

# VENTAS

*de artículos*

Programa para organizar las ventas de los artículos en una tienda en MongoDB

**REALIZADO POR:**

Alex Ferrandis Ros

**SUPERVISADO POR:**

Juan Bautista Talents

# ÍNDICE

## 01 Introducción

1.1 Introducción	03
------------------	----

## 02 Herramientas y métodos

2.1 Herramientas	04
2.2 Métodos	04

## 03 Perspectiva Estática

3.1 E/R (Entidad-Relación)	05
3.2 Pasos a Document	05
3.3 DDL	06
3.4 DML	06
3.5 DQL	06
3.6 DCL	06

## 04 Perspectiva Dinámica

4.1 Sketch	07
4.2 Casos de Uso (Métodos)	07

## 05 Conclusiones

5.1 Resumen de los resultados	08
5.2 Reflexión y futuras mejoras	08

## 06 Bibliografía y Webgrafía

6.1 Referencias utilizadas	08
6.2 Enlaces a recursos	08

# INTRODUCCIÓN

El programa proporcionado es un sistema de gestión de inventario para una tienda, escrito en Python. Permite a los usuarios realizar varias operaciones relacionadas con los artículos de la tienda, como introducir nuevos artículos, realizar ventas, mostrar información detallada, borrar artículos individuales o todos los artículos, y salir del programa.

La clase `Article` se utiliza para crear objetos que representan los artículos de la tienda, con atributos para el nombre, precio y cantidad vendida.

El programa también incluye una serie de funciones para interactuar con el usuario y realizar las operaciones mencionadas. Por ejemplo, la función `menú` muestra las opciones disponibles y recoge la elección del usuario, mientras que la función `mostrarTabla` muestra una tabla con la información de todos los artículos, incluyendo el total de ingresos y el artículo más vendido.

El bucle principal del programa permite a los usuarios seleccionar una opción del menú hasta que decidan salir, asegurándose de que la opción sea válida y ejecutando la acción correspondiente.

Es un programa interactivo que facilita la gestión de un inventario de una manera sencilla y estructurada.

# HERRAMIENTAS

Este programa se encarga de organizar los artículos de una tienda, introducir artículos nuevo, hacer ventas, mostrar información sobre el artículo y borrar artículos.

He utilizado Visual Studio Code para hacer el programa en python, para la base de datos he usado el SQLite, además de las diferentes paginas webs para informarme.

# MÉTODOS

El introducir artículos nuevos se encarga de introducir la información del artículo nuevo, como el nombre y precio.

Hacer ventas se encarga de ir incrementando el numero de ventas de el artículo vendido y calcular los beneficios con ese artículo.

Mostrar información, muestra una tabla con todos los artículos y su información ordenada además de el beneficio total.

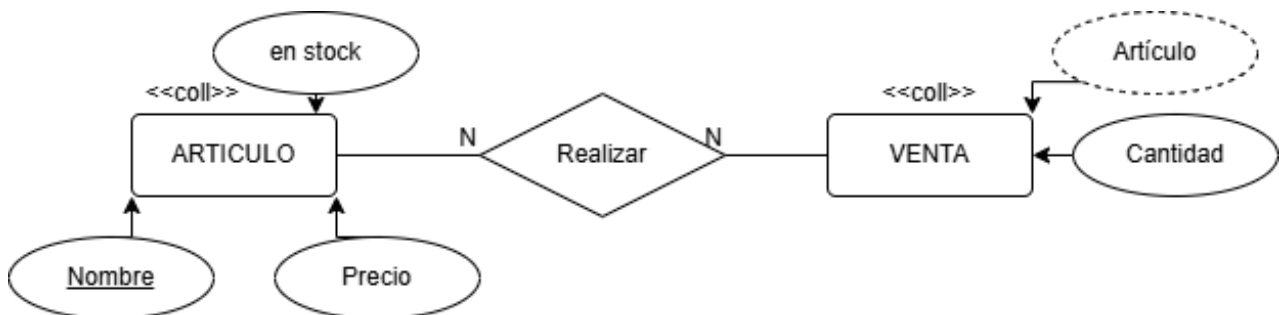
Borrar un artículo , básicamente borra la información y el artículo que se le indique en el momento.

Borrar todos los artículos, simplemente se encarga de borrar todos los artículos en la base de datos.

Salir, solo es para salir del programa una vez no tengas nada que hacer.

