CT2106

Object Orientated Programming



Title:

Assignment 1

DeadLine:

28/09/2018

23:59pm

By:

Killian O'Dálaigh – 18101573

Paul Kirwan - 17321313

Table of Contents

Code Brief -------------------------------------------------------------------------- Page 2

Code Design ------------------------------------------------------------------------ Page 2

Code --------------------------------------------------------------------------------- Page 2

Outputs ------------------------------------------------------------------------------ Page 9

Contributions ----------------------------------------------------------------------- Page 10

Evaluation -------------------------------------------------------------------------- Page 10

Code Brief:

A company called ‘MkX’ cars have designed a revolutionary new modular car. You can design the car online and choose a Car body, a particular Engine, Door, and Wheel types. MkX have designed a simple software simulation in Java to demonstrate how different car/engine/door/wheel configurations work.

Code Design:

The Car class must have an Engine and Door Class. The Engine Class must in turn have a Wheel class. This means that the Car class can access the Engine, Door and Wheel class, however for example the Door class can not access the Wheel class as it has no realation.

Our code is very simple. Having inputed the data of the car it creates a Car class. This Classes constructor in turn creates an Door Class and an Engine Class, the later of which creates a Wheel Class in its Constructor.

The Cars fuel is then set to an input to calculate how far it can drive on a tank of fuel.

Then the drive method calculates how far the car can drive.

This value is then printed out to the console.

Code:

/\*

\* Title: Assignment 1

\* Description: Creates a car with variable parts and calculates the range of this car

\* Author: Killian O'Dálaigh, Paul Kirwan

\* Student Num: 18101573, 17321313

\*/

/\*

\* Title: Main

\* Description: Running Class

\*/

**public** **class** Main {

/\*

\* Title: main

\* Parameters: String args

\* Return: Void

\* Description: Running method

\*/

**public** **static** **void** main(String[] args) {

// Creates a new instance of obj Car

Car car = **new** Car("Mark7", "PL9", 9, "UP2", 2, "Wichelin8", 8);

// Sets fuel to 50

car.setFuel(25);

// Calculates the distance covered

car.drive();

// Prints out the total Km's covered

System.***out***.println("Kilometers Completed = " + car.getKmCompleted() + "Km");

}// End main

}// End Class Main

/\*

\* Title: Car

\* Description: The Class for the Car of the car

\*/

**public** **class** Car {

// Initializes variables

**private** String name; // Holds the name

**private** **double** kmCompleted; // Holds the km completed by the car

**private** Engine engine; // Holds the Class engine

**private** Door door; // Holds the Class Door

/\*

\* Title: Class Constructor

\* Parameters:

\* - String name -> Name of model

\* - String eng -> Name of Engine model

\* - int fuelEff -> Fuel efficiency of the engine

\* - String drName -> Name of Door model

\* - int drNum -> Number of doors

\* - String whNm -> Name of Wheel model

\* - int rad -> Radius of the wheel

\*

\* Description: Creates an instance of class Car

\*/

**public** Car(String name, String eng, **int** fuelEff, String drName, **int** drNum, String whNm, **int** rad)

{

// Initalizes variable

**this**.name = name;

**this**.kmCompleted = 0;

// Creates new instance of Class Engine

engine = **new** Engine(eng,fuelEff, whNm, rad);

// Creates new instance of Class Door

door = **new** Door(drName, drNum);

}// End Constructor

/\*

\* Title: drive

\* Parameters: None

\* Return: Void

\* Description: Calculates the total distance covered by the car

\*/

**public** **void** drive() {

// Checks if there is fuel to drive the car

**if**(engine.getFuelLevel()<=0) {

//If not print error

System.***out***.println("Error 01: No fuel in tank");

}// End if

**else** {

// Drives car until tank is empty

**while** ((engine.getFuelLevel())>0) {

// Calculates the amount of km driven per liter expended (fuellEff is in Turns per Liter)

kmCompleted += (engine.getWheel().getCircumference())\*(engine.getFuelEff());

engine.setFuelLevel((engine.getFuelLevel()-1));

}// End while

}// End else

}// End drive

/\*

\* Getters and Setters for Variables

\*/

**public** **void** setFuel(**int** fuel) {

**this**.engine.setFuelLevel(fuel);

}// End setFuel

**public** String getName() {

**return** **this**.name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **double** getKmCompleted() {

**return** **this**.kmCompleted;

}

**public** **void** setKmCompleted(**double** kmCompleted) {

**this**.kmCompleted = kmCompleted;

}

**public** Engine getEngine() {

**return** **this**.engine;

}

**public** **void** setEngine(Engine engine) {

**this**.engine = engine;

}

**public** Door getDoor() {

**return** **this**.door;

}

**public** **void** setDoor(Door door) {

**this**.door = door;

}

}// End Class Car

/\*

\* Title: Engine

\* Description: The Class for the Engine of the car

\*/

**public** **class** Engine {

**private** **double** fuelLevel; // Holds the fuel level

**private** String name; // Holds the name of the engine

**private** **double** fuelEff; // Turns per Litre

**private** Wheel wheel; // Holds the wheel class

/\*

\* Title: Class Constructor

\* Parameters: String, int

\* Description: Creates an instance of class Engine

\*/

**public** Engine(String name, **int** fuelEff, String whName, **int** rad) {

// Initializes variables

**this**.fuelEff = fuelEff;

**this**.name = name;

**this**.fuelLevel = 0;

// Creates new instance of type Wheel

wheel = **new** Wheel(whName, rad);

}// End Constructor

/\*

\* Getters and Setters for Variables

\*/

**public** **double** getFuelLevel() {

**return** fuelLevel;

}

**public** **void** setFuelLevel(**double** fuelLevel) {

**this**.fuelLevel = fuelLevel;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **double** getFuelEff() {

**return** fuelEff;

}

**public** **void** setFuelEff(**double** fuelEff) {

**this**.fuelEff = fuelEff;

}

**public** Wheel getWheel() {

**return** wheel;

}

**public** **void** setWheel(Wheel wheel) {

**this**.wheel = wheel;

}

}// End Class Engine

/\*

\* Title: Door

\* Description: The Class for the Door of the car

\*/

**public** **class** Door {

// Variables

**private** String name; // Holds the name

**private** **int** quantity; // Holds the quantity

/\*

\* Title: Class Constructor

\* Parameters: String, int

\* Description: Creates an instance of the class

\*/

**public** Door(String drName, **int** drNum) {

**this**.name = drName;

**this**.quantity = drNum;

}// End Constructor

/\*

\* Getters and Setters for Variables

\*/

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **int** getQuantity() {

**return** quantity;

}

**public** **void** setQuantity(**int** quantity) {

**this**.quantity = quantity;

}

}// End Class Door

/\*

\* Title: Wheel

\* Description: The Class for the Wheel of the car

\*/

**public** **class** Wheel {

// Variables

**private** **int** radius; // holds the radius of the wheel

**private** String name; // holds the name of the wheel

**private** **double** circumference; // holds the circumference

/\*

\* Title: Class Constructor

\* Parameters: String, int

\* Description: Creates an instance of class Wheel

\*/

**public** Wheel(String name, **int** radius) {

// Initializes variables

**this**.name = name;

**this**.radius = radius;

**this**.circumference = (2\*Math.***PI***\***this**.radius);

}// End Constructor

/\*

\* Getters and Setters for Variables

\*/

**public** **double** getCircumference() {

**return** circumference;

}// End getDistance

**public** **int** getRadius() {

**return** radius;

}

**public** **void** setRadius(**int** radius) {

**this**.radius = radius;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **void** setCircumference(**double** circumference) {

**this**.circumference = circumference;

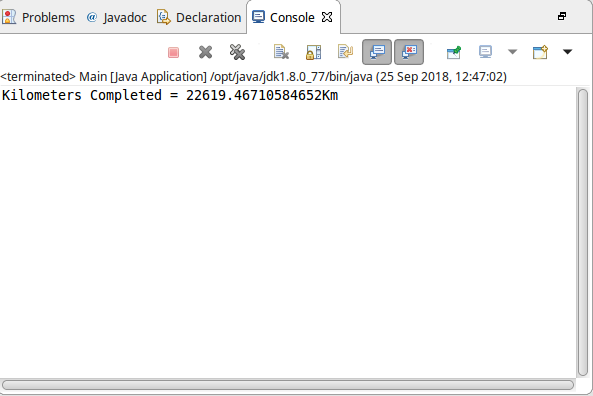
}

}// End Class Wheel

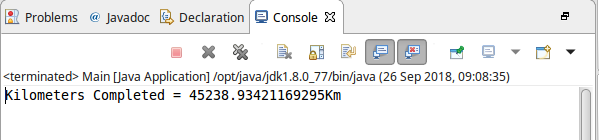
Outputs:

The following are various outputs using different inputs.

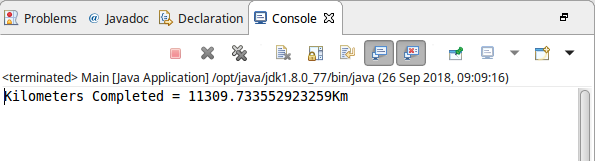
1. This is the output from the above code with the fuel set to 50



2. This is the output from the above code with the fuel set to 100



3. This is the output from the above code with the fuel set to 25



Killian O'Dálaigh Contribution:

As I was the more experienced with Java, my contribution was mostly the overall design and building of the code. I applied techniqued I learned from the lectures and some from personal experience in Java, such as creating instances of classes in the constructor of other classes. This created a well balanced and well designed piece of code which filled the brief. The design of the class interaction was core to the project which Paul helped with.

Paul Kirwan Contribution:

As I was less experienced with Java, my contribution was largely based on applying the new techniques that I had recently learned from going to the lectures. This involved creating accessor methods so that values from one class could be returned to another class. Similarly, I was able to create mutator methods so that values of the instance variables in a class could be edited. This was essential for completing the assignment.

Overall Evaluation:

We had no difficulty with this project, however a slight modification we could have made would have been to recieve input from the user using System.in and the Scanner objects from java.util.Scanner. However this was not a requirement for this project and thus we made the decision to not include it into my final code, as we felt it might over complicate the assignment.