

## Modelos de datos

### Ejercicio 1

Modelo semiestructurado

Deberás leer el archivo csv de la práctica 1 y pasarlo a alguna estructura del modelo semiestructurado (json, xml, html etcétera), recomendando json por ser el más usado.

#### Archivo CSV

ID	Nombre	Categoría	Precio	Cantidad en Existencia
AMOANA	Amoxicilina	Antibioticos	150.0	24
IBUAOS	Ibuprofeno	Antibioticos	50.0	50
OMEOS	Omeprazol	Otros	20.0	30
ASPAOS	Aspirina	Analgesicos	40.0	60
VITAOS	Vitaminas	Alimentos	15.0	200
AGUBAS	Agua Mineral	Bebidas	5.0	120
CREHTE	Crema Hidratante	Desinflamantes	25.0	40
SUEBAS	Suerox	Bebidas	15.5	10
LLAOOS	Llavero	Otros	15.5	90
PALAOOS	Paletas	Alimentos	12.0	30

#### Archivo JSON

```
1  {
2      [
3          {
4              "ID": "AMOANA",
5              "Nombre": "Amoxicilina",
6              "Categoría": "Antibioticos",
7              "Precio": 150.0,
8              "Cantidad en Existencia": 24
9          },
10         {
```

```
11         "ID": "IBUAOS",
12         "Nombre": "Ibuprofeno",
13         "Categoria": "Antibioticos",
14         "Precio": 50.0,
15         "Cantidad en Existencia": 50
16     },
17     {
18         "ID": "OMEOOS",
19         "Nombre": "Omeprazol",
20         "Categoria": "Otros",
21         "Precio": 20.0,
22         "Cantidad en Existencia": 30
23     },
24     {
25         "ID": "ASPAOS",
26         "Nombre": "Aspirina",
27         "Categoria": "Analgesicos",
28         "Precio": 40.0,
29         "Cantidad en Existencia": 60
30     },
31     {
32         "ID": "VITAOS",
33         "Nombre": "Vitaminas",
34         "Categoria": "Alimentos",
35         "Precio": 15.0,
36         "Cantidad en Existencia": 200
37     },
38     {
39         "ID": "AGUBAS",
40         "Nombre": "Agua Mineral",
41         "Categoria": "Bebidas",
42         "Precio": 5.0,
43         "Cantidad en Existencia": 120
44     },
45     {
46         "ID": "CREHTE",
47         "Nombre": "Crema Hidratante",
48         "Categoria": "Desinflamantes",
```

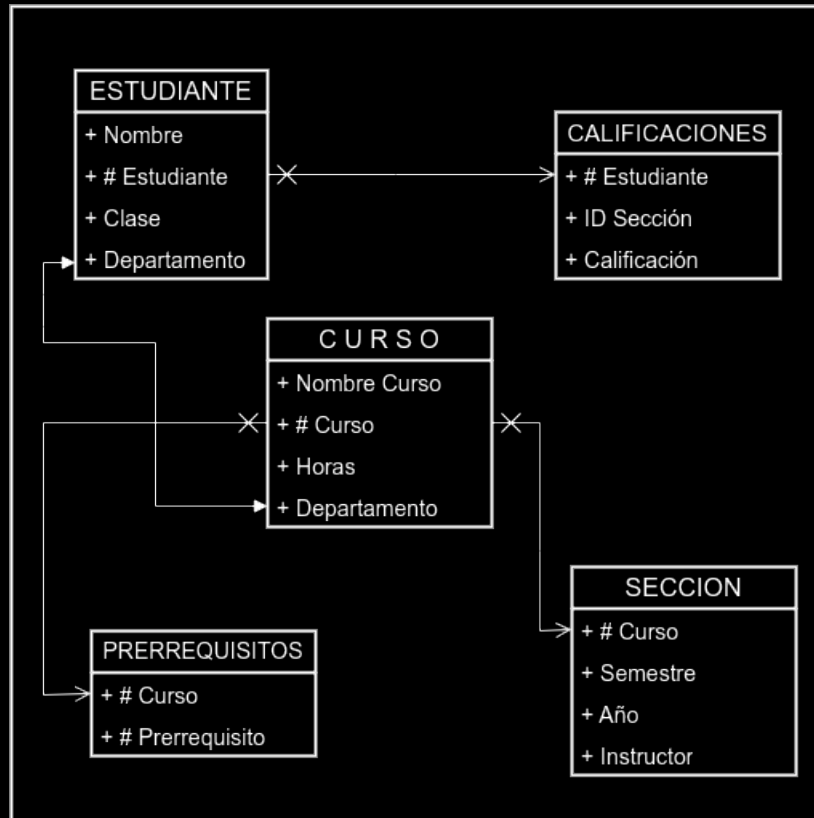
```
49         "Precio": 25.0,  
50         "Cantidad en Existencia": 40  
51     },  
52     {  
53         "ID": "SUEBAS",  
54         "Nombre": "Suerox",  
55         "Categoria": "Bebidas",  
56         "Precio": 15.5,  
57         "Cantidad en Existencia": 10  
58     },  
59     {  
60         "ID": "LLAOOS",  
61         "Nombre": "Llaverero",  
62         "Categoria": "Otros",  
63         "Precio": 15.5,  
64         "Cantidad en Existencia": 90  
65     },  
66     {  
67         "ID": "PALAOS",  
68         "Nombre": "Paletas",  
69         "Categoria": "Alimentos",  
70         "Precio": 12.0,  
71         "Cantidad en Existencia": 30  
72     }  
73 ]  
74 }
```

