Лабораторная работа 4

# Функциональные требования:

Автомобиль представлен композицией **деталей и узлов.** Каждая деталь имеет собственный вес. Каждый узел состоит из нескольких деталей и тоже характеризуется совокупным весом.

|  |  |  |  |
| --- | --- | --- | --- |
| Assembly/Unit | Name | Quantity | **\***Weight |
| Car | Engine V6 | 1 |  |
|  | Wheel | 4 |  |
|  | Transmission | 1 |  |
|  | Differential | 2 |  |
|  | Door | 4 |  |
|  | Gas tank | 1 |  |
| Engine V6 | Piston system | 1 |  |
|  | Valve system | 1 |  |
|  | intake manifold | 1 |  |
|  | exhaust manifold | 2 |  |
|  | Radiator | 1 |  |
| Piston system | Piston | 4 |  |
|  | Crankshaft | 1 |  |
|  | Link | 4 |  |
|  | Gasket | 1 |  |
| Value system | Value | 24 |  |
|  | Camshaft | 2 |  |
| Wheel | Disk | 1 |  |
|  | Tire | 1 |  |
| Door | Glass | 1 |  |
|  | door card | 1 |  |
| Differential | gears | 5 |  |
|  | bearing | 3 |  |
|  | shank | 1 |  |

Реализовать решение, позволяющее клиенту (Plant) подсчитать конечный вес собранного автомобиля и вывести детализированную информацию о кол-ве деталей, их вес, состав узлов и их вес и общий вес автомобиля. **\***Вес узлов/деталей задайте самостоятельно.

# Не функциональные требования:

* Платформа: Консольное приложение для OS Windows.
* Язык реализации: C# / Java / C++.
* Вызов каждого метода должен сопровождаться простым выводом в консоль информации о действии метода/его результат.
* Структура проекта должна включать каталоги для размещения классов, интерфейсов и абстрактных классов с именованием согласно предметной области.