BLUE Cake Shop

BLUE Cake Shop is a small cake shop. Mr. Krauser, the manager of BLUE Cake Shop, wants to change the cashier system to be paperless. He needs a program that can handle his cake shop transaction. He asks you as a skillful programmer to make a program like this:

- The program always shows a **list of cakes** which contains **No**, **Cake Code**, **Cake Name**, **Available**, and **Price** of the available cakes.

```
BLUE CAKE SHOP CASHIER
======================

No. | Cake Code | Cake Name            | Available | Price
-------------------------------------------------------------
01. | CK001     | Blueberry Cake       |        13 | Rp. 25000,-
02. | CK009     | Chocochip Cake       |         5 | Rp. 20000,-
03. | CK017     | Mayonaise Cake       |        24 | Rp. 30000,-
04. | CK023     | Strawberry ShortCake |         7 | Rp. 17500,-
-------------------------------------------------------------
```

Figure 1. List of Cakes

- The program consists of **3 menus**, there are:
  1. Sell
  2. Add Stock
  3. Exit
- If user chooses **Sell** (**Menu 1**), then:
  - Ask user to input **cake code** he/she wants to sell**.** Validate the code must consist of **5 characters** and **available** in the **list of cakes**. If user inputs the cake code **other than the one on the list**, the program will show the message **"--- The Cake Code doesn't exist ---"** and ask user to **input again**. The cake code is **case-sensitive**.
  - Then the program will ask user to input the **quantity**. Validate the quantity must be **between 0 and x**, where **x = [quantity of cake available]**. User **cannot** sell the cakes **more than the quantity** available on the list. If user tries to do so, show the message **"...The quantity of cake is not enough..."**.
  - If user succeeds to sell, then show the **total price**.

Total Price = Rp [**price of cake**],- **x** [**quantity of sell**] = Rp [**total price**],-

  - After that, **subtract** the **available** cakes with the **quantity of cake** that **has been sold**.

```
Menu :
1. Sell
2. Add Stock
3. Exit
Input choice : 1

Input Cake Code [5 chars]: CK002

--- The Cake Code doesn't exist ---

Input Cake Code [5 chars]: CK001
Input Quantity [0..13]: 14

...The quantity of cake is not enough...
Input Quantity [0..13]: 5


Total Price is : Rp 25000,- x 5 = Rp 125000,-

--- Thank You ---
```

Figure 2. Sell Menu

- If user chooses **Add Stock** (**Menu 2**), then:
  - Ask user to input the **cake code** he/she wants to add to the list. Validate the code must c**onsist of 5 characters** and **available** in the **list of cakes**. If user inputs the cake code **other than the one on the list**, the program will show the message **"--- The Cake Code doesn't exist ---"** and ask user to input again. The cake code is **case-sensitive**.
  - If the inputted cake code is **valid**, then ask user to input the **quantity**. Validate the quantity of cake must be **between 1 and 10**.
  - If user succeeds to add stock, then show **"--- Adding Stock Success ---"**. Then **add** the **available** cakes with **the inputted quantity of cake**.

```
Menu :
1. Sell
2. Add Stock
3. Exit
Input choice : 2

Input Cake Code [5 chars]: CK002

--- The Cake Code doesn't exist ---

Input Cake Code [5 chars]: CK001
Input Quantity [1..10]: 11
Input Quantity [1..10]: 8


--- Adding Stock Success ---
```

Figure 3. Add Stock Menu

- If user chooses **Exit(Menu 3),** then the program will be **terminated**.

Please run the EXE file to see the sample program