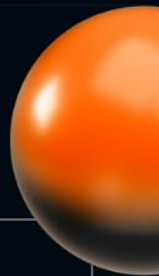
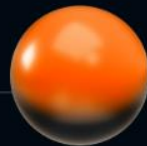


Как приручить автотестового монстра, или Dependency Injection в автотестах



Артём
Сидорук

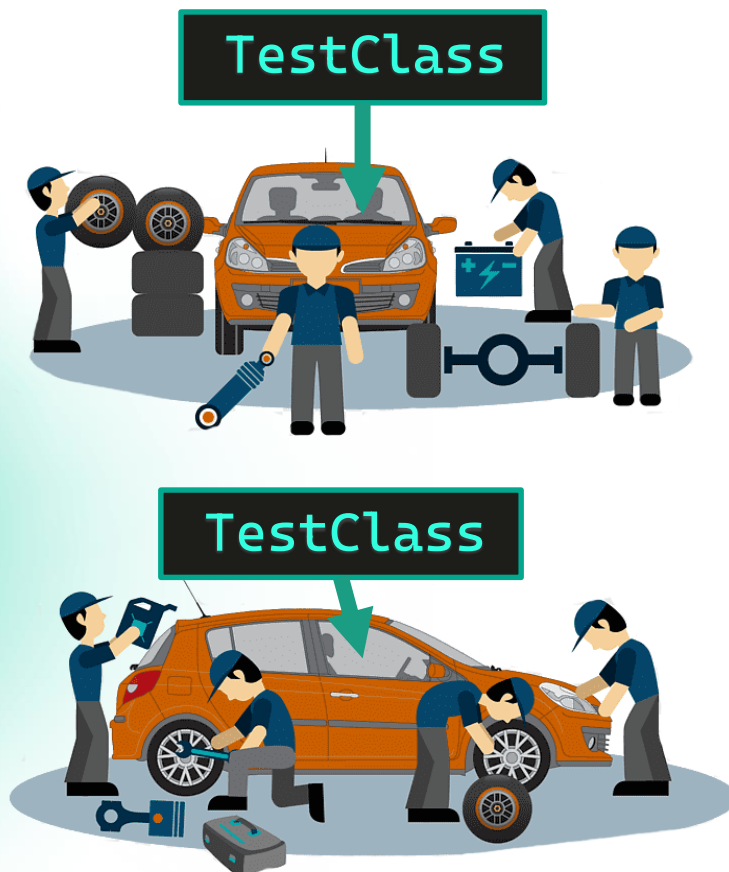
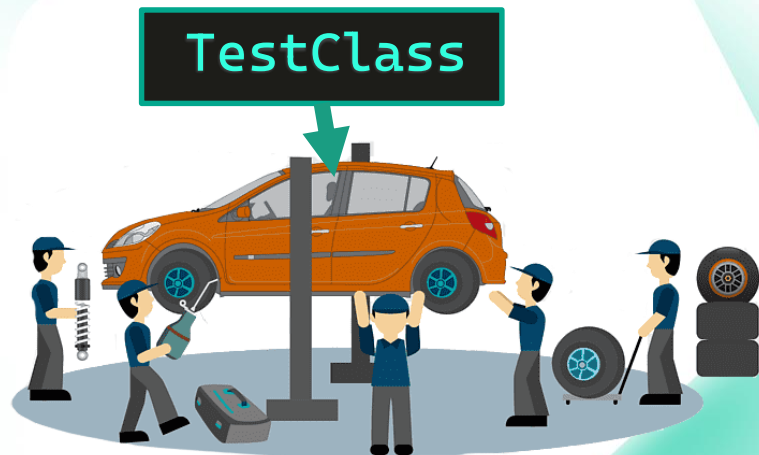
Kaspersky

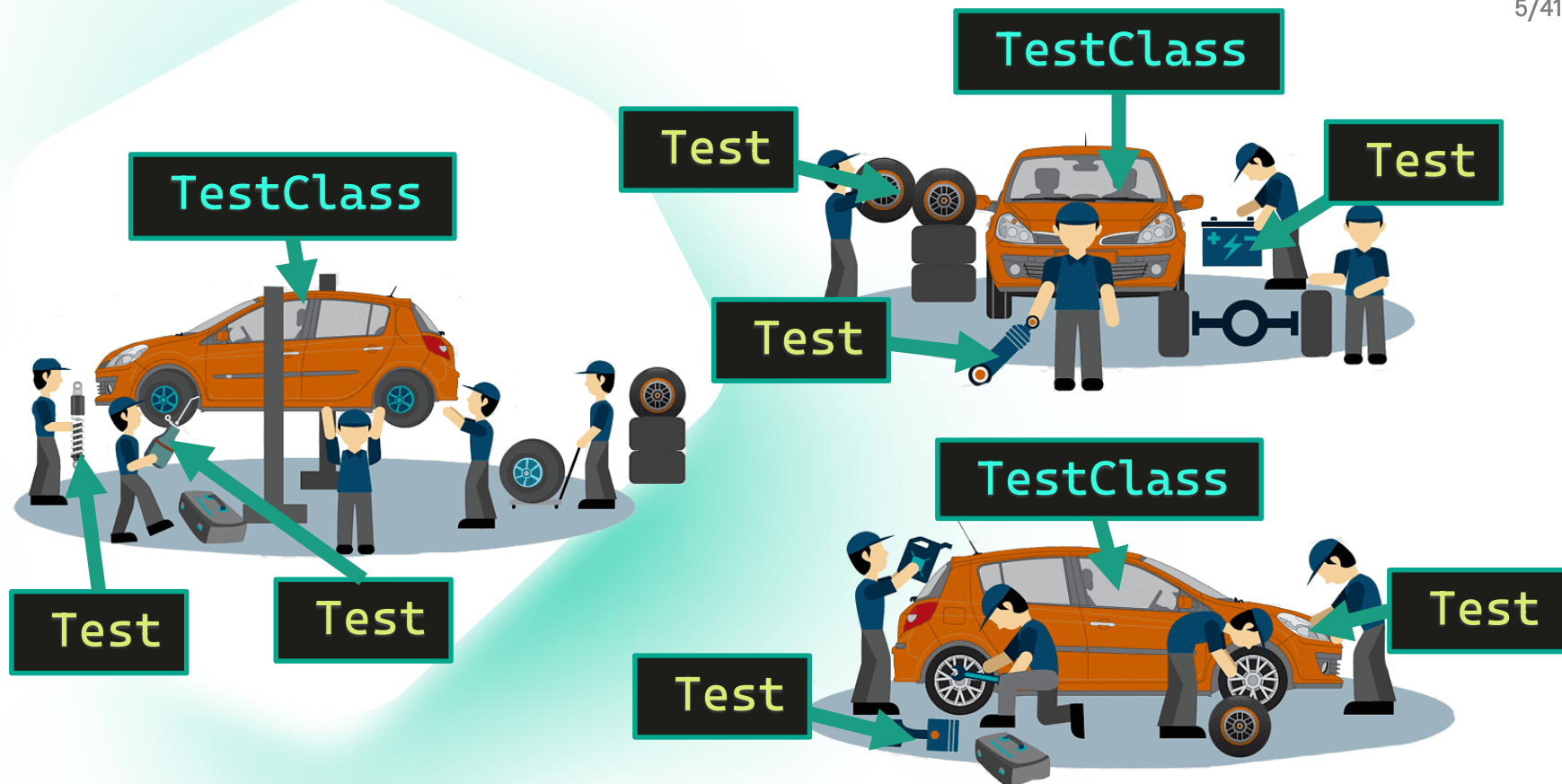


План доклада

- Посмотрим на проект с автотестами
- Определим его проблемные места
- Попытаемся их исправить



