

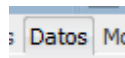
TAREA PARA BD05.**Ejercicio 1:**

Inserta un registro nuevo en la tabla **M_PACIENTES** utilizando la herramienta gráfica *Application Express* que ofrece Oracle Database Express. Los datos deben ser los siguientes:

DNIP	: 10000000B
NRO_SEG_SOC	: 11112
APELLIDOS	: Gómez Carrasco
NOMBRE	: Luisa
SEXO	: M
CENTRO SALUD	: Centro de Salud 2
DIRECCIÓN	: C/Mata, 21-1º A
POBLACIÓN	: Miguelturra
PROVINCIA	: Ciudad Real
TELEFONO	: 923 242424
MOVIL	: 611611611
EMAIL	: luisa.Gomez@gmail.com

Debes entregar una **captura de pantalla** de la ventana en la que estás introduciendo los datos, justo antes de pulsar el botón para guardarlos.

De forma gráfica en SqlDeveloper, selecciono la tabla M_PACIENTES, clic en “Datos”,

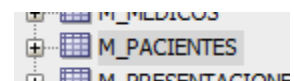


clicamos la pestaña “insertar fila”.



Nos aparecerá

una tabla con un nuevo registro a rellenar y ponemos los campos que se piden en el ejercicio:



	DNIP	NRO_SEG_SOC	APELLIDOS	NOMBRE	SEXO	CENTRO_SALUD	DIRECCION	POBLACION	PROVINCIA	TELEFONO	MOVIL	EMAIL
1	10000000A	11111	Papellido11 Papellido21	Pnombre1	H	CENTRO SALUD 1	C/COSO, 1	CIUDAD REAL	CIUDAD REAL	926111111	(null)	nombre1.papellido11@gmail.es
2	20000000B	22222	PAPELLIDO12 PAPELLIDO23	PNOMBRE2	M	CENTRO SALUD 2	C/MORA, 12-1A	CIUDAD REAL	Ciudad Real	(null)	611111111	nombre2.papellido12@gmail.es
3	30000000C	33333	PAPELLIDO13 PAPELLIDO23	PNOMBRE3	H	CENTRO SALUD 1	C/POZO, 51-1C	TOLEDO	TOLEDOL	926222222	(null)	nombre3.papellido13@hotmail.es
4	40000000D	44444	Papellido17 Papellido24	PNombre1	H	CENTRO SALUD 1	(null)	puertollano	CIUDAD REAL	926331123	622344556	nombre1.papellido17@gmail.es
5	50000000E	55555	Papellido15 Papellido25	PNOMBRE4	M	CENTRO SALUD 1	C/SALSIPUEDES, 21	Puertollano	Ciudad Real	926353535	62234336	nombre4.papellido15@afir.es
6	60000000F	66666	PAPELLIDO16 PAPELLIDO26	PNOMBRE5	M	CENTRO SALUD 2	(null)	Puertollano	CIUDAD REAL	(null)	622331146	nombre5.papellido16@palfa.es
7	70000000G	77777	PAPELLIDO12 PAPELLIDO21	PNOMBRE1	H	CENTRO SALUD 2	C/BARRIAL, 5	PUERTOLLANO	CIUDAD REAL	926232424	656334546	nombre1.apellido12@gmail.es
8	80000000H	88888	PAPELLIDO18 PAPELLIDO28	PNOMBRE8	M	CENTRO SALUD 3	C/MATA, 21-1A	DAIMIEL	CIUDAD REAL	926212424	(null)	nombre1.papellido18@gmail.es
9	90000000I	99999	PAPELLIDO11 PAPELLIDO29	PNOMBRE9	M	CENTRO SALUD 1	(null)	DAIMIEL	CIUDAD REAL	926222427	656334461	nombre1.papellido11@hotmail.es
+10	10000000B	11112	Gómez Carrasco	Luisa	M	Centro de S...	C/Mata, 21-1ª A	Miguelturra	Ciudad Real	: 923 ...	611611611	luisa.Gomez@gmail.com

Ejercicio 2:

Inserta varios registros en la tabla **M_MEDICOS** utilizando sentencias SQL. En la entrega de la tarea debes copiar las sentencias que has utilizado. Los datos deben ser los siguientes:

Tabla **M_MEDICOS**

DNIM	APELLIDOS	NOMBRE	CENTRO SALUD	POBLACIÓN	PROVINCIA	TELEFONO	MOVIL	EMAIL
11111112B	Sanz Hervás	Maria	Centro Salud 1	Ciudad Real	Ciudad Real	926212121		
11111113B	RAMOS CRUZ	JUAN	Centro Salud 3	DAIMIEL	CIUDAD REAL			
11111114B	LAOS MIS	PEDRO		PUERTOLLANO				PLM@gmail.com
11111115B	Lagos Cortés	Marina			TOLEDO		622622622	

Las columnas con los datos que aparecen en blanco no deben utilizarse en las sentencias.

```

1  INSERT INTO m_medicos (dnim, apellidos, nombre, centro_salud, poblacion, provincia, telefono) VALUES
2  ('11111112B', 'Sanz Hervás', 'Maria', 'Centro Salud 1', 'Ciudad Real', 'Ciudad Real', '926212121');
3  INSERT INTO m_medicos (dnim, apellidos, nombre, centro_salud, poblacion, provincia) VALUES
4  ('11111113B', 'RAMOS CRUZ', 'JUAN', 'Centro Salud 3', 'DAIMIEL', 'CIUDAD REAL');
5  INSERT INTO m_medicos (dnim, apellidos, nombre, poblacion, email) VALUES
6  ('11111114B', 'LAOS MIS', 'PEDRO', 'PUERTOLLANO', 'PLM@gmail.com');
7  INSERT INTO m_medicos (dnim, apellidos, nombre, provincia, movil) VALUES
8  ('11111115B', 'Lagos Cortés', 'Marina', 'TOLEDO', '622622622');
9
10
11
12

```

<div><div></div><div>Ordenar...</div><div>Filtrar:</div></div>									
DNIM	APELLIDOS		NOMBRE	CENTRO_SALUD	POBLACION	PROVINCIA	TELEFONO	MOVIL	EMAIL
1 11111111A	apellido01	apellido21	nombre1	CENTRO SALUD 1	CIUDAD REAL	CIUDAD REAL	926111111	(null)	nombre1.apellido01@sescam.es
2 22222222B	APELLIDO12	APELLIDO23	NOMBRE2	CENTRO SALUD 2	CIUDAD REAL	Ciudad Real	926222222	611111111	nombre2.apellido12@sescam.es
3 33333333C	APELLIDO13	APELLIDO23	NOMBRE3	CENTRO SALUD 1	CIUDAD REAL	CIUDAD REAL	926222222	(null)	nombre3.apellido13@sescam.es
4 44444444D	Apellido17	Apellido24	Nombre1	CENTRO SALUD 1	puertollano	CIUDAD REAL	926331123	622344556	nombre1.apellido17@sescam.es
5 55555555E	apellido15	apellido25	NOMBRE4	CENTRO SALUD 1	Puertollano	Ciudad Real	926353535	62234336	nombre4.apellido15@sescam.es
6 66666666f	APELLIDO16	APELLIDO26	NOMBRE5	CENTRO SALUD 2	Puertollano	CIUDAD REAL	926232323	622331146	nombre5.apellido16@sescam.es
7 77777777G	APELLIDO12	APELLIDO21	NOMBRE1	CENTRO SALUD 2	PUERTOLLANO	CIUDAD REAL	926232424	656334546	nombre1.apellido12@sescam.es
8 88888888H	APELLIDO18	APELLIDO28	NOMBRE8	CENTRO SALUD 3	DAIMIEL	CIUDAD REAL	926212424	656334462	nombre1.apellido18@sescam.es
9 99999999I	APELLIDO11	APELLIDO29	NOMBRE9	CENTRO SALUD 1	DAIMIEL	CIUDAD REAL	926222427	656334461	nombre1.apellido11@sescam.es
10 11111112B	Sanz Hervás		Maria	Centro Salud 1	Ciudad Real	Ciudad Real	926212121	(null)	(null)
11 11111113B	RAMOS CRUZ		JUAN	Centro Salud 3	DAIMIEL	CIUDAD REAL	(null)	(null)	(null)
12 11111114B	LAOS MIS		PEDRO	(null)	PUERTOLLANO	(null)	(null)	(null)	PLM@gmail.com
13 11111115B	Lagos Cortés		Marina	(null)	(null)	TOLEDO	(null)	622622622	(null)

Los campos sin añadir se registrarán como null como se puede ver.

Ejercicio 3:

Utiliza la herramienta gráfica, entregando con la tarea una **captura de pantalla** de la pestaña Datos de las tablas, donde se aprecien todos los cambios que has realizado.

Inserta en la tabla **M_VENTAS_MED** las siguientes ventas:

ID_VENTA	ID_MED	Fecha Venta	Unidades
25	9	01/02/2017	4
26	10	02/02/2017	5
27	11	20/03/2017	10
28	12	30/01/2017	1

The screenshot shows a database management tool interface with the table **M_VENTAS_MED** open. The table has four columns: **ID_VENTA**, **ID_MED**, **FECHA_VENTA**, and **UNIDADES**. The existing data is listed in the table, and four new rows (25 to 28) are highlighted in green, indicating they have been added. The new rows are:

+25	25	9	01/02/17	4
+26	26	10	02/02/17	5
+27	27	11	20/03/17	10
+28	28	12	30/01/17	1

The interface also shows a 'Mensajes - Log' panel at the bottom.

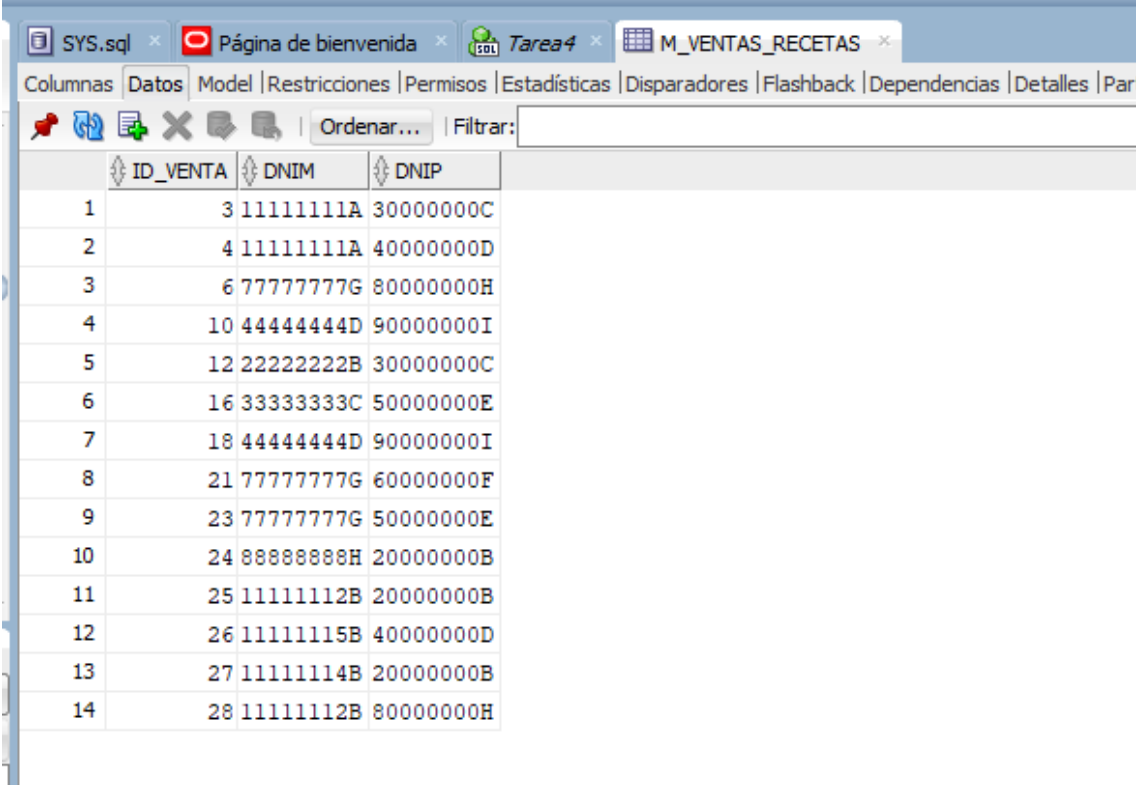
Hago clic en Confirmar cambios (F11).

Inserta en la tabla **M_VENTAS_RECETAS** los registros correspondientes a las ventas anteriores pero sin asignarle médicos ni pacientes.

	ID_VENTA	DNIM	DNIP
1	3	11111111A	30000000C
2	4	11111111A	40000000D
3	6	77777777G	80000000H
4	10	44444444D	90000000I
5	12	22222222B	30000000C
6	16	33333333C	50000000E
7	18	44444444D	90000000I
8	21	77777777G	60000000F
9	23	77777777G	50000000E
10	24	88888888H	20000000B
11	25	(null)	(null)
12	26	(null)	(null)
13	27	(null)	(null)
14	28	(null)	(null)

Modifica los registros de la tabla **M_VENTAS_RECETAS** para asignar a cada venta introducida anteriormente médico y un paciente. El profesorado que debes asignar a cada curso es:

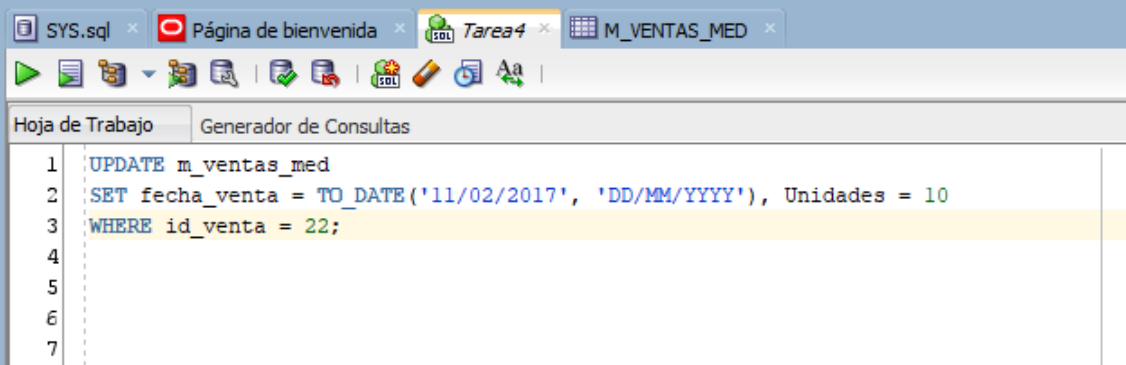
ID_VENTA	DNI Médico	DNI Paciente
25	11111112B	20000000B
26	11111115B	40000000D
27	11111114B	20000000B
28	11111112B	80000000H



	ID_VENTA	DNIM	DNIP
1	3	11111111A	30000000C
2	4	11111111A	40000000D
3	6	77777777G	80000000H
4	10	44444444D	90000000I
5	12	22222222B	30000000C
6	16	33333333C	50000000E
7	18	44444444D	90000000I
8	21	77777777G	60000000F
9	23	77777777G	50000000E
10	24	88888888H	20000000B
11	25	11111112B	20000000B
12	26	11111115B	40000000D
13	27	11111114B	20000000B
14	28	11111112B	80000000H

Ejercicio 4:

Modifica de la tabla **M_VENTAS_MED** el registro cuyo ID_VENTA es 22, cambiando el valor de la fecha de venta: 11/02/2017 y las unidades 10. Debes hacerlo usando un sola sentencia SQL que debes copiar para la entrega de la tarea



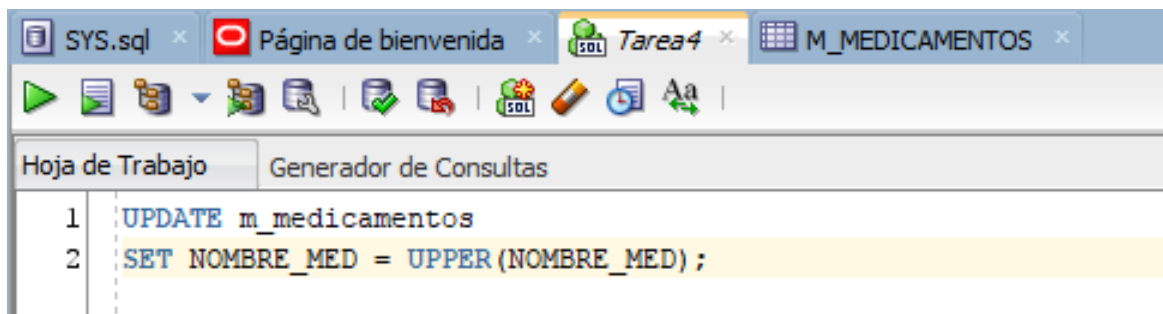
	ID_VENTA	FECHA_VENTA	UNIDADES
21	21	8/25/03/16	7
22	22	4/11/02/17	10
23	23	9/12/10/16	3

Cambios:

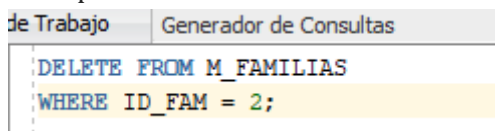
21	21	8/25/03/16	7
22	22	4/11/02/17	10
23	23	9/12/10/16	3

Ejercicio 5:

Modifica la columna NOMBRE_MED de la tabla **M_MEDICAMENTOS** convirtiendo todos los nombres a mayúsculas. Debes hacerlo usando un sola sentencia SQL que debes copiar para la entrega de la tarea.

**Ejercicio 6:**

Elimina de la tabla **M_FAMILIAS** el registro cuyo ID_FAM es 2 ¿te deja eliminarlo?. Razona la respuesta.



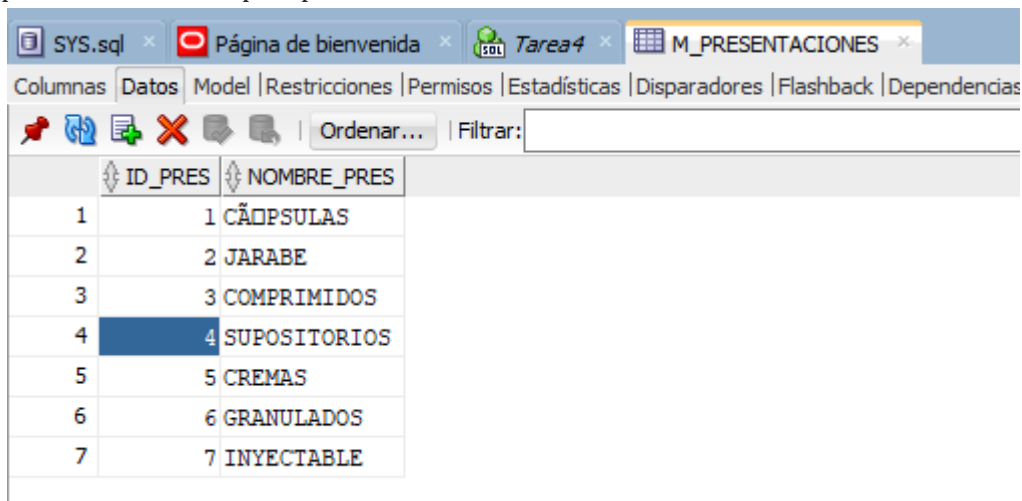
No nos deja eliminarlo resultando en el siguiente error:

```
Error que empieza en la línea: 1 del comando :
DELETE FROM M_FAMILIAS
WHERE ID_FAM = 2
Informe de error -
ORA-02292: restricción de integridad (SYS.SYS_C008402) violada - registro secundario encontrado
```

Este error se da cuando intentamos borrar una fila (DELETE) de una tabla hacia la que hay dirigida un foreign key.

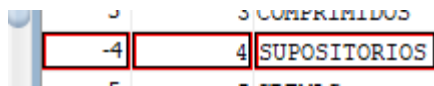
Ejercicio 7:

Elimina de la tabla **M_PRESENTACIONES** el registro del curso que tiene el ID_PRES es 4. Debes realizar esta acción desde la herramienta gráfica. Debes entregar una **captura de pantalla** de la ventana en la que vas a eliminar cada el registro, justo antes y después de pulsar el botón Aceptar para confirmar el borrado.



