UMBRACOSPARK #SPARKQA

dotnet new umbraco

BY PAUL SEAL

About me

About me

Paul Seal

ClerksWell.

About me

ClerksWell.

Paul Seal

Umbraco Tech Lead

About me

ClerksWell.

Paul Seal

Umbraco Tech Lead

Certified Master

Umbraco MVP x 5

About ClerksWell

ClerksWell.

Umbraco Gold Partners

Paul Seal

Umbraco Tech Lead

About ClerksWell

ClerksWell.

Umbraco Gold Partners

Microsoft Gold Partners

Paul Seal

Umbraco Tech Lead

What we will cover

Basics

dotnet

Umbraco

What we will cover

Basics

dotnet

Um

What we will cover

Basics

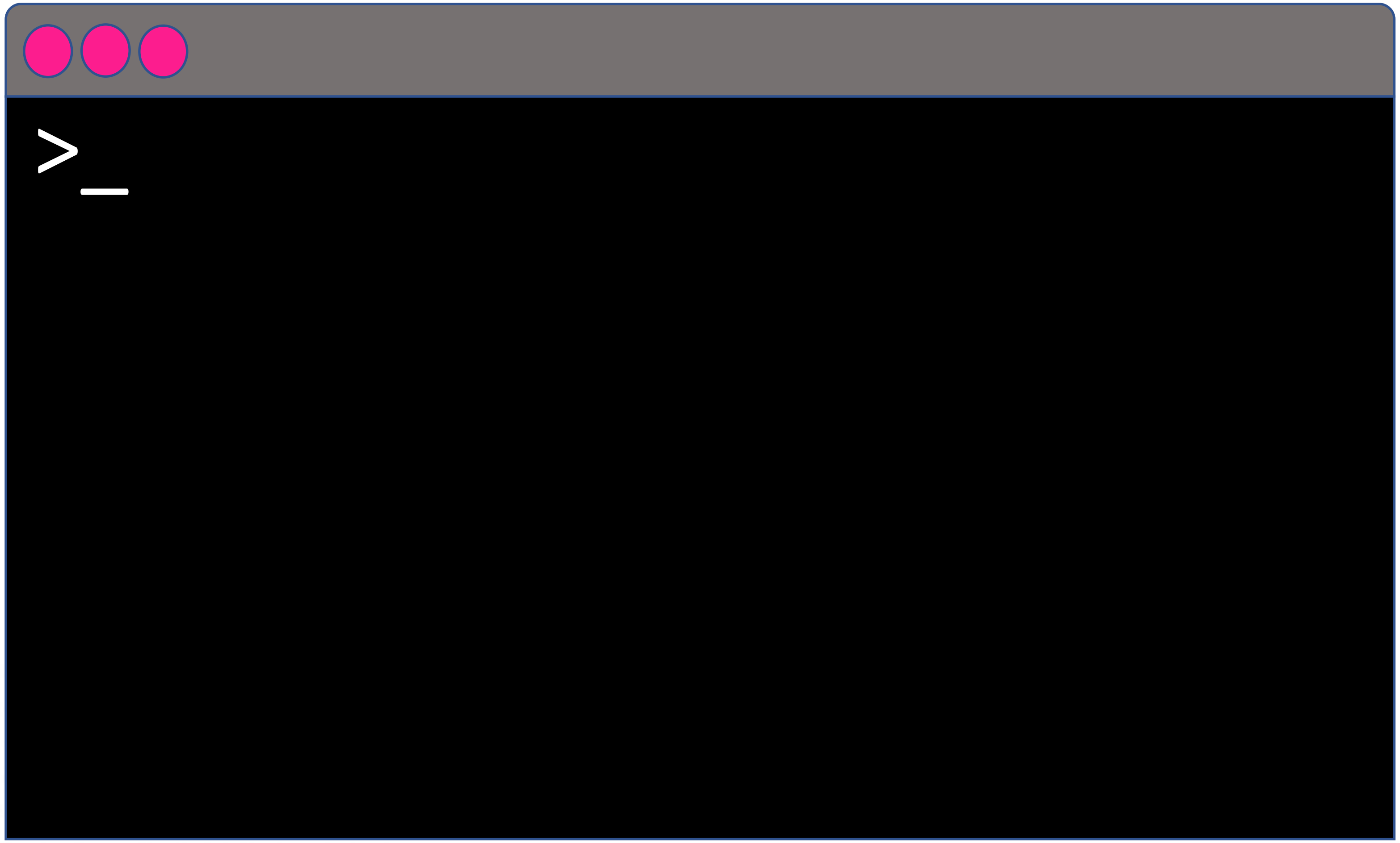
dotnet

Umbrac

What we will cover

Show of hands

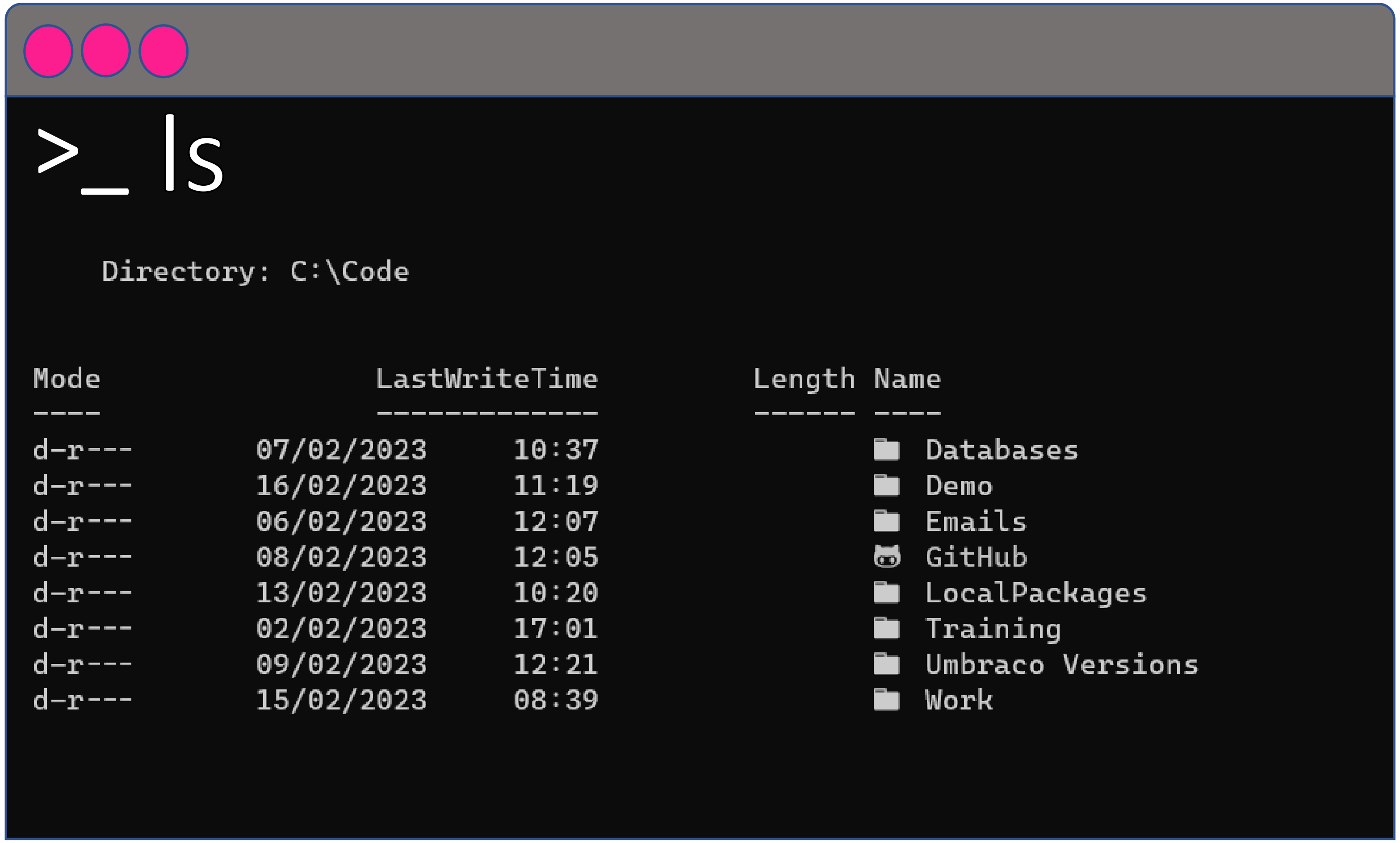
Who already uses the command line?



```
>_
```

