Class Diagrams Notes

Introduction

Planning is the first step to writing well thought out code.
If you get stuck in without much forethought and you'll regret it later.

Unfortunately this is a difficult lesson to learn, not only do you need to learn not to dive in without planning but you need to learn how to plan effectively which takes experience and practice.

In this lesson we're going to help you on your way and cover a few tools that you can use to set yourself up for success.

We will have another lesson revolving around planning just before your first projects but for now the main tools I'll be introducing you to are Class and Object Diagrams.

Why Use Diagrams?

Drawing a diagram is a great way to visualise the structure of your code.

Doing this before writing a single line of code can help you structure the logic and behaviour of your code as well as spot any flaws before implementation.

It will also help you develop multiple solutions and narrow down which is the most optimal choice for the task at hand.

So there are a few options for drawing diagrams, draw.io, figma, photoshop etc but for now we're going to stick with the traditional platform.

A pencil and a piece of paper.

What Is UML

Today the diagrams we'll be covering are part of the UML or Unified Modelling Language.

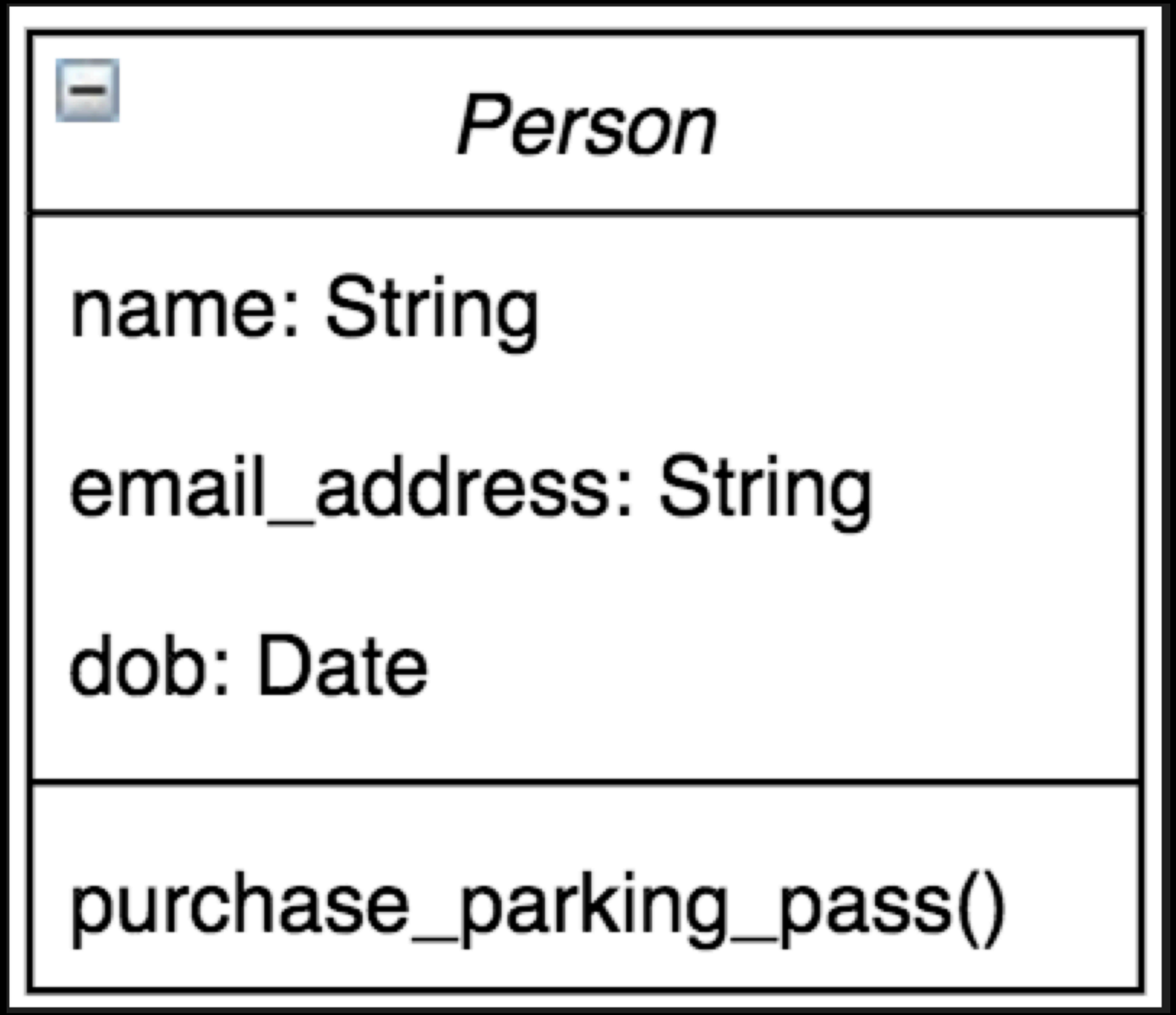
This is a language which aims to provide a standardised way of modelling software systems.

