

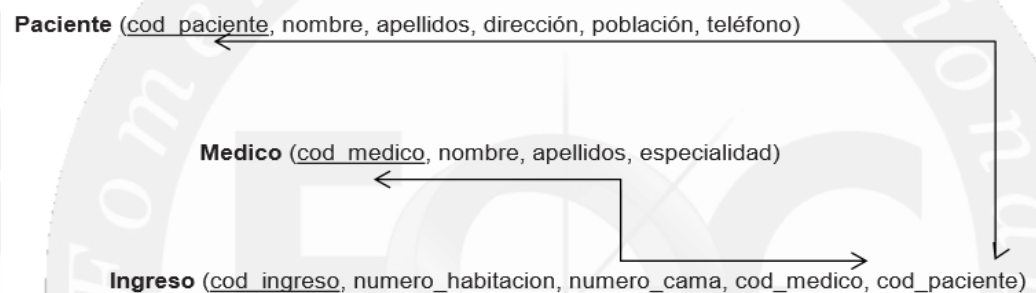
# Módulo: Bases de Datos

## Unidad 5: Tratamiento de Datos

### Sesión 1: Inserción de datos. Guiones

#### Descripción:

Se pide realizar la implementación del siguiente grafo relacional, para ello deberá determinar las sentencias SQL que permitirán crear las tablas asociadas a cada una de las relaciones (tablas) presentes en el grafo relacional, incluyendo las restricciones que crea necesarias y por último insertando, actualizando y borrando datos almacenados en la tabla.



- Crear las tablas especificadas en el grafo relacional e incluir las restricciones que se crean necesarias
- Realizar las siguientes **inserciones** de datos:

- Tabla Paciente:

cod_paciente	nombre	apellidos	dirección	población	teléfono
1	Juan	Fernández	C/Paloma	Granada	635219623
2	José	Perez	C/Mirlo	Granada	698532114
3	Francisco	Ruiz	C/Águila	Granada	NULL
4	María	Cano	C/Perdiz	Málaga	NULL
5	Luisa	Martinez	C/Amapola	Sevilla	625489321

- Tabla Médico

cod_medico	nombre	apellidos	especialidad
1	Olivia	Lorca	Traumatología
2	Adolfo	Salazar	Neurología
3	Pedro	Aguilera	Dermatología

- Tabla Ingreso.

cod_ingreso	numero_habitacion	Numero_cama	Cod_medico	Cod_paciente
1	21	1	1	1
2	13	2	3	2
3	35	2	2	3
4	50	1	3	4

Además, necesitamos crear una tabla con los pacientes de dermatología que llamaremos Paciente\_Derma y tendrá el código, nombre y apellidos de los pacientes que está siendo tratados por médicos con la especialidad dermatología.

### **Criterios de Evaluación:**

- RA04\_a: Se han identificado las herramientas y sentencias para modificar el contenido de la base de datos.
- RA04\_b: Se han insertado, borrado y actualizado datos en las tablas.
- RA04\_c: Se ha incluido en una tabla la información resultante de la ejecución de una consulta.
- RA04\_d: Se han diseñado guiones de sentencias para llevar a cabo tareas complejas.

### **Objetivos:**

- Crear tablas.
- Incluir restricciones sobre las tablas.
- Realizar inserciones de datos.

