

Министерство науки и высшего образования Российской Федерации  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ  
В.Ф.УТКИНА»  
Рязанский станкостроительный колледж РГРТУ

Лабораторная работа 36

**Дополнительная профессиональная программа  
повышения квалификации  
«Тестирование программного обеспечения (с учетом стандарта Ворлдскиллс по  
компетенции «Программные решения для бизнеса»)»**

Мазуренко Валерия Витальевна

Рязань 2022

## Практическая работа №6: Selenium WebDriver

Цель: Целью данной работы является знакомство с автоматизацией функционального тестирования при помощи C#, NUnit и Selenium WebDriver. Задание:

1. Выбрать web-site (например, rssk.ru или rsreu.ru).
2. Составить план функционального тестирования для выбранного сайта.
3. Реализовать набор тестов при помощи NUnit и Selenium WebDriver.

План тестирования должен включать в себя как минимум:

1. Проверка заголовка страницы.
2. Проверка видимости объектов
3. Переход по ссылке.
4. Заполнение текстового поля.
5. Эмуляция нажатия на кнопку.

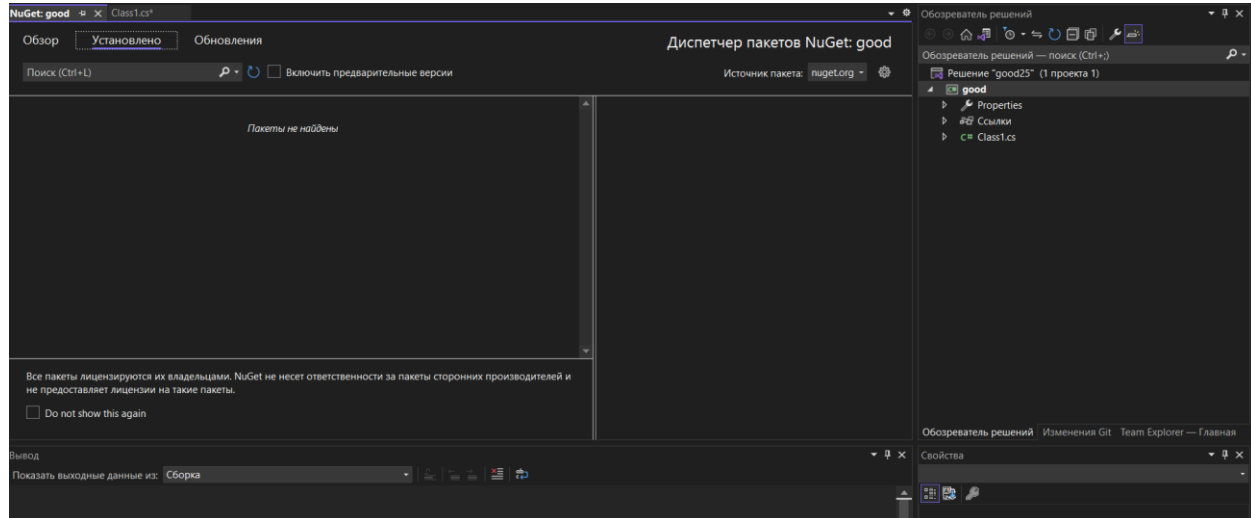
4. Процесс разработки должен вестись в локальном репозитории. В процессе разработки необходимо:

1. Совершать сохранение изменений в локальный репозиторий после реализации каждой отдельной функции.
2. Синхронизировать локальный и удалённый репозиторий в конце каждого сеанса работы над проектом.

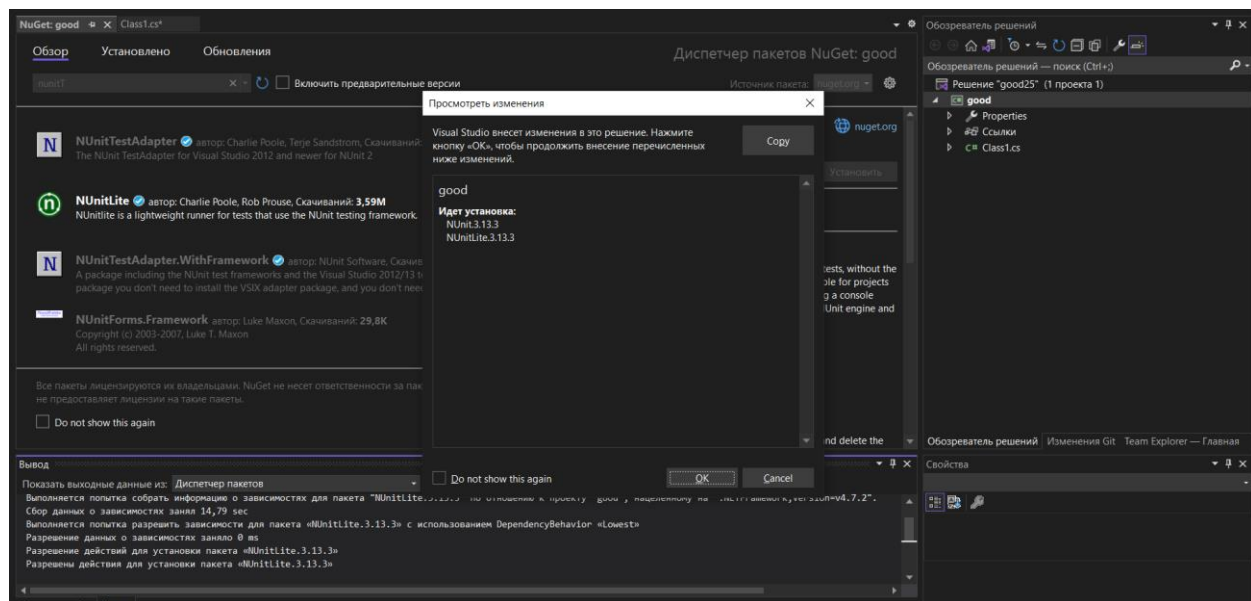
## Лабораторная работа №6: Selenium WebDriver