UML has a standard set of diagrams that can be used for modelling code in any language.

Our diagrams are going to look similar to the official UML diagrams however we're not as concerned with strictly following the rules.

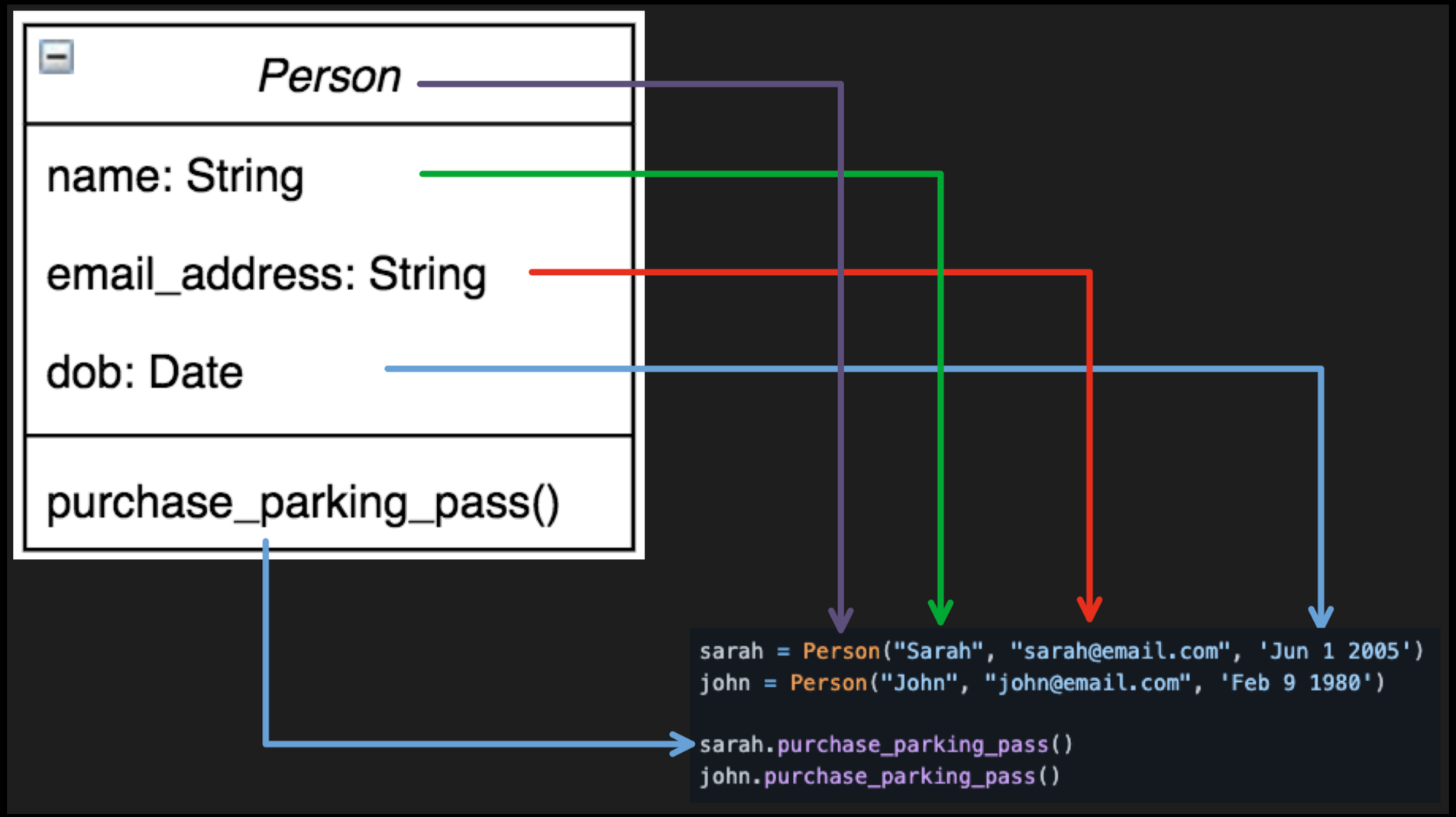
Your First Class Diagram

So we’re going to make a person class for a Web App. The class takes a name, email and date of birth. It also has