**EXAMEN INTEGRADOR 2025**

**Base de datos de una clínica médica**

**Materia: Bases de Datos**

**Profesor: Nicolás Quiros**

**Integrantes: Flores Leandro, Aciar Agustin, Vera Santiago, Ramirez Rodrigo, Franco Camila.**

# Instrucciones generales

* Este trabajo debe realizarse en **grupos de 4 a 6 integrantes**.
* Deberán **presentar los resultados en una exposición oral de hasta 10 minutos**, donde expliquen las decisiones tomadas, el diseño de la base de datos, los resultados obtenidos y las justificaciones técnicas.
* El trabajo está basado en un escenario realista y deben resolverlo aplicando los conocimientos adquiridos durante la cursada.

# 

# Contexto del caso

Una clínica médica privada desea informatizar la gestión de atención a pacientes. Se requiere un sistema que permita registrar pacientes, turnos médicos, especialidades, tratamientos realizados y resultados de consultas. Además, debe facilitar el análisis del rendimiento médico, cantidad de turnos por especialidad, y seguimiento de pacientes crónicos. También se necesita filtrar la información según especialidades específicas.

Asumiremos que:

* Cada paciente puede tener una sola obra social, o bien puede no tener ninguna.
* Cada médico puede tener una sola especialidad.

*Todas las asumpciones que no estén listadas arriba se pueden manejar como deseen, siempre y cuando las detallen en su presentación oral.*

# Campos requeridos (a distribuir en tablas)

Se provee una lista de campos obligatorios junto con su tipo de dato que deberán formar parte del modelo. Cada grupo decidirá cómo distribuirlos en diferentes tablas y si debe renombrarlos o no. Podrán agregar campos si lo consideran necesario:

| **Nombre del campo** | **Tipo de dato** | **Descripción** |
| --- | --- | --- |
| id\_paciente | INT | Identificador único del paciente. |
| nombre\_paciente | VARCHAR | Primer nombre del paciente. |
| apellido\_paciente | VARCHAR | Apellido nombre del paciente. |
| dni\_paciente | INT | Documento nacional de identidad del paciente. |
| email\_paciente | VARCHAR | Dirección de correo electrónico del paciente. |
| telefono\_paciente | VARCHAR | Número de teléfono del paciente. |
| fecha\_nacimiento\_paciente | DATE | Fecha de nacimiento del paciente. |
| obra\_social\_paciente | VARCHAR | Obra social del paciente. |
| numero\_afiliado\_paciente | VARCHAR | Número de afiliado del paciente en su obra social. |
| historia\_paciente | VARCHAR | Historia clínica del paciente. |
| tiene\_enfermedad\_cronica\_paciente | VARCHAR | Booleano que adquiere valor de “True” si el paciente tiene una enfermedad crónica y “False” en caso contrario. |
| id\_medico | INT | Identificador único del médico. |
| nombre\_medico | VARCHAR | Primer nombre del médico. |
| apellido\_medico | VARCHAR | Apellido nombre del médico. |
| matricula\_medico | VARCHAR | Número de matrícula del médico. |
| especialidad\_medico | VARCHAR | Especialidad del médico. |
| id\_turno | INT | Identificador único del turno médico. |
| fecha\_turno | DATE | Fecha en la que el paciente asistirá o asistió al turno médico. |
| hora\_turno | TIME | Horario en el que el paciente asistirá o asistió al turno médico. |
| estado\_turno | VARCHAR | Estado del turno médico. Sólo puede tomar los valores “Asignado”, “Atendido” o “Cancelado”. |
| motivo\_turno | VARCHAR | Motivo por el cual el paciente asiste al turno médico. |
| id\_secretario | VARCHAR | Identificador único del secretario a cargo de asignar el tuno médico. |
| nombre\_secretario | VARCHAR | Primer nombre del secretario. |
| apellido\_secretario | VARCHAR | Apellido del secretario. |
| sede\_secretario | VARCHAR | Sede de la clínica donde trabaja el secretario (Ciudad de Mendoza, Luján de Cuyo, etc). |

# Actividades

## 1. Diseño conceptual y lógico

1. Definir qué tablas van a construir. Especificar para cada tabla:
   * Su(s) clave(s) primaria(s).
   * Su(s) clave(s) foránea(s).
   * Qué restricciones o contraints se deben utilizar. Por ejemplo: “el campo X no puede ser nulo, el campo Y solo puede tener estos valores determinados, etc”.
   * Su forma normal. Justificar.
   * Decidir si representa un hecho o una dimensión. Justificar.
2. Realizar el diagrama entidad-relación (ER), incluyendo:
   * Entidades.
   * Al menos 3 atributos por entidad.
   * Relaciones, detallando cardinalidad y ordinalidad.

