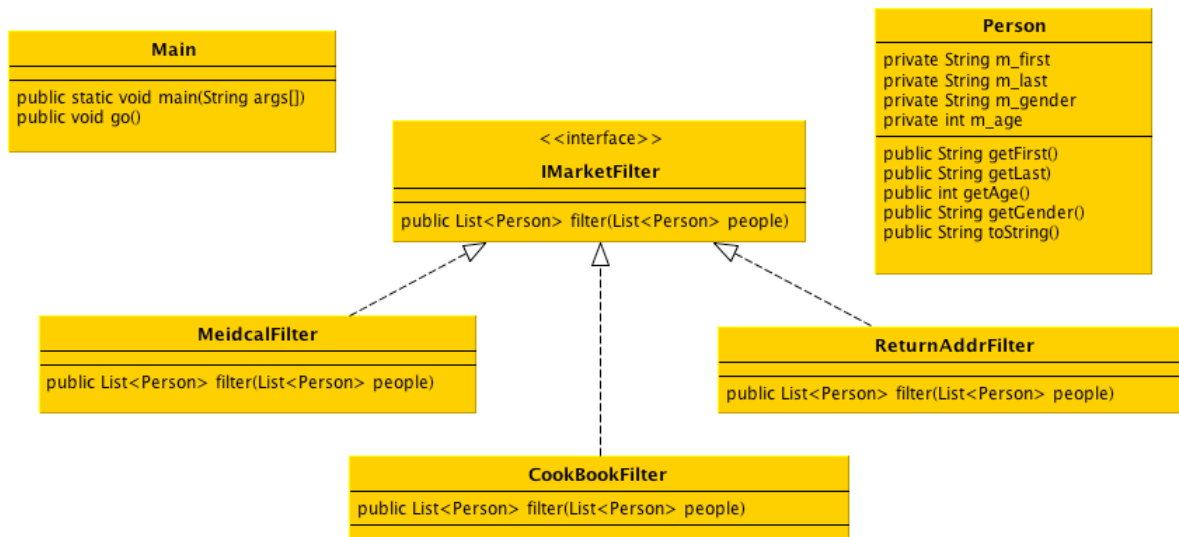# CSCI328 - Intro to OOD Exam

A marketing company has hired you to analyze marketing data and locate people who are good targets for certain advertisements. Specifically, they are interested in identifying three groups: people who may be interested in buying return address labels; people who may be interested in purchasing an Italian cookbook; and people who may be interested in medical equipment. The company has provided you with a file (names.txt) containing a long list of potential customers. A line in this file will look something like this:

```
Marisa,Schoonderbeek,F,38
```

The design for your application has already been completed and is shown in the UML diagram below, along with a short description of the classes and interfaces in the UML diagram. Your program should match all of the details in the UML diagram exactly.

Main: This class contains the static main method for your program as well as a non-static method called go(). Your static main method should instantiate an instance of Main and call go() to solve this problem. The go() method should read names.txt (posted on Canvas) and create a Person object for each line in the file. It should place each Person in an ArrayList and then pass this list to instances of the three different market filters (Medical, CookBook, and ReturnAddr) to identify each person that is a potential customer for each category. Your program should print the number of people in each of the three lists returned by the filters.

Person: This class represents a person in the file. It should contain the attributes and methods defined in the UML diagram.

IMarketFilter: This interface defines an abstraction for all classes that are capable of searching through the list of people and finding those that meet a specific customer profile. The customers that meet the criteria should be returned in a list. This interface should contain just the filter method shown in the UML diagram.

ReturnAddrFilter: This class should search the specified list and find people whose full names are at least 18 characters long. These people should be returned in a new list.

MedicalFilter: This class should search the specified list and find people in the file whose age is greater than or equal to 30 plus the average age of all of the people in the list (age >= ave+30).   These people should be returned in a new list.

CookBookFilter: This class should search the specified list and find people in the file whose full name contains at least seven vowels (a, e, i, o , u). These people should be returned in a new list.