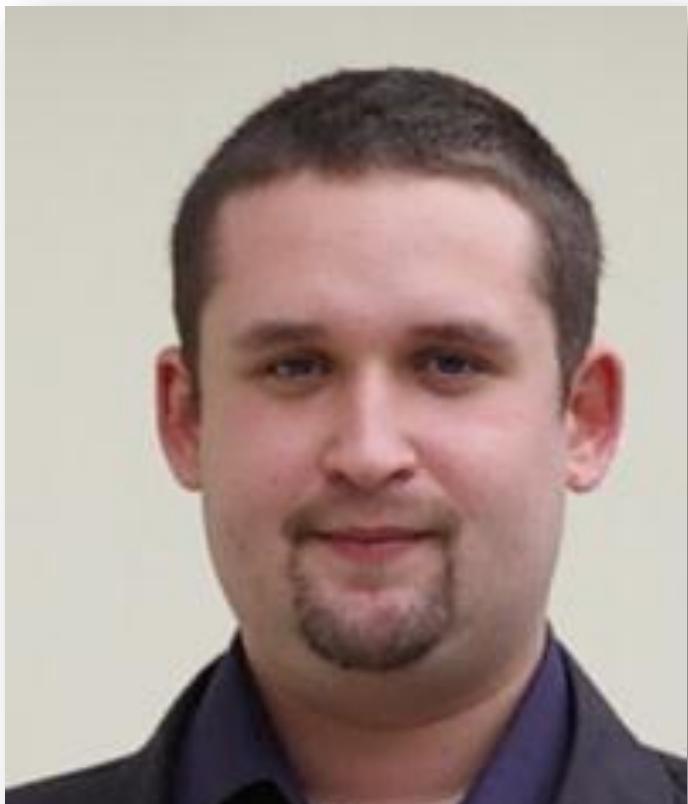


За пределами PageObject

Дмитрий Жарий



Давайте познакомимся!



Дима Жарий

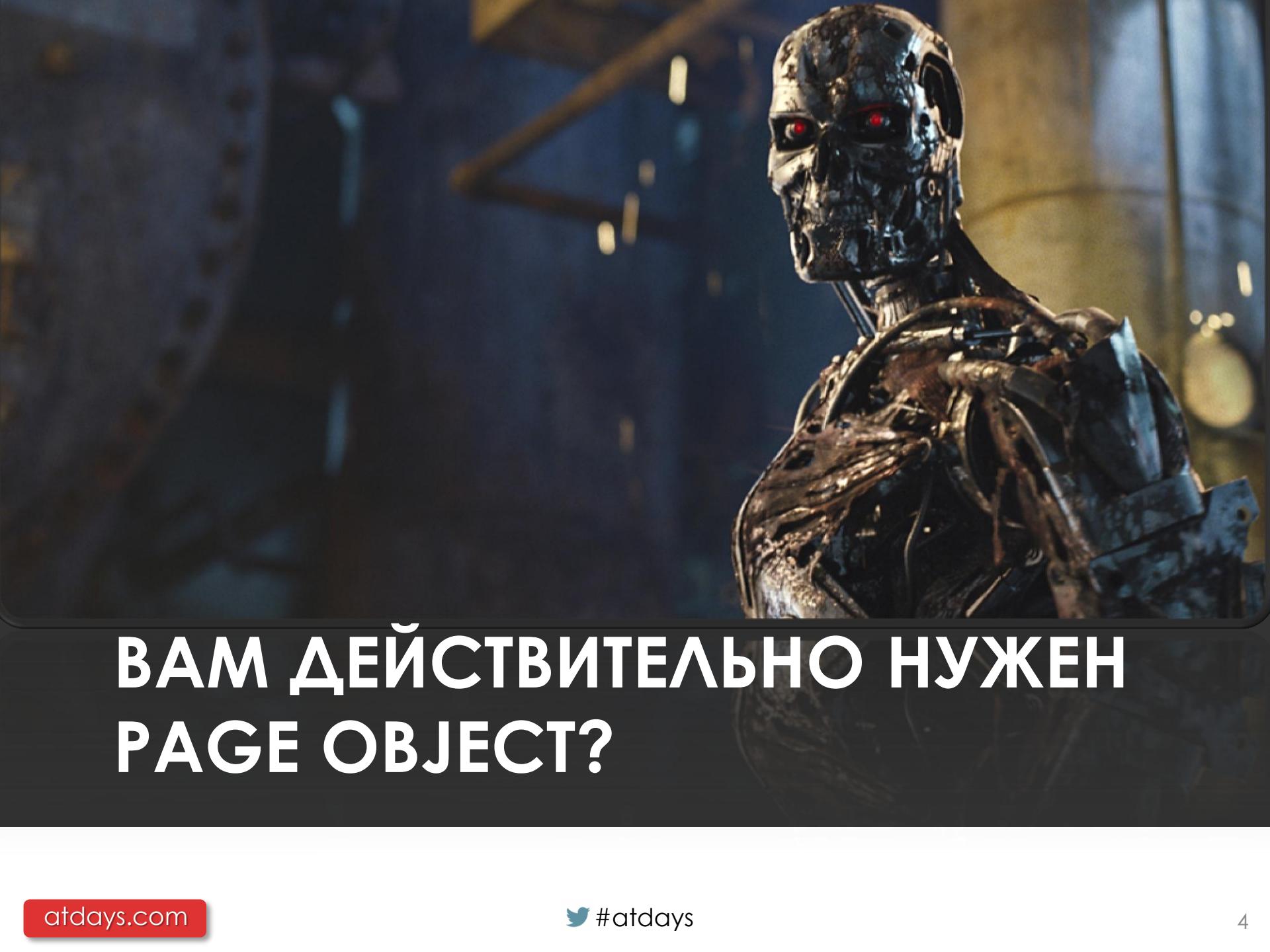


Аджендиум...

[OK]

1. Вам не нужен PageObject
2. Вам нужен PageObject
3. PageObject – это паттерн, шаблон... ИДЕЯ!
4. Статический PageObject
5. Динамический PageObject
6. За пределами:
 - Паттерн «Цепочка ответственности» и .Invoke()
 - Интерфейс IHaveExpectedControls
 - Наследование в тестах



A close-up, low-angle shot of the T-800 model from the Terminator franchise. The robot's head is metallic and heavily damaged, with visible internal components and a skeletal structure. Its red, glowing eyes are prominent. It appears to be in a dark, industrial or post-apocalyptic setting, with pipes and structural elements visible in the background.

ВАМ ДЕЙСТВИТЕЛЬНО НУЖЕН
PAGE OBJECT?