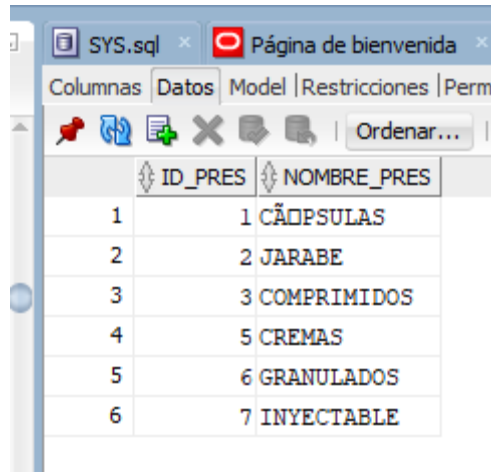
Selecciono el registro, y hacemos clic derecho y suprimir filas seleccionadas o pulsamos la "X" del panel de arriba.

Se marcará en rojo antes de guardar cambios para indicarnos que se va a borrar el registro marcado:



1	3	COMPRIMIDOS
2	4	JARABE
3	5	COMPRIMIDOS
4	6	CREMAS
5	7	GRANULADOS
6	8	INYECTABLE

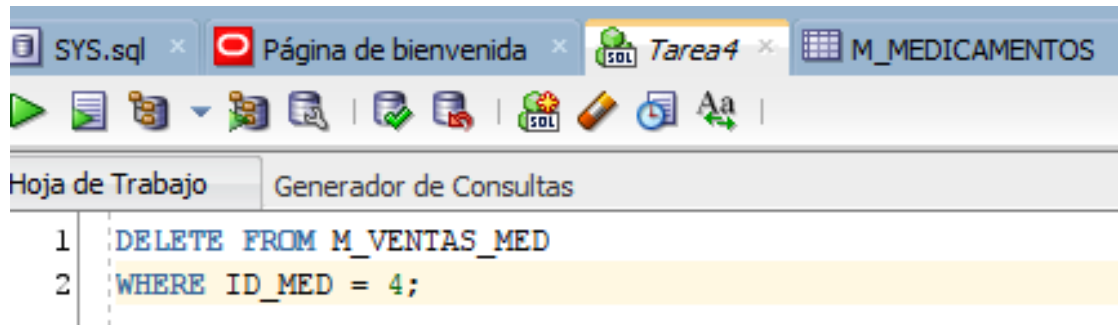
Al pulsar en guardar cambios:



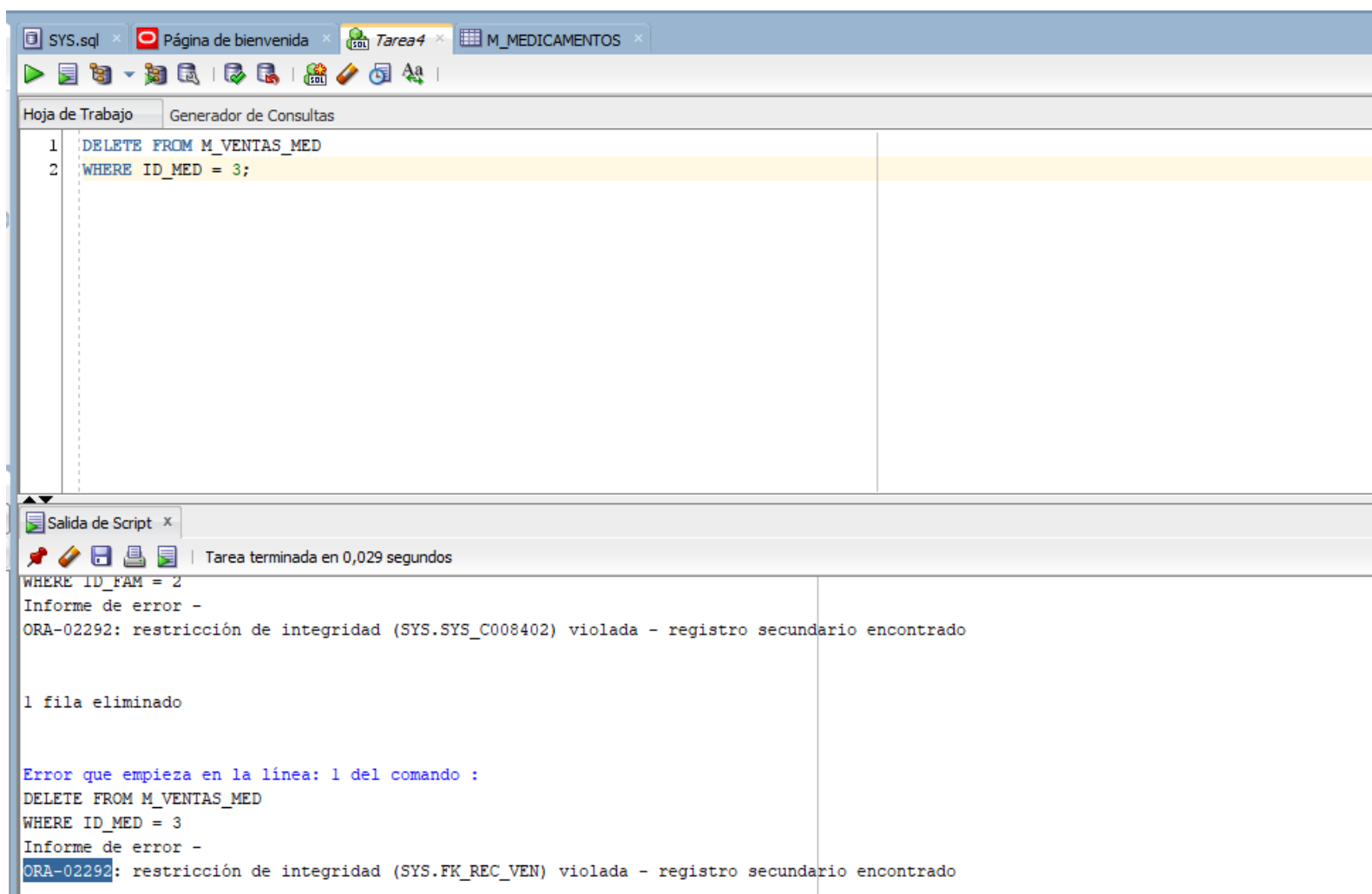
ID_PRES	NOMBRE_PRES
1	1 CÁPSULAS
2	2 JARABE
3	3 COMPRIMIDOS
4	4 CREMAS
5	5 GRANULADOS
6	7 INYECTABLE

