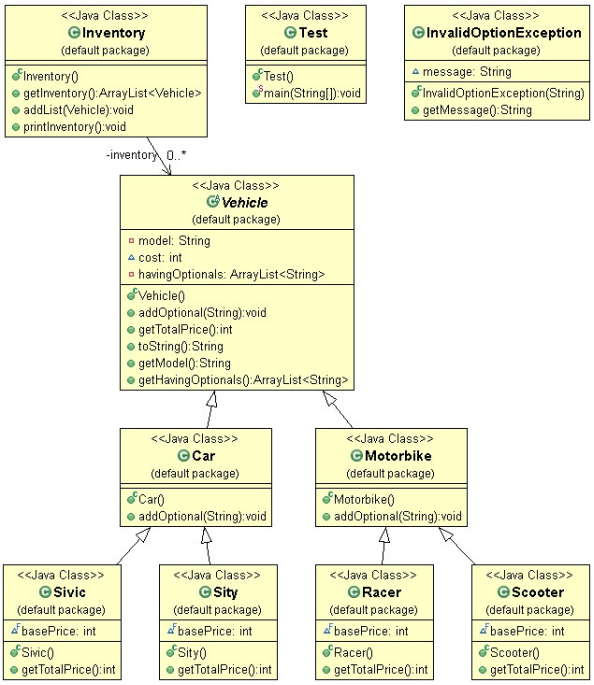
Final Project

Gökhan Mutlu

June 13, 2021

A) Provide the class diagram of your implementation.



B) What are the differences between an abstract class and a concrete class? Which class(es) can be defined as abstract classes which of them should be concrete in your code?

In an abstract class, the object of such class cannot be created directly using the new keyword. there can be an abstract method or not in abstract

class. if there is a abstract method it can not contain body. If we look at concrete class, concrete classes can create directly an object with using new keyword. if there is a method, it must be have a body. In my code, Vehicle class can be defined as abstract class. Other classes should be concrete class.

C) What is Encapsulation? Have you applied encapsulation in your implementation? Explain where and how you applied.

Encapsulation hides the implementation details from users. If an attribute is private it means it can only be accessed within the same class. If we use public getter and setter methods to update and read the private attributes then the outside class can access those private attributes via getter and setter. I used it in Vehicle and Inventory classes. For model attribute in vehicle class i used it as getter in inventory class. For havingOptionals, i used it as a getter in motorbike and car classes to add new option to the list.

D) What are the advantages of Inheritance in Object Oriented Programming? Have you used inheritance in your implementation? Which super class(es) have you used and what did you benefit from these super classes.

One of the biggest advantages of inheritance is that it prevents us from getting repetitive. For example, it saves you from writing the same code over and over. Also increases the readability of the code.The Super classes I used are Car, Motorbike, and top class Vehicle. I only used a function in the top class to output the subclasses, the others were not needed. Also i have used a function(addOptional) in Motorbike and Car classes to add specifications and the subclasses(Sivic, Sity, Racer, Scooter) benefited from there.

E) Explain the usage of final keyword in java. Can you use final keyword in any part of your implementation? Explain why or why not?

After using the final keyword, whatever the method or variable is cannot be changed during coding, it can only be accessed. Also, you can't do overriding. I used the last keyword to define baseprice in each of the subclasses(Sivic, Sity, Racer, Scooter), because baseprice of these vehicles will never change.

F) Have you benefited from polymorphism in your implementation. If yes, copy the code segment where you use polymorphism to your report and give the name of the polymorphic variable.

I have benefited of polymorphism. The place I benefited:

```
Vehicle vehicle = new Sivic();
vehicle = new Sivic();
vehicle = new Sity();
vehicle = new Racer();
vehicle = new Scooter();
```

All of my object was created on the same variable.