' || C.APE_MATERNO || ', ' || C.NOMBRE AS NOMBRE_COMPLETO,
V.FECHA_VENTA
FROM CLIENTE C, VENTA V
WHERE V.ID_CLIENTE = C.ID_CLIENTE
AND FECHA_VENTA BETWEEN to_date('01/08/2019', 'dd/mm/yyyy') AND
to_date('30/09/2019', 'dd/mm/yyyy')
ORDER BY C.APE_PATERNO ASC, C.APE_MATERNO ASC;
```

	NOMBRE_COMPLETO	FECHA_VENTA
1	Prado Garcia, Maria	30/09/19
2	Zupe Alvarado, Anna	15/08/19
3	Zupe Alvarado, Anna	01/08/19

8. Muestre la cantidad de ítems realizados en cada venta

```
SELECT V.ID_VENTA, COUNT(ID_PRODUCTO) AS ITEMS
FROM DETALLE_VENTA DV, VENTA V
WHERE DV.ID_VENTA = V.ID_VENTA
GROUP BY V.ID_VENTA
ORDER BY ID_VENTA ASC;
```

	ID_VENTA	ITEMS
1	1	2
2	2	1
3	3	1
4	4	1
5	5	3

9. Muestre el monto total de cada una de las ventas ordenados de manera descendente por el monto

```
SELECT V.ID_VENTA, COUNT(ID_PRODUCTO) AS ITEMS, SUM(TOTAL) AS TOTAL
FROM DETALLE_VENTA DV, VENTA V
WHERE DV.ID_VENTA = V.ID_VENTA
GROUP BY V.ID_VENTA
ORDER BY 3 DESC;
```

	ID_VENTA	ITEMS	TOTAL
1	1	2	7
2	2	1	6
3	3	1	4
4	4	1	16
5	5	3	14

10. Muestre los productos únicos vendidos entre los meses de Agosto y Setiembre del 2019.

```
SELECT DISTINCT P.ID_PRODUCTO, P.NOMBRE
FROM DETALLE_VENTA DV, VENTA V, PRODUCTO P
WHERE DV.ID_VENTA = V.ID_VENTA
AND DV.ID_PRODUCTO = P.ID_PRODUCTO
AND V.FECHA_VENTA BETWEEN to_date('01/08/2019', 'dd/mm/yyyy') AND
to_date('30/09/2019', 'dd/mm/yyyy');
```

11. Mostrar los clientes que han comprado el producto Shampoo para ofrecerles un descuento en su siguiente compra.

```
SELECT DISTINCT ( C.APE_PATERNO || ' ' || C.APE_MATERNO || ', ' || C.NOMBRE) AS
NOMBRE_COMPLETO
FROM DETALLE_VENTA DV, VENTA V, PRODUCTO P, CLIENTE C
WHERE DV.ID_VENTA = V.ID_VENTA
AND DV.ID_PRODUCTO = P.ID_PRODUCTO
AND C.ID_CLIENTE = V.ID_CLIENTE
AND P.NOMBRE = 'Shampoo';
```

NOMBRE_COMPLETO
1 Zupe Alvarado, Anna

12. Actualizar el total en el detalle de la venta del producto Shampoo, debido a que hubo un problema al calcular el monto total de este.

Nota: Para visualizar los nuevos valores puede hacer una consulta a la tabla usados en la sentencia.

```
SELECT * FROM PRODUCTO
WHERE NOMBRE = 'Shampoo';
--ID PRODUCTO = 5
--PRECIO = 16

UPDATE DETALLE_VENTA DV
SET DV.TOTAL = DV.CANTIDAD * 16
WHERE DV.ID_PRODUCTO = 5;
```

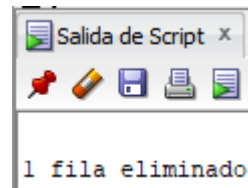
13. Mostrar la cantidad de clientes que nacieron antes del año 1980.