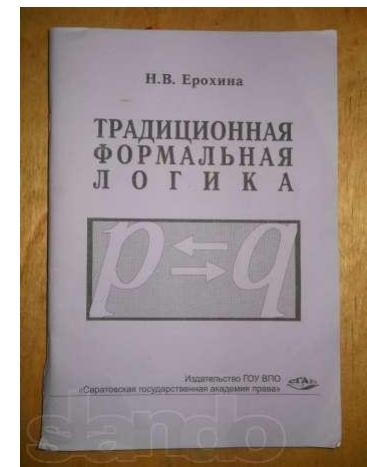
Нет. Если у вас парочка несложных тестов

```
public void Wikipedia_Smart_Search_Test()
{
    RemoteWebDriver driver = new InternetExplorerDriver();

    driver.Navigate()
        .GoToUrl("http://en.wikipedia.org/wiki/Main_Page");

    driver.FindElementByCssSelector(@"div#simpleSearch")
        .SendKeys("Webdriver Selenium");

    Assert.AreEqual("http://en.wiki {...} Driver"
        , driver.Url);
}
```

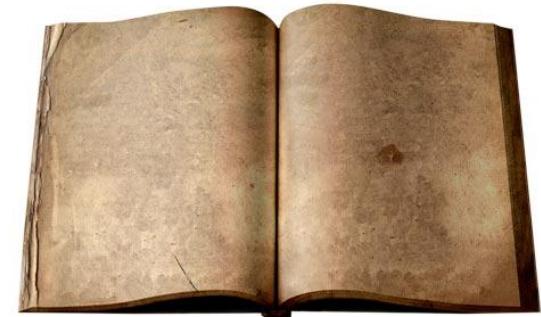


Нет. Если у вас несколько сложных тестов

```
public void TestMethod2()
{
    RemoteWebDriver driver = new InternetExplorerDriver();
    driver.Navigate().GoToUrl("http://localhost:8085/");
    Thread.Sleep(2000);
    driver.Manage().Cookies.DeleteAllCookies();
    Thread.Sleep(2000);
    driver.FindElement(By.XPath("//a[text()='Log On']")).Click();
    Thread.Sleep(2000);
    Assert.That(driver.Title, Is.EqualTo("Task Tracker - Log On"));
    driver.FindElement(By.Id("UserName")).SendKeys("testuser");
    driver.FindElement(By.Id("Password")).SendKeys("password");
    driver.FindElement(By.Id("LogOn")).Click();
    Thread.Sleep(2000);

    Assert.That(driver.Title, Is.EqualTo("Task Tracker - Home"));
    Assert.Throws<NoSuchElementException>(
        () => driver.FindElement(By.XPath("//a[text()='Log On']")));

    driver.FindElement(By.XPath("//a[text()='Log Off']")).Click();
    Thread.Sleep(2000);
    Assert.That(driver.Title, Is.EqualTo("Task Tracker - Home"));
    Assert.Throws<NoSuchElementException>(
        () => driver.FindElement(By.XPath("//a[text()='Log Off']")));
    Assert.Throws<NoSuchElementException>(
        () => driver.FindElement(By.XPath("//span[@welcome=''][text()='Welcome, test user!']")));
}
```



Ведь код можно улучшить

```
public void HomeIndex_LogIn_LogOut()
{
    RemoteWebDriver driver = new InternetExplorerDriver();
    // Navigate to the base url.
    driver.Navigate().GoToUrl("http://localhost:8085/");
    Wait();

    // Delete all cookies on the profile.
    driver.Manage().Cookies.DeleteAllCookies();
    Wait();

    // Click on the Log On link.
    IWebElement logOnLink = driver.FindElement(By.XPath("//a[text()='Log On']"));
    logOnLink.Click();
    Wait();

    // Check that the page title is that of the User/Logon page.
    Assert.That(driver.Title, Is.EqualTo("Task Tracker - Log On"));

    // Enter the username and password.
    IWebElement userName = driver.FindElement(By.Id("UserName"));
    userName.SendKeys("testuser");
    // ...
}
```

Как улучшить?

- ✓ Вынести часто используемый функционал в **общедоступные** методы
- ✓ Отформатировать код
- ✓ Добавить комментарии

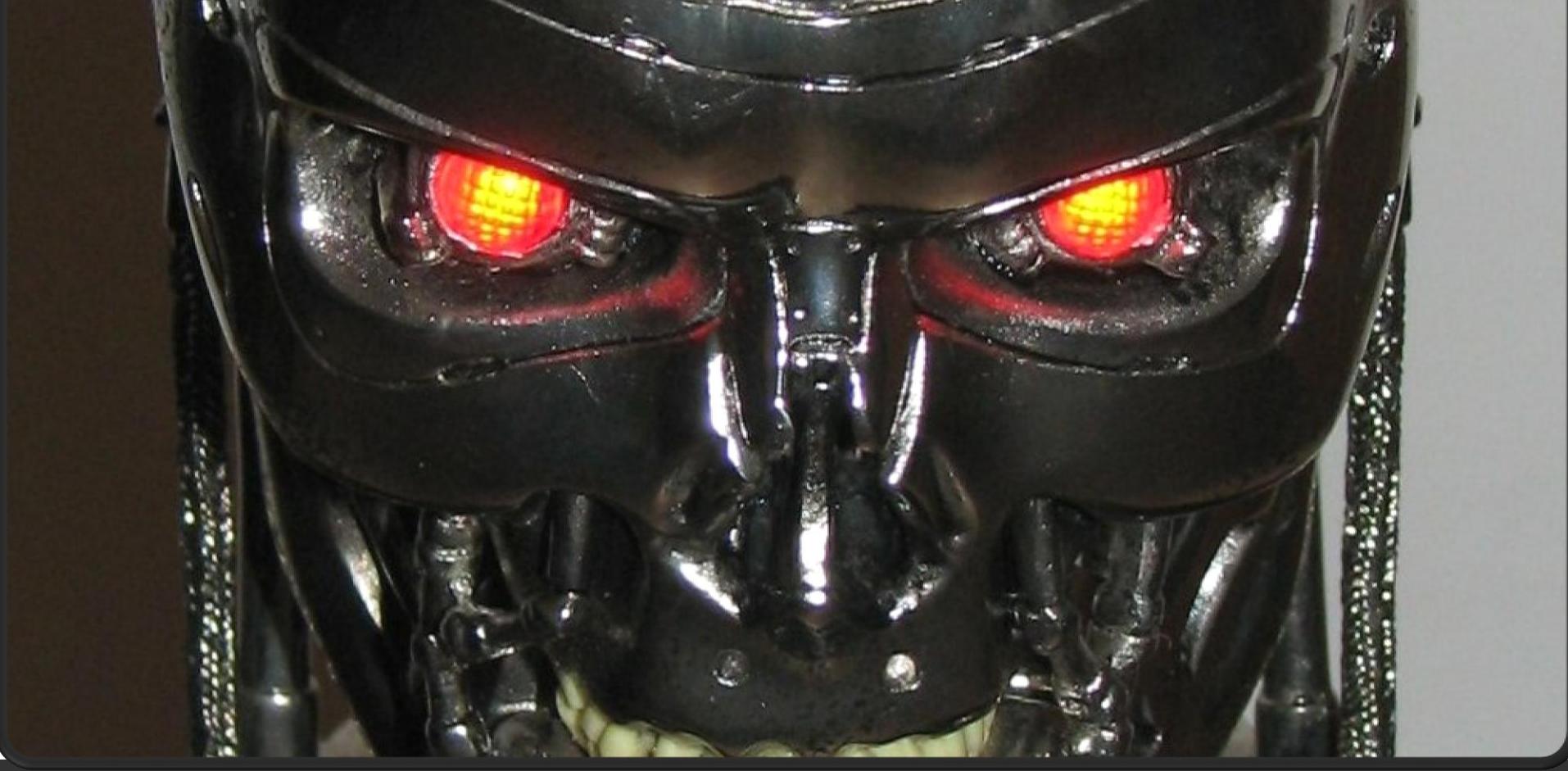
```
driver.FindElement(By.XPath("//a[text()='Log On']"))  
.Click();
```



```
// Click on the Log On link.  
var logOnLink = driver.FindElement(  
    By.XPath("//a[text()='Log On']"));  
logOnLink.Click();
```