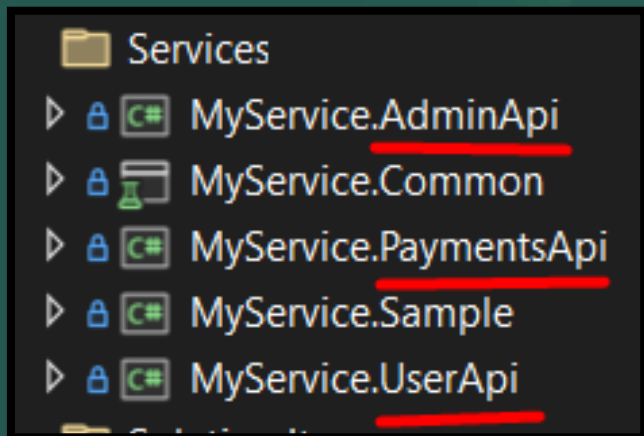




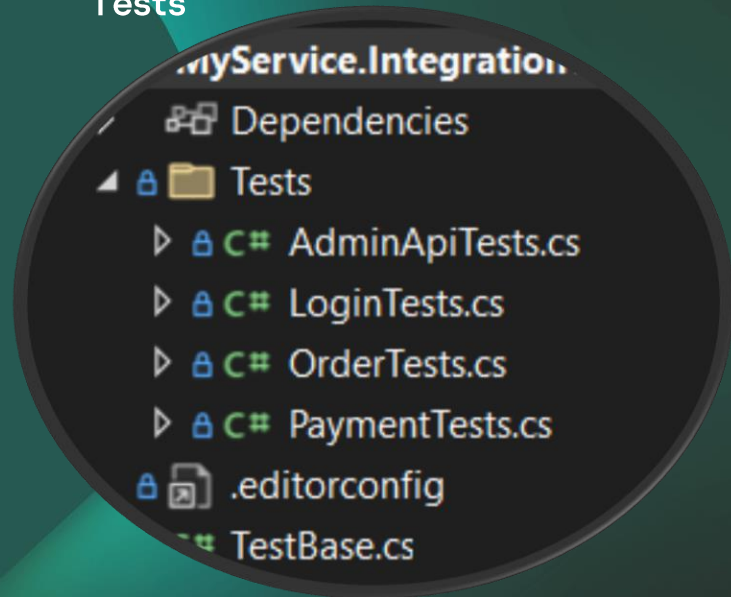
Solution

6/41

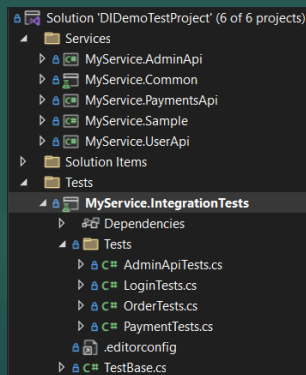
Services



Tests



Solution



Services

7/41

```
namespace MyService.UserApi
```

```
{
```

```
11 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
public class MyServiceUserApi : IMyServiceUserApi
```

```
{
```

```
public class MyServiceUserApi : IUserApi
```

```
    _baseUrl = baseUrl;
```

```
}
```

```
7 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
public UserSession Login(UserInfo user)...
```

```
7 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
public Order CreateOrder()...
```

```
3 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
public Order GetOrder(int orderNumber)
```

```
{
```

```
    return _client.Get<Order>($"{BaseUrl}/orders?order={orderNumber}");
```

```
}
```

```
3 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
public void DeleteOrder(int orderNumber)...
```

```
4 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
public string BaseUrl => $"{_baseUrl}/user";
```

```
private readonly string _baseUrl;
```

```
private readonly IRestHttpClient _client;
```

```
}
```

```
}
```

Services

8/41

```
namespace MyServiceUserApi
{
    public MyServiceUserApi(
        IRestHttpClient httpClient,
        string baseUrl)
    {
        _client = httpClient;
        _baseUrl = baseUrl;
    }
}
```

```
private readonly string _baseUrl;

private readonly IRestHttpClient _client;
```


Services

9/41

```
namespace MyService.UserApi
```

```
{
```

```
11 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
public class MyServiceUserApi : IMyServiceUserApi
```

```
{
```

```
10 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
public MyServiceUserApi(IRestHttpClient httpClient, string baseUrl)
```

```
{
```

```
    _httpClient = httpClient;
```

```
    _baseUrl = baseUrl;
```

```
public Order GetOrder(int orderNumber)
```

```
{
```

```
    var url = $"{_baseUrl}/orders?order={orderNumber}";
```

```
    return _httpClient.Get<Order>(url);
```

```
}
```

```
4 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
public string BaseUrl => $"{_baseUrl}/user";
```

```
private readonly string _baseUrl;
```

```
private readonly IRestHttpClient _httpClient;
```

```
}
```

```
}
```

Services

10/41

```
namespace MyService.UserApi
```

```
{  
    11 references | Artem Sidoruk, 1 day ago | 1 author, 1 change  
    public class MyServiceUserApi : IMyServiceUserApi  
    {
```

```
        public class MyServiceUserApi : IUserApi
```

```
        {  
            _baseUrl = baseUrl;
```

```
            public class MyServiceAdminApi : IAdminApi
```

```
            {  
                public class MyServicePaymentsApi : IPaymentsApi
```

```
                {  
                    return _client.Get<Order>($"{{BaseUrl}}/orders?order={{orderNumber}}");  
                }  
            }  
        }
```

```
        3 references | Artem Sidoruk, 1 day ago | 1 author, 1 change  
        public void DeleteOrder(int orderNumber)...
```

```
        4 references | Artem Sidoruk, 1 day ago | 1 author, 1 change  
        public string BaseUrl => $"{{_baseUrl}}/user";
```

```
        private readonly string _baseUrl;
```

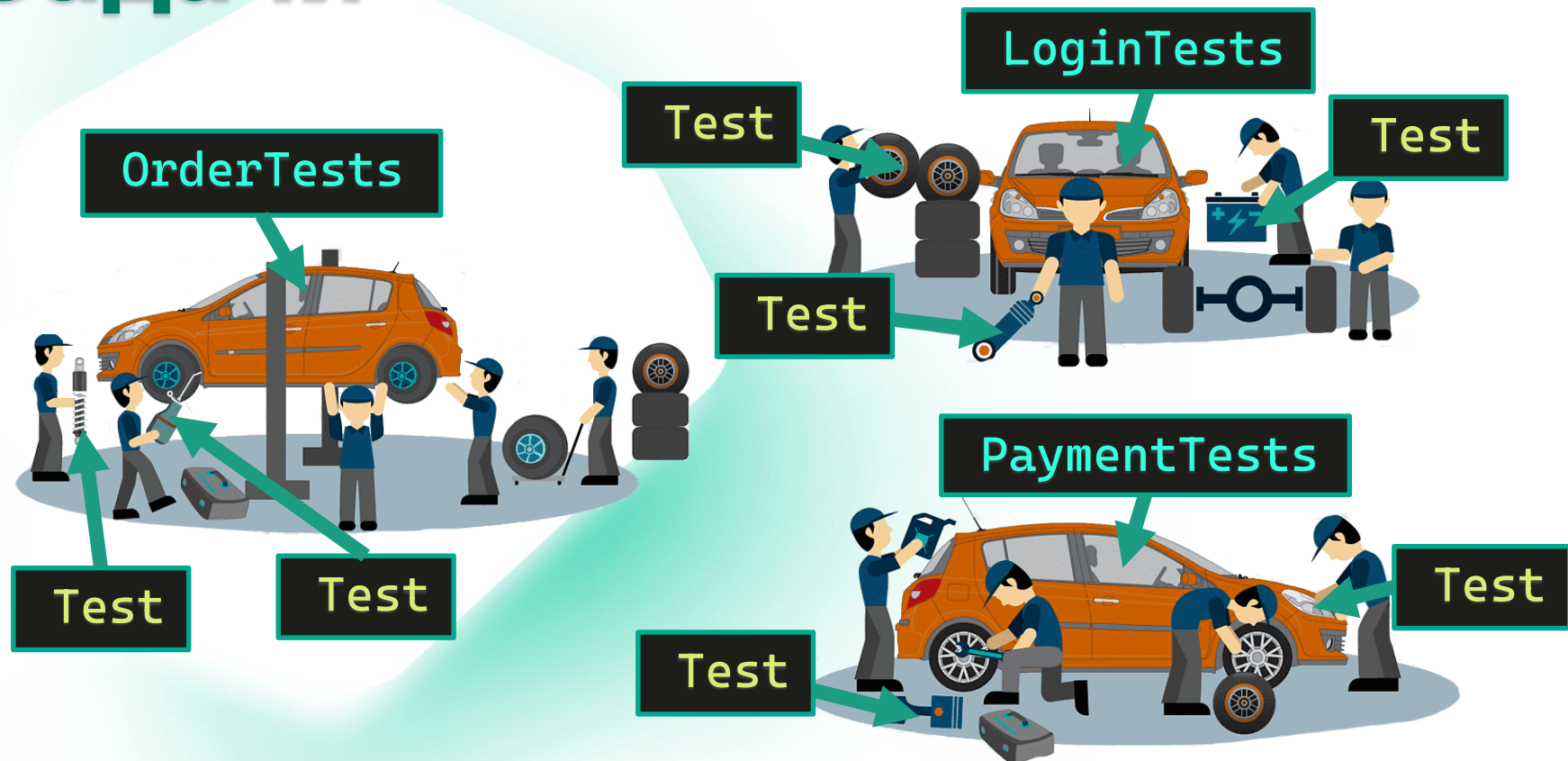
```
        private readonly IRestHttpClient _client;
```

```
    }
```

```
}
```

Задачи

11/41



AdminApiTests

12/41

```
5 namespace MyService.IntegrationTests.Tests
```

```
6 [Parallelizable(ParallelScope.All)]
```

```
7 public class AdminApiTests : TestBase
```

```
8 {
```

```
9     [SetUp]
```

```
10     public class AdminApiTests : TestBase
```

```
11     {
```

```
12         0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
13         public void AdminApi_CreateNewUser_UserLoginIsEmail()...
```

```
14         [Test]
```

```
15         0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
16         public void AdminApi_CreateNewUser_UserPasswordNotEmpty()...
```

```
17         [Test]
```

```
18         0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
19         public void AdminApi_DeleteUser_UserDeleted()...
```

```
20         [TearDown]
```

```
21         0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
22         public void Cleanup()...
```

```
23     }
```

```
24 }
```

AdminApiTests

13/41

```
5 namespace MyService.IntegrationTests.Tests
6 {
7     [TestFixture]
8     [Parallelizable(ParallelScope.All)]
9     0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
10     public class AdminApiTests : TestBase
11     {
12         [SetUp]
13         0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
14         public void Setup()
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
```

[Test]

public void

AdminApi_DeleteUser_UserDeleted()...