Se permite y sugiere trabajar con datos inventados para ejemplificar el funcionamiento de las estructuras propuestas.

## 1. Diseño conceptual y lógico

### **Tablas principales**

#### Tabla: **paciente**

* Clave primaria: id\_paciente
* Claves foráneas: id\_obra\_social → obra\_social.id\_obra\_social
* Restricciones: dni y email son UNIQUE, nombre, apellido, dni y fecha\_nac son NOT NULL
* Forma normal: 3FN (todos los atributos dependen solo de la clave primaria y no entre ellos)
* Tipo: Dimensión (describe entidades, no hechos)

#### 

#### 

#### Tabla: **obra\_social**

* Clave primaria: id\_obra\_social
* Restricciones: nombre es UNIQUE y NOT NULL
* Forma normal: 3FN
* Tipo: Dimensión

#### Tabla: **medico**

* Clave primaria: id\_medico
* Clave foránea: id\_especialidad → especialidad.id\_especialidad
* Restricciones: matricula es UNIQUE, todos los campos son NOT NULL
* Forma normal: 3FN
* Tipo: Dimensión

#### Tabla: **especialidad**

* Clave primaria: id\_especialidad
* Restricciones: nombre es UNIQUE y NOT NULL
* Forma normal: 3FN
* Tipo: Dimensión

#### Tabla: **turno**

* Clave primaria: id\_turno
* Claves foráneas: id\_paciente, id\_medico, id\_secretario, id\_estado, id\_sede
* Restricciones: fecha, hora, estado son NOT NULL, estado con CHECK si no se normaliza
* Forma normal: 3FN
* Tipo: Hecho (representa acciones medibles)

#### Tabla: **estado**

* Clave primaria: id\_estado
* Restricciones: nombre es UNIQUE y NOT NULL
* Tipo: Dimensión (catálogo para evitar errores y normalizar)

#### Tabla: **secretario**

* Clave primaria: id\_secretario
* Clave foránea: id\_sede
* Tipo: Dimensión

#### Tabla: **sede**

* Clave primaria: id\_sede
* Tipo: Dimensión

#### Tabla: **enfermedades\_cronicas**

* Clave primaria: id\_enfermedad
* Tipo: Dimensión

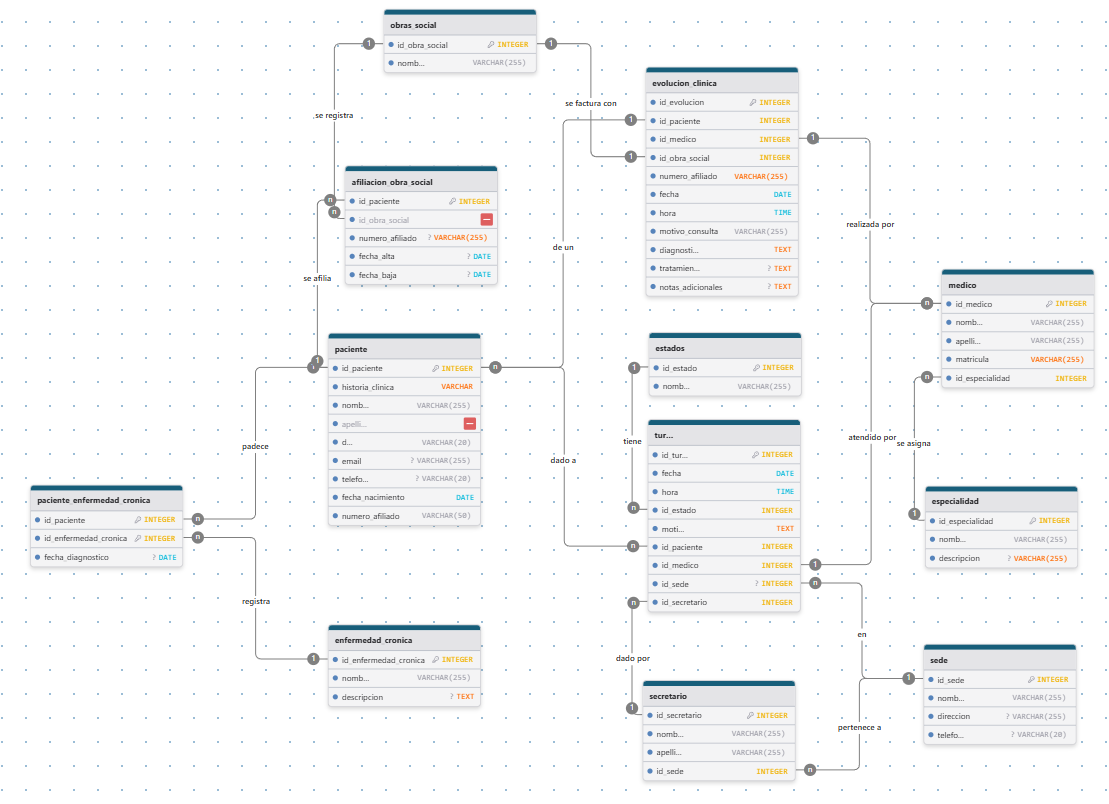
#### Tabla: **pacientes\_enfermedades\_cronicas**