И, если у вас все под
контролем...





БАМ НЕ НУЖЕН PAGE OBJECT!



```
var ddlMonthSelect =  
new  
SelectElement(driver.Fin  
dElementByName(@"EXPIRYD  
ATE_MM"));  
    var  
ddlYearSelect = new  
SelectElement(driver.Fin  
dElementByName(@"EXPIRYD  
ATE_YY"));  
    var  
txtSecurityCode =  
driver.FindElementByName  
(@"CVV");  
    var  
btnContinue =  
driver.FindElementById("br  
nSubmit");  
  
ddlMonthSelect.SelectByT  
ext("05");  
  
ddlYearSelect.SelectByTe  
xt("15");
```

```
RemoteWebDriver driver = new InternetExplorerDriver();  
// Navigate to the base url.  
driver.Navigate().GoToUrl(@"http://en.wikipedia.org");  
|  
// Main Page  
var lnkDonate = driver.FindElementByCssSelector(@"a[title='Support us ']");  
lnkDonate.Click();  
  
// Donate Page  
var rbtnDonate50 = driver.FindElementByCssSelector(@"input[name='amount'][value='50']");  
rbtnDonate50.Click();  
  
var btnMakeDonation = driver.FindElementByCssSelector(@"input[value='Donate by credit/debit card']");  
btnMakeDonation.Click();  
  
// Donation Payments  
var txtFirstName = driver.FindElementByName("fname");  
var txtLastName = driver.FindElementByName("lname");  
var txtStreet = driver.FindElementByName("street");  
var txtPostalCode = driver.FindElementByName("zip");  
var txtCity = driver.FindElementByName("city");  
var txtEmailAddress = driver.FindElementByName("emailAdd");  
var rbtnCardVisa = driver.FindElementByCssSelector(@"input[name='cardtype'][value='visa']");  
  
txtFirstName.Clear();  
txtFirstName.SendKeys("John")  
  
txtLastName.Clear();  
txtLastName.SendKeys("Doe")  
  
txtStreet.Clear();  
txtStreet.SendKeys("123 Main Street")  
  
txtPostalCode.Clear();  
txtPostalCode.SendKeys("12345")  
  
ddlMonthSelect.SelectByText("05");  
ddlYearSelect.SelectByText("15");  
txtSecurityCode.SendKeys("555");  
  
Assert.IsTrue(btnContinue.Displayed, "There is a Continue button");  
  
txtEmailAddress.Clear();  
txtEmailAddress.SendKeys("johndoe@example.com");  
driver.SwitchTo().DefaultContent();  
  
rbtnCardVisa.Click();  
var wait = new WebDriverWait(driver, TimeSpan.FromSeconds(30));  
var frameCardNumber=wait.Until(ExpectedConditions.ElementIsVisible(By.CssSelector(@"iframe[src]))  
driver.SwitchTo().Frame(frameCardNumber);  
var txtName = wait.Until(ExpectedConditions.ElementIsVisible(By.Name("CREDITCARDNUMBER")));  
txtName.Clear();  
txtName.SendKeys("4556767453966453");  
  
var ddlMonthSelect = new SelectElement(driver.FindElementByName(@"EXPIRYDATE_MM"));  
var ddlYearSelect = new SelectElement(driver.FindElementByName(@"EXPIRYDATE_YY"));  
var txtSecurityCode = driver.FindElementByName(@"CVV");  
  
var txtSecurityCode = driver.FindElementByName(@"CVV");  
var btnContinue = driver.FindElementById("btnSubmit");  
  
Assert.AreEqual("Donate-error - Payments", driver.Title);  
var lblFirstHeader = driver.FindElementById(@"firstHeading");  
Assert.AreEqual("Donate-error", lblFirstHeader.Text);
```

Прости, ня! 😞

Когда приходит Хаос...

- ✓ "Тест-простыней" становится **СЛИШКОМ МНОГО**
- ✓ В коде не разобраться без пол-литру
- ✓ На **поддержку** уходит уж **слишком много времени**
- ✓ Легче **всё переписать** заново
- ✓ Начинаете **ненавидеть разработчиков**, которые вынесли div из span

