**EXSM 3936: JavaScript II - Inheritance & Polymorphism**

Deadline: Sunday May 29, 2022 at 11:59 PM

<https://classroom.github.com/a/XjJ5VTR8>

Introduction

This assignment is intended to cement your understanding of inheritance and polymorphism.

Instructions

You are to upgrade and divide your class from Assignment 1 that simulates a car. The classes should be structured as follows (data types are intended, they do not need to be enforced):

**Vehicle:**

1. *method:* constructor(make [string], model [string], engineCylCount [number], engineCylConfig [string], transmissionType [string])
2. *property:* make [string]
   1. Initialized from constructor parameter.
3. *property:* model [string]
   1. Initialized from constructor parameter.
4. *property:* engine [object]
   1. *property*: cylCount [number]
      * Initialized from constructor parameter.
   2. *property*: cylConfig [string]
      * Initialized from constructor parameter.
   3. *property:* running [boolean]
      * Initialized to false.
5. *property:* transmission [object]
   1. *property:* type [string]
      * Initialized from constructor parameter.
6. *property:* odometerKM [number]
   1. Initialized to 0.
7. *method:* startEngine()
   1. Sets the engine’s running property to true, if it is false.
   2. If the engine’s running property is already true, throw an exception.
8. *method:* stopEngine()
   1. Sets the engine’s running property to false, if it is true.
   2. If the engine’s running property is already false, do nothing.
9. *method:* drive(distance [number])
   1. If the engine is currently not running, throw an exception.
   2. Increase the odometerKM by a value equal to the distance parameter.
10. method: toString()
    1. Returns an english-readable sentence describing the vehicle.

**Car** (inherits from Vehicle):

1. method: toString()
   1. Overrides the toString() method of the parent, describing the same properties but specifying that the object is a car.

**PickupTruck** (inherits from Vehicle)**:**

1. *method:* constructor(make [string], model [string], engineCylCount [number], engineCylConfig [string], transmissionType [string], payload [number], bedLength [number])
   1. Calls the parents’ constructor after setting unique properties.
2. *property:* payload [number]
   1. Initialized from constructor parameter.
3. *property:* bedLength [number]
   1. Initialized from constructor parameter.
4. method: toString()
   1. Overrides the toString() method of the parent, describing the same properties plus the additional ones unique to this class, and specifying that the object is a pickup truck.

**Van** (inherits from Vehicle)**:**

1. *method:* constructor(make [string], model [string], engineCylCount [number], engineCylConfig [string], transmissionType [string], style [string])
   1. Calls the parents’ constructor after setting unique properties.
2. *property:* style [string]
   1. Initialized from constructor parameter.
3. method: toString()
   1. Overrides the toString() method of the parent, describing the same properties plus the additional ones unique to this class, and specifying that the object is a van.

Once your class is constructed, **using the console template in your repository**, create a script that will (safely, remember to catch potential exceptions):

1. Instantiate an array containing a car, a pickup truck, and a van.
2. Loop through the array and:
   1. Turn on the engine of the vehicle.
   2. Drive for a random distance between 50km and 100km (inclusive, round to the nearest tenth).
   3. Turn off the engine of the vehicle.
3. Loop through the array a second time and output the toString() results of each object implicitly.

Example:

A Ford Taurus (car) with a 6 Cylinder V6 engine that is not currently running, a 5 Speed Manual transmission, and 67 kilometers on the odometer.

A Mitsubishi Mighty Max (1/4 ton pickup truck) with a 4 Cylinder Inline-4 engine that is not currently running, a 3 Speed Automatic transmission, a 116 inch bed, and 89 kilometers on the odometer.

A Dodge Grand Caravan (passenger van) with a 6 Cylinder V6 engine that is not currently running, a 6 Speed Automatic transmission, and 53 kilometers on the odometer.

Criteria

| Requirement | Marks Available | Mark(s) Awarded | Comments |
| --- | --- | --- | --- |
| Script displays the values expected, and the steps outlined are followed. | 1 |  |  |
| Class Declarations |  |  |  |
| Each of the 3 child classes inherit from Vehicle (0.5 marks, all or nothing). | 0.5 |  |  |
| Each of the 3 child classes implement their constructors correctly (0.5 each). | 1.5 |  |  |
| Each of the 3 child classes implement toString() correctly (1 mark, all or nothing). | 1 |  |  |
| Final output is achieved implicitly (no direct calling of toString()). | 0.5 |  |  |
| Distance driven is determined randomly between the given bounds (inclusive, rounded). | 0.5 |  |  |
| Deductions: |  |  |  |
| README.md file is not present in the docs directory, or does not contain a reasonable minimum amount of information. | -1 |  |  |
| Version control is not used appropriately (-1 for vague or nonsensical commit messages, up to -2 for insufficient commits, -3 for a single commit regardless of message). | -3 |  |  |
| Whitespace is not managed (tab stops / formatting). | -1 |  |  |
| Variable, class, constant, property and/or method names do not meet [the code style requirements](https://docs.google.com/document/d/12iEtYtYiKwRAjhLayI-JVNVqXkud3X_PtOE9D0sU9Cw/edit) (-0.5 each declaration). | -3 |  |  |
| Missing semicolons (-0.5 each). | -2 |  |  |
| Unhandled exception(s) generated (-1 each scenario that generates one). | -3 |  |  |

**Total: /5**