Ejercicio 8:

Elimina de la tabla **M_VENTAS_MED** aquellos registros asociados al medicamento con ID_MED 4. Debes hacerlo usando un sola sentencia SQL que debes copiar para la entrega de la tarea.



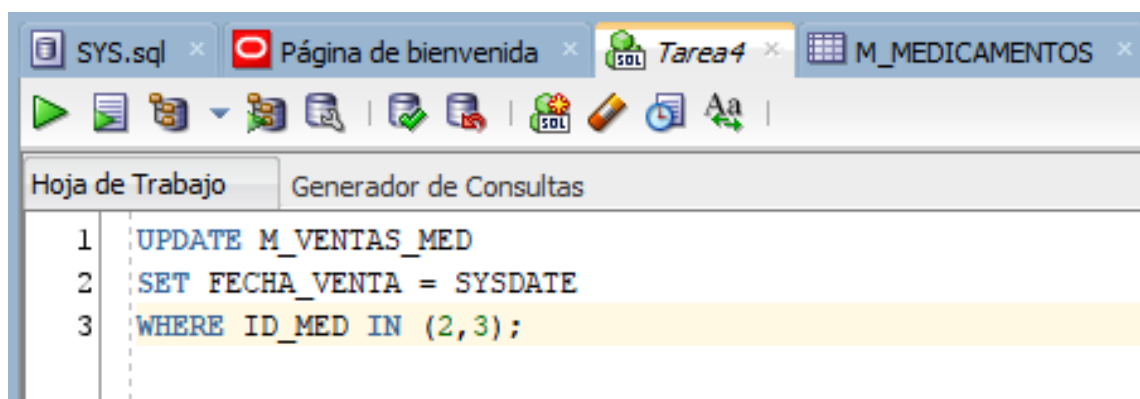
Elimina de la tabla **M_VENTAS_MED** aquellos registros asociados al medicamento con ID_MED 3. Debes hacerlo usando un sola sentencia SQL que debes copiar para la entrega de la tarea. ¿Deja eliminarlo? ¿Porqué?



No nos deja borrarlo ya que se repite el error del anterior apartado, este error se da cuando intentamos borrar una fila (DELETE) de una tabla hacia la que hay dirigida un foreign key.

Ejercicio 9:

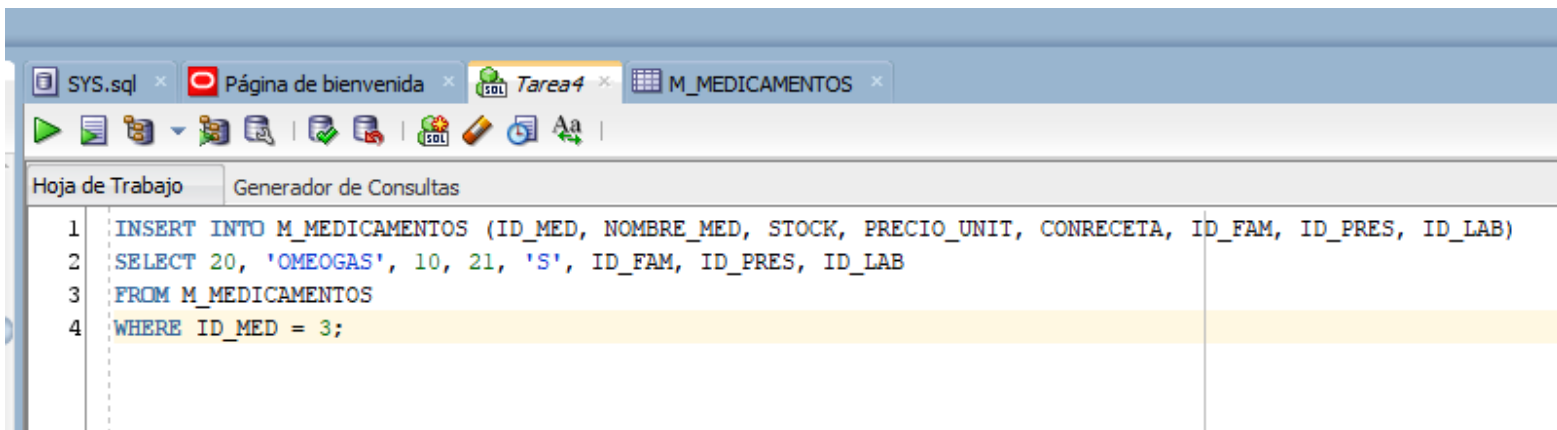
En la tabla **M_VENTAS_MED**, actualiza el campo **FECHA_VENTA** a la fecha del sistema a los medicamentos con **ID_MED** 2 y 3. Debes hacerlo usando un sola sentencia SQL y copiarla en la entrega de la tarea.