### Создание и настройка решения:

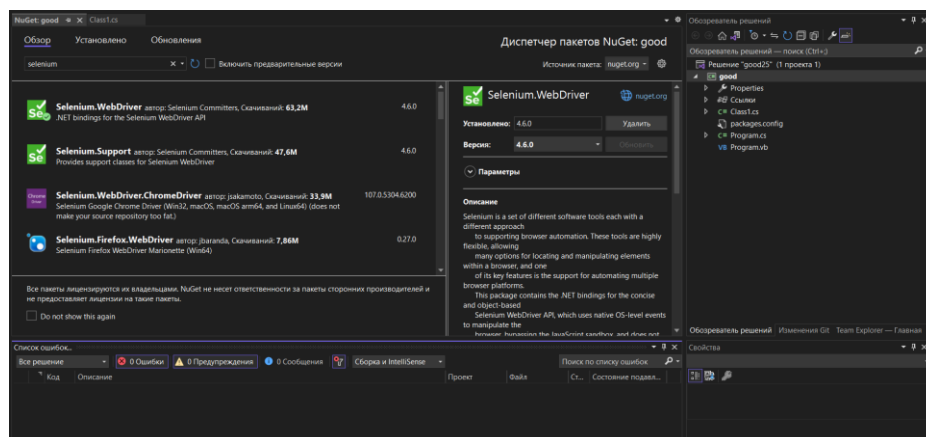
Разрабатываемый проект, по сути, не является программным приложением, поэтому, при создании решения, следует выбрать “Библиотека классов”, далее создаем проект, подключаем к нему NUnit при помощи пакетов NuGet



Выберите “Обзор” и введите в строке поиска NUnit:

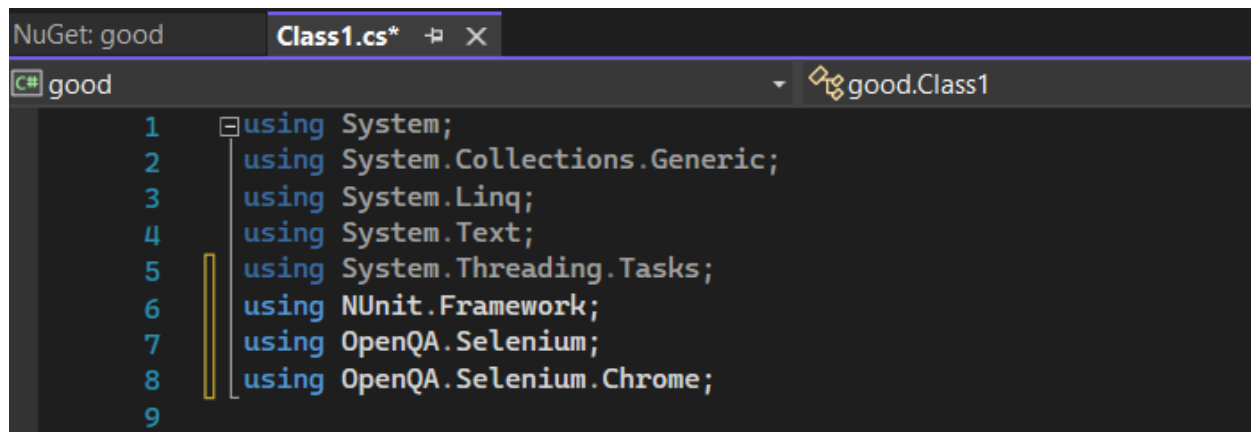


Затем, введите в строке поиска “Selenium” и установите Selenium.WebDriver:



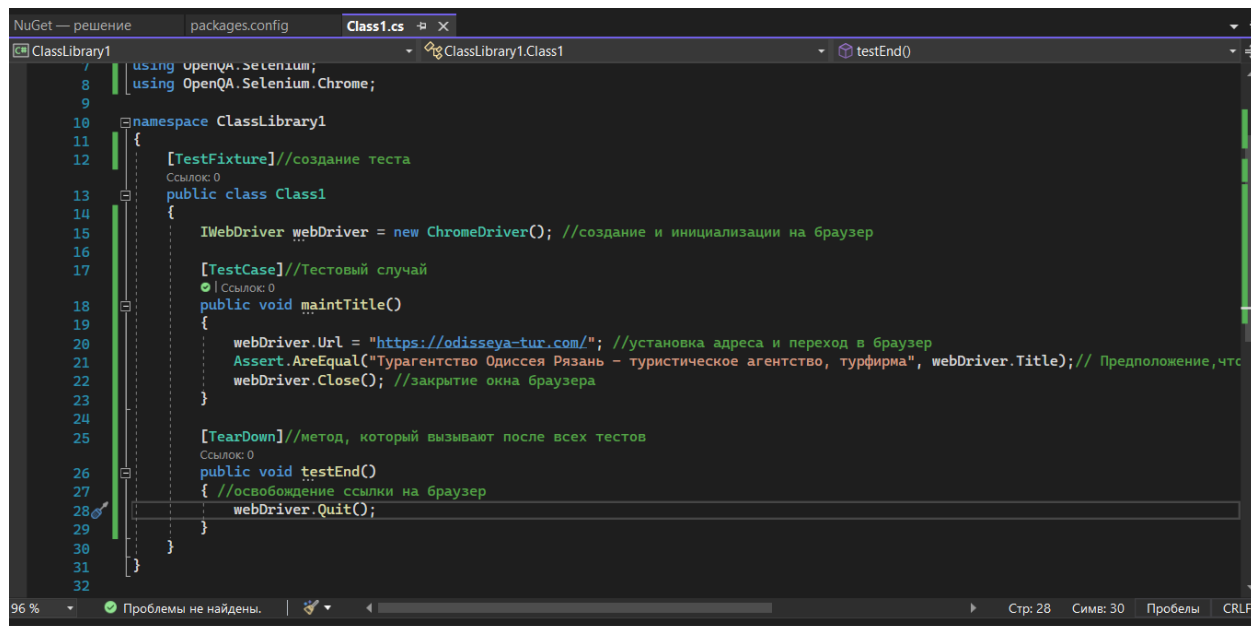
Подключение библиотек и написание простого теста:

Для того, чтобы получить доступ к функциям тестирования, подключите следующие пространства имён:



```
NuGet: good | Class1.cs* | X
C# | good | good.Class1
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6  using NUnit.Framework;
7  using OpenQA.Selenium;
8  using OpenQA.Selenium.Chrome;
9
```

В качестве простого теста, можно использовать проверку заголовка веб страницы <https://odissey-tur.com/>:



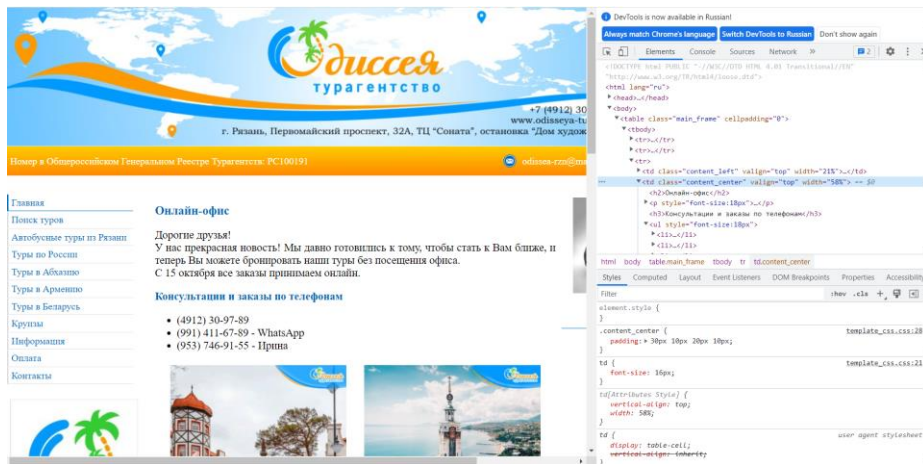
```
NuGet — решение | packages.config | Class1.cs | X
ClassLibrary1 | ClassLibrary1.Class1 | testEnd()
7  using OpenQA.Selenium;
8  using OpenQA.Selenium.Chrome;
9
10 namespace ClassLibrary1
11 {
12     [TestFixture]//создание теста
13     public class Class1
14     {
15         IWebDriver webDriver = new ChromeDriver(); //создание и инициализации на браузер
16
17         [TestCase]//Тестовый случай
18         public void maintTitle()
19         {
20             webDriver.Url = "https://odissey-tur.com/"; //установка адреса и переход в браузер
21             Assert.AreEqual("Турагентство Одиссея Рязань – туристическое агентство, турфирма", webDriver.Title); // Предположение, что
22             webDriver.Close(); //закрытие окна браузера
23         }
24
25         [TearDown]//метод, который вызывают после всех тестов
26         public void testEnd()
27         {
28             //освобождение ссылки на браузер
29             webDriver.Quit();
30         }
31     }
32 }
```

После сборки решения и запуска теста, будет открыто окно браузера Google Chrome, в нём будет открыт сайт “<https://odissey-tur.com/>”, будет выполнена проверка заголовка страницы, после чего, окно браузера будет закрыто.

