

Tema 2 - Exercici Avaluable 2

Una empresa vol fer un programa per gestionar les comandes que es fan dels seus productes. Per cada comanda es vol guardar el nom del client que fa la comanda, la data de la comanda, l'import total i el nº total de productes inclosos a la comanda. A més a més, per cadascun dels productes inclosos a la comanda s'ha de guardar el seu codi (com a tipus `string`), el nº d'unitats del producte incloses a la comanda i el preu per cada unitat del producte.

1. Declareu i implementeu una classe `ProducteComanda` per guardar i gestionar les dades de cadascun dels productes de la comanda. Aquesta classe ha de tenir tots els atributs necessaris per guardar el codi, el nº d'unitats i el preu del producte, tal com s'ha descrit abans, i també tots els mètodes necessaris per modificar i recuperar els valors dels atributs.
2. Declareu i implementeu una classe `Comanda` per guardar i gestionar les dades d'una comanda. Aquesta classe ha de tenir els atributs necessaris per guardar les dades d'una comanda que s'han descrit abans: nom del client, data, import total i les dades de tots els productes inclosos a la comanda. Per guardar les dades dels productes haureu d'utilitzar un array dinàmic.

Tema 2 - Exercici Avaluable 2

La classe Comanda haurà de tenir aquests mètodes públics:

- Un constructor que rebi com a paràmetres el nom del client, la data (com a string) i el número de productes de la comanda.
- El destructor de la classe.
- Un constructor de còpia.
- Un operador d'assignació.
- Mètodes getClient, getData i getImport per recuperar els valors del nom del client, la data i l'import total de la factura.
- Un mètode afegeixProducte que rebi com a paràmetre el codi, el nº d'unitats i el preu unitari d'un producte i l'afegeix a la llista de productes de la comanda. Heu de tenir en compte de modificar l'import total de la comanda cada cop que s'afegeixi un producte.
- Un mètode consultaProducte que rebi com a paràmetre el codi d'un producte i comprovi si aquest producte està inclòs a la comanda. Si està inclòs a la comanda retornarà true i també el nº d'unitats i el preu unitari utilitzant paràmetres per referència. Si no està inclòs a la comanda retornarà false.
- Un mètode eliminaProducte que rebi com a paràmetre el codi d'un producte i elimini aquest producte de la llista de productes inclosos a la comanda. Si està inclòs a la comanda i es pot eliminar retornarà true. Si no està inclòs a la comanda i no es pot eliminar retornarà false.

Tema 2 - Exercici Avaluable 1

Per poder guardar i gestionar totes les comandes de l'empresa us demanem crear una nova classe Empresa que contingui el següent:

- Un atribut per guardar la llista de totes les comandes de l'empresa, utilitzant la classe Comanda que heu implementat a l'exercici 4. Aquesta llista s'haurà de guardar com una llista dinàmica amb nodes enllaçats gestionada amb apuntadors.
- Un atribut per guardar la informació bàsica de tots els productes que ofereix l'empresa. La informació bàsica de cada producte és el nom, el codi del producte i el preu per unitat. Per poder guardar aquesta informació haureu de crear i implementar una nova classe Producte (diferent de la classe ProducteComanda anterior) amb els atributs necessaris i els mètodes bàsics per modificar i recuperar els valors dels atributs. Per guardar tots els productes heu d'utilitzar la classe vector de la llibreria estàndard.

Tema 2 - Exercici Avaluable 1

Els mètodes públics de la classe Empresa seran els següents:

- `void llegeixProductes(const string& nomFitxer);`

Llegeix d'un fitxer les dades amb la informació bàsica de tots els productes de l'empresa i les guarda a la llista de productes. El fitxer té aquest format:

```
NOM_PROD_1 CODI_PROD_1 PREU_PROD_1
...
NOM_PROD_N CODI_PROD_N PREU_PROD_N
```

- `bool afegeixComanda(const string& nomFitxer);`

Llegeix d'un fitxer les dades d'una comanda i les afegeix a la llista de comandes. El fitxer té aquest format:

```
NOM_CLIENT
DATA_COMANDA (en format DD/MM/YYYY)
Nº_DE_PRODUCTES_DE_LA_COMANDA
NOM_PROD_1 UNITATS_PROD_1
...
NOM_PROD_N UNITATS_PROD_N
```

Per tots els productes de la comanda s'haurà de verificar que existeixen a la llista de productes. Si existeix es recuperarà el codi i el preu unitari per afegir les dades d'aquest producte a la comanda. Si algun producte no existeix no s'afegirà la comanda a la llista de comandes i es retornarà false. Si tots els productes existeixen s'afegirà la comanda a la llista de comandes i es retornarà true.

Tema 2 - Exercici Avaluable 1

Els mètodes públics de la classe Empresa seran els següents:

- `bool importComanda(const string& nomClient, const string& data, float &import);`

Busca si existeix a la llista de comandes una comanda del client i la data que es passen com a paràmetres. Si existeix retorna true i al paràmetre per referència import, retorna l'import total de la comanda. Si no existeix retorna false.

- `bool DetallProducteComanda(const string& nomClient, const string& data, const string& codiProducte, int& nUnitats, float& preu);`

Busca si existeix a la llista de comandes una comanda del client i la data que es passen com a paràmetres i comprova que aquesta comanda contingui un producte amb el codi que es passa com a paràmetre. Si existeix retorna true i als paràmetre per referència nUnitats i preu, retorna el nº d'unitats i el preu del producte a la comanda. Si no existeix retorna false.

- `bool eliminaComanda(const string& nomClient, const string& data);`

Elimina de la llista de comandes la comanda del client i la data que es passen coma paràmetres. Si existeix i es pot eliminar retorna true. Si no existeix retorna false.