# CPSC1012 Advanced Portfolio 5 – Classes Relationships

**Weight: 5% of your final mark**

## Car Instrument Simulator

Design a set of classes that work together to simulate a car’s fuel gauge and odometer. The classes you will design are the following:

* The **FuelGauge** Class: This class will simulate a fuel gauge. Its responsibilities are as follows:
  + To know the car’s current amount of fuel, in liters.
  + To report the car’s current amount of fuel, in liters.
  + To be able to increment the amount of fuel by 1 liter. This simulates putting fuel in the car. (The car can hold as maximum of 60 liters.)
  + To be able to decrement the amount of fuel by 1 liter, if the amount of fuel is greater than 0 liters. This simulates burning fuel as the car runs.
* The **Odometer** Class: This class will simulate the car’s odometer. Its responsibilities are as follows:
  + To know the car’s current kilometer.
  + To report the car’s current kilometer.
  + To be able to increment the kilometer by 1 kilometer. The maximum kilometer the odometer can store is 999 kilometers. When this amount is exceeded, the odometer resets the current kilometer to 0.
  + To be able to work with a **FuelGauge** object. It should decrease the **FuelGauge** object’s current amount of fuel by 12 liters for every 100 kilometers traveled. (The car’s fuel economy is 12 liters per 100 kilometers).

Demonstrate the classes by creating a program that instantiate an Odometer and simulate filling the car up with fuel, and then run a loop that increments the odometer until the car runs out of fuel. During each loop iteration, print the car’s current kilometer and amount of fuel. Create a menu to allow the user to add fuel, drive the car, or exit (see sample output).

## Marking Guide

|  |  |  |
| --- | --- | --- |
| **Description** | **Marks Possible** | **Marks Earned** |
| Correctness   * FuelGauage Class – cannot fill beyond 60 liters of fuel * Odometer Class – odometer resets to 0 after 999 kilometers | 2 |  |
| Structure   * Classes have correct data fields, properties, constructors, and methods * Loop to add fuel 1 liter per iteration * Loop to drive 1 km per iteration and burn correct amount of fuel * Stop driving if out of fuel * Display current kilometer and fuel while driving * Menu works correctly (validate **ALL** input) | 6 |  |
| Style and Readability   * Horizontal and vertical white space * Meaningful identifiers | 1 |  |
| Documentation   * Opening documentation * Source code comments | 1 |  |
| **Total:** | **10** |  |

## Coding Requirements

The following coding standards must be followed when developing your program:

* Your C# Console App project must be named as **AdvancedPortfolio05-*YourFullName*** (e.g.: AdvancedPortfolio05-CodeGuru)
* Opening documentation at the beginning of the source file describing the **purpose**, **input**, **process**, **output, author, last modified date** of the program.
* Write only one statement per line.
* Write only one declaration per line.
* Use camelCase for local variable names and method parameter name.
* Use PascalCase for method names and constant variable names.
* If continuation lines are not indented automatically, indent them one tab stop (four spaces).
* Do NOT use the goto statement.
* There can only be one exit point for a loop, do not use the break statement inside a loop.
* Do NOT use static variables.

## Demonstration and Submission Requirements

* Demonstrate to your instructor your working program before submitting to Moodle. Be prepared to answer questions about your code after the demonstration. **No marks will be given** if you are unable to explain your code or if you submit your project without a demonstration of your working program to your instructor.

## Sample Output

|------------------------------|

| CPSC1012 Adv. Portfolio #5 |

|------------------------------|

| 1. Add Fuel |

| 2. Drive Car |

| 0. Exit |

|------------------------------|

Option: 4

Please enter a valid number between 0 and 2 inclusive

Option: a

Please enter a valid number between 0 and 2 inclusive

Option: 1

Enter amount of fuel to add: 60

|------------------------------|

| CPSC1012 Adv. Portfolio #5 |

|------------------------------|

| 1. Add Fuel |

| 2. Drive Car |

| 0. Exit |

|------------------------------|

Option: 1

Enter amount of fuel to add: 60

|------------------------------|

| CPSC1012 Adv. Portfolio #5 |

|------------------------------|

| 1. Add Fuel |

| 2. Drive Car |

| 0. Exit |

|------------------------------|

Option: 2

Enter distance to drive: 400

Driving the car

Km: 1 Fuel: 59.88 l

Km: 2 Fuel: 59.76 l

. . .

. . .

. . .

Km: 399 Fuel: 12.12 l

Km: 400 Fuel: 12.00 l

|------------------------------|

| CPSC1012 Adv. Portfolio #5 |

|------------------------------|

| 1. Add Fuel |

| 2. Drive Car |

| 0. Exit |

|------------------------------|

Option: 1

Enter amount of fuel to add: 30

|------------------------------|

| CPSC1012 Adv. Portfolio #5 |

|------------------------------|

| 1. Add Fuel |

| 2. Drive Car |

| 0. Exit |

|------------------------------|

Option: 2

Enter distance to drive: 400

Driving the car

Km: 401 Fuel: 41.88 l

Km: 402 Fuel: 41.76 l

. . .

. . .

. . .

Km: 749 Fuel: 0.12 l

Km: 750 Fuel: 0.00 l

Out of Fuel ... please add fuel

|------------------------------|

| CPSC1012 Adv. Portfolio #5 |

|------------------------------|

| 1. Add Fuel |

| 2. Drive Car |

| 0. Exit |

|------------------------------|

Option: 1

Enter amount of fuel to add: 60

|------------------------------|

| CPSC1012 Adv. Portfolio #5 |

|------------------------------|

| 1. Add Fuel |

| 2. Drive Car |

| 0. Exit |

|------------------------------|

Option: 2

Enter distance to drive: 450

Driving the car

Km: 751 Fuel: 59.88 l

Km: 752 Fuel: 59.76 l

. . .

. . .

. . .

Km: 998 Fuel: 30.24 l

Km: 999 Fuel: 30.12 l

Km: 0 Fuel: 30.00 l

Km: 1 Fuel: 29.88 l

. . .

. . .

. . .

Km: 199 Fuel: 6.12 l

Km: 200 Fuel: 6.00 l

|------------------------------|

| CPSC1012 Adv. Portfolio #5 |

|------------------------------|

| 1. Add Fuel |

| 2. Drive Car |

| 0. Exit |

|------------------------------|

Option: 0

Goodbye - please come again ...

**Note**: Your output does not have to exactly match this sample output; this is only a guide as to how this can be done.