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

public void AdminApi_DeleteUser_UserDeleted()...

[TearDown]

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

public void Cleanup()...

AdminApiTests

14/41

```
5 namespace MyService.IntegrationTests.Tests
```

```
6 [Setup]
```

```
7 public void Setup()...
```

```
8 0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
9 public void Setup()
```

```
10 [Test]
```

```
11 public void
```

```
12 AdminApi_DeleteUser_UserDeleted()...
```

```
13 0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
14 public void AdminApi_DeleteUser_UserDeleted()...
```

```
15 [TearDown]
```

```
16 public void Cleanup()...
```

CreateOrder test

15/41

```
namespace MyService.IntegrationTests.Tests
```

```
{
```

```
[Test]
```

```
public void UserApi_CreateNewOrder_OrderCreated()
```

```
{
```

```
    // arrange
```

```
    IRestHttpClient httpClient = new RestHttpClient();
```

```
    IUserApi userApi = new MyServiceUserApi(httpClient, MyServiceUrl);
```

```
    userApi.Login(User);
```

```
    // act
```

```
    var order = userApi.CreateOrder();
```

```
    // assert
```

```
    order.Id.Should().BeGreaterThan(0, "Order number should has positive value");
```

```
    order.Name.Should().NotBeNullOrEmpty("Order name should be not empty text");
```

```
}
```

```
}
```

CreateOrder test

16/41

```
namespace MyService.IntegrationTests.Tests
```

```
{  
    [Test]  
    public void UserApi_CreateNewOrder_OrderCreated()  
    {
```

```
        // arrange
```

```
        IRestHttpClient httpClient = new RestHttpClient();
```

```
        IUserApi userApi = new MyServiceUserApi(httpClient, MyServiceUrl);
```

```
        userApi.Login(User);
```

```
        // act
```

```
        var order = userApi.CreateOrder();
```

```
        // assert
```

```
        order.Id.Should().BeGreaterThan(0, "Order number should has positive value");
```

```
        order.Name.Should().NotBeNullOrEmpty("Order name should be not empty text");
```

```
    }  
}
```


CreateOrder test

17/41

```
namespace MyService.IntegrationTests.Tests
```

```
{
```

```
[Test]
```

```
public void UserApi_CreateNewOrder_OrderCreated()
```

```
{
```

```
    // arrange
```

```
    IRestHttpClient httpClient = new RestHttpClient();
```

```
    IUserApi userApi = new MyServiceUserApi(httpClient, MyServiceUrl);
```

```
    userApi.Login(User);
```

```
    // act
```

```
    var order = userApi.CreateOrder();
```

```
    // assert
```

```
    order.Id.Should().BeGreaterThan(0, "Order number should has positive value");
```

```
    order.Name.Should().NotBeNullOrEmpty("Order name should be not empty text");
```

```
}
```

```
}
```

CreateOrder test

18/41

```
namespace MyService.IntegrationTests.Tests
```

```
{
```

```
[Test]
```

```
public void UserApi_CreateNewOrder_OrderCreated()
```

```
{
```

```
    // arrange
```

```
    IRestHttpClient httpClient = new RestHttpClient();
```

```
    IUserApi userApi = new MyServiceUserApi(httpClient, MyServiceUrl);
```

```
    userApi.Login(User);
```

```
    // act
```

```
    var order = userApi.CreateOrder();
```

```
    // assert
```

```
    order.Id.Should().BeGreaterThan(0, "Order number should has positive value");
```

```
    order.Name.Should().NotBeNullOrEmpty("Order name should be not empty text");
```

```
}
```

```
}
```

CreateOrder test

19/41

```
namespace MyService.IntegrationTests.Tests
```

```
{
```

```
[Test]
```

```
public void UserApi_CreateNewOrder_OrderCreated()
```

```
{
```

```
    // arrange
```

```
    IRestHttpClient httpClient = new RestHttpClient();
```

```
    IUserApi userApi = new MyServiceUserApi(httpClient, MyServiceUrl);
```

```
    userApi.Login(User);
```

```
    // act
```

```
    var order = userApi.CreateOrder();
```

```
    // assert
```

```
    order.Id.Should().BeGreaterThan(0, "Order number should has positive value");
```

```
    order.Name.Should().NotBeNullOrEmpty("Order name should be not empty text");
```

```
}
```

```
}
```

OrdersTests setup

20/41

namespace MyService.IntegrationTests.Tests

{

[TestFixture]

[Parallelizable(ParallelScope.All)]

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

[SetUp]

public void Setup()

{

IRestHttpClient client = new RestHttpClient();

IAdminApi adminApi = new MyServiceAdminApi(client, MyServiceUrl);

User = adminApi.CreateUser();

}

[Test]

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

public void UserApi_DeleteOrder_OrderDeleted()...

[TearDown]

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

public void Cleanup()...

}

}

OrdersTests setup

21/41

namespace MyService.IntegrationTests.Tests

{

[TestFixture]

[Parallelizable(ParallelScope.All)]

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

[SetUp]

public void Setup()

{

IRestHttpClient client = new RestHttpClient();

IAdminApi adminApi = new MyServiceAdminApi(client, MyServiceUrl);

User = adminApi.CreateUser();

}

[Test]

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

public void UserApi_DeleteOrder_OrderDeleted()...

[TearDown]

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

public void Cleanup()...

}

}

PayOrder test

22/41

```
11 namespace MyService.IntegrationTests.Tests
12 {
13     [TestFixture]
14     [Test]
15     [TestCase("RUR")]
16     [TestCase("EUR")]
17     [TestCase("USD")]
18     public void PayApi_PayForExistentOrder_PaymentSucced(string currency)
19     {
20         // arrange
21         var sum = 100.500;
22
23         // act
24         var result = LoginAndCreateOrderAndPay(sum, currency);
25
26         // assert
27         result.IsSuccess.Should().BeTrue(0, "Existent order can be paid");
28     }
29
30     protected UserSession Session...
31 }
32
33
34
35
36
37
38
39
40
41
42
43
44
45
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
101
102
103
104
105
106
107
108
109
```

PayOrder test

23/41

```
11 namespace MyService.IntegrationTests.Tests
12 {
13     [TestFixture]
14     [Test]
15     [TestCase("RUR")]
16     [TestCase("EUR")]
17     [TestCase("USD")]
18     public void PayApi_PayForExistentOrder_PaymentSucced(string currency)
19     {
20         // arrange
21         var sum = 100.500;
22
23         // act
24         var result = LoginAndCreateOrderAndPay(sum, currency);
25
26         // assert
27         result.IsSuccess.Should().BeTrue(0, "Existent order can be paid");
28     }
29 }
30
31 protected UserSession Session...
```

TestBase class

24/41

```
9 namespace MyService.IntegrationTests
10 {
11     4 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
12     public abstract class TestBase
13     {
14         1 reference | Artem Sidoruk, 1 day ago | 1 author, 1 change
15         protected IMyServiceUserApi CreateUserAndLogin()...
16
17         1 reference | Artem Sidoruk, 1 day ago | 1 author, 1 change
18         protected PayResult LoginAndCreateOrderAndPay(double sum, string currency)...
19
20         1 reference | Artem Sidoruk, 1 day ago | 1 author, 1 change
21         protected PayResult LoginAndCreateOrderAndPay(double sumInEuro)...
22
23         22 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
24         protected UserInfo User
25         {
26             get => NUnitHelper.GetTestContextProperty<UserInfo>(nameof(User));
27             set => NUnitHelper.SetTestContextProperty(nameof(User), value);
28         }
29
30         protected string MyServiceUrl = "http://localhost:9091";
31     }
32 }
```