---

## Ejercicio 2

### Modelo relacional

Basándote en las figuras 1.2 y 2.1 diseña un esquema de base de datos y coloca algunos registros de ejemplo sobre un universo de discurso de tu elección.



## Ejercicio 3

Completa el modelo con los siguientes aspectos:

- ¿Qué tipo de información adicional y restricciones podrías representar en este esquema? Inventa al menos 5 restricciones y agrega al menos 5 campos distribuidos en distintas tablas, también si lo consideras necesario, puedes agregar una tabla adicional.
  - **Restricción de Unicidad en Departamentos:** Agregar una restricción única para el campo *Departamento* en la tabla de *Departamento* para garantizar que no haya departamentos con el mismo nombre.
  - **Fecha de Inicio en Secciones:** Agregar un campo *Fecha de Inicio* en la tabla de *Sección* para registrar la fecha en que una sección comienza.
  - **Restricción de Calificaciones:** Agregar una restricción de rango en la tabla de *Calificaciones* para garantizar que las calificaciones estén dentro del rango específico (A, B, C, D, F).

- **Restricción de Edad en Estudiantes:** Agregar un campo *Fecha de Nacimiento* en la tabla de *Estudiante* para calcular automáticamente la edad y aplicar una restricción de edad mínima.
- **Tabla de Profesores:** Agregar una nueva tabla llamada *Profesor* que almacene información sobre los profesores. Se relaciona con la tabla de *Sección* para saber quién imparte cada sección.

## Tablas

### Tabla estudiante

Nombre	Numero_Estudiante	Clase	Departamento	Fecha_Nacimiento
Juan	4	1	CS	1990 – 03 – 15
Luis	17	2	CS	1992 – 08 – 20

### Tabla Curso

Nombre_Curso	Numero_Curso	Horas	Departamento
Introducción a Computación	CS1310	4	CS
Estructura de datos	CS3320	4	CS
Matemáticas Discretas	MATH2410	3	MATH
Bases de datos	CS3380	3	CS

### Tabla Sección

Numero_Curso	Semestre	Año	Profesor	Fecha_Inicio
MATH2410	Otoño	07	Yoda	2023 – 09 – 01
CS1310	Primavera	08	Profesor X	2023 – 01 – 15

### Tabla Calificaciones

Número_Estudiante	ID_Seccion	Calificacion
2	1	A
1	2	B

Tabla Prerrequisitos

Numero_Curso	Numero_Prerrequisito
CS1310	CS3320

Tabla Profesor

Nombre	Numero Identificacion	Departamento
Profesor X	12345	CS
Yoda	67890	MATH

2. Menciona al menos 3 usuarios de tu base de datos, define los privilegios que cada uno tendría en la BD y describe el tipo de información que cada uno de ellos utilizaría.
- **Administrador de la Base de Datos:** Los administradores tienen privilegios completos sobre la base de datos. Puede crear, modificar y eliminar tablas, así como otorgar permisos a otros usuarios. Su función es mantener la integridad y la estructura de la base de datos.
  - **Profesor:** Los profesores tienen acceso a la información de las tablas *Sección* y *Calificaciones*. Tienen permiso para ver y actualizar las calificaciones de los estudiantes en las secciones que imparten. También pueden consultar información sobre las secciones que están a cargo.
  - **Departamento de Admisiones:** Ellos tiene acceso a la tabla *Estudiante* para registrar nuevos estudiantes en la base de datos. También puede consultar información sobre los departamentos y cursos ofrecidos.

