# Упражнения: Дефиниране на класове

## Задача 1. Един магазин

Създайте клас **Product** с полета за име на продукта и баркод – и двете са текстови низове, цена – double и количество - double. Създайте статичен клас, който да поддържа информация за продуктите в магазина и следните функционалности:

* Продажба на продукт – приема за параметри баркода и продаваното количество. Не допускайте продажба на продукта, ако той има по-малка наличност от желаното количество. Изведете подходящо съобщение на екрана (**Sell**).
  + Командата ще има вида: Sell <**код**> <**количество**>
  + Ако продуктът не съществува или няма достатъчно количество, изведете „Not enough quantity”.
* Добавяне на нов продукт – добавя се информация за продукта; баркод, име, цена и количество (**Add**)
  + Командата ще има вида: Add <**код**> <**име**> <**количество**>
* Зареждане на продукт – добавя се количество от даден продукт; параметрите са баркода и самото количество; не допускайте зареждане на продукт, ако той изобщо не съществува към момента (**Update**)
  + Командата ще има вида: Update <**код**> <**количество**>
  + Ако такъв продукт не съществува, изведете „Please add your product first!”
* Изпечатване на налични продукти по азбучен ред (**PrintA**)
* Изпечатване на информация за неналични продукти по азбучен ред (**PrintU**)
* Изпечатване на всички продукти по намаляща наличност - тези от които има най-много са в началото (**PrintD**)
* Изчисляване на стойността на всички налични продукти (**Calculate**)

За всичко това трябва да се създаде и програма, която приема команди и изпълнява съответните действия. Името на всяка команда е записано в скоби по-горе. Командата, която приключва въвеждането е „**Close**”. Когато се въведе тя програмата приключва. Всички реални числа се извеждат закръглене и с точно 2 знака след запетаята.

### Примери

|  |  |  |  |
| --- | --- | --- | --- |
| **Вход** | **Изход** | **Коментар** |  |
| Add 359293 ProductA 3.50 8.0  PrintA  Sell 359293 8.0  PrintA  Update 359293 5.0  PrintA  Add 555 ProductB 5.50 3.0  PrintA  Sell 359293 4.5  PrintD  Calculate | ProductA (359293)  ProductA (359293)  ProductA (359293)  ProductB (555)  ProductB (555)  ProductA (359293)  18.25 | Първата команда за печат идва след като само и единствено сме добавили продукта, затова на нея съответства само първия ред от изхода.  След това продукта бива продаден и извикваме отново PrintA. Този път понеже няма налични продукти не трябва да отпечатваме нищо.  След това зареждаме ново количество от продукта и поради това следващото PrintA го отпечатва отново.  Добавяме нов продукт, след което отново отпечатваме резултата – този път имаме два налични продукта. При PrintD първо излиза ProductB, понеже преди това сме продали почти цялото количество от ProductA, а условието за командата PrintD гласи информацията да се изпечата в намалящ ред, спрямо количеството. При извикването на calculate показваме сумата от всички налични продукти: (0.5 \* 3.50) + (3\*5.50) = 18.25 |  |

## Задача 2. Банкер

Създайте класа BankAccount

Този клас трябва да има полета за:

* id: int
* balance: double

Класът трябва да има свойства за:

* ID: int
* Balance: double

Създайте методите:

* Deposit(Double amount): void – който да вкарва пари в сметката
* Withdraw(Double amount): void – който да изтегля пари от сметката

Заменете метода ToString(), като в този метод изпечатвайте информация за банковата сметка

Създайте статичния клас Bank.

В този клас трябва да създадете следните функционалности:

* Теглене на средства – на този метод трябва да подадете ID-то, списъка с всички създадени сметки и желаната сума за теглене. Ако сумата я няма в наличност или сметката не съществува, изведете подходящо съобщение – за тази функционалност може да се наложи да реализирате няколко метода.
* Внасяне на средства – на този метод трябва да подадете ID-то, списъка с всички създадени сметки и желаната сума за внасяне. Ако сметката не съществува, изведете подходящо съобщение – за тази функционалност може да се наложи да реализирате няколко метода или да използвате вече реализирани такива от предходната точка

Изберете адекватна структура и логика за реализирането на желаните функционалности