Тест прошёл успешно

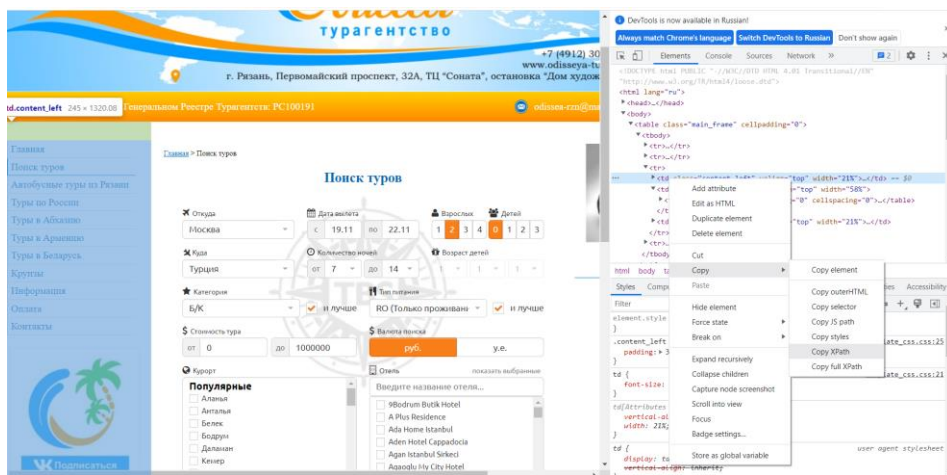
## 1.Получение доступа к веб элементам:

Для того, чтобы получить доступ к веб элементу страницы, удобнее всего использовать XPath. Для того, чтобы получить XPath объекта, кликните по нему правой кнопкой мыши и нажмите “Просмотреть исходный код”:

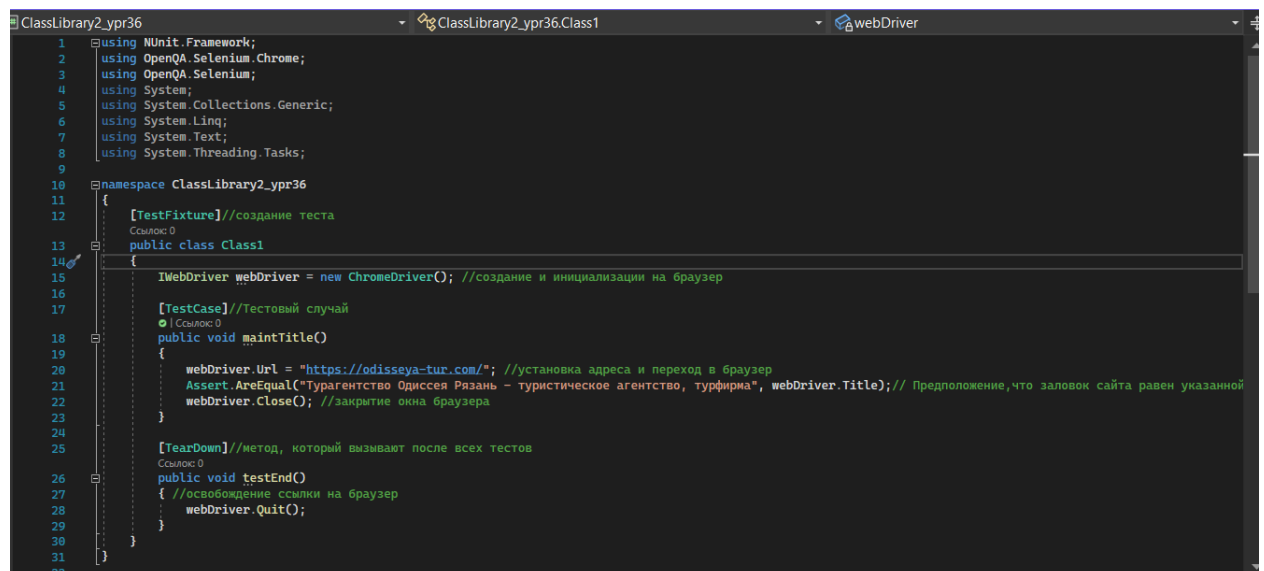


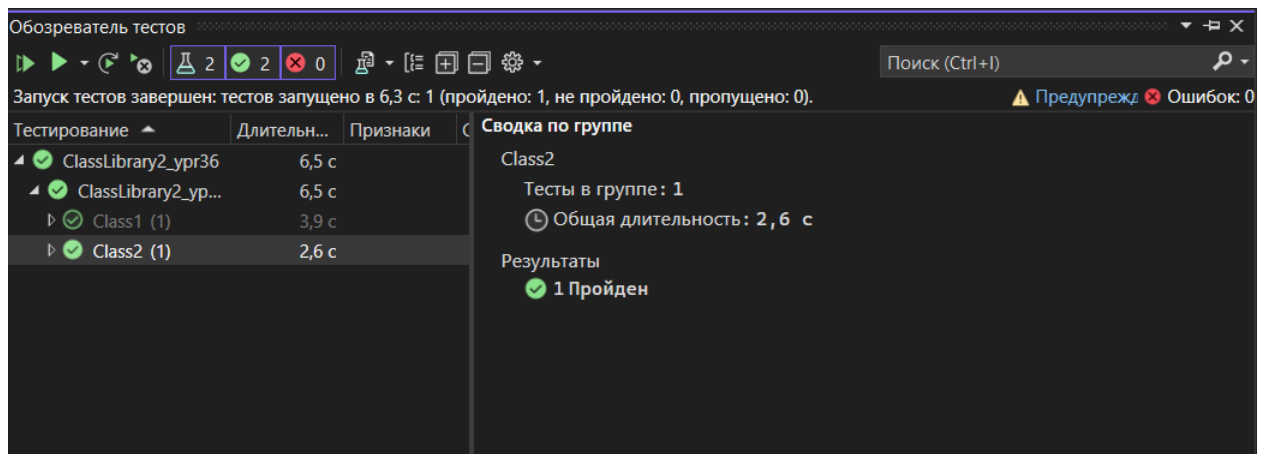
В появившемся окне, нажмите правой кнопкой мыши на нужный элемент → Сору->Сору

XPath:

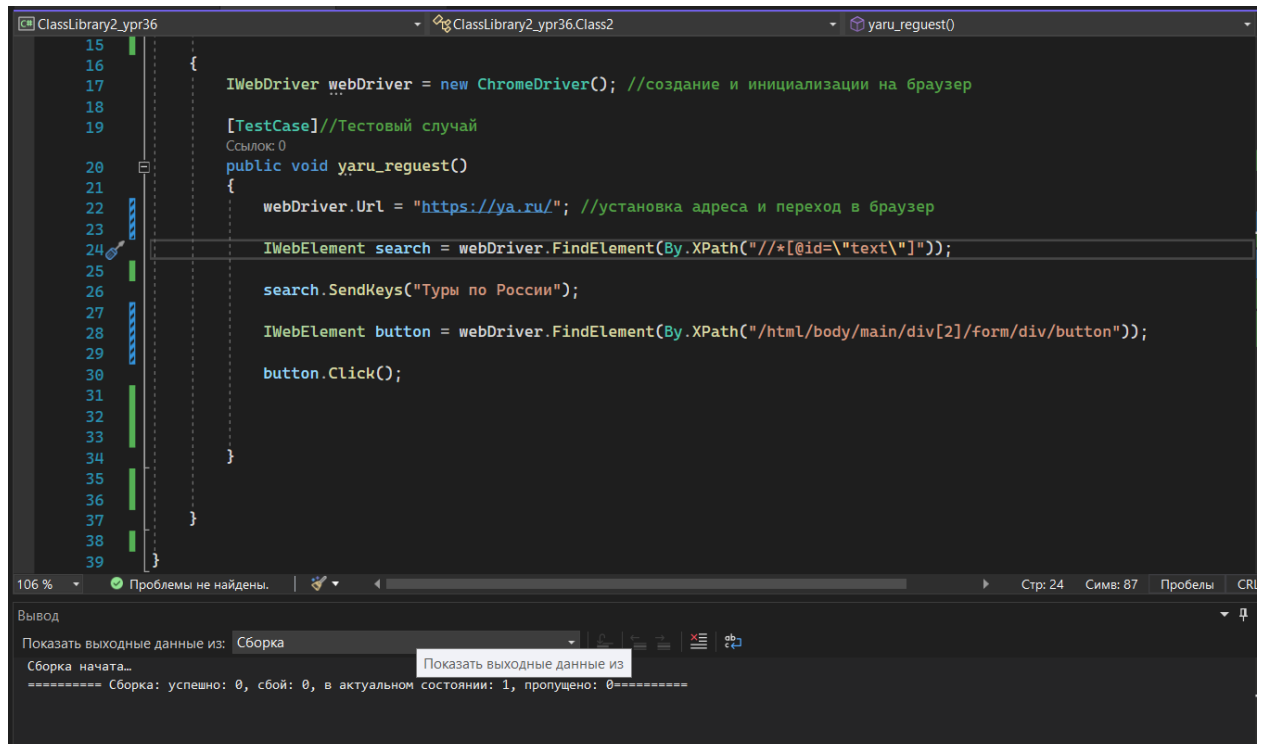


В исходном коде:

