



A6.EA1: creació de múltiples classes

Instruccions

Creeu el projecte **ShopProject** amb els *packages* **shop.models**, **shop.controllers** i **shop.main**.

Per entregar l'exercici només haureu d'entregar la carpeta *src* comprimida en zip.

En tots els exercicis cal tenir en compte els següents aspectes importants:

1. Tota interacció amb l'usuari es farà des del main (petició i presentació de dades)
2. Els atributs i els mètodes de la classe han de tenir la visibilitat adequada per garantir l'encapsulació i la protecció de les dades.

Exercici 1.

Creeu la classe **Product** dins del package **shop.models**. Aquesta classe representarà un producte dels que es venen a la botiga que esteu implementant.

Atributs

De cada producte se'n vol guardar les dades següents:

- Codi de barres (IBAN de 13 caràcters)
- Nom
- Preu per unitat

Mètodes

- Creeu els tres constructors bàsics de la classe
- Implementeu els *getters* i *setters*
- Implementeu les funcions `equals()` i `toString()`

Exercici 2.

Creeu la classe **Warehouse** dins del package **shop.models**. Aquesta classe representarà el magatzem de la botiga.

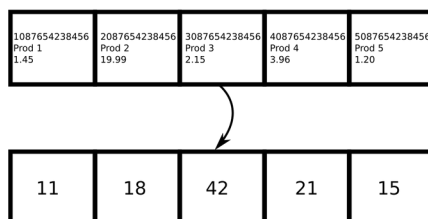
Atributs

El magatzem ha de gestionar les dades següents:

- Un *ArrayList* amb tots els productes que hi té emmagatzemats
- Un *ArrayList* amb la quantitat emmagatzemada de cada producte

Vegeu l'exemple següent on s'indica que el magatzem conté 42 unitats del producte "Prod 3".

ArrayList de prodctes



ArrayList de quantitats

Mètodes



- Creeu els tres constructors bàsics de la classe.
- Implementeu les funcions `equals()` i `toString()`.
- Implementeu la funció **`boolean addProduct(Product prod)`**. Si el producte "prod" no existeix dins del magatzem, l'afegirà amb una quantitat de 0 unitats i retornarà `true`. Si el producte ja existeix, no farà res i retornarà `false`.
- Implementeu la funció **`void addProduct(Product prod, int quantity)`**. Si el producte "prod" no existeix dins del magatzem, l'afegirà amb una quantitat de `quantity` unitats. Si el producte ja existeix, afegirà `quantity` unitats a les ja existents.
- Implementeu la funció **`boolean deleteProduct(String barcode)`**. Si el producte amb el codi de barres `barcode` existeix, l'esborrarà (tant de l'`ArrayList` de productes com de l'`ArrayList` de quantitats) i retornarà `true`. En cas contrari, no farà res i retornarà `false`.
- Implementeu la funció **`boolean reduceProduct(String barcode, int quantity)`**. Reduirà la quantitat del producte amb codi de barres `barcode` `quantity` unitats i retornarà `true`. Si el producte no existeix o no se'n té la quantitat demanada, no farà res i retornarà `false`.
- Implementeu la funció **`ArrayList<Product> getProductsForReestock()`**. Aquesta funció retornarà tots els productes dels quals se'n tingui 5 o menys unitats.
- Implementeu la funció **`ArrayList<Product> getProducts()`**. Aquesta funció retornarà tots els productes que tinguin estoc (com a mínim, n'hi ha d'haver 1 unitat).
- Implementeu la funció **`ArrayList<Product> getAllProducts()`**. Aquesta funció retornarà tots els productes que gestiona el magatzem (tinguin o no tinguin estoc actualment).
- Implementeu la funció **`Product getProduct(String barcode)`**. Aquesta funció retornarà el producte amb el codi de barres `barcode` o `null`, en cas que el producte no existeixi.

Exercici 3.

Creeu la classe **CashRegister** dins del *package* **shop.models**. Aquesta classe representarà la caixa enregistradora de la botiga.

Atributs

La caixa enregistradora ha de gestionar les dades següents:

- Diners inicials del dia
- Total d'entrada de diners (diners derivats de vendes)
***Atenció:** tindrà en compte el valor de la factura real*
- Total de sortida de diners (diners derivats de compres a proveïdors)
***Atenció:** tindrà en compte el valor de la factura real*
- Diners acumulats durant el dia (diners inicials + vendes – compres)
***Atenció:** tindrà en compte el moviment real de diners (amb els errors que es puguin provocar en els pagaments i el retorn de canvis)*

Mètodes

- Creeu els constructor per defecte de la classe, tot inicialitzant els diners inicials del dia a 3000.0€ i la resta d'atributs a 0.0€.
- Implementeu la funció `toString()`.
- Implementeu la funció **`void manageSale(float value)`**. Aquesta funció modificarà el total de diners d'entrada (segons la factura real) i els diners acumulats durant el dia tenint en compte que s'ha realitzat una venda per valor de `value` euros. Per modificar els diners acumulats durant el dia, tingueu en compte que en els pagaments es poden produir petits



errors. Per representar-ho, demaneu un valor aleatori dins del rang [1, 100]. Si aquest valor és imparell, considereu que hi ha hagut un error en el pagament. Si és parell, el pagament ha estat exacte. En cas d'error, incrementeu o minveu el valor `value` dins d'un marge d'un 1% del valor total de la venda.

