LAB 3

LINKED LIST

In POS systems, product data is the base to decide the success, so it is definitely important to data scientists researching on marketing and sale management. Besides, *products* are grouped by function and feature, called by ***product categories***. Assuming that, technicians had utilized singly linked list to store products and their categories.

Task 1.

1.1. Declare a structure to store a product category (like a node in linked list) in categories list. A category information consists of category ID (a string), category name (string).

1.2. Declare a structure to store a product (like a node in linked list) in product list. Product information consists of ID (a string), name (a string), selling price (an integer), importing price (an integer), quantity (a real number), a pointer to its product category. A category includes many products, but only a product belongs to a category (1 – N relationship).

Task 2.

Implement two functions to append a new category to categories list and a new product to product list.

Task 3.

Implement a function to traverse only product list to print all categories to monitor. The format you have to follow bellowing example:

There are three (3) categories in systems. The result we see in monitor:

```
3 categories
--------------------
ID: C001
Name: Cosmetics
--------------------
ID: C002
Name: Food
--------------------
ID: C003
Name: Beverage
```

Task 4.

Implement a function to delete all products with quantity less than a number. This function returns true if at least one product are deleted, otherwise returns false. Prototype: `bool deleteProducts(Product* &pHead, double quantity);`

Task 5.

Implement a function to read selling data from a file and decrease quantity of product after selling. This file (in CSV file) consists of two columns: product ID, selling quantity. This function returns total amount in price. Prototype:

```
double sellingFromFile(Product* &pHead, string filename);
```