a bit of behaviour. A method for purchasing a parking pass.


Person
name: String
email_address: String
dob: Date
purchase_parking_pass()

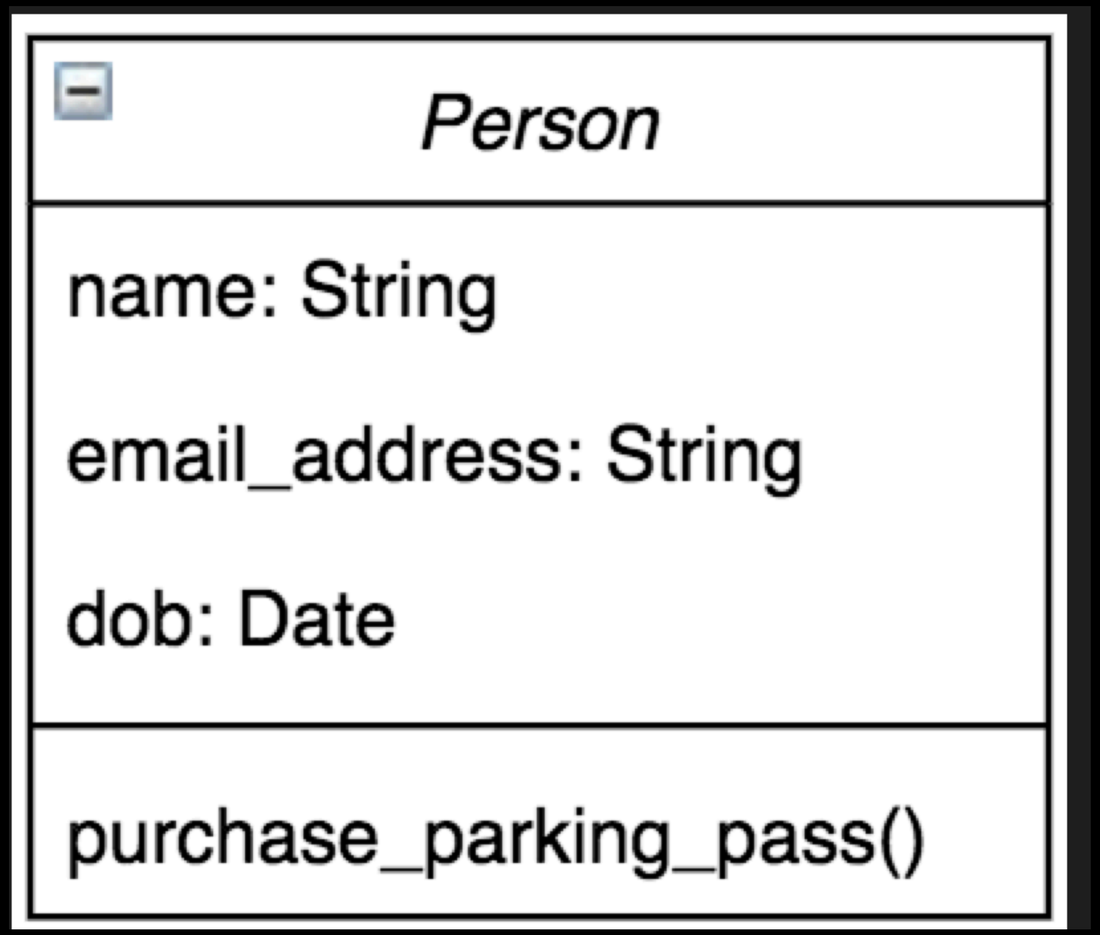
```
sarah = Person("Sarah", "sarah@email.com", 'Jun 1 2005')
john = Person("John", "john@email.com", 'Feb 9 1980')

sarah.purchase_parking_pass()
john.purchase_parking_pass()
```

How Does A Class Diagram Relate To A Class?