TestBase class

```
9 namespace MyService.IntegrationTests
10 {
11     4 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
12     public abstract class TestBase
13     {
14         0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
15         protected IServiceUserApi CreateUserAndLogin()...
16
17         1 reference | Artem Sidoruk, 1 day ago | 1 author, 1 change
18         protected PayResult LoginAndCreateOrderAndPay(double sum, string currency)...
19
20     }
21
22     protected UserInfo User
23     {
24         get => ...GetTestContextProperty...
25         set => ...SetTestContextProperty...
26     }
27
28     protected const string MyServiceUrl = "http://localhost:9091";
```

TestBase class

```
9 namespace MyService.IntegrationTests
10 {
11     4 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
12     public abstract class TestBase
13     {
14         0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
15         protected IMyServiceUserApi CreateUserAndLogin()
16     }
22     protected IMyServiceUserApi CreateUserAndLogin() ...
23
44     protected PayResult LoginAndCreateOrderAndPay(double sum, string currency) ...
45
66     protected PayResult LoginAndCreateOrderAndPayWithEuro(double sum) ...
67
68     {
69         get => NUnitHelper.GetTestContextProperty<UserInfo>(nameof(User));
70         set => NUnitHelper.SetTestContextProperty(nameof(User), value);
71     }
72
73     protected string MyServiceUrl = "http://localhost:9091";
74 }
75 }
```

Tests

27/41

Test	Duration	Tra
▲ ✓ MyService.IntegrationTests (16)	2,1 min	
▲ ✓ MyService.IntegrationTests.Tests (16)	2,1 min	
▶ ✓ AdminApiTests (3)	15,6 sec	
▶ ✓ LoginTests (3)	20,8 sec	
▶ ✓ OrderTests (3)	22 sec	
▶ ✓ PaymentTests (7)	1,2 min	

Оптимизация

28/41



OrderTests



LoginTests

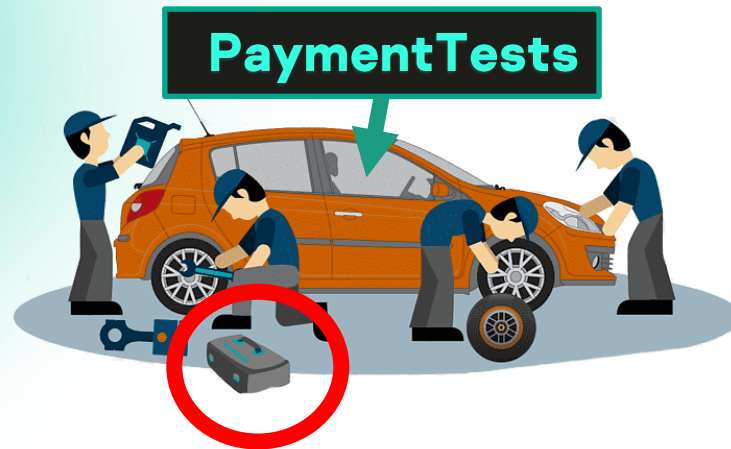
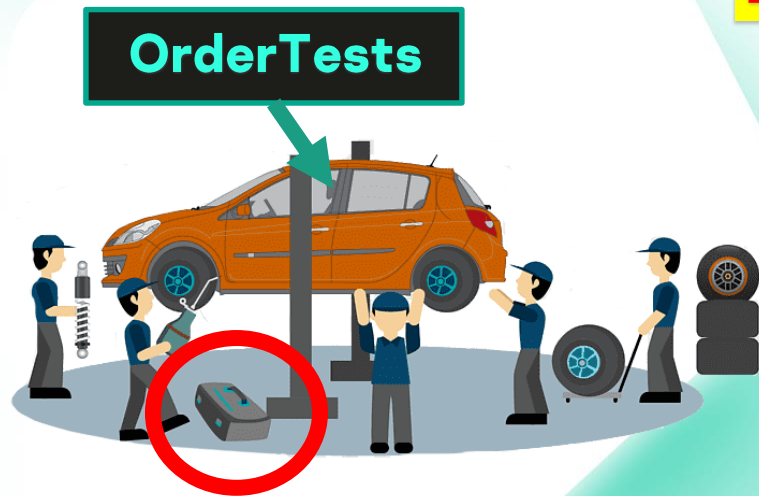


PaymentTests



А где инструменты?

30/41



Новый сотрудник

31/41



```
[SetUp]
public void Setup()
{
    IRestHttpClient client = new RestHttpClient();
    IAdminApi adminApi = new MyServiceAdminApi(client, MyServiceUrl);
    User = adminApi.CreateUser();
}
```



```
[SetUp]
```

```
public void Setup()
```

```
[Test]
```

```
public void UserApi_CreateNewOrder_OrderCreated()
```

```
{
```

```
    IRestHttpClient client = new RestHttpClient();
```

```
    IUserApi userApi = new MyServiceUserApi(client, MyServiceUrl);
```

```
    var order = userApi.CreateOrder();
```

```
}
```

```
[SetUp]
```

```
public void Setup()
```

```
[Test]
```

```
public void UserApi_CreateNewOrder_OrderCreated()
```

```
[TearDown]
```

```
public void Cleanup()
```

```
{
```

```
    IRestHttpClient client = new RestHttpClient();
```

```
    IAdminApi adminApi = new MyServiceAdminApi(client, MyServiceUrl);
```

```
    adminApi.DeleteUser(User);
```

```
}
```

```
IRestHttpClient client = new RestHttpClient();  
IUserApi userApi = new MyServiceUserApi(client, MyServiceUrl);
```

№1 Явные вызовы конструкторов

36/41

```
IRestHttpClient client = new RestHttpClient();  
IUserApi userApi = new MyServiceUserApi(client, MyServiceUrl);
```

Dependency Injection / Inversion of Control ^{37/41}

```
public MyServiceUserApi(  
    IRestHttpClient httpClient,  
    string baseUrl)  
{  
    _client = httpClient;  
    _baseUrl = baseUrl;  
}
```

Dependency Injection / Inversion of Control ^{38/41}

```
public MyServiceUserApi(  
    IRestHttpClient httpClient,  
    string baseUrl)  
{  
    _client = httpClient;  
    _baseUrl = baseUrl;  
}
```

```
public MyServiceUserApi(string baseUrl)  
{  
    _client = new RestHttpClient(baseUrl);  
    _baseUrl = baseUrl;  
}
```

Dependency Injection Container

39/41


DI (Dependency Injection)

Service Locator

IoC (Inversion Of Control)

DI Containers

40/41



Autofac

PHP-DI

container

Butterfly Container

Castle Windsor

bottlejs

Guice

Pico Container

Ninject

dependency_injector

Spring (Bean, DI)

DI Containers

41/41

A word cloud of various Dependency Injection (DI) containers. The words are arranged in a roughly triangular shape, with 'Castle Windsor' being the largest and most prominent, circled in red. Other containers include 'Autofac', 'Butterfly Container', 'Guice', 'Pico Container', 'Ninject', 'Spring (Bean, DI)', 'bottlejs', 'PHP-DI', 'dependency_injector', and 'container'.

Autofac

PHP-DI

container

Butterfly Container

Castle Windsor

bottlejs

Guice

Pico Container

Ninject

dependency_injector

Spring (Bean, DI)

Create DI Container

42/41

```
1  using Castle.MicroKernel.Registration;
2  using Castle.Windsor;
3
4  public class ContainerInitializer
5
6  using MyService.UserApi;
7
8  namespace MyService.IntegrationTests.DI
9  {
10     1 reference | Artem Sidoruk, 1 day ago | 1 author, 1 change
11     public class ContainerInitializer
12     {
13         1 reference | Artem Sidoruk, 1 day ago | 1 author, 1 change
14         public IWindsorContainer GetContainer()...
15
16         1 reference | Artem Sidoruk, 1 day ago | 1 author, 1 change
17         public void Install(IWindsorContainer container)...
```

Create DI Container

43/41

```
1 using Castle.MicroKernel.Registration;  
2 using Castle.Windsor;
```

```
3  
4 public class ContainerInitializer
```

```
5  
6 public void IWindsorContainer GetContainer()  
7 {  
8  
9     var container = new WindsorContainer();  
10    Install(container);  
11    return container;  
12  
13 }  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51 }
```

Create DI Container

```
1  using Castle.MicroKernel.Registration;  
2  using Castle.Windsor;  
3  using MyService.AdminApi;  
4  using MyService.Common;  
5  using MyService.PaymentsApi;  
6  
7  private void Install(IWindsorContainer container)  
8  {  
9  
10     container.Register(  
11         XXX,  
12         YYY,  
20         ZZZ);  
21  
50 }  
51
```

1
2
3
4
5
6
7
8
9

10
11

12
20

21
50
51

using Castle Windsor:

Create DI Container

```
1 using Castle.MicroKernel.Registration;
2 using Castle.Windsor;
3 container.Register(
4     Component
5         .For<IRestHttpClient>()
6         .ImplementedBy<RestHttpClient>()
7         .LifestyleXXX(),
8     Component
9         .For<IAdminApi>()
10        .ImplementedBy<MyServiceAdminApi>()
11        .LifestyleXXX(),
12    );
```


Create DI Container

```
1 using Castle.MicroKernel.Registration;
2 using Castle.Windsor;
3 container.Register(
4     Component
5         .For<IRestHttpClient>()
6         .ImplementedBy<RestHttpClient>()
7         .LifestyleXXX(),
8     Component
9         .For<IAdminApi>()
10        .ImplementedBy<MyServiceAdminApi>()
11        .LifestyleXXX(),
12    Component
13        .For<IMyService>()
14        .ImplementedBy<MyService>()
15        .LifestyleXXX(),
16    Component
17        .For<IMyService2>()
18        .ImplementedBy<MyService2>()
19        .LifestyleXXX(),
20    Component
21        .For<IMyService3>()
22        .ImplementedBy<MyService3>()
23        .LifestyleXXX(),
24    Component
25        .For<IMyService4>()
26        .ImplementedBy<MyService4>()
27        .LifestyleXXX(),
28    Component
29        .For<IMyService5>()
30        .ImplementedBy<MyService5>()
31        .LifestyleXXX(),
32    Component
33        .For<IMyService6>()
34        .ImplementedBy<MyService6>()
35        .LifestyleXXX(),
36    Component
37        .For<IMyService7>()
38        .ImplementedBy<MyService7>()
39        .LifestyleXXX(),
40    Component
41        .For<IMyService8>()
42        .ImplementedBy<MyService8>()
43        .LifestyleXXX(),
44    Component
45        .For<IMyService9>()
46        .ImplementedBy<MyService9>()
47        .LifestyleXXX(),
48    Component
49        .For<IMyService10>()
50        .ImplementedBy<MyService10>()
51        .LifestyleXXX(),
```

```

1  using Castle.MicroKernel.Registration;
2  using Castle.Windsor;
3  container.Register(
4      Component
5          .For<IRestHttpClient>()
6          .ImplementedBy<RestHttpClient>()
7          .LifestyleXXX(),
8      Component
9          .For<IAdminApi>()
10         .ImplementedBy<MyServiceAdminApi>()
11         .LifestyleXXX(),
12
13     );
14
15     // 2. Add the service to the container
16     container.Register(
17         Component
18             .For<IRestHttpClient>()
19             .ImplementedBy<RestHttpClient>()
20             .LifestyleXXX(),
21         Component
22             .For<IAdminApi>()
23             .ImplementedBy<MyServiceAdminApi>()
24             .LifestyleXXX(),
25
26         );
27
28     // 3. Add the service to the container
29     container.Register(
30         Component
31             .For<IRestHttpClient>()
32             .ImplementedBy<RestHttpClient>()
33             .LifestyleXXX(),
34         Component
35             .For<IAdminApi>()
36             .ImplementedBy<MyServiceAdminApi>()
37             .LifestyleXXX(),
38
39         );
40
41     // 4. Add the service to the container
42     container.Register(
43         Component
44             .For<IRestHttpClient>()
45             .ImplementedBy<RestHttpClient>()
46             .LifestyleXXX(),
47         Component
48             .For<IAdminApi>()
49             .ImplementedBy<MyServiceAdminApi>()
50             .LifestyleXXX(),
51
52         );
53
54     // 5. Add the service to the container
55     container.Register(
56         Component
57             .For<IRestHttpClient>()
58             .ImplementedBy<RestHttpClient>()
59             .LifestyleXXX(),
60         Component
61             .For<IAdminApi>()
62             .ImplementedBy<MyServiceAdminApi>()
63             .LifestyleXXX(),
64
65         );
66
67     // 6. Add the service to the container
68     container.Register(
69         Component
70             .For<IRestHttpClient>()
71             .ImplementedBy<RestHttpClient>()
72             .LifestyleXXX(),
73         Component
74             .For<IAdminApi>()
75             .ImplementedBy<MyServiceAdminApi>()
76             .LifestyleXXX(),
77
78         );
79
80     // 7. Add the service to the container
81     container.Register(
82         Component
83             .For<IRestHttpClient>()
84             .ImplementedBy<RestHttpClient>()
85             .LifestyleXXX(),
86         Component
87             .For<IAdminApi>()
88             .ImplementedBy<MyServiceAdminApi>()
89             .LifestyleXXX(),
90
91         );
92
93     // 8. Add the service to the container
94     container.Register(
95         Component
96             .For<IRestHttpClient>()
97             .ImplementedBy<RestHttpClient>()
98             .LifestyleXXX(),
99         Component
100             .For<IAdminApi>()
101             .ImplementedBy<MyServiceAdminApi>()
102             .LifestyleXXX(),
103
104         );
105
106     // 9. Add the service to the container
107     container.Register(
108         Component
109             .For<IRestHttpClient>()
110             .ImplementedBy<RestHttpClient>()
111             .LifestyleXXX(),
112         Component
113             .For<IAdminApi>()
114             .ImplementedBy<MyServiceAdminApi>()
115             .LifestyleXXX(),
116
117         );
118
119     // 10. Add the service to the container
120     container.Register(
121         Component
122             .For<IRestHttpClient>()
123             .ImplementedBy<RestHttpClient>()
124             .LifestyleXXX(),
125         Component
126             .For<IAdminApi>()
127             .ImplementedBy<MyServiceAdminApi>()
128             .LifestyleXXX(),
129
130         );
131
132     // 11. Add the service to the container
133     container.Register(
134         Component
135             .For<IRestHttpClient>()
136             .ImplementedBy<RestHttpClient>()
137             .LifestyleXXX(),
138         Component
139             .For<IAdminApi>()
140             .ImplementedBy<MyServiceAdminApi>()
141             .LifestyleXXX(),
142
143         );
144
145     // 12. Add the service to the container
146     container.Register(
147         Component
148             .For<IRestHttpClient>()
149             .ImplementedBy<RestHttpClient>()
150             .LifestyleXXX(),
151         Component
152             .For<IAdminApi>()
153             .ImplementedBy<MyServiceAdminApi>()
154             .LifestyleXXX(),
155
156         );
157
158     // 13. Add the service to the container
159     container.Register(
160         Component
161             .For<IRestHttpClient>()
162             .ImplementedBy<RestHttpClient>()
163             .LifestyleXXX(),
164         Component
165             .For<IAdminApi>()
166             .ImplementedBy<MyServiceAdminApi>()
167             .LifestyleXXX(),
168
169         );
170
171     // 14. Add the service to the container
172     container.Register(
173         Component
174             .For<IRestHttpClient>()
175             .ImplementedBy<RestHttpClient>()
176             .LifestyleXXX(),
177         Component
178             .For<IAdminApi>()
179             .ImplementedBy<MyServiceAdminApi>()
180             .LifestyleXXX(),
181
182         );
183
184     // 15. Add the service to the container
185     container.Register(
186         Component
187             .For<IRestHttpClient>()
188             .ImplementedBy<RestHttpClient>()
189             .LifestyleXXX(),
190         Component
191             .For<IAdminApi>()
192             .ImplementedBy<MyServiceAdminApi>()
193             .LifestyleXXX(),
194
195         );
196
197     // 16. Add the service to the container
198     container.Register(
199         Component
200             .For<IRestHttpClient>()
201             .ImplementedBy<RestHttpClient>()
202             .LifestyleXXX(),
203         Component
204             .For<IAdminApi>()
205             .ImplementedBy<MyServiceAdminApi>()
206             .LifestyleXXX(),
207
208         );
209
210     // 17. Add the service to the container
211     container.Register(
212         Component
213             .For<IRestHttpClient>()
214             .ImplementedBy<RestHttpClient>()
215             .LifestyleXXX(),
216         Component
217             .For<IAdminApi>()
218             .ImplementedBy<MyServiceAdminApi>()
219             .LifestyleXXX(),
220
221         );
222
223     // 18. Add the service to the container
224     container.Register(
225         Component
226             .For<IRestHttpClient>()
227             .ImplementedBy<RestHttpClient>()
228             .LifestyleXXX(),
229         Component
230             .For<IAdminApi>()
231             .ImplementedBy<MyServiceAdminApi>()
232             .LifestyleXXX(),
233
234         );
235
236     // 19. Add the service to the container
237     container.Register(
238         Component
239             .For<IRestHttpClient>()
240             .ImplementedBy<RestHttpClient>()
241             .LifestyleXXX(),
242         Component
243             .For<IAdminApi>()
244             .ImplementedBy<MyServiceAdminApi>()
245             .LifestyleXXX(),
246
247         );
248
249     // 20. Add the service to the container
250     container.Register(
251         Component
252             .For<IRestHttpClient>()
253             .ImplementedBy<RestHttpClient>()
254             .LifestyleXXX(),
255         Component
256             .For<IAdminApi>()
257             .ImplementedBy<MyServiceAdminApi>()
258             .LifestyleXXX(),
259
260         );
261
262     // 21. Add the service to the container
263     container.Register(
264         Component
265             .For<IRestHttpClient>()
266             .ImplementedBy<RestHttpClient>()
267             .LifestyleXXX(),
268         Component
269             .For<IAdminApi>()
270             .ImplementedBy<MyServiceAdminApi>()
271             .LifestyleXXX(),
272
273         );
274
275     // 22. Add the service to the container
276     container.Register(
277         Component
278             .For<IRestHttpClient>()
279             .ImplementedBy<RestHttpClient>()
280             .LifestyleXXX(),
281         Component
282             .For<IAdminApi>()
283             .ImplementedBy<MyServiceAdminApi>()
284             .LifestyleXXX(),
285
286         );
287
288     // 23. Add the service to the container
289     container.Register(
290         Component
291             .For<IRestHttpClient>()
292             .ImplementedBy<RestHttpClient>()
293             .LifestyleXXX(),
294         Component
295             .For<IAdminApi>()
296             .ImplementedBy<MyServiceAdminApi>()
297             .LifestyleXXX(),
298
299         );
300
301     // 24. Add the service to the container
302     container.Register(
303         Component
304             .For<IRestHttpClient>()
305             .ImplementedBy<RestHttpClient>()
306             .LifestyleXXX(),
307         Component
308             .For<IAdminApi>()
309             .ImplementedBy<MyServiceAdminApi>()
310             .LifestyleXXX(),
311
312         );
313
314     // 25. Add the service to the container
315     container.Register(
316         Component
317             .For<IRestHttpClient>()
318             .ImplementedBy<RestHttpClient>()
319             .LifestyleXXX(),
320         Component
321             .For<IAdminApi>()
322             .ImplementedBy<MyServiceAdminApi>()
323             .LifestyleXXX(),
324
325         );
326
327     // 26. Add the service to the container
328     container.Register(
329         Component
330             .For<IRestHttpClient>()
331             .ImplementedBy<RestHttpClient>()
332             .LifestyleXXX(),
333         Component
334             .For<IAdminApi>()
335             .ImplementedBy<MyServiceAdminApi>()
336             .LifestyleXXX(),
337
338         );
339
340     // 27. Add the service to the container
341     container.Register(
342         Component
343             .For<IRestHttpClient>()
344             .ImplementedBy<RestHttpClient>()
345             .LifestyleXXX(),
346         Component
347             .For<IAdminApi>()
348             .ImplementedBy<MyServiceAdminApi>()
349             .LifestyleXXX(),
350
351         );
352
353     // 28. Add the service to the container
354     container.Register(
355         Component
356             .For<IRestHttpClient>()
357             .ImplementedBy<RestHttpClient>()
358             .LifestyleXXX(),
359         Component
360             .For<IAdminApi>()
361             .ImplementedBy<MyServiceAdminApi>()
362             .LifestyleXXX(),
363
364         );
365
366     // 29. Add the service to the container
367     container.Register(
368         Component
369             .For<IRestHttpClient>()
370             .ImplementedBy<RestHttpClient>()
371             .LifestyleXXX(),
372         Component
373             .For<IAdminApi>()
374             .ImplementedBy<MyServiceAdminApi>()
375             .LifestyleXXX(),
376
377         );
378
379     // 30. Add the service to the container
380     container.Register(
381         Component
382             .For<IRestHttpClient>()
383             .ImplementedBy<RestHttpClient>()
384             .LifestyleXXX(),
385         Component
386             .For<IAdminApi>()
387             .ImplementedBy<MyServiceAdminApi>()
388             .LifestyleXXX(),
389
390         );
391
392     // 31. Add the service to the container
393     container.Register(
394         Component
395             .For<IRestHttpClient>()
396             .ImplementedBy<RestHttpClient>()
397             .LifestyleXXX(),
398         Component
399             .For<IAdminApi>()
400             .ImplementedBy<MyServiceAdminApi>()
401             .LifestyleXXX(),
402
403         );
404
405     // 32. Add the service to the container
406     container.Register(
407         Component
408             .For<IRestHttpClient>()
409             .ImplementedBy<RestHttpClient>()
410             .LifestyleXXX(),
411         Component
412             .For<IAdminApi>()
413             .ImplementedBy<MyServiceAdminApi>()
4
```

