Name: proj5.cpp, Car.cpp, Vehicle.cpp

Author: Dylan Wong

Class: CS 202.1001

Purpose: Demonstrate an understanding of inheritance by creating classes to be utilized by the driver code provided.

Vehicle (base) Class:

* The rental car class has a default, parameterized, and copy constructor, the first takes no arguments, the second takes a vin and an LLA, the third takes a Vehicle object address.
* The setters and getters do as there name implies changing and retrieving the values of the member variables as well as static data.
* Overload and insertion operators were utilized to print out the object to the screen in a readable format and to change the LLA of one Vehicle object to that of another.
* “availibleVin” method takes a possible vin as an argument and returns a valid vin either from the function parameter, s\_idgen, or s\_deleted\_vins (if any are present).
* “Move” method does nothing but produce debug output.
* Private variables: m\_vin, m\_lla, s\_idgen, s\_deleted\_vins.

Car (derived) Class:

* The rental car class has a default, parameterized, and copy constructor, the first takes no arguments, the second takes a vin and an LLA, the third takes a Car object address.
* The setters and getters do as there name implies changing and retrieving the values of the member variables as well as static data.
* Overload and insertion operators were utilized to print out the object to the screen in a readable format and to change the Plates and LLA of one Car object to that of another.
* “Drive” method sets m\_throttle to a given value.
* “Move” method changes a Car object's lla property and calls Drive() at a speed of 75.
* Private variables: m\_plates, m\_throttle.

The main function was provided and creates Car objects, Vehicle objects, and tests other class methods enumerated above.

Ways the program could be improved:

* I could have let the user have control over the throttle value when Drive()-ing the Car/Vehicle.