# PERSPECTIVA ESTÁTICA

E/R (Entidad-Relación)



Pasos a documentos

```
// Document de la col·lecció "article"
{
  "_id": ObjectId("32f24g"),
  "nom": "Airpods",
  "preu": 100.0,
  "stock": 50
}
```

```
// Document de la col·lecció "vendes"
{
  "_id": ObjectId("frgg542"),
  "article": ObjectId("32f24g"),
  "quantitat": 10
}
```

# PERSPECTIVA ESTÁTICA

DDL (Data Definition Language)

El DDL de mi base de datos es el mismo concepto que el implementado anteriormente ya que no ha habido ningún cambio significativo en la base de datos y no he visto que haga falta.

DML (Data Manipulation Language)

Insertar Datos en "ARTICULO":

```
articles.insert_one({'nom': article, 'preu': 3.5, 'stock': 10})
```

Insertar Datos en "VENTA":

```
vendes.insert_one({'nom': article, 'quantitat': 3})
```

Actualizar Datos en "ARTICULO":

```
articles.update_one(
    {'nom': article_nom},
    {'$inc': {'stock': -quantitat}}
)
```

DQL (Data Query Language)

```
articles = coleccion.find({})
```

DCL (Data Control Language)

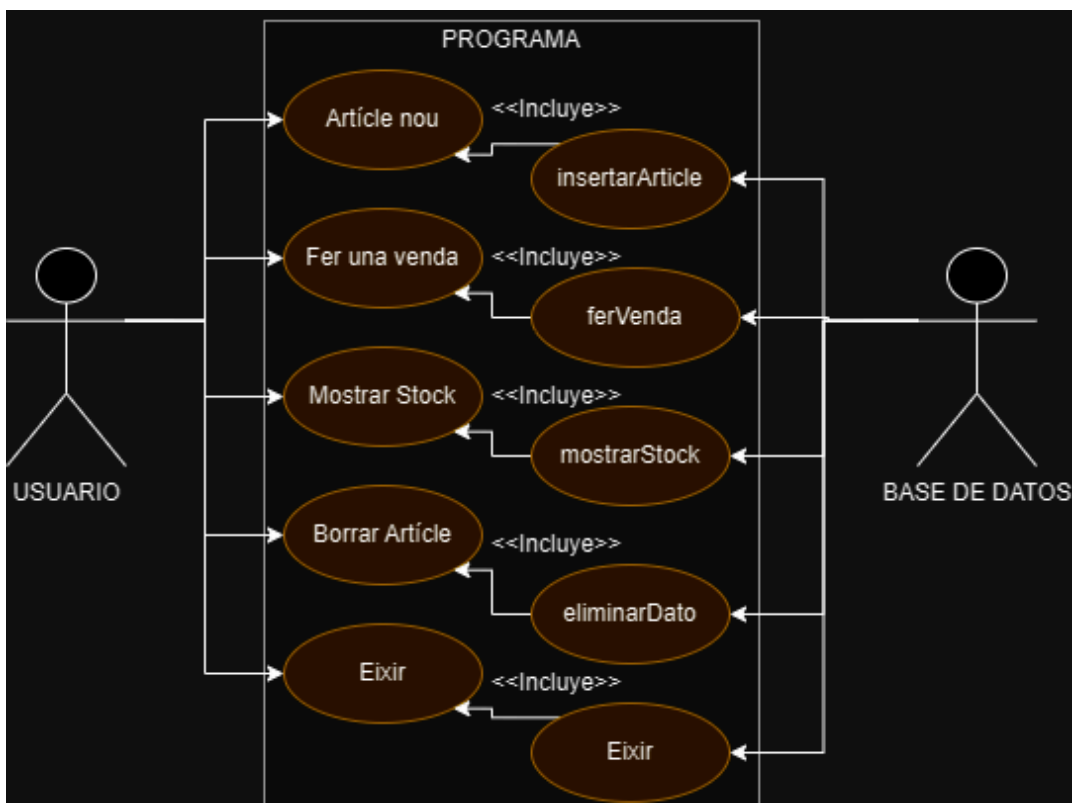
En mi caso no uso ningún comando de control de datos en mi base de datos

# PERSPECTIVA DINAMICA

SKETCH:



CASOS DE USO:



# CONCLUSIONES

## RESUMEN DE RESULTADOS:

El programa hecho en Python usando interfaz de ventanas y enlace con una base de datos, ha resultado bastante funcional para los requisitos mínimos para el proceso de ventas en una tienda.

## REFLEXIONES SOBRE EL PROCESO:

El proceso ha sido tedioso ya que no tengo muchos conocimientos sobre la librería de tkinter ni nada sobre la interfaz grafica, pero ha servido para aprender un poco en el proceso de desarrollamiento de la propia aplicación, además de enseñarme a enlazar los datos de una Base de datos en un programa en Python con la librería SQLite.

# BIBLIOGRAFIA

- ChatGPT
- GitHub

<https://github.com/aleexfrr/MiniProyecto-II.git>