Init container

51/41

```
public abstract class TestBase
```

```
protected IWindsorContainer Container =>  
    _container ??= new ContainerInitializer()  
        .GetContainer();
```

```
private static IWindsorContainer _container;
```

```
protected UserHelper UserHelper => Container.Resolve<UserHelper>();
```

5 references | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

```
protected TestContext TestContext => Container.Resolve<TestContext>();
```

7 references | Sugrob57, 2 days ago | 1 author, 1 change | 1 work item

```
protected IWindsorContainer Container => _container ??= new ContainerInitializer().GetContainer();
```

```
private static IWindsorContainer _container;
```

Resolve components

```
public abstract class TestBase
```

```
protected IAdminApi AdminApi =>  
    Container.Resolve<IAdminApi>();
```

```
protected IUserApi UserApi =>  
    Container.Resolve<IUserApi>();
```

```
protected IPaymentsApi PaymentsApi =>  
    Container.Resolve<IPaymentsApi>();
```

Resolve components

```
public abstract class TestBase
```

```
protected IAdminApi AdminApi =>  
    Container.Resolve<IAdminApi>();
```

```
protected IUserApi UserApi =>  
    Container.Resolve<IUserApi>();
```

```
protected IPaymentsApi PaymentsApi =>  
    Container.Resolve<IPaymentsApi>();
```

Resolve components

```
public abstract class TestBase
```

```
protected IAdminApi AdminApi =>  
    Container.Resolve<IAdminApi>();
```

```
protected IUserApi UserApi =>  
    Container.Resolve<IUserApi>();
```

```
protected IPaymentsApi PaymentsApi =>  
    Container.Resolve<IPaymentsApi>();
```

Очищаем тесты

55/41

```
[Test]
public void PayApi_PayForExistentOrder_PaymentSucced()
{
    // arrange
    IRestHttpClient httpClient = new RestHttpClient();
    IUserApi userApi = new MyServiceUserApi(httpClient, MyServiceUrl);
    userApi.Login(User);
    ...
}
```

Очищаем тесты

56/41

```
[Test]
public void PayApi_PayForExistentOrder_PaymentSucced()
{
    // arrange
    IRestHttpClient httpClient = new RestHttpClient();
    TUserApi userApi = new MyServiceUserApi(httpClient, MyServiceUrl);
    userApi.Login(User);
    ...
}
```


Очищаем тесты

57/41

```
[Test]
public void PayApi_PayForExistentOrder_PaymentSucced()
{
    // arrange
    IRestHttpClient httpClient = new RestHttpClient();
    IUserApi userApi = new MyServiceUserApi(httpClient, MyServiceUrl);
    userApi.Login(User); ➔ UserApi.Login(User);
    ...
}
```

~~№1 Явные вызовы конструкторов~~

58/41

```
IRestHttpClient client = new RestHttpClient();  
IUserApi userApi = new MyServiceUserApi(client, MyServiceUrl);
```

Container.Resolve<IUserApi>();

```
[SetUp]
public void SetUp()
{
    IRestHttpClient client = new RestHttpClient();
    IAdminApi adminApi = new MyServiceAdminApi(client, MyServiceUrl);
    User = adminApi.CreateUser();
}
```

```
[SetUp]
```

```
public void Setup()
```

```
{
```

```
    IRestHttpClient client = new RestHttpClient();
```

```
    IAdminApi adminApi = new MyServiceAdminApi(client, MyServiceUrl);
```

```
    User = adminApi.CreateUser();
```

```
[TearDown]
```

```
public void Cleanup()
```

```
{
```

```
    IRestHttpClient client = new RestHttpClient();
```

```
    IAdminApi adminApi = new MyServiceAdminApi(client, MyServiceUrl);
```

```
    adminApi.DeleteUser(User);
```

№2 Хранение объектов

61/41

[SetUp]

```
... new MyServiceUserApi(client, MyServiceUrl);
```

[Test]

```
... new MyServiceUserApi(client, MyServiceUrl);
```

[TearDown]

```
... new MyServiceAdminApi(client, MyServiceUrl);
```

LifeStyles

62/41

LifeStyles

- Singleton



LifeStyles

- Singleton
- Transient



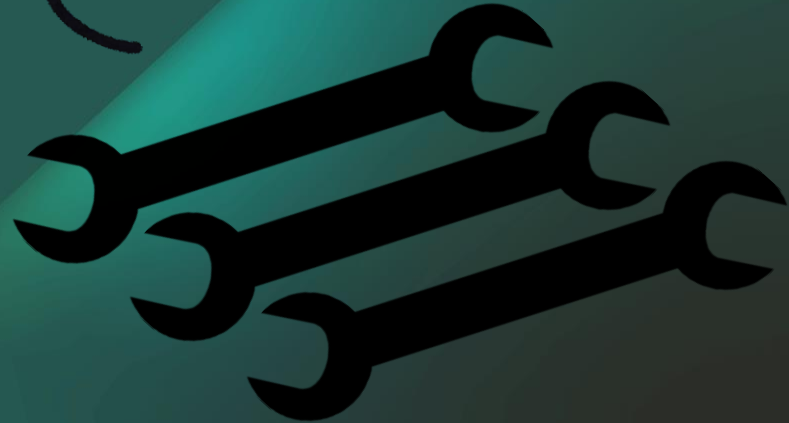
LifeStyles

- Singleton
- Transient
- Scoped



LifeStyles

- Singleton
- Transient
- Scoped
- Pooled



Add LifeStyles

67/41

Component

```
.For<IAdminApi>()  
.ImplementedBy<MyServiceAdminApi>()  
.LifestyleSingleton(),
```

Add LifeStyles

68/41

Component

```
.For<IAdminApi>()  
.ImplementedBy<MyServiceAdminApi>()  
.LifestyleSingleton(),
```

Add LifeStyles

69/41

Component

```
.For<IAdminApi>()  
.ImplementedBy<MyServiceAdminApi>()  
.LifestyleSingleton(),
```

IPaymentsApi

Add LifeStyles

70/41

Component

```
.For<IAdminApi>()  
.ImplementedBy<MyServiceAdminApi>()  
.LifestyleSingleton(),
```

```
IPaymentsApi => Transient
```

Add LifeStyles

71/41

Component

```
.For<IAdminApi>()  
.ImplementedBy<MyServiceAdminApi>()  
.LifestyleSingleton(),
```

IPaymentsApi => Transient

IUserApi

Add LifeStyles

72/41

Component

```
.For<IAdminApi>()  
.ImplementedBy<MyServiceAdminApi>()  
.LifestyleSingleton(),
```

IPaymentsApi => Transient

IUserApi => Scoped

Begin scope

73/41

4 references | Artem Sidoruk, 1 day ago | 2 authors, 4 changes | 2 work items

public abstract class TestBase

{

```
public abstract class TestBase
```

{
 TestScope = Container.BeginScope();

```
[SetUp]
```

```
public void BaseSetUp()
```

```
{
```

```
    Container.BeginScope();
```

```
}
```

```
protected IServiceUserApi UserApi => Container.Resolve<IServiceUserApi>();
```

0 references | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

```
protected IMyServicePaymentsApi PaymentsApi => Container.Resolve<IMyServicePaymentsApi>();
```

№2 Хранение объектов

74/41

```
[SetUp]
```

```
... new [Test]
```

```
... new MyServiceAdminApi(client, MyServiceUrl);
```

```
[TearDown]
```

```
... new MyServiceAdminApi(client, MyServiceUrl);
```

```
.ImplementedBy<MyServiceAdminApi>()  
.LifestyleSingleton(),
```

```
9 namespace MyService.IntegrationTests
10 {
```

```
1 public abstract class TestBase
```

```
2     protected IUserApi
3         CreateUserAndLogin() ...
```

```
4     protected PayResult
5         LoginAndCreateOrderAndPay(double sum, string currency) ...
```

```
6     protected PayResult
7         LoginAndCreateOrderAndPayWithEuro(double sum) ...
```

```
75 }
```

№3 «Все подряд» в базовом классе

76/41

```
9 namespace MyService.IntegrationTests
10 {
```

```
1 public abstract class TestBase
```

```
2     protected IUserApi
```

```
3         CreateUserAndLogin() ...
```

```
4     protected PayResult
```

```
5         LoginAndCreateOrderAndPay(double sum, string currency) ...
```

```
6     protected PayResult
```

```
7         LoginAndCreateOrderAndPayWithEuro(double sum) ...
```

```
74     }
75 }
```

TestServices

77/41

```
public class PaymentHelper
{
    public PaymentHelper(
        IMyServiceUserApi userApi,
        IMyServicePaymentsApi paymentsApi,
        TestContext testContext,
        UserHelper userHelper)
    {
        _userApi = userApi;
        _paymentsApi = paymentsApi;
        _testContext = testContext;
        _userHelper = userHelper;
    }

    public PayResult LoginAndCreateOrderAndPay(double sum, string currency = "Euro")...

    public PayResult Pay(int orderId, PayRequest payRequest)...

    private readonly IMyServiceUserApi _userApi;
    private readonly IMyServicePaymentsApi _paymentsApi;
    private readonly TestContext _testContext;
    private readonly UserHelper _userHelper;
}
```

TestServices

78/41

```
public class PaymentHelper
```

```
{
```

```
public class PaymentTestService
```

```
    IMyServicePaymentsApi paymentsApi,  
    TestContext testContext,  
    UserHelper userHelper)
```

```
{
```

```
    _userApi = userApi;  
    _paymentsApi = paymentsApi;  
    _testContext = testContext;  
    _userHelper = userHelper;
```

```
}
```

```
public PayResult LoginAndCreateOrderAndPay(double sum, string currency = "Euro")...
```

```
public PayResult Pay(int orderId, PayRequest payRequest)...
```

```
private readonly IMyServiceUserApi _userApi;
```

```
private readonly IMyServicePaymentsApi _paymentsApi;
```

```
private readonly TestContext _testContext;
```

```
private readonly UserHelper _userHelper;
```

```
}
```

TestServices

79/41

```
public class PaymentHelper
```

```
{
```

```
public class PaymentTestService
```

```
    IMyServicePaymentsApi paymentsApi,  
    TestContext testContext,  
    UserHelper userHelper)
```

```
{
```

```
    _userApi = userApi;  
    _paymentsApi = paymentsApi;  
    _testContext = testContext;  
    _userHelper = userHelper;
```

```
}
```

```
public PayResult LoginAndCreateOrderAndPay(double sum, string currency) ...
```

```
public PayResult LoginAndCreateOrderAndPayWithEuro(double sum) ...
```

```
private readonly TestContext _testContext;  
private readonly UserHelper _userHelper;
```

```
}
```

TestServices

80/41

```
public class PaymentHelper
```

```
{
```

```
    public class PaymentTestService
```

```
    {  
        IMyServicePaymentsApi paymentsApi
```

```
        public PaymentTestService(  
            IUserApi userApi,
```

```
            IPaymentsApi paymentApi) { ... }
```

```
        public PayResult LoginAndCreateOrderAndPay(double sum, string currency) ...
```

```
        public PayResult LoginAndCreateOrderAndPayWithEuro(double sum) ...
```

```
        private readonly TestContext _testContext;
```

```
        private readonly UserHelper _userHelper;
```

```
    }
```


Register test services

```
using Castle.Windsor;  
using MyService.AdminApi;  
using MyService.Helpers.Services;  
using MyService.Interfaces;  
using MyService.Models;
```

```
namespace MyService
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
{
```

```
container.Register(  
    Component  
        .For<PaymentTestService>()  
        .ImplementedBy<PaymentTestService>()  
        .LifestyleTransient(),  
    Component  
        .For<UserHelper>()  
        .ImplementedBy<UserHelper>()  
        .LifestyleTransient();  
}
```

№3 «Все подряд» в базовом классе

82/41

```
protected IUserApi
    CreateUserAndLogin() ...

protected PayResult
    LoginAndCreateOrderAndPay(double sum, string currency) ...

protected PayResult
    LoginAndCreateOrderAndPayWithEuro(double sum) ...
```

Nº3 «Все подряд» в базовом классе

83/41

```
protected IUserApi  
    CreateUserAndLogin() ...  
  
protected PayResult  
    LoginAndCreateOrderAndPay(double sum, string currency) ...  
  
protected PayResult  
    LoginAndCreateOrderAndPayWithEuro(double sum) ...
```

Container.Resolve<UserTestService>();

```
[TearDown]
public void CleanUp()
{
    AdminApi.DeleteUser(User);
}
```

```
[TearDown]
public void CleanUp()
{
    UserApi.Logout();
}
```

№4 Не везде выполняется очистка

```
[TearDown]
public void CleanUp()
{
    AdminApi.DeleteUser(User);
}
```

```
[TearDown]
public void CleanUp()
{
    UserApi.Logout();
}
```

Dispose scope

86/41

4 references | Artem Sidoruk, 1 day ago | 2 authors, 4 changes | 2 work items

`public abstract class TestBase`

`{`

`[SetUp]`

`public void BaseSetUp()`

`{`

`Container.BeginScope();`

`}`

7 references | 3/3 passing | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

`protected IMyServiceAdminApi AdminApi => Container.Resolve<IMyServiceAdminApi>();`

12 references | 6/6 passing | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

`protected IMyServiceUserApi UserApi => Container.Resolve<IMyServiceUserApi>();`

0 references | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

`protected IMyServicePaymentsApi PaymentsApi => Container.Resolve<IMyServicePaymentsApi>();`

Dispose scope

87/41

4 references | Artem Sidoruk, 1 day ago | 2 authors, 4 changes | 2 work items

`public abstract class TestBase`

`{`


`[SetUp]`

`public void BaseSetUp()`


`{`

`TestScope = Container.BeginScope();`

`}`

7 references |  3/3 passing | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

`protected IMyServiceAdminApi AdminApi => Container.Resolve<IMyServiceAdminApi>();`

12 references |  6/6 passing | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

`protected IMyServiceUserApi UserApi => Container.Resolve<IMyServiceUserApi>();`

0 references | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

`protected IMyServicePaymentsApi PaymentsApi => Container.Resolve<IMyServicePaymentsApi>();`

Dispose scope

88/41

4 references | Artem Sidoruk, 1 day ago | 2 authors, 4 changes | 2 work items

```
public abstract class TestBase
```

```
{
```

```
[SetUp]
```

```
public void TestScope()
{
    private IDisposable TestScope
    {
        get => ...GetTestContextProperty...
        set => ...SetTestContextProperty...
    }
}
```

7 references

```
protected
```

12 references | 6/6 passing | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

```
protected IMyServiceUserApi UserApi => Container.Resolve<IMyServiceUserApi>();
```

0 references | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

```
protected IMyServicePaymentsApi PaymentsApi => Container.Resolve<IMyServicePaymentsApi>();
```


Dispose scope

89/41

4 references | Artem Sidoruk, 1 day ago | 2 authors, 4 changes | 2 work items

`public abstract class TestBase`

`{`

`[SetUp]`

`private IDisposable TestScope`

`{`

`{`

`[TearDown]`

`public void BaseCleanup()`

`{`

`}`

`TestScope.Dispose();`

`}`

7 references

`protected`

12 references | 6/6 passed

`protected IMySer`

0 references | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

`protected IMyServicePaymentsApi PaymentsApi => Container.Resolve<IMyServicePaymentsApi>();`

Implement IDisposable

90/41

```
namespace MyService.UserApi
```

```
{
```

```
public class MyServiceUserApi : IUserApi, IDisposable
```

```
{
```

```
    _client = httpCLeint;
```

```
    _baseUrl = baseUrl;
```

```
}
```

5 references | 3/3 passing | Artem Sidoruk, 2 days ago | 1 author, 1 change

```
public UserSession Login(UserInfo user)...
```

5 references | 4/4 passing | Artem Sidoruk, 2 days ago | 1 author, 1 change

```
public Order CreateOrder()...
```

3 references | 2/2 passing | Artem Sidoruk, 2 days ago | 1 author, 1 change

```
public Order GetOrder(int orderNumber)...
```

3 references | 2/2 passing | Artem Sidoruk, 2 days ago | 1 author, 1 change

```
public void DeleteOrder(int orderNumber)...
```

1 reference | Artem Sidoruk, 1 day ago | 1 author, 1 change

```
public void Logout()...
```

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

```
void IDisposable.Dispose()
```

```
{
```

```
    Logout();
```

```
}
```

5 references | Artem Sidoruk, 2 days ago | 1 author, 1 change

```
public string BaseUrl => $"{_baseUrl}/user";
```

```
private readonly string _baseUrl;
```

Implement IDisposable

91/41

```
namespace MyService.UserApi
```

```
{
```

```
public class MyServiceUserApi : IUserApi, IDisposable
```

```
{
```

```
    {  
        client = httpCleint;
```

```
void IDisposable.Dispose()
```

```
{
```

```
    Logout();
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change
```

```
void IDisposable.Dispose()
```

```
{
```

```
    Logout();
```

```
}
```

```
5 references | Artem Sidoruk, 2 days ago | 1 author, 1 change
```

```
public string BaseUrl => $"{_baseUrl}/user";
```

```
private readonly string _baseUrl;
```

Implement IDisposable

92/41

```
namespace MyService.UserApi
```

```
{
```

```
public class MyServiceUserApi : IUserApi, IDisposable
```

```
{
```

```
    client = httpCleint;
```

```
void IDisposable.Dispose()
```

```
{
```

```
}
```

```
void IDisposable.Dispose()
```

```
{
```

```
    _adminApi.DeleteUser(_user.Login);
```

```
}
```

```
0 references | API
```

```
void IDisposable
```

```
{
```

```
    Logout()
```

```
}
```

```
5 references | Artem Sidoruk, 2 days ago | 1 author, 1 change
```

```
public string BaseUrl => $"{_baseUrl}/user";
```

```
private readonly string _baseUrl;
```

№4 Не везде выполняется очистка

93/41

```
[TearDown]
```

```
public void CleanUp()
```

```
{
```

```
Ad
```

```
}
```

```
[TearDown]
```

```
public void CleanUp()
```

```
{
```

```
UserApi.Logout();
```

```
}
```

~~Nº4 Не везде выполняется очистка~~

```
[TearDown]
```

```
public void Cleanup()
```

```
{
```

```
Ad
```

```
}
```

```
[TearDown]
```

```
public void Cleanup()
```

```
{
```

```
UserApi.Logout();
```

```
}
```

TestScope.Dispose();

Tests

95/41

Test	Duration	Tra
▲ ✓ MyService.IntegrationTests (16)	2,1 min	
▲ ✓ MyService.IntegrationTests.Tests (16)	2,1 min	
▷ ✓ AdminApiTests (3)	15,6 sec	
▷ ✓ LoginTests (3)	20,8 sec	
▷ ✓ OrderTests (3)	22 sec	
▷ ✓ PaymentTests (7)	1,2 min	

DI Container != Service Locator

96/41

[TearDown]

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

```
protected IWindsorContainer Container =>
    _container ??= new ContainerInitializer()
        .GetContainer();
```

```
private static IWindsorContainer _container;
```

```
protected PaymentHelper PaymentHelper => Container.Resolve<PaymentHelper>();
```

0 references | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

```
protected UserHelper UserHelper => Container.Resolve<UserHelper>();
```

5 references | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

```
protected TestContext TestContext => Container.Resolve<TestContext>();
```

7 references | Sugrob57, 2 days ago | 1 author, 1 change | 1 work item

```
protected IWindsorContainer Container => _container ??= new ContainerInitializer().GetContainer();
```

```
private static IWindsorContainer _container;
```


DI Container != Service Locator

97/41

[TearDown]

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

```
protected IWindsorContainer Container =>
    _container ??= new ContainerInitializer()
        .GetContainer();
```

```
public static
private static IWindsorContainer _container;
```

```
protected PaymentHelper PaymentHelper => Container.Resolve<PaymentHelper>();
```

0 references | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

```
protected UserHelper UserHelper => Container.Resolve<UserHelper>();
```

```
TestBase._container.Resolve<...>
```

```
private static IWindsorContainer _container;
```

DI Container != Service Locator

98/41

[TearDown]

0 references | Artem Sidoruk, 1 day ago | 1 author, 1 change

```
protected IWindsorContainer Container =>
    _container ??= new ContainerInitializer()
        .GetContainer();
```

```
public static
private static IWindsorContainer _container;
```

```
protected PaymentHelper PaymentHelper => Container.Resolve<PaymentHelper>();
```

0 references | Sugrob57, 1 day ago | 1 author, 1 change | 1 work item

```
protected UserHelper UserHelper => Container.Resolve<UserHelper>();
```

```
TestBase._container.Resolve<...>
```

```
private static IWindsorContainer _container;
```

Выводы

- Явные вызовы конструкторов
- Хранение компонентов
- Большой базовый класс
- Очистка после тестов



Арте́м
Сидору́к



t.me@Sugrob57



Вакансии QA в Kaspersky



https://github.com/Sugrob57/DI_in_Tests_Demo