1. We are given a school. In the school there are classes of students. Each class has a set of teachers. Each teacher teaches a set of disciplines. Students have name and unique class number. Classes have unique text identifier. Teachers have name. Disciplines have name, number of lectures and number of exercises. Both teachers and students are people. Students, classes, teachers and disciplines could have optional comments (free text block).

Your task is to identify the classes (in terms of OOP) and their attributes and operations, encapsulate their fields, define the class hierarchy and create a class diagram with Visual Studio.

2. Define abstract class Human with first name and last name. Define new class Student which is derived from Human and has new field – grade. Define class Worker derived from Human with new property WeekSalary and WorkHoursPerDay and method MoneyPerHour() that returns money earned by hour by the worker. Define the proper constructors and properties for this hierarchy. Initialize a list of 10 students and sort them by grade in ascending order (use LINQ or OrderBy() extension method). Initialize a list of 10 workers and sort them by money per hour in descending order. Merge the lists and sort them by first name and last name.

3. Create a hierarchy Dog, Frog, Cat, Kitten, Tomcat and define useful constructors and methods. Dogs, frogs and cats are Animals. All animals can produce sound (specified by the ISound interface). Kittens and tomcats are cats. All animals are described by age, name and sex. Kittens can be only female and tomcats can be only male. Each animal produces a specific sound. Create arrays of different kinds of animals and calculate the average age of each kind of animal using a static method (you may use LINQ).