PageObject – он как книжная полка

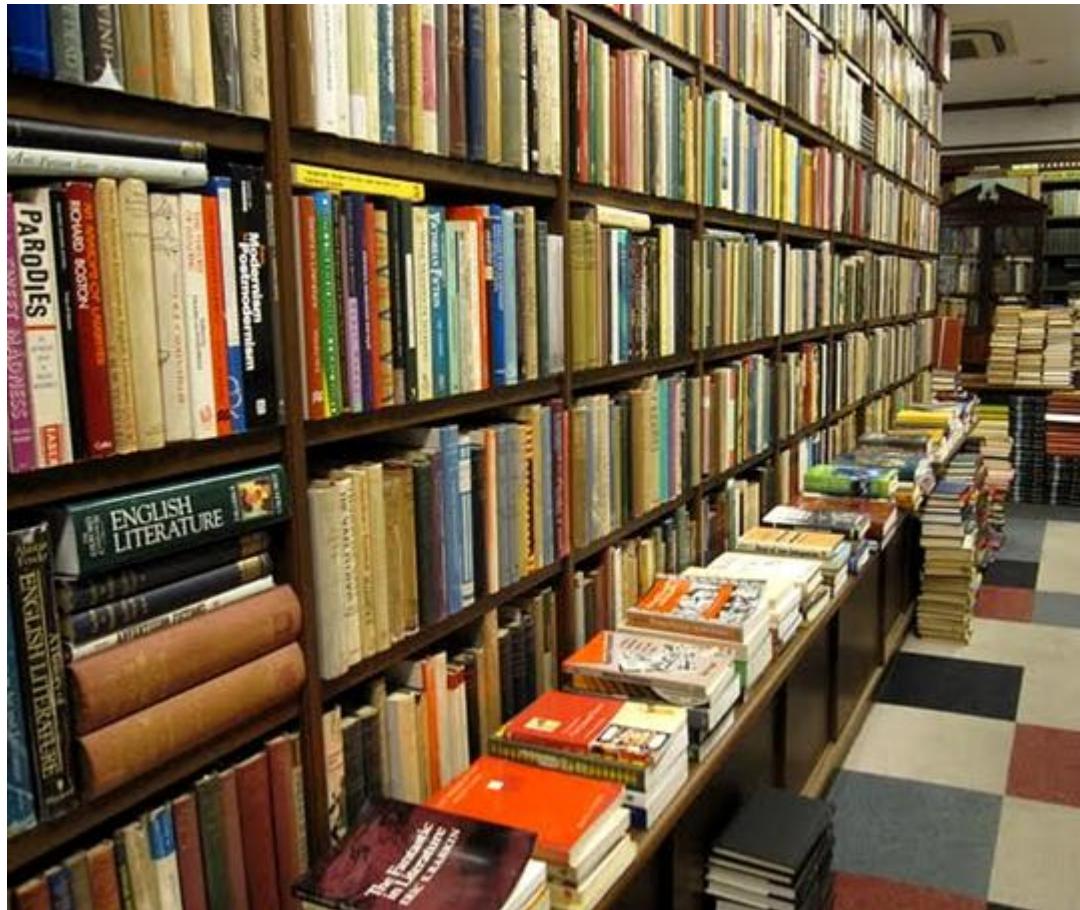
Pages

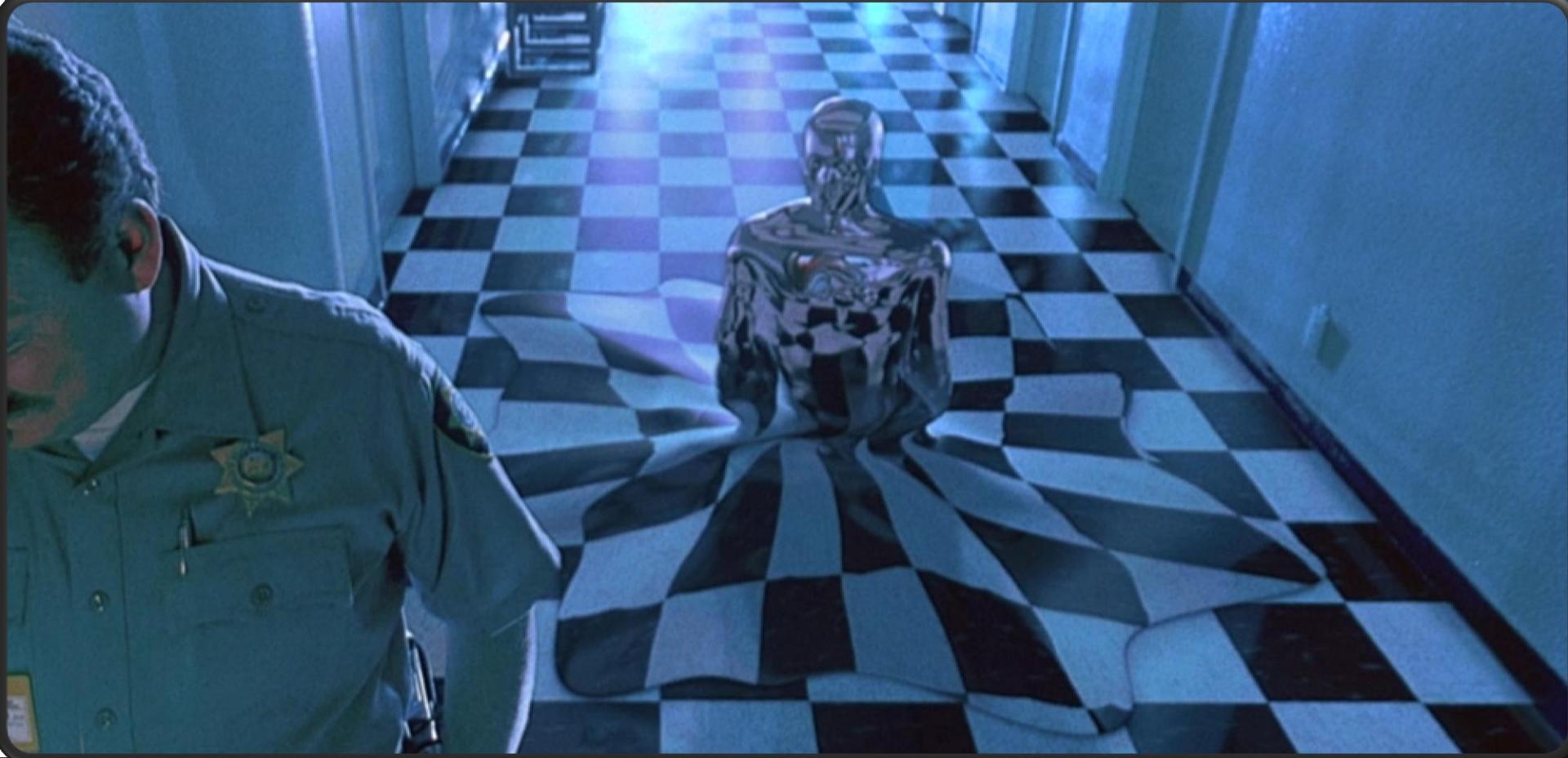
LoginPage

→ **Login(name, passwd)**

MainPage

→ **LogOut()**
→ **GotoProjects()**
→ **GotoUserProfile()**
→ **Search(text)**





PAGE ОБЈЕСТ – ОН ГИБКИЙ

PageObject – это:



**Концентрация
на языке задачи,
а не на языке решения**

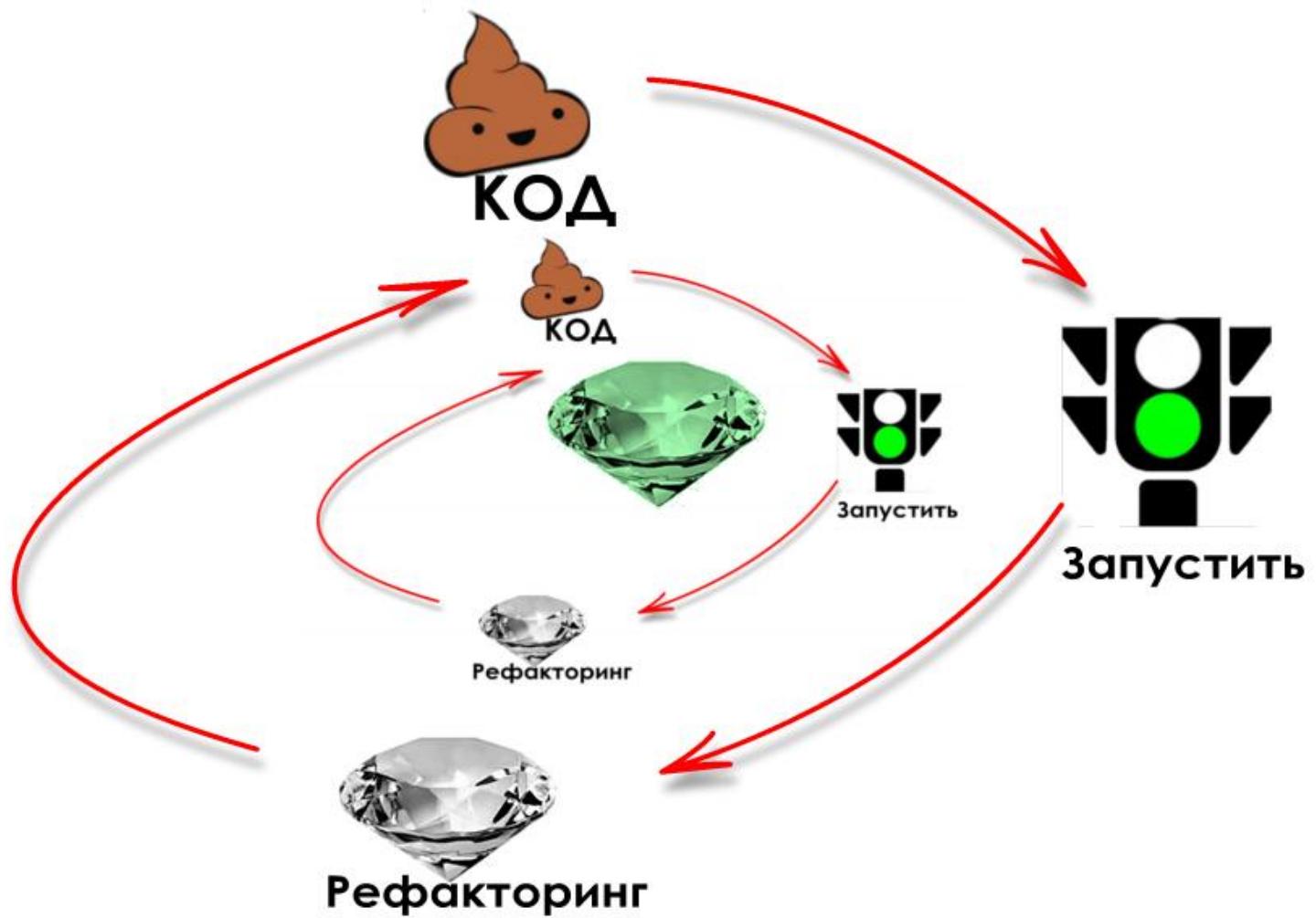
PageObject – главная цель

- ✓ Обеспечить **хранение локаторов** в отдельном классе
- ✓ Обеспечить **повторное использование** локаторов и/или действий над страницей без дублирования кода
- ✓ Обеспечить **слой абстракции** от «драйвера» так, чтобы в тестах не использовались физические элементы идентификации элементов управления приложением

А ВЫГЛЯДИТ ВСЁ ВОТ ТАК



Демо!





ЧТО ЭТО БЫЛО? –
СТАТИЧЕСКИЙ PAGE OBJECT!

Позитив 01: Тест стал читабелен

```
public static RemoteWebDriver Driver { get { return WebBrowser.Driver; } }

[TestMethod]
public void Donate_test_static()
{
    MyPages.MainPage.Open();
    MyPages.MainPage.GoToDonatePage();
    MyPages.DonatePage.Donate_50_UAH_Using_Debit_Card();
    MyPages.DonationPaymentsForm.FillDonationForm
    (
        firstName : "Vasya",
        lastName : "Pupkin",
        securityCode : "555"
    );
    // CUT
}
```

Позитив 2: Удобная «точка входа» в страницы

```
19 [TestMethod]
20 public void Donate_test_static()
21 {
22
23     MyPages.
24     MyPages.    □ DonatePage
25     MyPages.    □ DonationPaymentsForm
26     MyPages.    □
27     MyPages.    □
28     MyPages.    □
29     MyPages.    □
30     MyPages.    □
31     MyPages.    □
32
33     namespace DemoTest.StaticPageobject
34     {
35         public static class MyPages
36         {
37             public static RemoteWebDriver Driver
38             {
39                 get { return WebBrowser.Driver; }
40             }
41
42             public static class MainPage... □
43             public static class DonatePage... □
44             public static class DonationPaymentsForm... □
45             public static class PaymentResultPage... □
46         }
47     }
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100 }
```

Позитив 3: Driver стал доступен отовсюду!

```
public static class WebBrowser
{
    public static RemoteWebDriver _driver = null;

    public static RemoteWebDriver Driver
    {
        get
        {
            _driver = _driver ?? new InternetExplorerDriver();
            return _driver;
        }
    }
}
```

```
public static RemoteWebDriver Driver {
    get { return WebBrowser.Driver; }
}
```

И еще раз о позитиве

- ✓ Теперь **жизнь** браузера (**драйвера**) контролирует класс **WebBrowser**.
- ✓ **Текущий драйвер** доступен из любого участка кода
- ✓ Любая **страница доступна** из **MyPages**.*
- ✓ Наш тест теперь помещается на один экран монитора