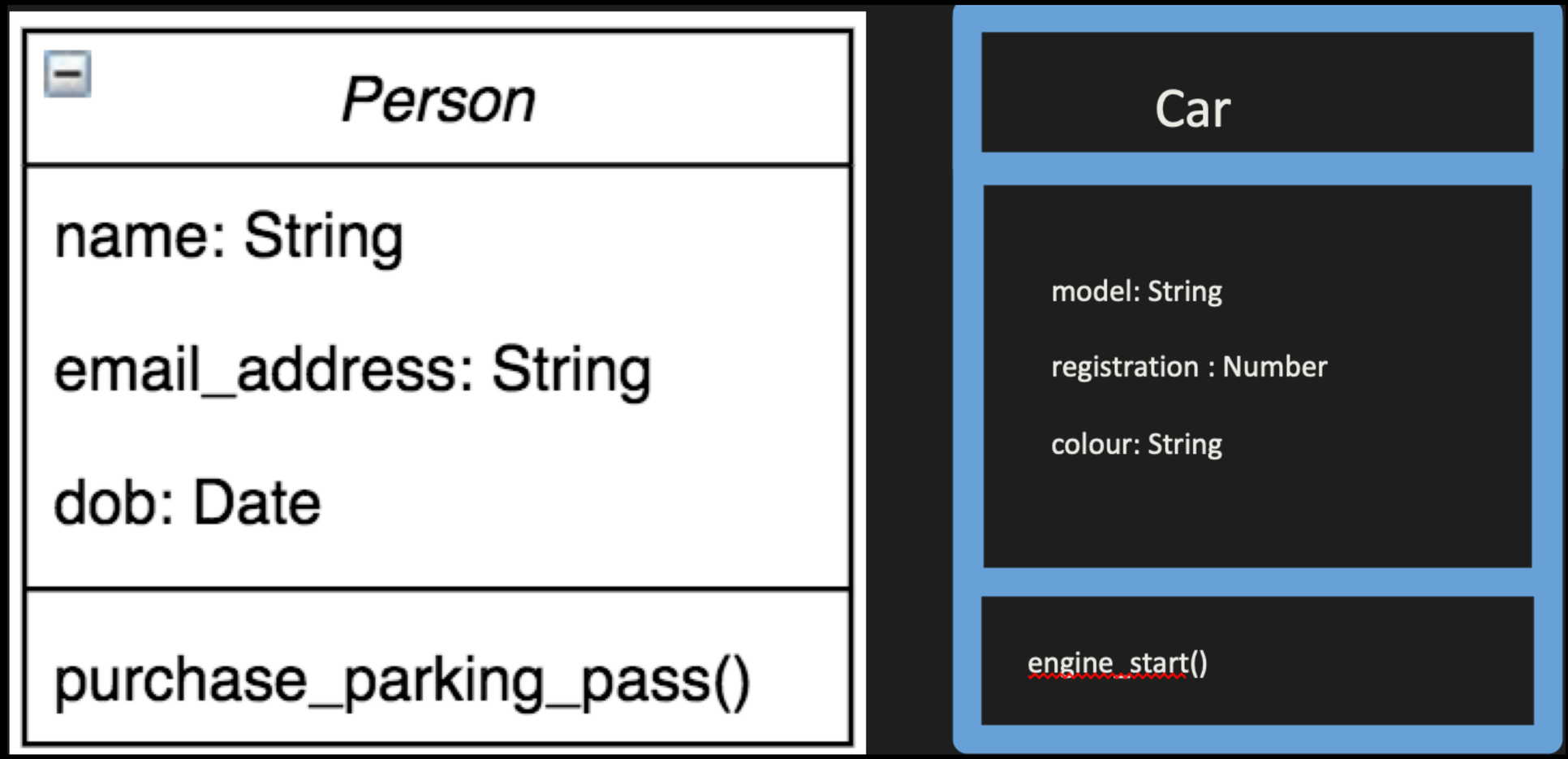
TASK 10 min

Draw a Class Diagram For A Car Class



TASK 10 min

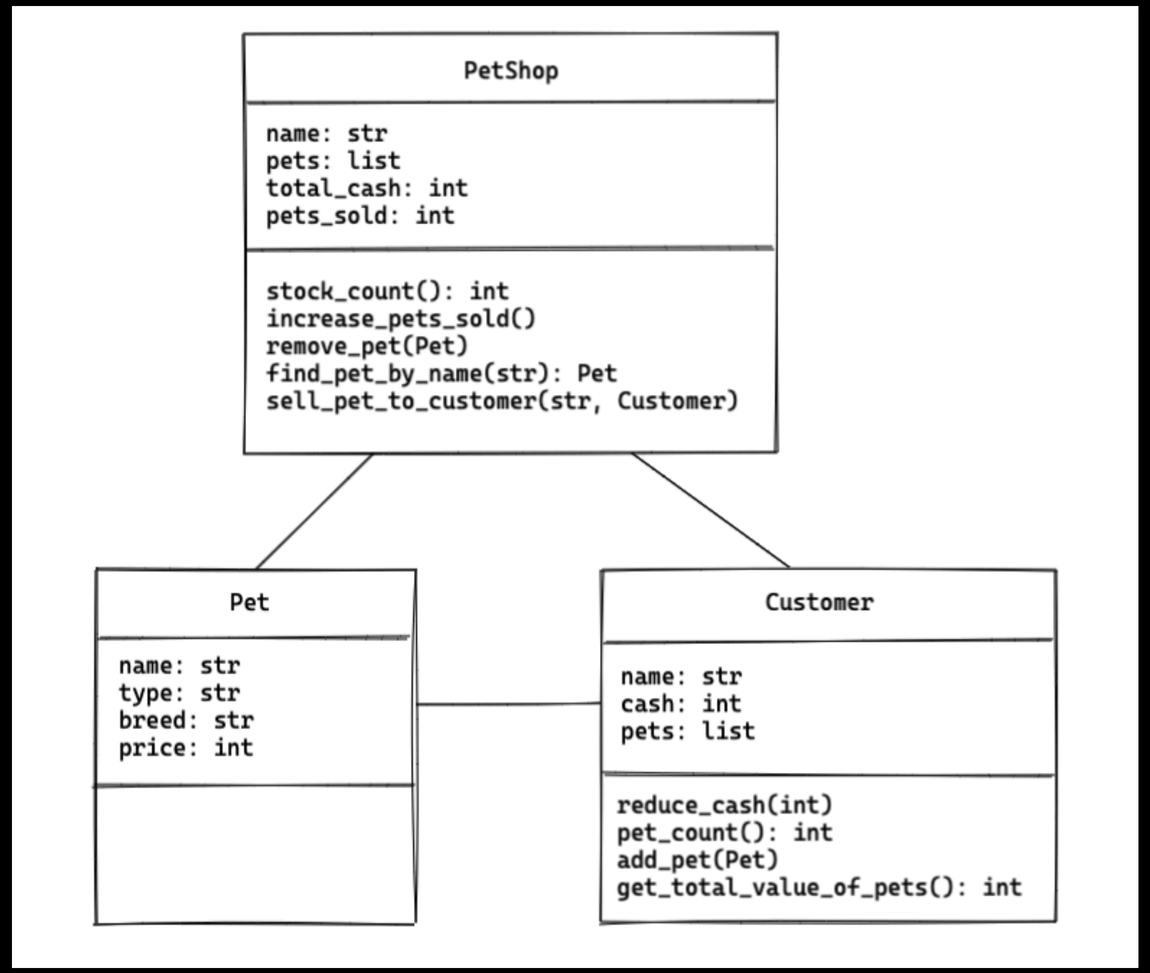
