**Task\_1:**

Suppose you are designing a program to model a grocery store's inventory.

Each item has a unique ID, a name, a price, and a quantity. Design a class for

representing an item in the grocery store's inventory. The class should getter and

setter methods where needed. It should also have a method for displaying the

item's details. Additionally, implement a constructor that takes arguments for

the item's ID, name, price, and quantity and sets them in the object. Make a

default constructor too. And also a destructor (which prints the message object

destroyed). The program should first ask the user to enter the total no of items to

store, then create an array of Items of that size in main. The main should be a

menu-driven program with the following choices:

1). Display Items

2). Update item details

3). Search for items by ID

The above methods should be declared outside the class. There must be

functions for all the operations. Display\_items() takes the array of objects and

should display all the items with their details(ID, Name, Price, Quantity).

2). UpdateItem() should take the item as an argument and further ask whether to

update the price or the quantity. And do accordingly in the function. Think

whether it should be pass by value or reference.

3). Search for items by ID should take array of items, size, and the ID\_NO. If

the item is found, display its details, else display “item not found”

s

**Task\_2:**

In a smart farming project, a company has developed an advanced Greenhouse Monitoring System designed to make it easier for farmers to control the climate within their greenhouses. Each greenhouse has its own ClimateController, which is responsible for adjusting the temperature, humidity, and light levels to create the perfect growing environment. The ClimateController is made up of three key sensors: a TemperatureSensor, a HumiditySensor, and a LightSensor. These sensors are tightly integrated with the controller, meaning they are created and destroyed together. The design ensures that the sensors and the controller can only exist within the context of a greenhouse. if the greenhouse is removed, the controller and its sensors are also automatically deleted. The entire system's components are dependent on the existence of the greenhouse.