Ejercicio 10:

Inserta un medicamento con los siguientes valores :

ID_MED=20, NOMBRE_MED='OMEOGAS', STOCK= 10, PRECIO_UNIT= 21, CONRECETA= 'S',
Las columnas ID_FAM, ID_PRES, ID_LAB serán los mismos que el medicamento cuyo ID_MED es 3. Debes hacerlo usando un sola sentencia SQL y copiarla en la entrega de la tarea.



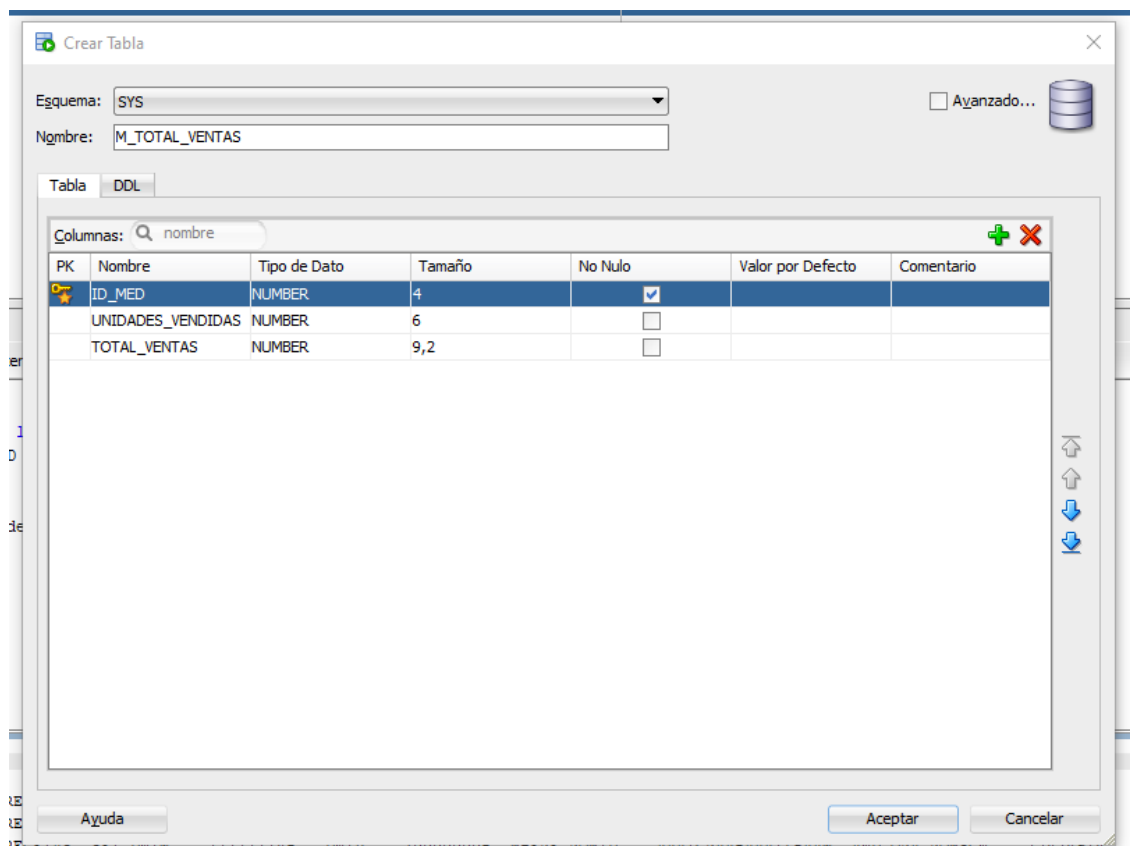
```
INSERT INTO M_MEDICAMENTOS (ID_MED, NOMBRE_MED, STOCK, PRECIO_UNIT,
CONRECETA, ID_FAM, ID_PRES, ID_LAB)
SELECT 20, 'OMEOGAS', 10, 21, 'S', ID_FAM, ID_PRES, ID_LAB
FROM M_MEDICAMENTOS
WHERE ID_MED = 3;
```

Utilizo SELECT para obtener los valores de las columnas ID_FAM, ID_PRES, ID_LAB de un registro existente en la tabla M_MEDICAMENTOS con un valor de ID_MED igual a 3.