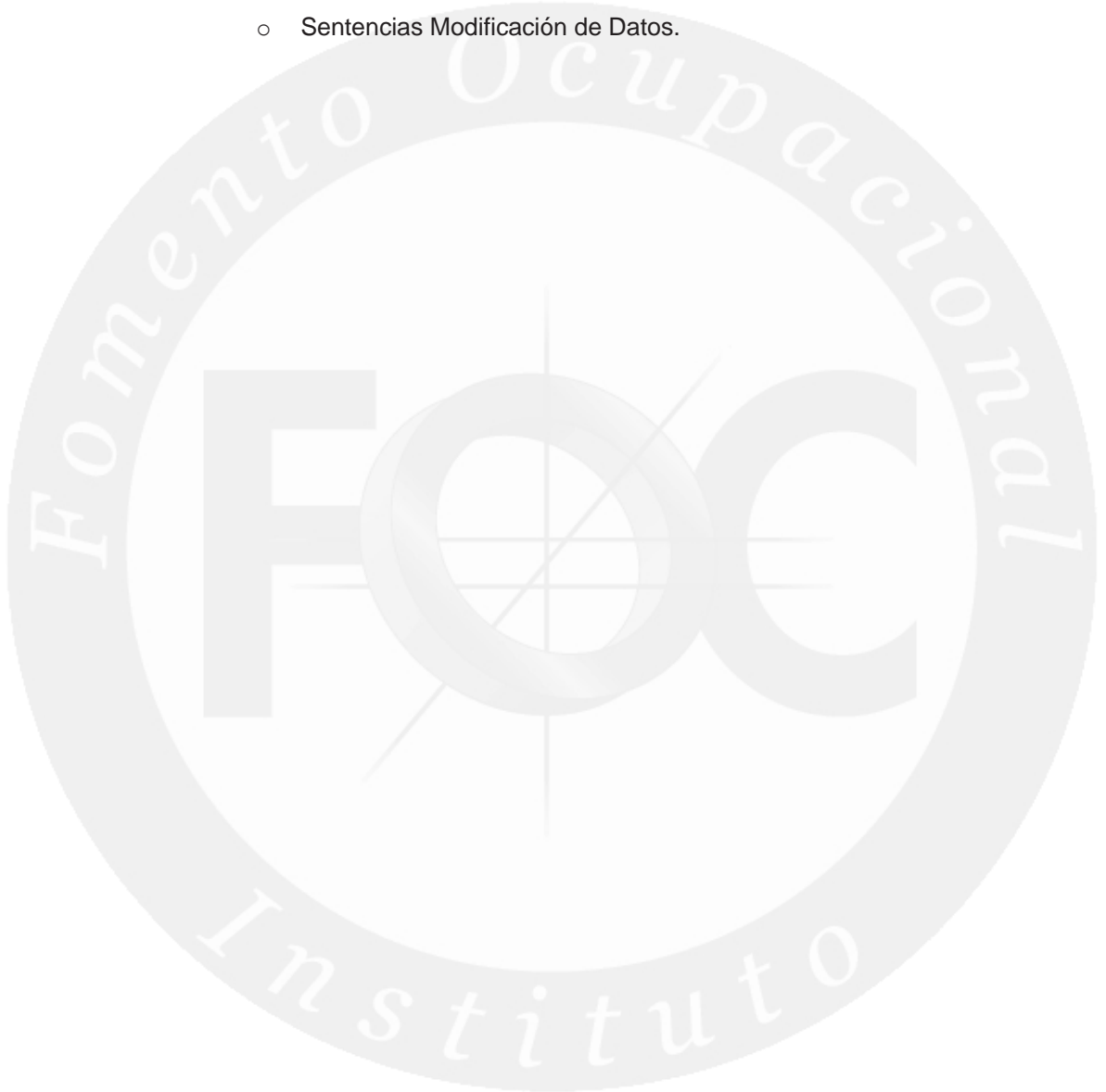
### **Recursos:**

- Acceso a Internet.
- Oracle Express.
- SQL Developer

### **Conceptos a revisar previamente:**

- Realizar el estudio de los apartados de la unidad:
  - Herramientas gráficas proporcionadas por el sistema gestor para la edición de la información.
  - Inserción, borrado y modificación de registros.
  - Guion. Concepto y tipos

- Ver los video conceptos:
  - Insert.
  
- Realizar ejercicio resuelto
  - Sentencias Modificación de Datos.



## Resolución de la práctica:

### Creación de la tabla Paciente

```
CREATE TABLE Paciente (
  cod_paciente NUMBER(4) CONSTRAINT pk_paciente PRIMARY KEY,
  nombre VARCHAR2(50) CONSTRAINT nnull_nombre_paciente NOT NULL,
  apellidos VARCHAR2(50) CONSTRAINT nnull_apellidos_paciente NOT NULL,
  direccion VARCHAR2(50),
  poblacion VARCHAR2(50),
  telefono NUMBER(9),
  CONSTRAINT uk_tlf_paciente UNIQUE (telefono));
```