* Clave primaria compuesta: (id\_paciente, id\_enfermedad)
* Tipo: Relación N:M

#### Tabla: **evoluciones\_clinicas**

* Clave primaria: id\_evolucion
* Tipo: Hecho (seguimiento médico)

b)



<https://github.com/Coders-UTN/integrador-bd-clinica-medica/blob/main/diagrama-clinica-medica.svg>

## 2. Modelado dimensional

1. Partiendo del modelo de entidad-relación realizado en el punto anterior, realizar el modelo de hechos y dimensiones de la base de datos. Incluir en el diagrama:
   * Clave(s) primaria(s) de cada tabla.
   * Clave(s) foránea(s) de cada tabla.
   * Relaciones, detallando cardinalidad y ordinalidad.
2. Explicar si el modelo resultante sigue el esquema de estrella o de copo de nieve. Justificar.
3. Incluir el código SQL de creación de al menos dos tablas relevantes, con sus respectivos tipos de datos y restricciones (claves primarias, claves foráneas, NOT NULL, etc.).

## 2. a) Modelo de hechos y dimensiones:

* Hecho central: **fact\_turno**
* Dimensiones: **dim\_paciente**, **dim\_medico**, **dim\_especialidad**, **dim\_sede**, **dim\_fecha**, **dim\_estado**
* Relaciones: 1:N desde cada dimensión a **fact\_turno**
* Tipo de esquema: Estrella (dimensiones desnormalizadas)

b) El modelo resultante sigue el esquema de copo de nieve.

Ya que no hay datos duplicados en las tablas de dimensiones cada tabla tiene su propia clave primaria y se relacionan mediante su clave foránea.

Se prioriza la normalización y no la facilidad de acceso y las dimensiones están divididas en tablas relacionadas como lo son paciente con una obra social o medico con una especialidad.

c) Dos tablas relevantes:

CREATE TABLE paciente (

id\_paciente INT PRIMARY KEY AUTO\_INCREMENT,

nombre VARCHAR(50) NOT NULL,

apellido VARCHAR(50) NOT NULL,

dni VARCHAR(20) UNIQUE NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

fecha\_nac DATE NOT NULL,

id\_obra\_social INT REFERENCES obra\_social(id\_obra\_social)

);

—-----------------------------------------------------------------------------------------

CREATE TABLE turno (

id\_turno INT PRIMARY KEY AUTO\_INCREMENT,

id\_paciente INT REFERENCES paciente(id\_paciente),

id\_medico INT REFERENCES medico(id\_medico),

id\_secretario INT REFERENCES secretario(id\_secretario),

id\_estado INT REFERENCES estado(id\_estado),

id\_sede INT REFERENCES sede(id\_sede),

fecha DATE NOT NULL,

hora TIME NOT NULL,

motivo TEXT

);

3. Vistas

1. Crear un código de SQL que genere dos vistas específicas:
   * Una destinada a los médicos, donde se puedan ver todos los datos del paciente.

