Design an object model for a given subject area. Use (demonstrate the ability to work with):  
- classes (abstract classes - if possible),  
- interfaces  
- inheritance  
- polymorphism  
- encapsulation  
- collections.  
Each class, method and variable should have an exhaustive meaning in name and informative composition.  
You need to think carefully about which classes are needed to solve the problem.  
Inheritance should only be applied when it makes sense.  
Classes should be correctly packaged.  
Work with the console or the console menu should be minimal (only the necessary data to enter, display only what is requested in the task).  
The task is some kind of subject area in which it is required to implement the necessary hierarchy of classes and implement it using OOP (using inheritance, if necessary, or implementing interfaces).  
Each class should have fields and methods that you consider necessary.  
The program should create objects of various classes in the selected subject area, combine them into some set of objects (use collections).  
Typically, a task requires you to perform some action on objects in a collection of objects according to specified criteria.  
Chef. Define a hierarchy of vegetables. Make a salad. Count calories. Sort vegetables for salad based on one of the parameters. Find vegetables in the salad that match a given range of parameters.