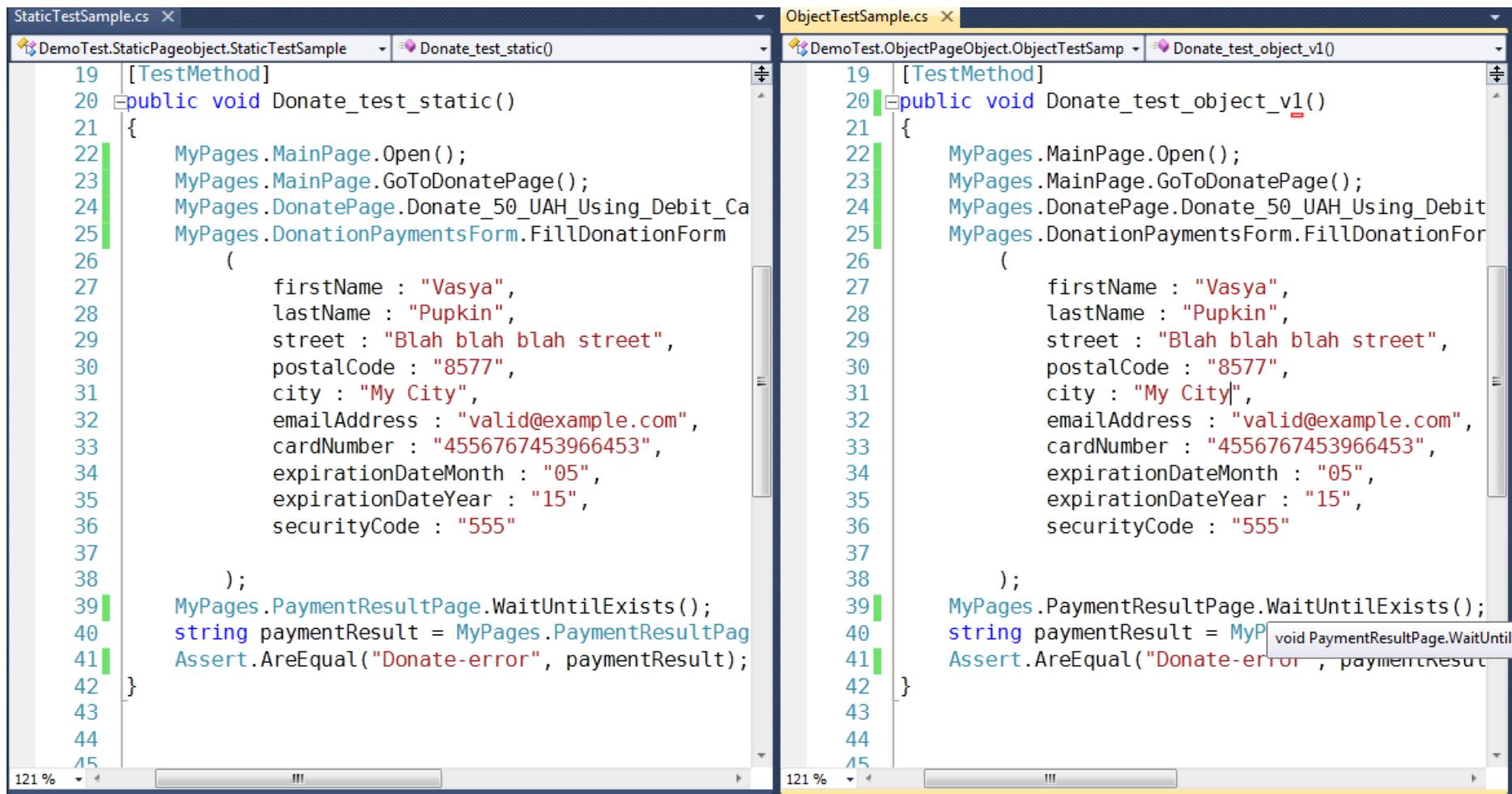
Ограничения статического класса в C#

	class	static class
Наследоваться	можно	нельзя
Породить много разных	можно	нельзя
Инициализировать параметрами через конструктор	можно	нельзя
Завалить	можно	полностью нельзя. Очень живучие
Присвоить переменной	можно	нельзя
Передать как параметр другому методу	можно	нельзя
Реализовать интерфейс	можно	нельзя
Использовать статические методы	можно	можно
Использовать статические поля	можно	можно
Инициализировать при помощи Webdriver PageFactory	можно	нельзя "из коробки", нужно написать свою фабрику



ГИБКОСТЬ – ЭТО ВАЖНО.
PAGE OBJECT КАК OBJECT

Тест не изменился! Спасибо, MyPages



The image shows two side-by-side code editors in a development environment, likely Visual Studio. Both editors display nearly identical C# code for testing a donation process on the MyPages platform.

Left Editor (StaticTestSample.cs):

```
19 [TestMethod]
20 public void Donate_test_static()
21 {
22     MyPages.MainPage.Open();
23     MyPages.MainPage.GoToDonatePage();
24     MyPages.DonatePage.Donate_50_UAH_Using_Debit_Ca
25     MyPages.DonationPaymentsForm.FillDonationForm
26     (
27         firstName : "Vasya",
28         lastName : "Pupkin",
29         street : "Blah blah blah street",
30         postalCode : "8577",
31         city : "My City",
32         emailAddress : "valid@example.com",
33         cardNumber : "4556767453966453",
34         expirationDateMonth : "05",
35         expirationDateYear : "15",
36         securityCode : "555"
37
38     );
39     MyPages.PaymentResultPage.WaitUntilExists();
40     string paymentResult = MyPages.PaymentResultPag
41     Assert.AreEqual("Donate-error", paymentResult);
42 }
```

Right Editor (ObjectTestSample.cs):

```
19 [TestMethod]
20 public void Donate_test_object_v1()
21 {
22     MyPages.MainPage.Open();
23     MyPages.MainPage.GoToDonatePage();
24     MyPages.DonatePage.Donate_50_UAH_Using_Debit_Ca
25     MyPages.DonationPaymentsForm.FillDonationForm
26     (
27         firstName : "Vasya",
28         lastName : "Pupkin",
29         street : "Blah blah blah street",
30         postalCode : "8577",
31         city : "My City",
32         emailAddress : "valid@example.com",
33         cardNumber : "4556767453966453",
34         expirationDateMonth : "05",
35         expirationDateYear : "15",
36         securityCode : "555"
37
38     );
39     MyPages.PaymentResultPage.WaitUntilExists();
40     string paymentResult = MyP void PaymentResultPage.WaitUntil
41     Assert.AreEqual("Donate-error", paymentResult);
42 }
```

The code in both files is identical, demonstrating a static test method for a static page and an object-oriented test method for the same scenario. The right file contains a partially visible line starting with 'void PaymentResultPage.WaitUntil'.

Изменения в MyPages

```
public static class MyPages
```

```
{
```

```
    public static MainPage MainPage
    {
        get { return new MainPage(); }
    }
```

```
    public static DonatePage DonatePage
```

```
{
```

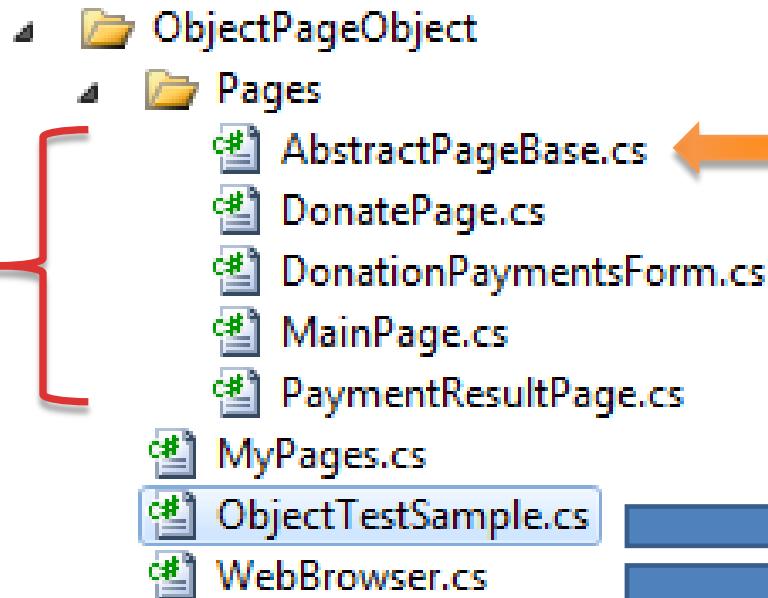
```
        get { return new DonatePage(); }
```

```
}
```