Draw a Class Diagram For A Car Class



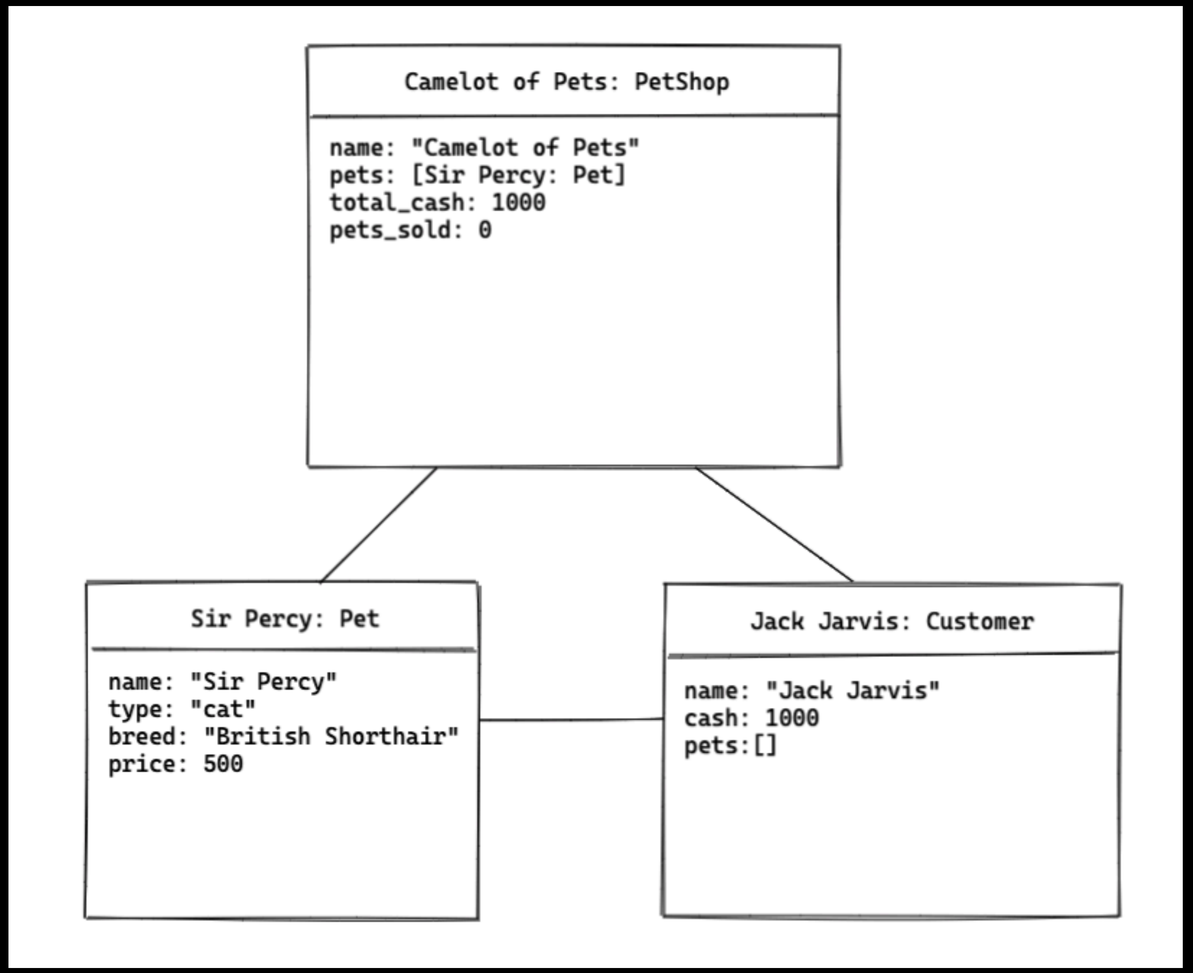
Object Diagram

An Object is an instance of a Class. Therefore an Object Diagram is an instance of a class diagram. This is good for demonstrating, visualising or simulating the inputs and outputs of our classes.

Class Diagram



Object Diagram



HOMEWORK