### Creación de la tabla Médico

```
CREATE TABLE Medico (
  cod_medico NUMBER(4),
  nombre VARCHAR2(50) CONSTRAINT nnull_nombre_medico NOT NULL,
  apellidos VARCHAR2(50) CONSTRAINT nnull_apellidos_medico NOT NULL,
  especialidad VARCHAR2(50) CONSTRAINT nnull_apellidos NOT NULL,
  CONSTRAINT pk_medico PRIMARY KEY (cod_medico));
```

### Creación de la tabla Ingreso

```
CREATE TABLE Ingreso (
  cod_ingreso NUMBER(4) CONSTRAINT pk_ingreso PRIMARY KEY,
  numero_habitacion NUMBER(4) CONSTRAINT nnull_num_hab NOT NULL,
  numero_cama NUMBER(4) CONSTRAINT nnull_num_cama NOT NULL,
  cod_medico NUMBER(4),
  cod_paciente NUMBER(4) CONSTRAINT fk_paciente_ingreso REFERENCES
Paciente(cod_paciente),
  CONSTRAINT ck_numero_habitacion CHECK (numero_habitacion>0),
  CONSTRAINT ck_numero_cama CHECK (numero_cama>0),
  CONSTRAINT fk_medico_ingreso FOREIGN KEY (cod_medico) REFERENCES
Medico(cod_medico));
```

### Inserciones en la tabla Paciente

```
INSERT INTO Paciente (cod_paciente, nombre, apellidos, direccion, poblacion,
telefono) VALUES (1, 'Juan', 'Fernández', 'C/Paloma', 'Granada', 635219623);
INSERT INTO Paciente VALUES (2, 'José', 'Perez', 'C/Mirlo', 'Granada', 698532114);
INSERT INTO Paciente VALUES (3, 'Francisco', 'Ruiz', 'C/Aguila', 'Granada', NULL);
INSERT INTO Paciente (nombre, cod_paciente, apellidos, direccion, poblacion )
VALUES ('María', 4, 'Cano', 'C/Perdiz', 3);
INSERT INTO Paciente (nombre, cod_paciente, apellidos, direccion, poblacion,
telefono) VALUES ('Luisa', 5, 'Martinez', 'C/Amapola', 'Sevilla', 625489321);
```

### Tabla Médico

```

INSERT INTO Medico (cod_medico, nombre, apellidos, especialidad) VALUES
(1,'Olivia','Lorca','Traumatología');
INSERT INTO Medico VALUES (2,'Adolfo','Salazar','Neurología');
INSERT INTO Medico (cod_medico, especialidad, nombre, apellidos) VALUES
(3,'Dermatología','Pedro','Aguilera');

```

#### Tabla Ingreso

```

INSERT INTO Ingreso (cod_ingreso, numero_habitacion, numero_cama, cod_medico,
cod_paciente) VALUES (1,21,1,1,1);
INSERT INTO Ingreso (cod_ingreso, numero_habitacion, numero_cama, cod_medico,
cod_paciente) VALUES (2,13,2,3,2);
INSERT INTO Ingreso (cod_ingreso, numero_habitacion, numero_cama, cod_medico,
cod_paciente) VALUES (3,35,2,2,3);
INSERT INTO Ingreso (cod_ingreso, numero_habitacion, numero_cama, cod_medico,
cod_paciente) VALUES (4,50,1,3,4);

```

#### Tabla de pacientes de dermatología

```

CREATE TABLE Paciente_Derma (
cod_paciente NUMBER(4) CONSTRAINT pk_paciente_derma PRIMARY KEY,
nombre VARCHAR2(50) CONSTRAINT nnull_nombre_paciente_derma NOT NULL,
apellidos VARCHAR2(50) CONSTRAINT nnull_apellidos_paciente_derma NOT NULL
);

```

#### Inserción usando una subconsulta en la tabla de pacientes de dermatología

```

INSERT INTO Paciente_Derma
(SELECT p.cod_paciente, p.nombre, p.apellidos
FROM paciente p JOIN Ingreso i ON (p.cod_paciente=i.cod_paciente)
JOIN medico m on (m.cod_medico=i.cod_medico)
WHERE m.especialidad ='Dermatología')

```