

Taller de Introducción a C++

Gestión de Inventario de Productos usando `std::vector`

Objetivo

Este ejercicio tiene como objetivo aplicar conceptos de programación orientada a objetos en C++ usando `std::vector` para gestionar dinámicamente un inventario de productos en una tienda. Practicarás el uso de **estructuras (struct)**, **clases**, **punteros** y la gestión dinámica de objetos en un contenedor de la STL.

Descripción del Ejercicio

Implementa un programa en C++ que permita gestionar un inventario de productos. Cada producto tiene las siguientes propiedades:

- **Nombre:** el nombre del producto (tipo `string`).
- **Precio:** el precio del producto (tipo `float`).
- **Stock:** la cantidad en stock del producto (tipo `int`).

El programa debe permitir al usuario realizar las siguientes operaciones:

1. **Agregar un nuevo producto** al inventario.
2. **Modificar el stock** de un producto existente.
3. **Consultar la información** de un producto específico.
4. **Mostrar el inventario completo**.
5. **Salir del programa**

Instrucciones

Paso 1: Definir el Struct Producto

Define un `struct` llamado `Producto` que almacene la información de cada producto, incluyendo su nombre, precio y cantidad en stock.

Paso 2: Crear la Clase Inventario

Define una clase llamada `Inventario` que almacene los productos usando un `std::vector` de `Producto`.

Métodos requeridos en la clase `Inventario`:

`void agregarProducto(const std::string& nombre, float precio, int stock)` Agrega un nuevo producto al inventario usando el método `push_back` de `std::vector`.

`void modificarStock(const std::string& nombre, int nuevoStock)` Busca un producto por su nombre y actualiza su cantidad en stock. Si el producto no existe, muestra un mensaje de error.

`void consultarProducto(const std::string& nombre) const` Busca un producto por su nombre y muestra su información (nombre, precio y cantidad en stock). Si el producto no existe, muestra un mensaje de error.

`void mostrarInventario() const` Muestra la lista completa de productos en el inventario, incluyendo su nombre, precio y cantidad en stock. Si el inventario está vacío, muestra un mensaje de aviso.

Ejemplo de Uso

1. Agregar producto
2. Modificar stock
3. Consultar producto
4. Mostrar inventario
5. Salir

Seleccione una opción: 1
Ingrese el nombre del producto: Lápiz
Ingrese el precio del producto: 1.5
Ingrese la cantidad en stock: 100

Producto agregado.

Seleccione una opción: 4
Inventario:
- Nombre: Lápiz, Precio: 1.5, Stock: 100

Notas Adicionales

- Usa `cin` para recibir entradas del usuario y `cout` para mostrar la información al usuario.
- El menú principal debe repetirse hasta que el usuario elija la opción de salir.
- Asegúrate de incluir validaciones de entrada básicas, como no permitir valores negativos para el stock y el precio.

Consideraciones de Evaluación

Este ejercicio será evaluado en función de:

- La correcta implementación de la clase `Inventario` y el struct `Producto`.
- El uso adecuado de `std::vector` para la gestión dinámica del inventario.
- La funcionalidad del menú interactivo y de las operaciones solicitadas.
- El cumplimiento de los requisitos especificados y la claridad del código.

Sugerencia

Consulta la documentación de C++ sobre `std::vector` si tienes dudas sobre el uso de este contenedor. Recursos útiles incluyen cppreference.com y la documentación oficial de C++ en línea.