## Cuestionario

1. ¿Cuáles son las 4 principales características de los modelos orientados a objetos?
- **Encapsulación:** Protege y controla el acceso a los datos en la base de datos, asegurando su integridad y privacidad.
  - **Herencia:** Permite crear nuevas estructuras de datos basadas en las existentes, promoviendo la reutilización y la organización eficiente de información.
  - **Abstracción:** Simplifica la representación de objetos del mundo real en la base de datos, centrándose en los aspectos clave y ocultando detalles innecesarios.
  - **Polimorfismo:** Permite tratar diferentes tipos de datos de manera uniforme a través de interfaces comunes, facilitando consultas y operaciones flexibles.

## 2. ¿Cuáles son las 4 principales propiedades de una relación?

- **Cardinalidad:** Describe el número de instancias de una entidad que pueden estar relacionadas con el número de instancias de otra entidad. Las posibles cardinalidades incluyen 1:1, 1:N, N:1 y M:N.
- **Participación:** Indica si la existencia de una entidad depende de su participación en una relación. Puede ser total (obligatoria) o parcial (opcional).
- **Atributos de relación:** Son atributos asociados a la relación misma, que almacenan información adicional sobre la relación en sí.
- **Grado de relación:** Indica cuántas entidades están involucradas en la relación. Las relaciones binarias implican dos entidades, mientras que las relaciones ternarias involucran tres, y así sucesivamente.

## 3. ¿Cuál es la principal diferencia entre el modelo entidad relación y el modelo semiestructurado?

El modelo entidad-relación (ER) se basa en una **estructura de datos tabular** y utiliza relaciones bien definidas entre entidades y atributos. Mientras el modelo semiestructurado es más flexible y permite **datos con estructura variable**, (JSON o XML), que no siguen una estructura tabular rígida. Entonces la diferencia radica la estructura de datos.

## 4. Nombra 3 ejemplos en el cual se utilice el modelo orientado a objetos.

### ■ **Gestión de Biblioteca:**

Para hacer la gestión de una biblioteca, los libros, los usuarios y los préstamos se modelan como objetos. La encapsulación se utiliza para proteger la información de los usuarios, la herencia se aplica para crear clases de libros basadas en género o autor, el polimorfismo permite a diferentes tipos de usuarios realizar operaciones de préstamo y devolución de libros.

### ■ **Videojuegos:**

En videojuegos modernos, los personajes, objetos, enemigos y entornos se modelan como objetos en un mundo virtual. Cada objeto tiene atributos y comportamientos específicos, lo que permite una simulación realista y dinámica del juego. Por ejemplo, un personaje en un juego de rol es un objeto con atributos como salud, nivel y habilidades.

### ■ **Gestión de Archivos Multimedia:**

En aplicaciones de gestión de activos digitales como sistemas de transmisión de contenido, se utilizan bases de datos orientadas a objetos para almacenar y organizar archivos multimedia, como imágenes, audio y video. Esto permite la indexación eficiente de metadatos y facilita la búsqueda y recuperación de contenido multimedia.

5. **Nombra 3 ejemplos en el cual se utilice el modelo entidad relación.**

■ **Ventas:**

En un sistema de ventas, se utiliza el modelo entidad-relación para representar clientes, productos y pedidos. Facilitando el seguimiento de las transacciones y ver el inventario

■ **Museos:**

Para museos o exposiciones de arte, el sistema maneja los registros de obras, exposiciones y visitantes dando seguimiento detallado a lo que entra y sale y líneas de tiempo.

■ **Recursos Humanos:**

Dentro de una empresa el poder llevar las relaciones internas ayudan a administrar quién trabaja en qué departamento, quién reporta a quién y el registro de horas trabajadas.

6. **Nombra 3 ejemplos en el cual se utilice el modelo semiestructurado.**

■ **Aplicaciones Web: (JSON):**

Los datos en JSON son fáciles de transmitir entre un servidor web y un navegador, lo que lo convierte en un formato ideal para el intercambio de datos en aplicaciones web. Se utiliza para representar información como configuraciones de usuario, respuestas de API y datos de formularios.

■ **Datos Empresariales (XML):**

Se emplea en la representación de datos estructurados y no estructurados en documentos, informes y flujos de trabajo empresariales.

■ **Bases de Datos NoSQL:**

Las bases de datos NoSQL, como MongoDB y CouchDB, almacenan datos en un formato semiestructurado permitiendo gestionar datos variados, como documentos JSON, datos de sensores, registros de redes sociales sin requerir una estructura de tabla rígida.