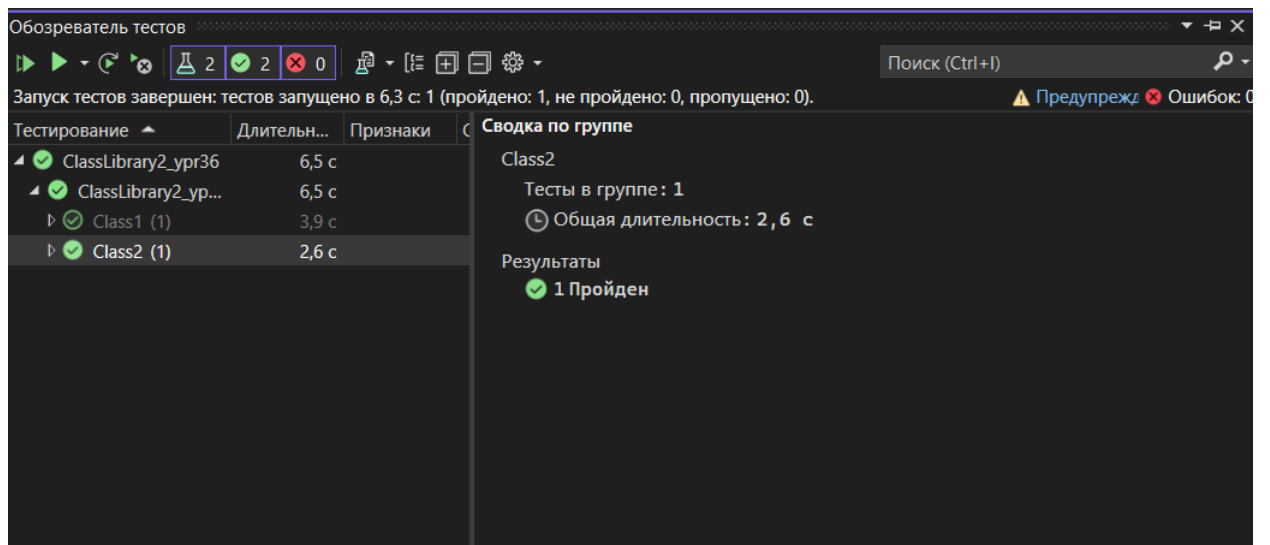




## 2. Заполнение текстового поля. и Эмуляция нажатия на кнопку.



## Тест



## 3.Переход по ссылке.

```
10 namespace ClassLibrary2_ypr36
11 {
12     [TestFixture]//создание теста
13     Ссылка: 0
14     public class Class3
15     {
16         IWebDriver webDriver = new ChromeDriver(); //создание и инициализации на браузер
17
18         [TestCase]//Тестовый случай
19         | Ссылка: 0
20         public void yaru_request()
21         {
22             webDriver.Url = "https://ya.ru/"; //установка адреса и переход в браузер
23
24             IWebElement search = webDriver.FindElement(By.XPath("//*[@id=\"text\"]"));
25
26             search.SendKeys("Турм по России");
27
28             IWebElement button = webDriver.FindElement(By.XPath("/html/body/main/div[2]/form/div/button"));
29
30             button.Click();
31
32             IWebElement ccl = webDriver.FindElement(By.XPath("//*[@id=\"search-result\"]/li[1]/div/div[1]/div[2]/div/a"));
33
34             ccl.Click();
35         }
36     }
37 }
38
39 87 % Проблемы не найдены. | Стр: 25 Симв: 13 Пробелы CRLF
Вывод
Показать выходные данные из: Сборка
Сборка начата...
----- Сборка: успешно: 0, сбой: 0, в актуальном состоянии: 1, пропущено: 0-----
```

Тест верный

Обозреватель тестов

Запуск тестов завершен: тестов запущено в 9,7 с: 1 (пройдено: 1, не пройдено: 0, пропущено: 0).

Тестирование	Длительн...	Признаки
ClassLibrary2_ypr36	12 с	
ClassLibrary2_ypr...	12 с	
Class1 (1)	3,9 с	
Class2 (1)	2,6 с	
Class3 (1)	5,5 с	

Сводка по группе

Class3

Тесты в группе: 1

Общая длительность: 5,5 с

Результаты

1 Пройден

#### 4. Проверка видимости объектов

```
10 namespace ClassLibrary2_ypr36
11 {
12     [TestFixture]//создание теста
13     Ссылка: 0
14     public class Class4
15     {
16         IWebDriver webDriver = new ChromeDriver(); //создание и инициализации на браузер
17
18         [TestCase]//Тестовый случай
19         | Ссылка: 0
20         public void yaru_request()
21         {
22             webDriver.Url = "https://ya.ru/"; //установка адреса и переход в браузер
23
24             IWebElement search = webDriver.FindElement(By.XPath("//*[@id=\"text\"]"));
25
26             search.SendKeys("Турм по России");
27
28             IWebElement button = webDriver.FindElement(By.XPath("/html/body/main/div[2]/form/div/button"));
29             Console.WriteLine(button.Displayed);
30             Console.WriteLine(button.Text);
31             Console.WriteLine(button.Selected);
32         }
33     }
34 }
35
36 96 % Проблемы не найдены. | Стр: 32 Симв: 10 Пробелы CRLF
Обозреватель решений
Обозреватель решений — поиск (Ctrl+I)
Решение "ClassLibrary2_ypr36" (1 проекта 1)
+ ClassLibrary2_ypr36
+ Properties
+ Сошли
+ Class1.cs
+ Class2.cs
+ Class3.cs
+ Class4.cs
+ packages.config
Обозреватель реше... Изменения Git — ур... Team Explorer — Гла...
```

