Introduction

In this project you will design and implement vector and vector operations for the management system. You will be provided with a starter template with necessary classes and templates. You need to add functions that are specified in the document. Do not change the main function, I have added test cases in it.

Food Pantry Management System

CSUF wants to make Inventory management and appointment management system to maintain data related to donations and students who take advantage of this service. They are asking newly admitted students to design a system that tracks all the operations. The pantry management has asked you to add features to manage inventory and appointments. Your task is to create functions which will add functionalities to the management system. This are some functional requirements:

You are given with two template classes for inventory management system:

Item - Item class template decides the properties of a particular item. It has four properties: name, expiration, category, quantity.

Inventory - Inventory function will take care of all the operations. Inventory class will have one vector which will store objects of Item.

1) **Add New Item**: Write a function **addNewItem()** which will add a new Item in your vector. While adding a new element in the vector, check whether it already exists in the vector or not. If it exists in a vector, then display the message as "Item is already present in inventory".

```
-------Inventory-------
Name              Expiration      Category      Quantity
Protien Bar       05/09/2023      4             Snacks
Milk              05/12/2023      2             Regular
Cerels            09/12/2023      5             Snacks
```

2) **Increase Quantity**: Write a function **increaseQuantity**() to increase the quantity of the item by the newly provided quantity. It will take two parameters: itemname and quantity.

i1.increaseQuantity("Cerels",10);

```
Quantity has increased: 15
-------Inventory-------
Name              Expiration      Category      Quantity
Protien Bar       05/09/2023      4             Snacks
Milk              09/24/2023      3             Regular
Cerels            09/12/2023      15            Snacks
```

3) **Update Item**: Write a function **updateItem()** which will update the quantity, expiration and category. updateItem() will take four parameters: itemname, expiration, quantity and category. It should search the item from the name and update the item with given parameters. We can only update the items that are already present in inventory. If time is not found throw an exception "Item not found".

i1.updateItem("Milk","09/24/2023","Regular",3);

```
ItemMilkupdated
-------Inventory-------
Name                Expiration      Category        Quantity
Protien Bar         05/09/2023      4               Snacks
Milk                09/24/2023      3               Regular
Cerels              09/12/2023      5               Snacks
```

4) **Remove Item** : Write a function **removeItem()** which will remove a particular item from the vector on the name of item. The function removeItem() takes a parameter as itemname. If an item is not in an inventory and you are trying to remove it from inventory, it will throw an exception "Item not found".

i1.removeItem("Milk");

```
-------Inventory-------
Name                Expiration      Category        Quantity
Protien Bar         05/09/2023      4               Snacks
Cerels              09/27/2023      4               Regular
```

5) **Total**: Write a function **Total()** to calculate the total number of items in inventory.

```
Total Number of items in inventory = 3
```

6) **Search Item**: Write a function **searchItem()** to search a particular item in the inventory. The function searchItem() takes a parameter itemname and based on that it will find an item. If an item is not present it will throw an exception "Item not found!!".

```
Query for Cerels
Item = Cerels
Expiration Date = 09/27/2023
Category= Regular
Quantity= 4
```

For appointment management you are provided with two classes:

Appointment- Appointment class template decides the properties of an appointment. It has four properties: name, ap_date, ap_time, CWID.

AppointmentSystem - This class is responsible for all the operations. AppointmentSystem class will have one vector and it will store objects of Appointment.

1) **Schedule:** Write a function **schedule()** to schedule an appointment. It will store an appointment in a vector. If CWID is already present in a vector, then display the message that "You have already scheduled an appointment!!!". It will store an object in a vector.

```
-------Appointments-------
Name                Date            Time            CWID
John Bob            09/12/2023      9:30AM          889456723
Jim Lee             09/12/2023      10:30AM         883476923
Chris Lynn          09/12/2023      12:00PM         879455714
Arnav Shah          09/12/2023      12:00PM         879459583
```

2) **Total_appointments:** Write a function **Total_appointments()** to calculate a total number appointments on a particular date and at a particular time. It will take two parameters: date and time.

```
Total Appointments = 2
```

3) **Remove recent:** Write a function **removeRecent()** to delete an appointment that is recently booked.

```
-------Appointments-------
Name                Date            Time            CWID
John Bob            09/12/2023      9:30AM          889456723
Jim Lee             09/12/2023      10:30AM         883476923
Chris Lynn          09/12/2023      12:00PM         879455714
```