Vista para médicos

CREATE VIEW vw\_medico\_pacientes AS

SELECT \* FROM paciente;

Vista de evoluciones Clínicas para médicos

CREATE VIEW evoluciones\_para\_medicos AS

SELECT

ec.id\_evolucion,

p.dni,

p.historia\_clinica,

p.apellido,

p.nombre,

p.fecha\_nacimiento

ec.fecha AS fecha\_evolucion,

ec.hora,

ec.motivo\_consulta,

ec.diagnostico,

ec.tratamiento,

ec.notas\_adicionales

FROM

evolucion\_clinica AS ec

LEFT JOIN

paciente AS p

ON

ec.id\_paciente = p.id\_paciente;

LEFT JOIN

obra social AS os

ON

os.id\_obra\_social = ec.id\_obra\_social

* + Otra destinada a los secretarios, donde se puedan ver todos los datos del paciente excepto su historia clínica.

Vista evoluciones clínicas para secretarios

CREATE VIEW evoluciones\_para\_administrativos AS

SELECT

ec.id\_evolucion

p.dni,

p.apellido,

p.nombre,

p.fecha\_nacimiento,

p.telefono,

p.mail,

os.nombre AS obra\_social,

p.numero\_afiliado

ec.fecha AS fecha\_evolucion,

ec.hora,

ec.motivo\_consulta,

ec.diagnostico

FROM

evolucion\_clinica AS ec

LEFT JOIN

paciente AS p

ON

ec.id\_paciente = p.id\_paciente;

LEFT JOIN

obra social AS os

ON

os.id\_obra\_social = ec.id\_obra\_social;

Vista para secretarios

CREATE VIEW vw\_secretario\_pacientes AS

SELECT

id\_paciente,

nombre,

apellido,

dni,

email,

fecha\_nac,

id\_obra\_social

FROM

paciente;

1. Justificar por qué es necesario usar vistas para este caso, en lugar de simplemente dejar que los estudiantes consulten las tablas originales.

Es necesario usar vistar en este caso por que estas permiten controlar el acceso a la información sensible (como tratamiento, patologias y notas adicionales medicas) dependiendo del rol del usuario, evitando errores o accesos indebidos a las tablas base.

## 4. Consultas analíticas

Elegir 3 de las siguientes preguntas y creen el código de SQL para responderlas:

* ¿Cuál fue la especialidad con más turnos cancelados en el año 2023?

SELECT

esp.nombre AS especialidad,

COUNT(t.id\_turno) AS total\_cancelados

FROM turno t

JOIN medico m ON t.id\_medico = m.id\_medico

JOIN especialidad esp ON m.id\_especialidad = esp.id\_especialidad

JOIN estado e ON t.id\_estado = e.id\_estado

WHERE

e.nombre = 'Cancelado'

AND YEAR(t.fecha) = 2023

GROUP BY esp.nombre

ORDER BY total\_cancelados DESC

LIMIT 1;

* ¿Qué médicos atendieron a más de 100 pacientes distintos?

SELECT

m.id\_medico,

m.nombre,

m.apellido,

COUNT(DISTINCT t.id\_paciente) AS pacientes\_unicos

FROM turno t

JOIN medico m ON t.id\_medico = m.id\_medico

JOIN estado e ON t.id\_estado = e.id\_estado

WHERE e.nombre = 'Atendido'

GROUP BY m.id\_medico, m.nombre, m.apellido

HAVING COUNT(DISTINCT t.id\_paciente) > 100;

* ¿Cuál es el porcentaje de pacientes con enfermedades crónicas?

SELECT

ROUND(

(SELECT COUNT(DISTINCT id\_paciente) FROM pacientes\_enfermedades\_cronicas)

\* 100.0 /

(SELECT COUNT(\*) FROM paciente),

2

) AS porcentaje\_con\_enfermedades\_cronicas;

* ¿Cuál es la obra social más utilizada en la clínica? ¿Y la menos utilizada?
* ¿En qué sede se realizaron más consultas durante el año 2020?

