



# Trabajo Práctico

Trabajar en forma: **Grupal** ▾

Entrega: **Individual** ▾

Fecha de presentación y entrega final: 19 de noviembre 7:00 p.m.

## Detalle de trabajo

### Aplicación de Gestión de Inventario

#### Objetivo:

Diseñar una base de datos sencilla para gestionar un inventario de productos y desarrollar una aplicación de escritorio en Python que permita realizar operaciones básicas de CRUD (Crear, Leer, Actualizar, Eliminar) sobre la base de datos.

#### Descripción:

Los alumnos deberán:

##### 1. Diseñar la base de datos:

- Identificar las entidades relevantes (productos, categorías, etc.).
- Definir los atributos de cada entidad (nombre, descripción, precio, stock, etc.).
- Establecer las relaciones entre las entidades.
- Crear un diagrama Entidad-Relación (E-R) para visualizar el diseño.
- Implementar la base de datos en un sistema de gestión de bases de datos (MySQL, PostgreSQL, SQLite).

##### 2. Desarrollar la aplicación en Python:

- Utilizar una librería para la conexión a la base de datos (e.g., `mysql.connector` para MySQL, `psycopg2` para PostgreSQL, `sqlite3` para SQLite).
- Crear una interfaz gráfica de usuario (GUI) sencilla utilizando una librería como Tkinter.



- Implementar las siguientes funcionalidades:
  - **Cargar datos:** Permitir al usuario ingresar nuevos productos al inventario con sus datos correspondientes.
  - **Modificar datos:** Permitir al usuario editar la información de un producto existente.
  - **Listar datos:** Mostrar una lista de todos los productos del inventario con sus detalles.
  - **Eliminar datos:** Permitir al usuario eliminar un producto del inventario.
- Incluir un menú sencillo para navegar entre las diferentes funcionalidades.
- Manejar errores y excepciones para asegurar la robustez de la aplicación.

### Ejemplo de menú de la aplicación:

----- Menú de opciones -----

1. Cargar nuevo producto
2. Modificar producto existente
3. Listar productos
4. Eliminar producto
5. Salir

### Aclaración sobre la interfaz gráfica:

Si desean crear una interfaz gráfica de usuario (GUI) para la aplicación, pueden utilizar alguna de las siguientes librerías:

- **Tkinter:** Es la librería estándar de Python para crear GUIs. Es relativamente sencilla de usar y está incluida en la mayoría de las distribuciones de Python.
  - **Recomendada si:** Buscan una opción simple y rápida para crear una interfaz básica.
  - **Recursos:** Tutoriales y documentación de Tkinter disponibles en la web.
- **PyQt:** Es una librería más completa y robusta que Tkinter, que ofrece una amplia variedad de widgets y herramientas para crear interfaces gráficas más avanzadas.
  - **Recomendada si:** Desean una interfaz con más funcionalidades y un aspecto más profesional.
  - **Recursos:** Documentación oficial de PyQt, tutoriales y ejemplos en línea.



- **Kivy:** Es una librería multiplataforma (Windows, macOS, Linux, Android, iOS) que permite crear interfaces gráficas modernas con un enfoque en el diseño táctil.
  - **Recomendada si:** Quieren una interfaz con un diseño atractivo y adaptable a diferentes dispositivos.
  - **Recursos:** Documentación oficial de Kivy, tutoriales y ejemplos en línea.