# Exercise 1:

All classes that you create should include as many features as possible from what we have learnt (different types, enums, inheritance, aggregation, constructors, gets & setters, abstract classes, interfaces). Add toString() methods to some of the classes.

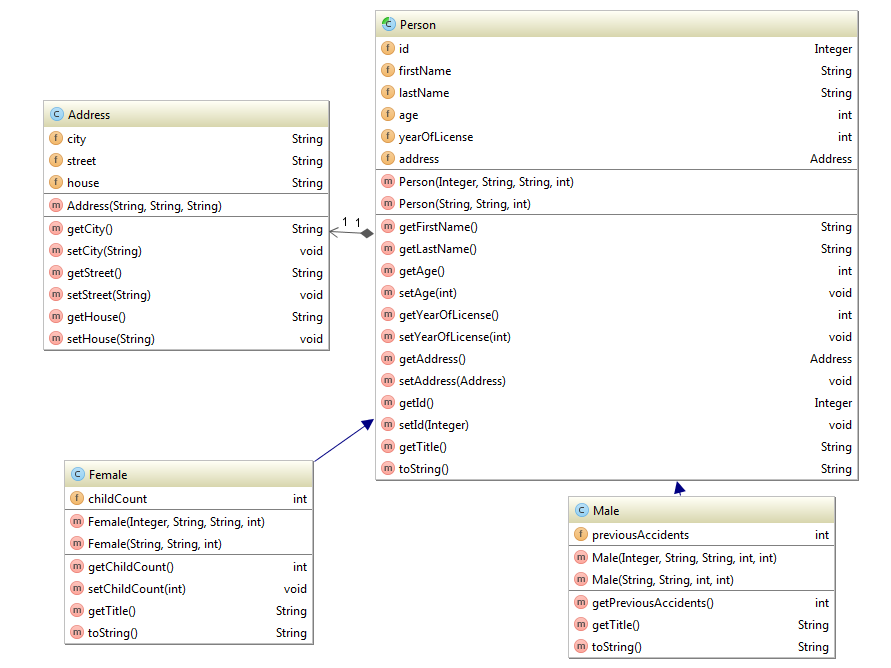
A standard application usually has the following layers: domain classes, data module (dao), service layer. Interfaces are usually not used in the domain classes but only in the other layers.

So when you create your classes try to put them in correct packages (in bigger projects we might even put them in different modules – jars).

## Domain classes

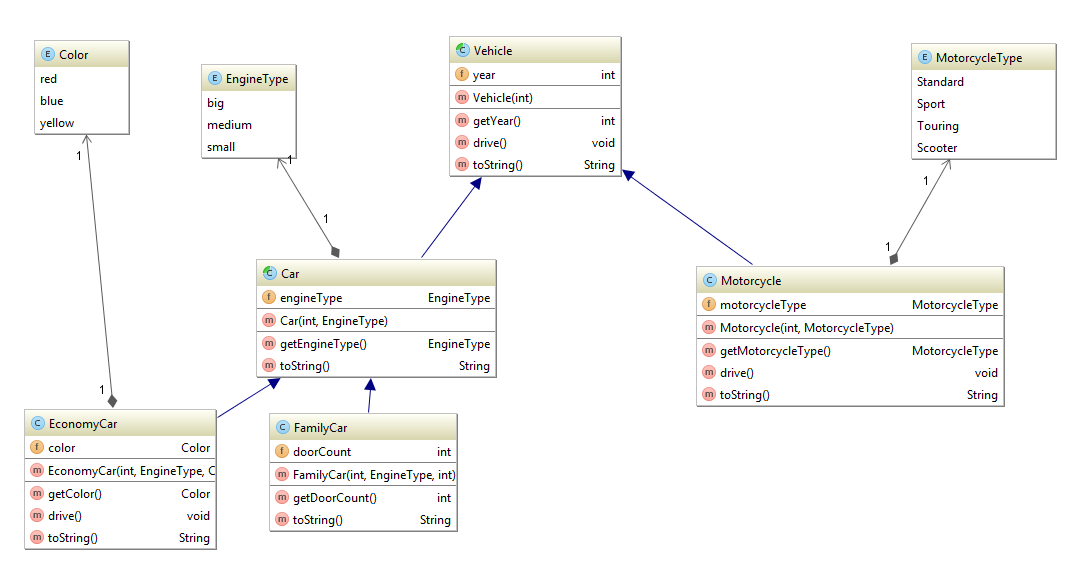
### People

The first sets of classes that we will create are for the people domain. For example:



### Vehicles

Create a set of classes for vehicles. An example class diagram for them is:



### Service

The service layer is the logic layer. Here we should add services that will query the data and return answers to logic questions. This is the layer that will evolve as the course progresses.

For this stage of the project all processing will be code in memory. As we continue in the course this layer will change to support spring and then also to connect to the database via hibernate.

When designing the service layer, you should be using interfaces so that the actual implementation is hidden from the calling code. As you can see in the diagram below the People interface has two implementations one in java 7 and one in java 8.