## 5. Visualización de resultados (extra)

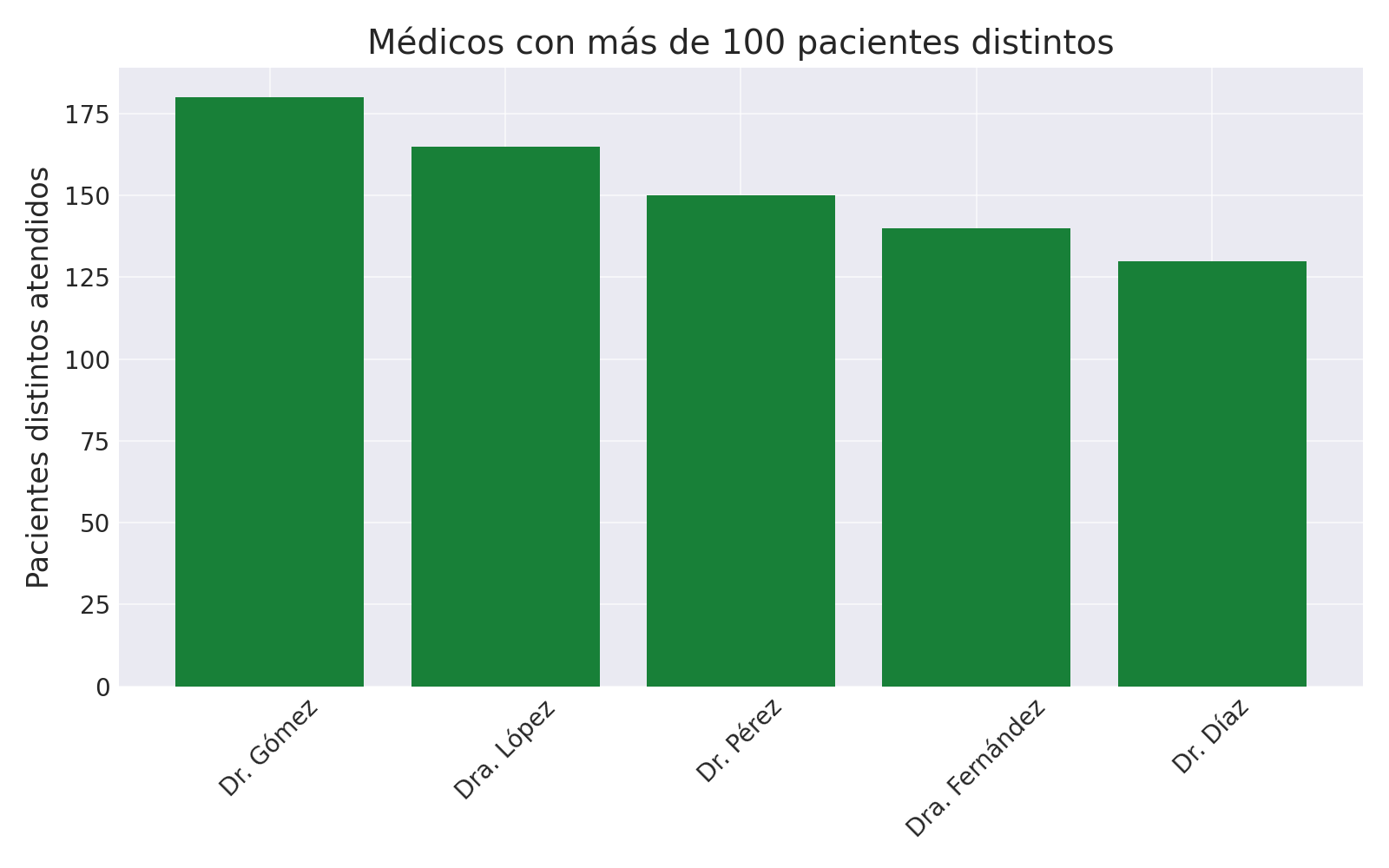
Sugerir **qué tipos de visualizaciones gráficas** podrían utilizar para representar los resultados obtenidos en la sección 4. Justificar la elección. Algunos ejemplos de visualización pueden ser:

* Gráfico de barras.
* Gráfico de torta.
* Gráfico de línea.
* Serie temporal.
* Mapa de calor.
* Tabla dinámica.

Para el primer resultado obtenido de la sección 4 sugerimos el gráfico de barras horizontales,permite ver cual es la especialidad con más turnos cancelados y también las que le siguen con poca diferencia.