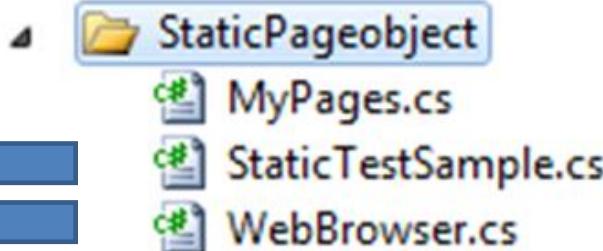
```
}
```

Новая структура проекта

✓ Object



✓ Static



AbstractPageBase

```
public abstract class AbstractPageBase
{
    public RemoteWebDriver Driver
    {
        get { return WebBrowser.Driver; }
    }
}
```

PaymentResultPage

```
public class PaymentResultPage : AbstractPageBase
{
    public void WaitUntilExists()
    {
        WebDriverWait wait = new WebDriverWait(Driver,
            TimeSpan.FromSeconds(30));
        wait.Until(ExpectedConditions.TitleContains(
            @"Donate-error - Payments"));
    }

    public string GetResultHeaderText()
    {
        var lblFirstHeader = Driver.
            FindElementById(@"firstHeading");
        return lblFirstHeader.Text;
    }
}
```

Changelog

- ✓ Появился **базовый класс** AbstractPageBase
- ✓ Из **MyPages.*** можно **по-прежнему** **получить любую страницу**
- ✓ Декларация страниц была вынесена из **MyPages** в отдельные файлы
- ✓ Мы лишились ограничений статических классов
- ✓ **ТЕПЕРЬ НАС НЕ ОСТАНОВИТЬ!**

Статический/Динамический – разница в перспективах



Статический



Динамический

T2 T-1000
www.toyhaven.blogspot.com



ПАТТЕРНЫ (РЕЦЕПТЫ) И ИНТЕРФЕЙСЫ (РОЛИ)

<http://bash.im/quote/420885>

xxx: тема письма в рабочей почте
" RE: FW: FW: RE: RE: FW: RE: СРОЧНО!!!!!"

yyy: Вот это проблема структурных организаций

ЦЕПОЧКА ОТВЕТСТВЕННОСТИ /CHAIN OF RESPONSIBILITY/

Chain of Responsibility



IInvokable

```
public interface IInvokable  
{  
    void Invoke();  
    bool Exists();  
}
```

Invoke – вызвать

Invokable – то, что можно вызвать

I invokable – Я вызываемый(-оя, -ое)!

НужнаяPage.Invoke():

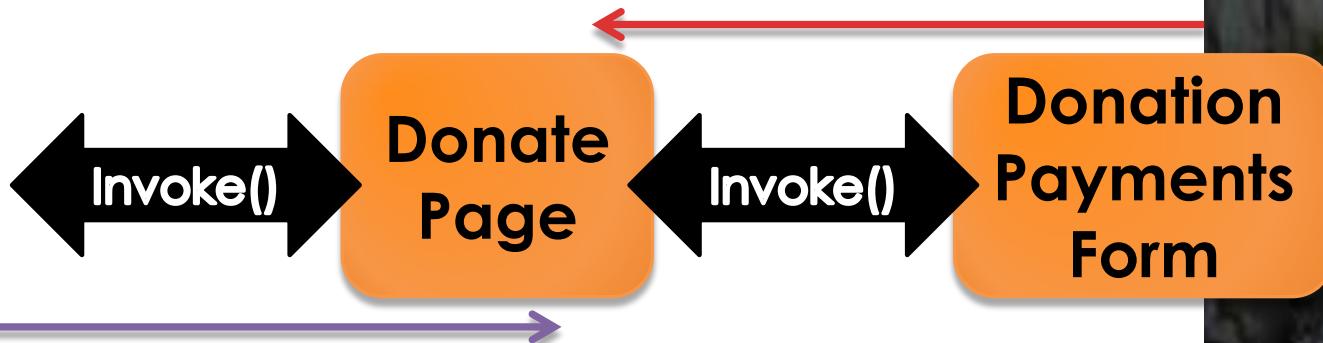
НужнаяPage.Invoke():

Если я уже **Exists()** – то вот она я!

Если я **не Exists()**, то я сделаю:

ПредыдущаяPage.Invoke()

Потом **что-то нажму** – вот она я!



Что дает нам `IInvokable`?

- ✓ Любую страницу можно вызвать с параметрами **по умолчанию**

```
var page = MyPages.  
    PaymentResultErrorInvalidCreditCard;  
  
page.Invoke();
```

Что дает нам IInvokable?

- ✓ Передать страницу как параметр метода

```
public void TestThatPageExists(IInvokable page)
{
    page.Invoke();
    Assert.IsTrue(page.Exists());
}
```

```
[TestMethod]
public void Test_PaymentResultErrorInvalidCreditCard()
{
```

```
    TestThatPageExists(
        MyPages.PaymentResultErrorInvalidCreditCard
    );
}
```

Что дает нам `IInvokable`?