Тесты

Произошла ошибка, и анализатор "CSharpWrappingCodeRefactoringProvider" отключен.Показать трассировку стекаВключитьВключить

Обозреватель тестов

Поиск (Ctrl+I)

Запуск тестов завершен: тестов запущено в 8,1 с: 1 (пройдено: 1, не пройдено: 0, пропущено: 0).1 предупреждОшибок: 0

Тестирование	Длительн...	Признаки
ClassLibrary2_ypr36	16 с	
ClassLibrary2_ypr...	16 с	
Class1 (1)	3,9 с	
Class2 (1)	2,6 с	
Class3 (1)	5,5 с	
yaru_request()	5,5 с	
Class4 (1)	4 с	
yaru_request()	4 с	

Подобная сводка по тесту

yaru\_request()

Источник: [Class4.cs](#) строка 19

Длительность: 4 с

Стандартный вывод:

True

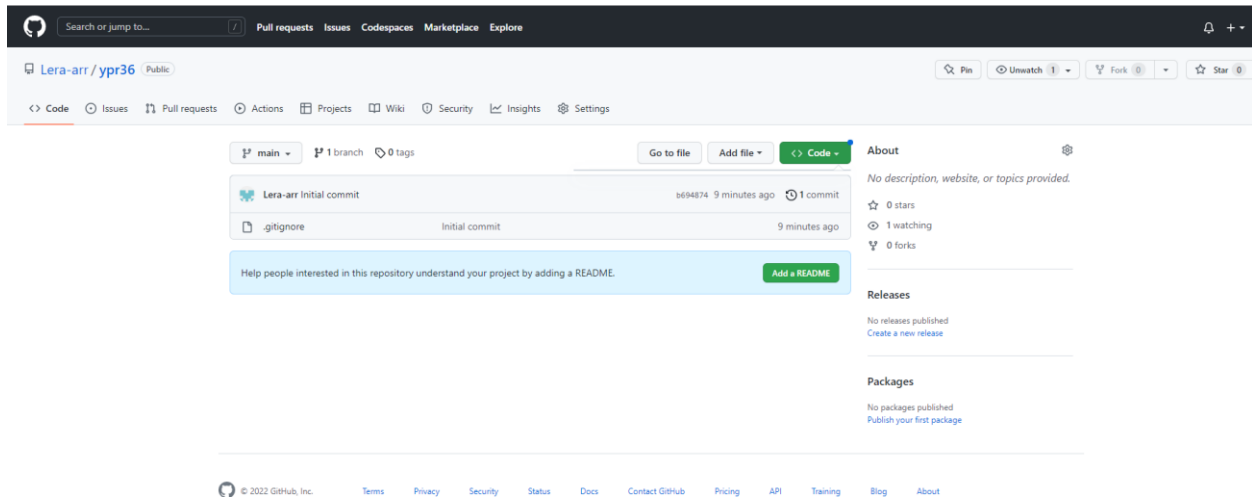
Найти

False

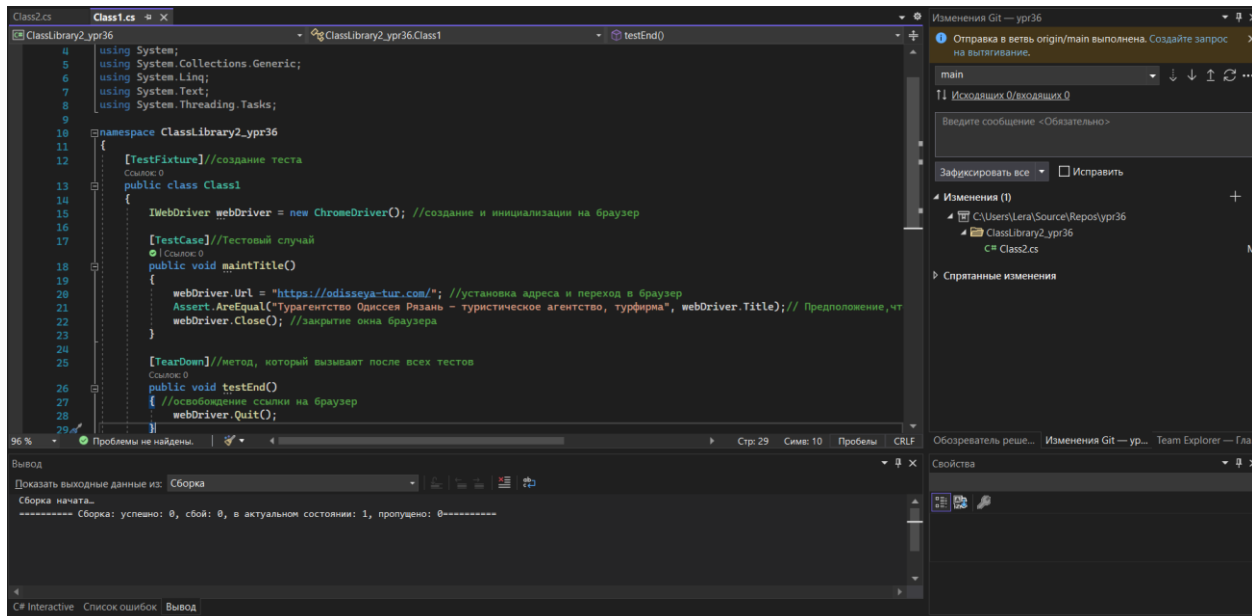
Панель элементовИсточники данныхОбозреватель тестов