```
SELECT COUNT(*)
FROM CLIENTE
WHERE FECHA_NACIMIENTO <= to_date('01/01/1980', 'dd/mm/yyyy');
```

COUNT(*)
1 2

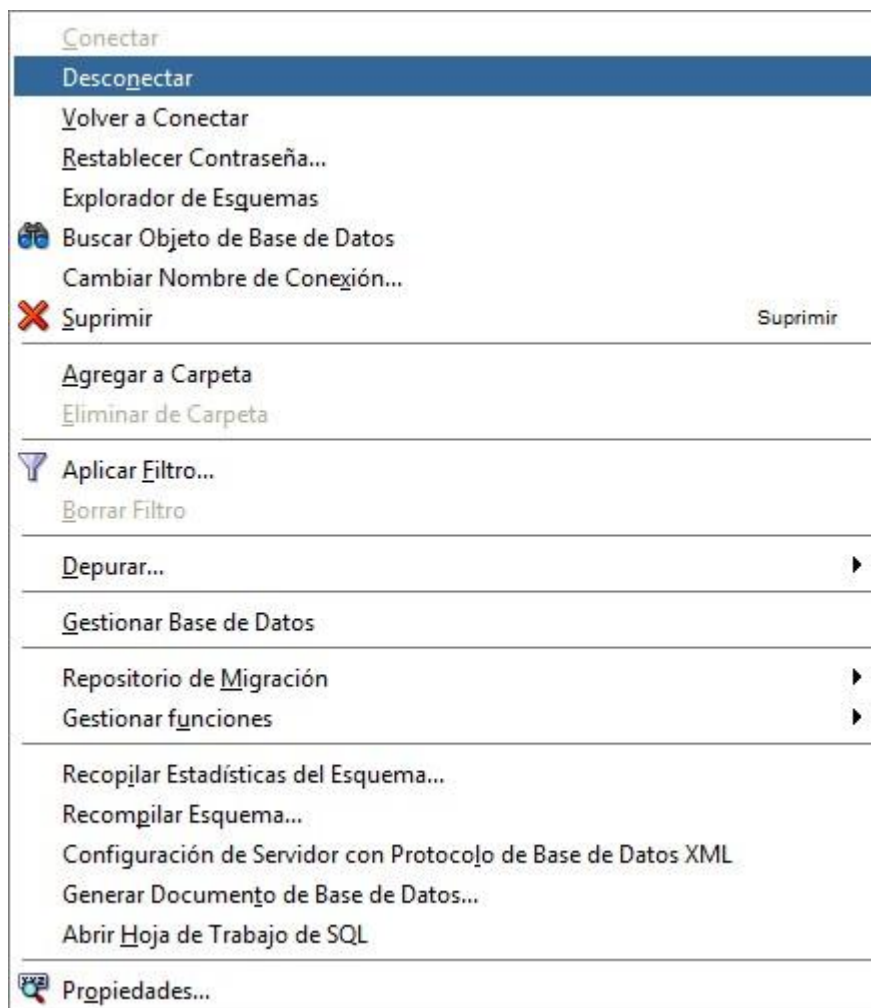
14. Eliminar los datos de los clientes que nacieron en el año 1976.


```
DELETE FROM CLIENTE
WHERE FECHA_NACIMIENTO BETWEEN to_date('01/01/1976', 'dd/mm/yyyy') AND
to_date('31/12/1976', 'dd/mm/yyyy');
```

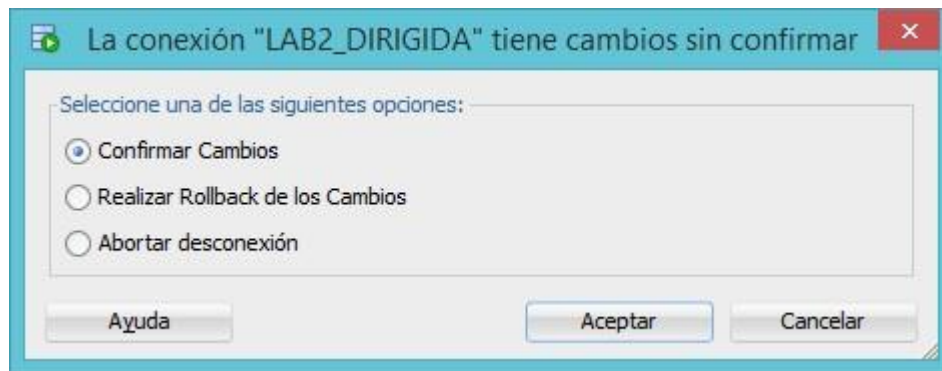


Desconectarse de una base de datos

Para desconectarse de la base de datos, haga clic derecho sobre el nombre de la conexión **LAB2_DIRIGIDA** y aparecerá el siguiente menú contextual:



Haga clic en la opción **Desconectar**. Si le sale un cuadro de diálogo como el que se muestra a continuación, seleccione la opción **Confirmar cambios** y haga clic en **Aceptar**.



30 de setiembre de 2020