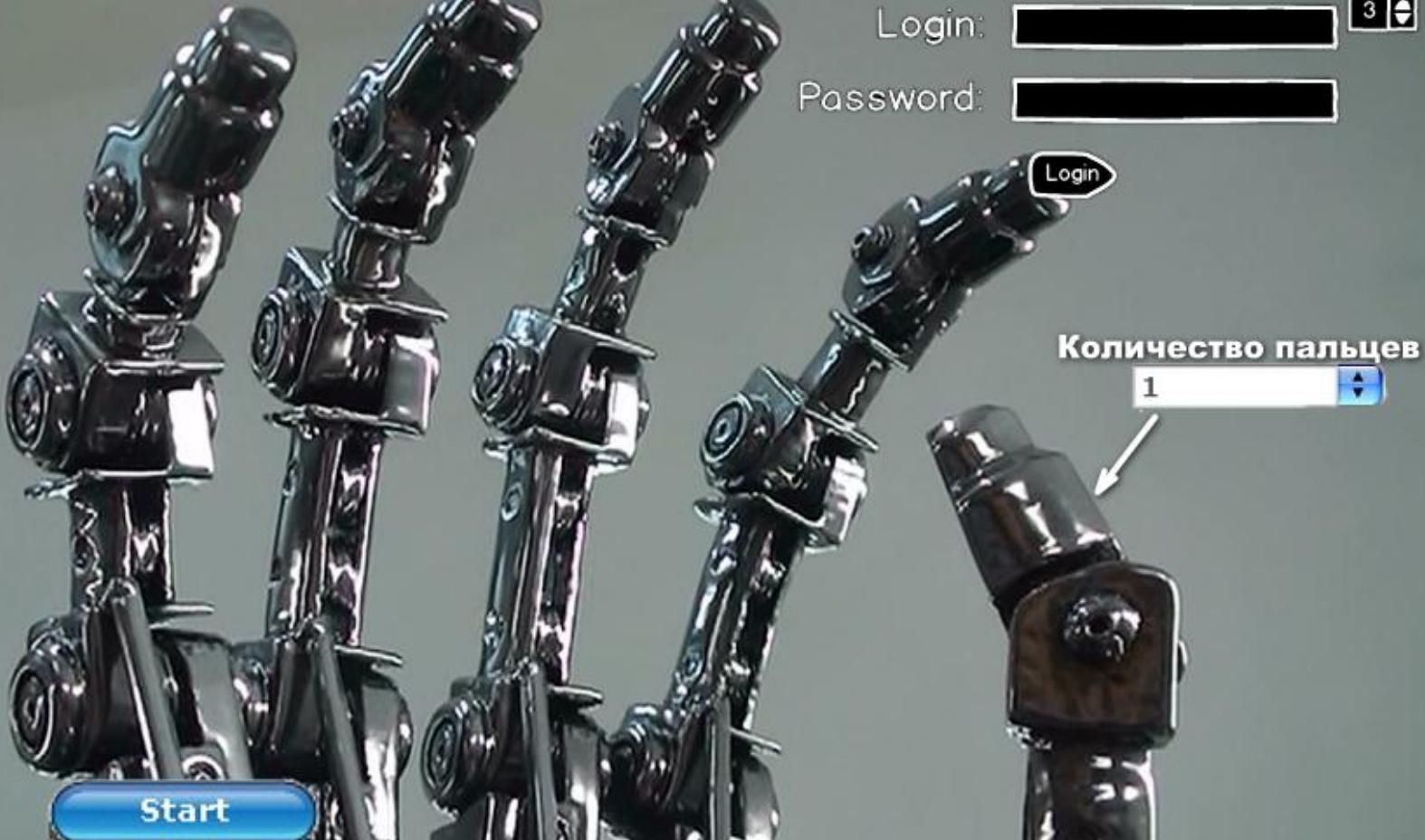
- ✓ Если нужная **страница уже открыта** (после предыдущего теста) – она **будет использована повторно**

```
public void Invoke()
{
    if (Exists() == false)
    {
        var mainPage = MyPages.MainPage;
        mainPage.Invoke();
        mainPage.GoToDonatePage();
    }
}
```

3

Login:

Password:



I HAVE EXPECTED CONTROLS

Сышишь, а есть какиета контролы?



```
public interface IHaveExpectedControls : IInvokable
{
    List<IWebElement> GetExpectedControls();
}
```



- .Invoke()
- .Exists()
- .GetExpectedControls()**

Get Expected Controls (PageObject)

```
public class DonatePage : AbstractPageBase, IHaveExpectedControls
{
    [FindsBy(Using=@"input[name='amount'][value='50']",  
        How = How.CssSelector)]  
    public IWebElement rbtnDonate50;  
  
    [FindsBy(Using=@"input[value='Donate by credit/debit card']",  
        How = How.CssSelector)]  
    public IWebElement btnMakeDonation;  
  
    public List<IWebElement> GetExpectedControls()  
    {  
        return new List<IWebElement>()  
        {  
            rbtnDonate50,  
            btnMakeDonation  
        };  
    }
}
```

Позволяет писать универсальные тесты

```
public virtual IHaveExpectedControls CurrentPage
{
    get { return null; }
}

[TestMethod]
public void TestExpectedControls()
{
    CurrentPage.Invoke();

    var expectedControls = CurrentPage.GetExpectedControls();
    foreach (var expectedControl in expectedControls)
    {
        Assert.IsTrue(expectedControl.Displayed);
    }
}
```

Авто-тесты для авто-тестов?



Легкие тесты, которые:

1. Открывают каждую страницу
2. Проверяет каждый важный элемент страницы

Основной тест-набор



Та да да дам!

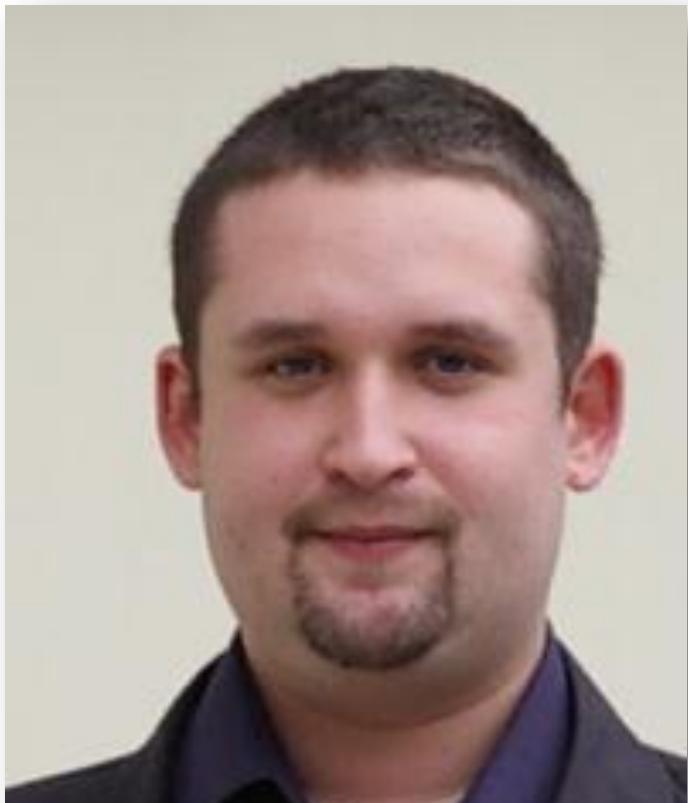
1. PageObject – это просто
2. PageObject – это СИЛА!
3. Пейдж-обжи больше – пиши меньше!
4. В Пейдж-объекте нет фокусов...
Есть только магия!



Спасибы:

- ✓ За то, что доклад состоялся:
Спасибо Вам!
- ✓ За Invoke()
Виктору Линчевскому
Леониду Артемьеву
- ✓ Помощь в подготовке доклада:
Михаил Поляруш
Андрей Ребров

Я не прощаюсь, я говорю: до свидания!



Дима Жарий

