Brandon Hobbs

CS-230

1-7-2022

Module 1: UML Class Diagrams

Diagram

Description automatically generated

*Briefly describe the object-oriented programming principles illustrated in the class diagram.*

The above UML diagram shows 4 classes and their associated variables and methods. If we look at the Bicycle class one can see that the 4 variables (gear, cost, etc.) are marked as private (- sign in front) which indicates that no other classes or functions, outside Bicycle, have access. This is an example of encapsulation. Furthermore, the methods within the class are public (+ sign) and are following the getter/setter technique to allow other classes to mutate/set and get the data in the private variables.

The open arrows from Bicycle to TwoWheeled to Vehicle indicate the OOP principle of inheritance. Even though TwoWheeled and Vehicle have no properties, in theory, Bicycle would inherit their attributes and have its own as part of any Bicycle object.

Bicycle also shows the OOP principle of polymorphism with the 4 constructors. The respective constructors will be called based upon what set of parameters are passed when the object is instantiated.

Finally, while not explicitly shown Abstraction is also implied in the UML Class diagram. For example, anyone consuming the program or expanding Driver to add more Bicycle objects need not know how Bicycle works, internally, beyond consuming the public methods.