

Pr02 - Projecte. Packs i productes

Material a entregar:

- Document de text a on es vegi l'enunciat de l'exercici, el codi del programa i el resultat de les proves realitzades i els problemes trobats.
- Els arxius .java de les classes desenvolupades.
- El codi ha d'estar comentat

Llegeix l'enunciat general i fes el següent desenvolupament:

1. Implementa la classe **ProducteAbstract** i posa-la com una classe abstracta (perquè no s'ha de permetre crear objectes d'aquesta classe).
 - Atributs: idproducte , nom, preu_venda
 - Constructors amb i sense paràmetres.
 - Mètodes: getters i setters; imprimir o toString;
 - Mètode **equals**(ProducteAbstract producte). Retorna true si un producte té el mateix nom que el que s'ha passat per paràmetre.
2. Implementa la classe **Producte** com a filla de **ProducteAbstract**:
 - No ha de permetre ser heretada.
 - Atributs: A part dels heretats de la classe **ProducteAbstract**:
 - stock del producte al magatzem
 - id del proveïdor habitual on es compra el producte (en el futur implementarem el manteniment de proveïdors)
 - Reescriu el mètode **imprimir()** o **toString()** fent servir el de la classe mare.
 - Pensa si cal reescriure el mètode **equals()**, és a dir, si canvia el criteri respecte la classe mare per decidir si dos productes són iguals.
3. Implementa la classe **Pack** com a filla de **ProducteAbstract**.
 - No ha de permetre ser heretada.
 - Atributs: A part dels heretats de la classe **ProducteAbstract**, els següents:
 - una llista (ArrayList<String>) on guardar els id's dels productes que formen un pack.
 - % de descompte a aplicar sobre el preu total de venda de tots els productes que el formen
 - Fixeu-vos que en aquest cas no guardem l'stock, i es que un pack està format per productes i aquest ja controlen el seu stock.
 - Constructors amb i sense paràmetres, que han de cridar als constructors de la classe mare.
 - Mètodes: **getters i setters; afegirProducte i esborrarProducte**, per afegir i eliminar un producte de la llista que forma el pack.

- Reescriu el mètode **imprimir()** o **toString()** fent servir el de la classe mare i imprimint el id i nom dels productes que formen el pack.
 - Reescriu el mètode **equals(Pack pack)**. Retorna true si el pack conté els mateixos productes que el pack passat per paràmetre.
4. Adapta la funcionalitat de la classe **ProductesDAO** desenvolupada a la pràctica anterior, ja que ara a la taula podem guardar tant productes com packs. Quin canvi has de fer? En quin dels principis de la POO es fonamenta aquest canvi?
 5. Adapta la classe **IniciVistaController** per realitzar el manteniment de productes i packs a la mateixa pantalla. A l'hora d'afegir, buscar, eliminar, imprimir, etc. heu de tractar tant productes com packs, és a dir, al moment d'afegir un producte heu de preguntar si és un producte normal o un pack per demanar els valors corresponents a cadascun d'ells.