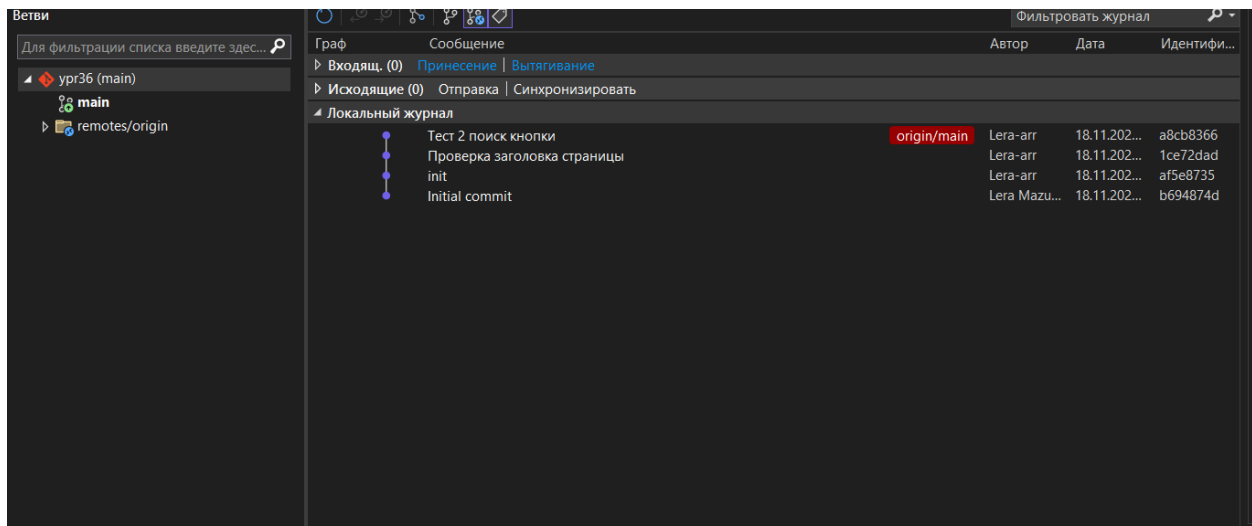
## 4. Подключаем к репозиторий



## Создаем класс библиотек



## Далее отправляем на репозиторий



Все тесты были отправлены на репозиторий

Class4.csClass3.csРепозиторий Git: ypr36Class2.csClass1.cs

Отправка в ветвь origin/main выполнена. Создайте запрос на вытягивание.

Ветви

Для фильтрации списка введите здесь...

ypr36 (main)  
main  
remotes/origin

Граф

Сообщение

Входящ. (0) [Принесение](#) | [Вытягивание](#)

Исходящие (0) [Отправка](#) | [Синхронизировать](#)

Локальный журнал

test 4

Тест 3

Тест 2 поиск кнопки

Проверка заголовка страницы

init

Initial commit

origin/main

Lera-arr

18.11.202...

4d8fe964

Lera-arr

18.11.202...

8cf3b731

Lera-arr

18.11.202...

a8cb8366

Lera-arr

18.11.202...

1ce72dad

Lera-arr

18.11.202...

af5e8735

Lera Mazu...

18.11.202...

b694874d

Lera-arr / ypr36Public

<> CodeIssuesPull requestsActionsProjectsWikiSecurityInsightsSettings

mainypr36 / ClassLibrary2\_ypr36 /

Go to fileAdd file...

Lera-arr тест 44d8fe96 2 minutes agoHistory

..

Properties

init

4 hours ago

drivers

Тест 3

34 minutes ago

Class1.cs

Проверка заголовка страницы

4 hours ago

Class2.cs

Тест 3

34 minutes ago

Class3.cs

тест 4

2 minutes ago

Class4.cs

тест 4

2 minutes ago

ClassLibrary2\_ypr36.csproj

тест 4

2 minutes ago

ClassLibrary2\_ypr36.sln

init

4 hours ago

packages.config

Проверка заголовка страницы

4 hours ago