**11 Полиморфизм**

Задания №1. Составить программу с одним родительским классом и двумя потомками. Потомки должны содержать виртуальные функции. Создать виртуальную функцию выдачи результатов расчета методов на экран монитора с указанием названий и полей, и их значений соответствующего объекта. Составить тестирующую программу с выдачей протокола на экран монитора. При этом создать объекты базового и производных типов, используя полиморфный контейнер - массив ссылок базового класса на объекты базового и производных классов (количество объектов>=5).

Листинг программы:

using System;

using System.Collections.Generic;

public class Automobile

{

public string Name { get; set; }

public Automobile(string name)

{

Name = name;

}

public virtual double CalculateFuelConsumption()

{

return 0;

}

public virtual void DisplayInfo()

{

Console.WriteLine($"Автомобиль: {Name}");

}

}

public class Truck : Automobile

{

public double LoadCapacity { get; set; }

public Truck(string name, double loadCapacity) : base(name)

{

LoadCapacity = loadCapacity;

}

public override double CalculateFuelConsumption()

{

return Math.Sqrt(LoadCapacity) \* 100;

}

public override void DisplayInfo()

{

Console.WriteLine($"Грузовой автомобиль: {Name}, Грузоподъемность: {LoadCapacity} т, Расход горючего: {CalculateFuelConsumption():F2} л/100км");

}

}

public class PassengerCar : Automobile

{

public double EngineVolume { get; set; }

public PassengerCar(string name, double engineVolume) : base(name)

{

EngineVolume = engineVolume;

}

public override double CalculateFuelConsumption()

{

return 2.5 \* EngineVolume;

}

public override void DisplayInfo()

{

Console.WriteLine($"Легковой автомобиль: {Name}, Объем двигателя: {EngineVolume} см^3, Расход горючего: {CalculateFuelConsumption()} л/100км");

}

}

class Program

{

static void Main()

{

List<Automobile> automobiles = new List<Automobile>

{

new Truck("Грузовик1", 5),

new Truck("Грузовик2", 10),

new PassengerCar("Легковой1", 2000),

new PassengerCar("Легковой2", 1500),

new PassengerCar("Легковой3", 1800)

};

foreach (var auto in automobiles)

{

auto.DisplayInfo();

}

}

}

Анализ результатов:

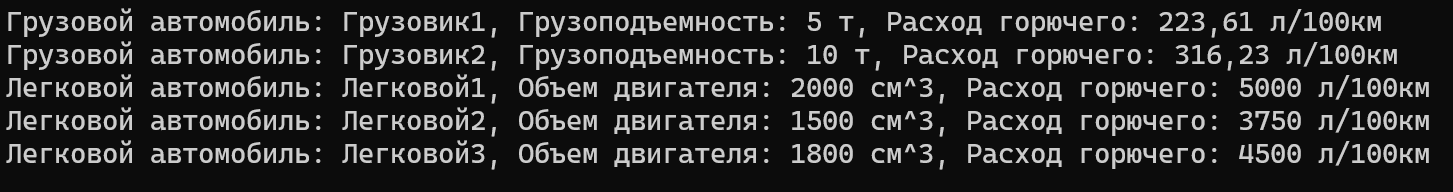


Рисунок 11.1 – Результаты работы программы