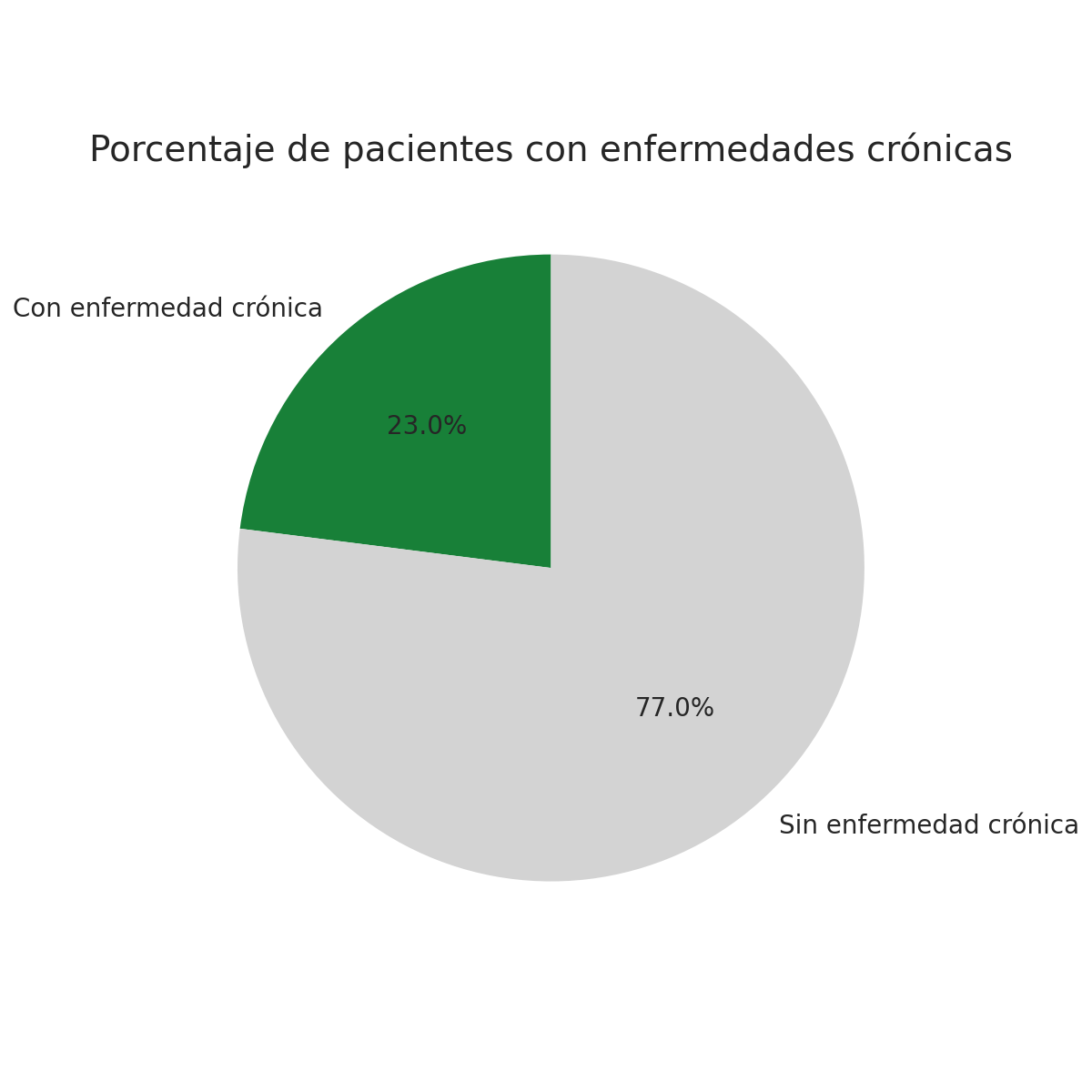
Para el segundo resultado sugerimos usar gráfico de columnas o barras verticales:



En este se puede ver claramente como todos los médicos superan los 100 pacientes distintos. Además se aprecia el orden descendente de pacientes únicos según cada médico.

Al igual que en el primer resultado,este tipo de gráficos son muy buenos para mostrar la información de manera sencilla.

Para el último resultado de ejemplo sugerimos utilizar un gráfico de torta:



Es una forma simple e intuitiva de mostrar porcentajes, y permite ver claramente la diferencia entre los pacientes con enfermedades crónicas y los que no..