- Implementeu la funció **`void managePurchase(float value)`**. Aquesta funció modificarà el total de diners de sortida (segons la factura real) i els diners acumulats durant el dia tenint en compte que s'ha realitzat una compra per valor de `value` euros. Per modificar els diners acumulats durant el dia, tingueu en compte que en els pagaments es poden produir petits errors. Per representar-ho, demaneu un valor aleatori dins del rang [1, 100]. Si aquest valor és imparell, considereu que hi ha hagut un error en el pagament. Si és parell, el pagament ha estat exacte. En cas d'error, incrementeu o minveu el valor `value` dins d'un marge d'un 1% del valor total de la venda.
- Implementeu la funció **`float closeCash()`**. Aquesta funció farà el tancament de caixa i comprovarà si els diners inicials més les vendes menys les compres equival als diners acumulats durant tot el dia. Retornarà la diferència obtinguda.

Exercici 4.

Creeu la classe **`ShopController`** dins del *package* **`shop.controllers`**. Aquesta classe representarà la botiga.

Atributs

La botiga ha de gestionar les dades següents:

- Magatzem
- Caixa enregistradora

Mètodes

- Creeu els constructor per defecte de la classe, inicialitzant un magatzem buit i una caixa enregistradora per defecte.
- Implementeu la funció **`toString()`**.
- Implementeu la funció **`void loadProducts(String filename)`**. Aquesta funció carregarà tots els productes a introduir al magatzem des d'un fitxer (trobareu el model d'aquest fitxer adjuntat a l'enunciat).
- Implementeu la funció **`ArrayList<Product> getProductsForReestock()`**. Aquesta funció retornarà tots els productes del magatzem dels quals se'n tinguin 5 o menys unitats.
- Implementeu la funció **`ArrayList<Product> getProducts()`**. Aquesta funció retornarà tots els productes del magatzem que tinguin estoc (com a mínim, n'hi ha d'haver 1 unitat).
- Implementeu la funció **`ArrayList<Product> getAllProducts()`**. Aquesta funció retornarà tots els productes que gestiona el magatzem (tinguin o no tinguin estoc actualment).
- Implementeu la funció **`void sellProduct(String barcode, int quantity)`**. Gestionarà la venda de `quantity` unitats del producte amb el codi de barres `barcode`. Si el producte existeix dins del magatzem i n'hi ha prou quantitat, farà la reducció d'estoc i modificarà la caixa enregistradora corresponentment.
- Implementeu la funció **`void buyProduct(String barcode, int quantity)`**. Gestionarà la compra de `quantity` unitats del producte amb el codi de barres `barcode`. Si el producte existeix dins del magatzem, n'incrementarà l'estoc; si no existeix, l'afegirà amb la quantitat especificada. A més a més modificarà la caixa enregistradora corresponentment.
- Implementeu la funció **`float closeCash()`** per fer el tancament de la caixa.



Exercici 5.

Creeu el programa **ShopMain** dins del *package* **garage.main** per gestionar les accions de la botiga a través del següent menú:

```
~~~~ Gestió de la Botiga ~~~~  
  
0. Sortir  
1. Carregar els productes des del fitxer  
2. Vendre producte  
3. Comprar producte (reestock)  
4. Eliminar un producte del magatzem  
5. Fer la caixa  
6. Mostrar la botiga
```

Aquest menú s'haurà de mostrar una i altra vegada mentre l'usuari no introdueixi l'opció 0. A més a més, també haurà d'informar dels errors produïts per la introducció d'opcions errònies.

Opció 1.

Els passos per implementar aquesta opció són els següents:

1. **Demaneu a l'usuari el fitxer** de productes que vol carregar
2. **Carregueu els productes a la botiga**, modificant correctament el magatzem.

Opció 2.

Els passos per implementar aquesta opció són els següents:

1. **Mostreu els productes amb estoc**
2. **Demaneu a l'usuari quin producte vol comprar i quantes unitats en vol.** Si el valor introduït no correspon a un producte vàlid, informeu de l'error i torneu al menú principal. En canvi, si el valor és correcte, procediu a realitzar la venda del producte (reducció de l'estoc i modificació de la caixa enregistradora). Cal tenir en compte que si no hi ha prou estoc, la compra no es podrà fer i s'haurà d'avisar a l'usuari.

Opció 3.

Els passos per implementar aquesta opció són els següents:

1. **Mostreu els productes que necessiten reestock**
2. **Demaneu a l'usuari quin producte entrarà al magatzem i quantes unitats n'arribaran.** Si el valor introduït no correspon a un producte vàlid, informeu de l'error i torneu al menú principal. En canvi, si el valor és correcte, procediu a realitzar el reestock (increment de l'estoc i modificació de la caixa enregistradora).

Opció 4.

Els passos per implementar aquesta opció són els següents:

1. **Mostreu tots els productes del magatzem**
2. **Demaneu a l'usuari quin producte vol eliminar.** Si el valor introduït no correspon a un producte vàlid, informeu de l'error i torneu al menú principal. En canvi, si el valor és correcte, procediu a eliminar el producte del magatzem (la caixa enregistradora no patirà cap canvi!).

Opció 5.

Els passos per implementar aquesta opció són els següents:

1. **Feu la caixa** (comprovant entrades i sortides de diners)
2. **Mostreu per pantalla l'error que s'hagi pogut cometre.**



Opció 6.

Els passos per implementar aquesta opció són els següents:

1. **Mostreu tots els productes del magatzem** (descripció i quantitat).
2. **Mostreu l'estat actual de la caixa enregistradora.**

INSTRUCCIONS D'ENTREGA

Per entregar aquest exercici només cal que comprimiu en **ZIP** (**si m'ho envieu en RAR no ho podré obrir!**) la carpeta *src* del projecte i la pugeu al Moodle.