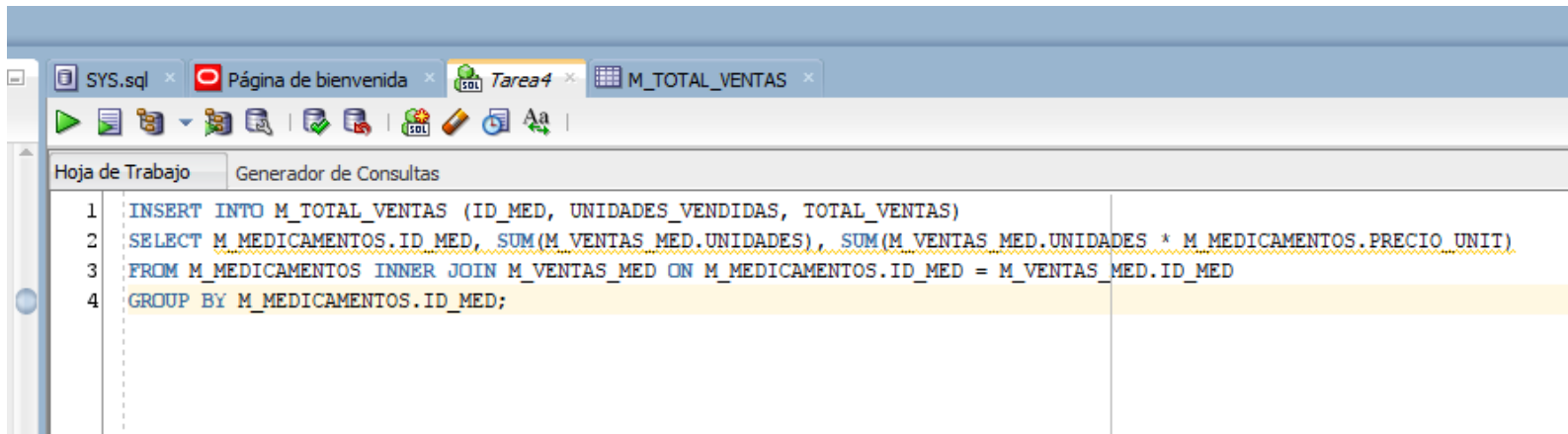
Ejercicio 11:

Crea una tabla llamada **M_TOTAL_VENTAS** que tenga las siguientes columnas:

ID_MED: Number(4), UNIDADES VENDIDAS: Number(6), TOTAL_VENTAS : Number(9,2)



Inserta en la tabla **M_TOTAL_VENTAS** un registro por cada medicamento, almacenando: ID_MED, el número de unidades totales vendidas de ese medicamento y total de todas las ventas (se calcula multiplicando las unidades por le precio unitario). Debes hacerlo usando un sola sentencia SQL y copiarla en la entrega de la tarea.



```

INSERT INTO M_TOTAL_VENTAS (ID_MED, UNIDADES_VENDIDAS, TOTAL_VENTAS)
SELECT      M_MEDICAMENTOS.ID_MED,      SUM(M_VENTAS_MED.UNIDADES),
SUM(M_VENTAS_MED.UNIDADES * M_MEDICAMENTOS.PRECIO_UNIT)
FROM M_MEDICAMENTOS INNER JOIN M_VENTAS_MED ON M_MEDICAMENTOS.ID_MED =
M_VENTAS_MED.ID_MED
GROUP BY M_MEDICAMENTOS.ID_MED;

```

The screenshot shows the same SQL IDE with the results of the query displayed in a table view. The table has four columns: ID_MED, UNIDADES_VENDIDAS, and TOTAL_VENTAS. The data is as follows:

	ID_MED	UNIDADES_VENDIDAS	TOTAL_VENTAS
1	1	13	74,1
2	2	7	56
3	3	2	6
4	5	4	28
5	6	6	60
6	8	7	77
7	9	8	96
8	10	5	20
9	11	10	66
10	12	7	68,6
11	14	11	231
12	16	3	69

Ejercicio 12:

Se quiere actualizar la columna STOCK de la tabla M_MEDICAMENTOS . Hay que sumar todas las unidades vendidas de ese medicamento y restarlas al STOCK. Debes hacerlo usando un sola sentencia SQL y copiarla en la entrega de la tarea.

```
UPDATE M_MEDICAMENTOS m
SET m.STOCK = m.STOCK - (
    SELECT NVL(SUM(v.UNIDADES), 0)
    FROM M_VENTAS_MED v
    WHERE v.ID_MED = m.ID_MED
)
WHERE EXISTS (
    SELECT 1
    FROM M_VENTAS_MED v
    WHERE v.ID_MED = m.ID_MED
);
```

Esta es la más complicada... Hago una subconsulta que calcula el total de unidades vendidas de cada medicamento en la tabla M_VENTAS_MED, y luego actualizo el STOCK de la tabla M_MEDICAMENTOS...

Ejercicio 13:

Elimina de la tabla **M_FAMILIAS** aquellos familias que no tengan ningún medicamento asignado . Debes hacerlo usando un sola sentencia SQL y copiarla en la entrega de la tarea.

```
DELETE m_familias WHERE id_fam NOT IN(SELECT id_fam FROM m_familias
NATURAL JOIN
m_medicamentos GROUP BY id_fam);
```

Ejercicio 14:

Incrementa en un 10% el precio del medicamento para aquellos medicamentos que se hayan vendido más de 3 veces. Debes hacerlo usando un sola sentencia SQL y copiarla en la entrega de la tarea.

```
UPDATE m_medicamentos SET precio_unit = precio_unit * 1.1 WHERE id_med IN
(SELECT id_med FROM m_medicamentos WHERE id_med IN (SELECT id_med ventas
FROM m_ventas_med
group by id_med having count(id_med)>3));
```