МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

НАБЕРЕЖНОЧЕЛНИНСКИЙ ИНСТИТУТ (ФИЛИАЛ) «КАЗАНСКОГО (ПРИВОЛЖСКОГО) ФЕДЕРАЛЬНОГО УНИВЕРСИТЕТА»

Кафедра Информационных систем

Отчет по лабораторным работам

дисциплина **«Программирование»**

Выполнил студент гр. 2231122

Башмаков В. В.

Проверил: преподаватель

Зубков Е. В.

Набережные Челны

2023 г.

Выполнил студент гр. 2231122

Башмаков В. В.

Проверил: ст. преподаватель

Булатова С. В.

Лабораторная работа №4

«Паттерн Наблюдатель»

Цель работы:

Практическая часть:

using System;

using System.Collections.Generic;

namespace Patterns {

public class Observer {

public Observer() {

Car car;

car = new Car();

Driver driver = new Driver("1");

Mechanic mechanic = new Mechanic("2");

car.Attach(driver);

car.Attach(mechanic);

car.StartEngine();

car.StopEngine();

car.Detach(driver);

car.Detach(mechanic);

car.StartEngine();

car.StopEngine();

}

public interface IObserver {

public void Update(string message);

}

public class Driver : IObserver {

private string \_name;

public Driver(string name) {

\_name = name;

}

public void Update(string message) {

Console.WriteLine(string.Format("Driver {0} has got message: {1}", \_name, message));

}

}

public class Mechanic : IObserver {

private string \_name;

public Mechanic(string name) {

\_name = name;

}

public void Update(string message) {

Console.WriteLine(string.Format("Mechanic {0} has got message: {1}", \_name, message));

}

}

public interface ISubject {

void Attach(IObserver observer);

void Detach(IObserver observer);

void Notify(string message);

}

public class Car : ISubject {

private List<IObserver> \_observers = new List<IObserver>();

private string \_state;

public void Attach(IObserver observer) {

\_observers.Add(observer);

}

public void Detach(IObserver observer) {

\_observers.Remove(observer);

}

public void Notify(string message) {

foreach (IObserver observer in \_observers) {

observer.Update(message);

}

}

public void StartEngine() {

\_state = "The engine is started";

Console.WriteLine(\_state);

Notify(\_state);

}

public void StopEngine() {

\_state = "The engine is stopped";

Console.WriteLine(\_state);

Notify(\_state);

}

}

}

}

Вывод: