

# C# Professional

Async & Await

# C# Starter

Автор курса



Александр Шевчук  
MCT



MCID: 9230440

# C# Professional

После урока обязательно



Повторите этот урок в видео формате на [ITVDN.com](http://itvdn.com)

Доступ можно получить через руководство вашего учебного центра



Проверьте как Вы усвоили данный материал на [TestProvider.com](http://testprovider.com)

# C# Professional

Tema

## Async & Await

# Async & Await

## C# 5.0 - What's new?

C# 2 = Generics, Lambda

C# 3 = var, LINQ

C# 4 = dynamic, TPL

C# 5 = Async

# Async & Await

## Синхронность: проблемы

```
private void GetButtonClick(object sender, RoutedEventArgs e)
{
    var req= (HttpWebRequest) WebRequest.Create("http://microsoft.com/");

    .....

    var resp= (HttpWebResponse) req.GetResponse();

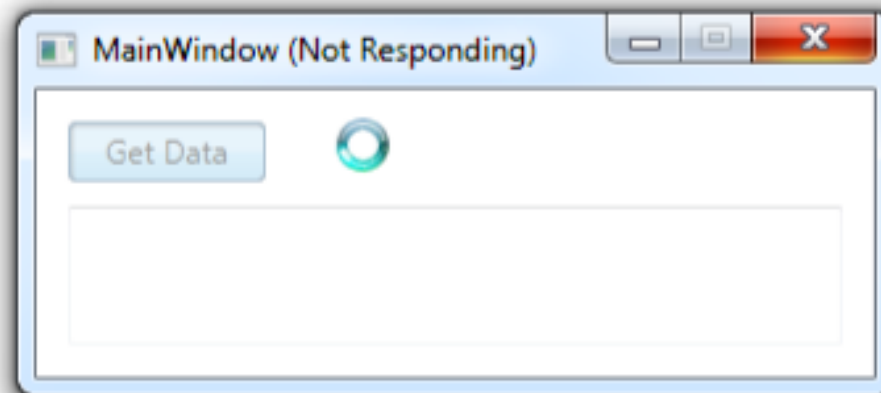
    dataTextBox.Text+= resp.Headers.ToString();
}
```



# Async & Await

## Синхронность: проблемы

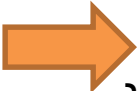
Вызывающий поток блокируется, до завершения длительной операции



# Async & Await

## Асинхронная модель до C# 5

```
var req = (HttpWebRequest)WebRequest.Create("http://www.google.com");  
  
req.Method = "GET";  
  
req.BeginGetResponse((asyncResult) =>  
    {  
        var resp = (HttpWebResponse)req.EndGetResponse(asyncResult);  
  
        string headersText = resp.Headers.ToString();  
  
        dataTextBox.Text += headersText;  
    },  
    null);
```





# Async & Await

## Асинхронная модель до C# 5

Попытка обновления UI из другого потока

The screenshot illustrates a cross-thread UI update attempt in a .NET application. The top thread dump shows the Main Thread (ID 6904) and several Worker Threads (IDs 13524, 7232, 11552). The code editor below shows the `AsyncFetch.MainWindow` class with a `GetButtonClick` event handler. An orange arrow points to line 24, where `dataTextBox.Text += headersText;` is executed. A red box highlights the entire interface. On the right, an exception message states: **InvalidOperationException was unhandled by user code**. The description reads: "The calling thread cannot access this object because a different thread owns it." Troubleshooting tips include a link to "Get general help for this exception." and a "Search for more Help Online..." link.

Thread ID	Priority	Thread Name	Parent Thread Name	Exception Message	Exception Type
13524	9	Worker Thread	.NET SystemEvents	[Managed to Native Transition]	Normal
7232	14	Worker Thread	Worker Thread	AsyncFetch.MainWindow.GetButtonClick.AnonymousMethod_0	Normal
6904	10	Main Thread	Main Thread	MS.Internal.DoubleUtil.RectHasNaN()	Normal
11552	3	Worker Thread	<No Name>	System.Threading.WaitHandle.WaitAny()	Normal

```
18 req.Method = GET;
19 req.BeginGetResponse(
20     (asyncResult) =>
21     {
22         var resp = (HttpWebResponse)req.EndGetResponse();
23         string headersText = resp.Headers.ToString();
24         dataTextBox.Text += headersText;
25     },
```

**InvalidOperationException was unhandled by user code**

The calling thread cannot access this object because a different thread owns it.

**Troubleshooting tips:**

[Get general help for this exception.](#)

[Search for more Help Online...](#)

# Async & Await

## Асинхронная модель: Обновление UI

Обновление UI из другого потока через [SynchronizationContext](#)

```
var sync = SynchronizationContext.Current;

req.BeginGetResponse(asyncResult=>
{
    var resp = (HttpWebResponse)req.EndGetResponse(asyncResult);

    sync.Post(delegate
    {
        // ОБНОВЛЕНИЕ UI
    },
    null);
},
null);
```

# Async & Await

## Асинхронная простота

Ключевое слово `async` указывает компилятору, что метод является асинхронным.

```
async void getButton_Click(object sender, RoutedEventArgs e)
{
    var w = new WebClient();

    string txt = await w.DownloadStringTaskAsync("...");

    dataTextBox.Text = txt;
}
```

`await` указывает компилятору, что в этой точке необходимо дождаться окончания асинхронной операции (при этом управление возвращается вызвавшему методу).

# Async & Await

## Асинхронная простота

```
async void DoDownloadAsync()
{
    using (var w = new WebClient())
    {
        string txt = await w.DownloadStringTaskAsync("http://www.microsoft.com/");
        dataTextBox.Text = txt;
    }
}

void DoDownload()
{
    using (var w = new WebClient())
    {
        string txt = w.DownloadString("http://www.microsoft.com/");
        dataTextBox.Text = txt;
    }
}
```

# Async & Await

## Исключения

Удобная обработка исключений

The screenshot displays the Visual Studio IDE with a task list at the top showing threads: Worker Thread (ID 13056, 8), Worker Thread (ID 9940, 9), and Main Thread (ID 9648, 10). The 'Parallel Tasks' window is open, showing the execution of 'AsyncFetch.MainWindow.DoDownloadAsync'. The code editor shows the following C# code:

```
18 DoDownloadAsync();  
19 Debug.WriteLine("DoDownload done");  
20 }  
21  
22 async void DoDownloadAsync()  
23 {  
24     using (var w = new WebClient())  
25     {  
26         string txt = await w.DownloadStringTaskAsync("http://www.microsoft.com/");  
27         dataTextBox.Text = txt;  
28     }  
29 }
```

An exception dialog box is open, displaying the error: **WebException was unhandled by user code**. The message states: "The remote name could not be resolved: 'www.microsoft.com'". It includes troubleshooting tips: "Check the Status property of the exception to determine what the error was.", "Check the Response property of the exception to determine the response from the server.", and "Get general help for this exception." There is also a link to "Search for more Help Online...". Under "Exception settings", the checkbox "Break when this exception type is user-unhandled" is checked.

Исключения «выбрасываются» в месте вызова асинхронной операции, а не Callback-метода!

# Async & Await

## Асинхронность и Многопоточность

`Task.WhenAll()` , `Task.WhenAny()`

```
var wc1 = new WebClient();  
var wc2 = new WebClient();  
  
Task<string> task1 = wc1.DownloadStringTaskAsync(url1);  
Task<string> task2 = wc2.DownloadStringTaskAsync(url2);  
  
string[] results = await Task.WhenAll(task1, task2);
```

# Смотрите наши уроки в видео формате

ITVDN.com



Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://itvdn.com) для закрепления пройденного материала.

Все курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics





# Проверка знаний

## TestProvider.com



The screenshot shows the TestProvider.com website. At the top, there is a navigation bar with the TestProvider logo, a tagline "Мы помогаем людям оценить себя", and links for "Регистрация" and "Войти". Below this is a secondary navigation bar with links: "Главная", "Каталог", "Сертификация Microsoft", "Поддержка", and "О нас". The main content area features a large blue banner with the heading "Тестирование" and subtitle "Языки программирования и информационные технологии". The Microsoft logo is prominently displayed. Below the logo is a list of technologies: C#, ASP.NET, MVC, JavaScript, Patterns, Of, Design, SQL, Architecture, Guide, WCF, HTML&CSS, XML, SEO, WPF, HTML5&CSS3, JQuery, XNA, SharePoint, GUI for Android, Windows, Azure, Platform, Microsoft Patterns&Practices, TFS, SCRUM, ReSharper, TDD, WWF, LINQ, Entity Framework, Windows Forms, Refactoring, Microsoft Expression Blend 4, Windows Phone 8, Windows 8, AppStore, Visual Studio, Tips&Tricks, MSF, MEF, SilverLight, AJAX, MEF, Service Oriented Architecture. A "Пройти тест" button is at the bottom of the banner. To the right of the banner is a photo of a woman smiling while using a laptop. Below the banner, there is a section titled "Наши партнеры" with logos for Microsoft Partner, CyberBionic, ITVDN, PROMETRIC TEST CENTER, PEARSON VUE Authorized Test Center, Windows Azure Cloud Partner, and EBA. At the bottom, a blue bar contains the text "Дополнительные ресурсы:" followed by icons and links for "Очное обучение", "On-line обучение", and "Видео обучение".

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](http://TestProvider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.





## Q&A

# Информационный видеосервис для разработчиков программного обеспечения

