Programming Fundamentals Using Python

2018

Problem Set 5

Most recent updated: July 12, 2018

Objectives

1. Understand what is object-oriented programming (OOP).
2. Learn classes and methods.

Note: Solve the programming problems listed using your favorite text editor. Make sure you save your programs in files with suitably chosen names, **and try as much as possible to write your code with good style (see the style guide for python code)**. In each problem find out a way to test the correctness of your program. After writing each program, test it, debug it if the program is incorrect, correct it, and repeat this process until you have a fully working program. Show your working program to one of the cohort instructors.

Problems: Cohort sessions

1. *Classes and Methods: Mixed Fraction* Create a classed called `MixedFraction` to represent fractions that have whole number, e.g. 1 1/2. You should make use of the class `Fraction` as its parent class. You will need to override `__init__()` and `__str__()` method. The initializer should allow either two or three arguments.

   ```
   >>> mf1 = MixedFraction(3,2)
   >>> mf2 = MixedFraction(1,1,2)
   ```

   And the str method should display the fraction with the whole number, e.g. 1 1/2. You need not store the whole number as an attribute, but you may want to create a helper method called `get_three_numbers()` that returns the whole number, numerator, and denominator from a Fraction's numerator and denominator. Test your `MixedFraction` class with all the operations that `Fraction` class supports.

2. *Classes: Vehicle Polymorphism* Create a super class Vehicle that takes in miles per gallon (mpg) and vehicle identification number (vin). The class should support a method to return the type of the Vehicle. There are three types: Car, Van, and Truck. It should also supports a method to get VIN, and description. The description method should return a string that states the attributes that are common to all vehicles, i.e. mpg and vin. It should also support a method to check if the vehicle is reserved or not, and a method to set the reservation status. The class Vehicle has three sub classes, i.e. Car, Van and Truck. Each of these classes have different arguments for its initialization. The method that gives the description is overridden in these child classes to give a more detailed description of the class.

3. *Classes: Vehicle Rental Program* Refer to the other handout and complete the Vehicle Rental Program.

End of Problem Set 5.

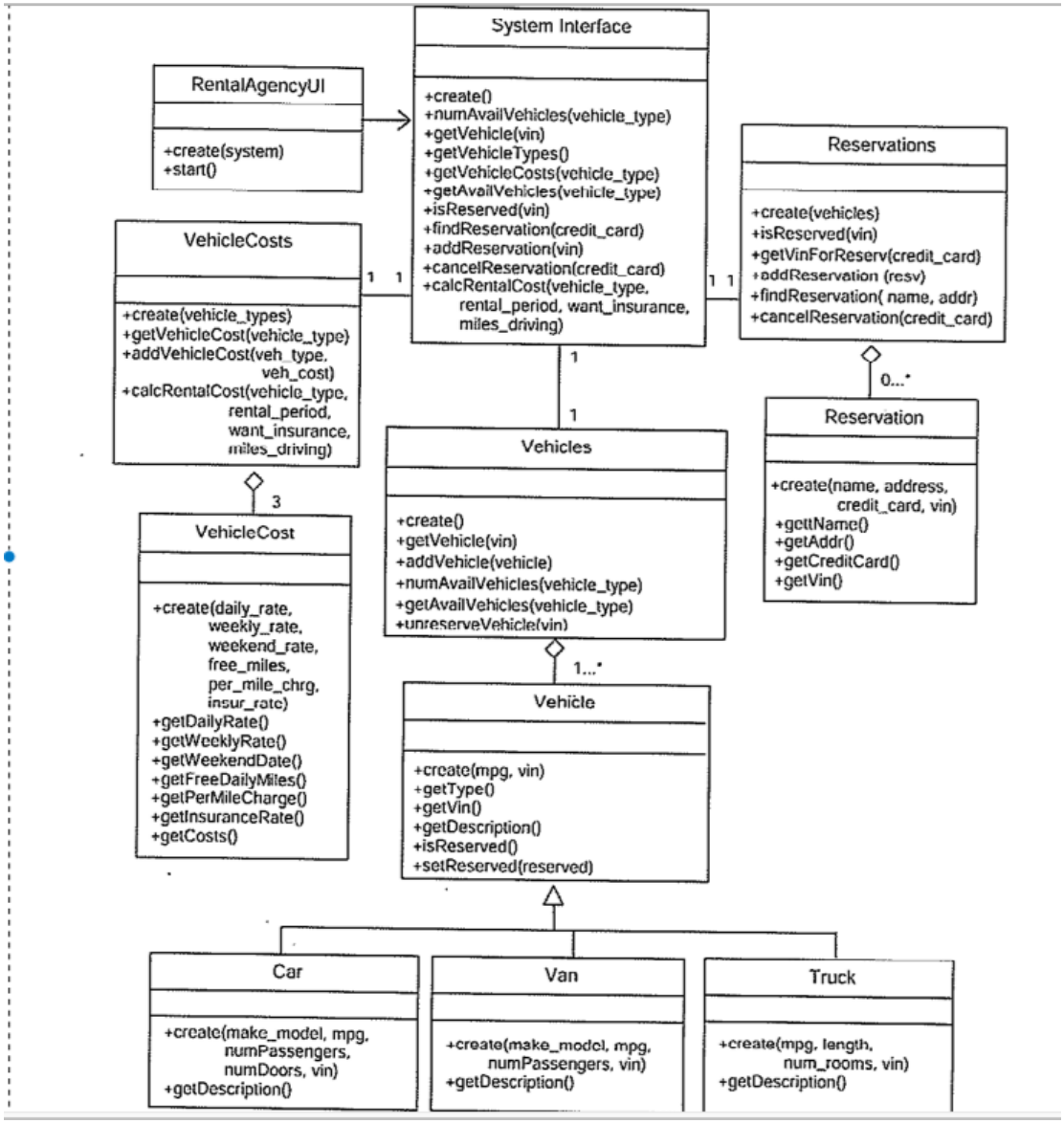

Figure 1: UML Diagram for Vehicle Rental Program.