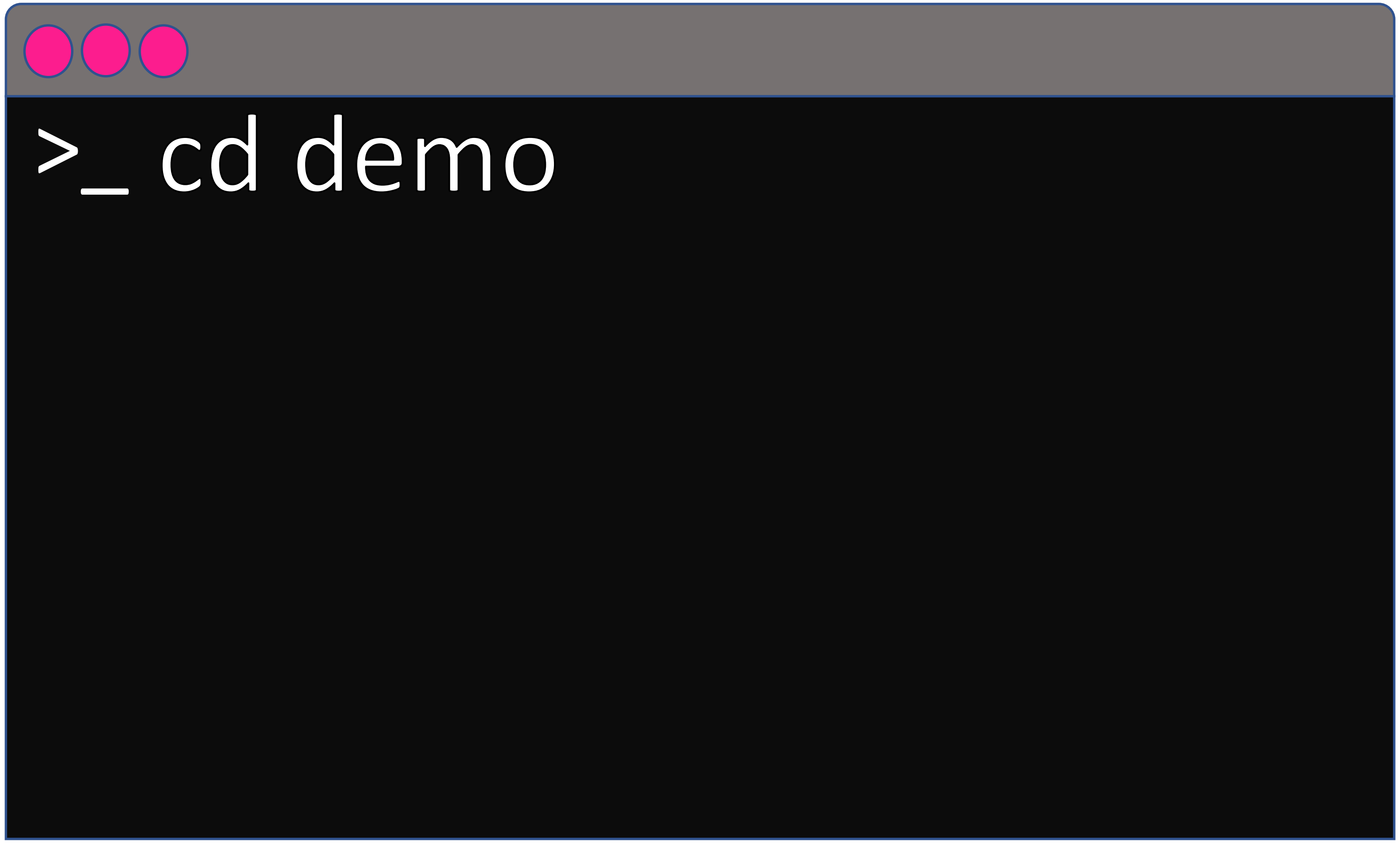
Starting with the basics

List out directory contents



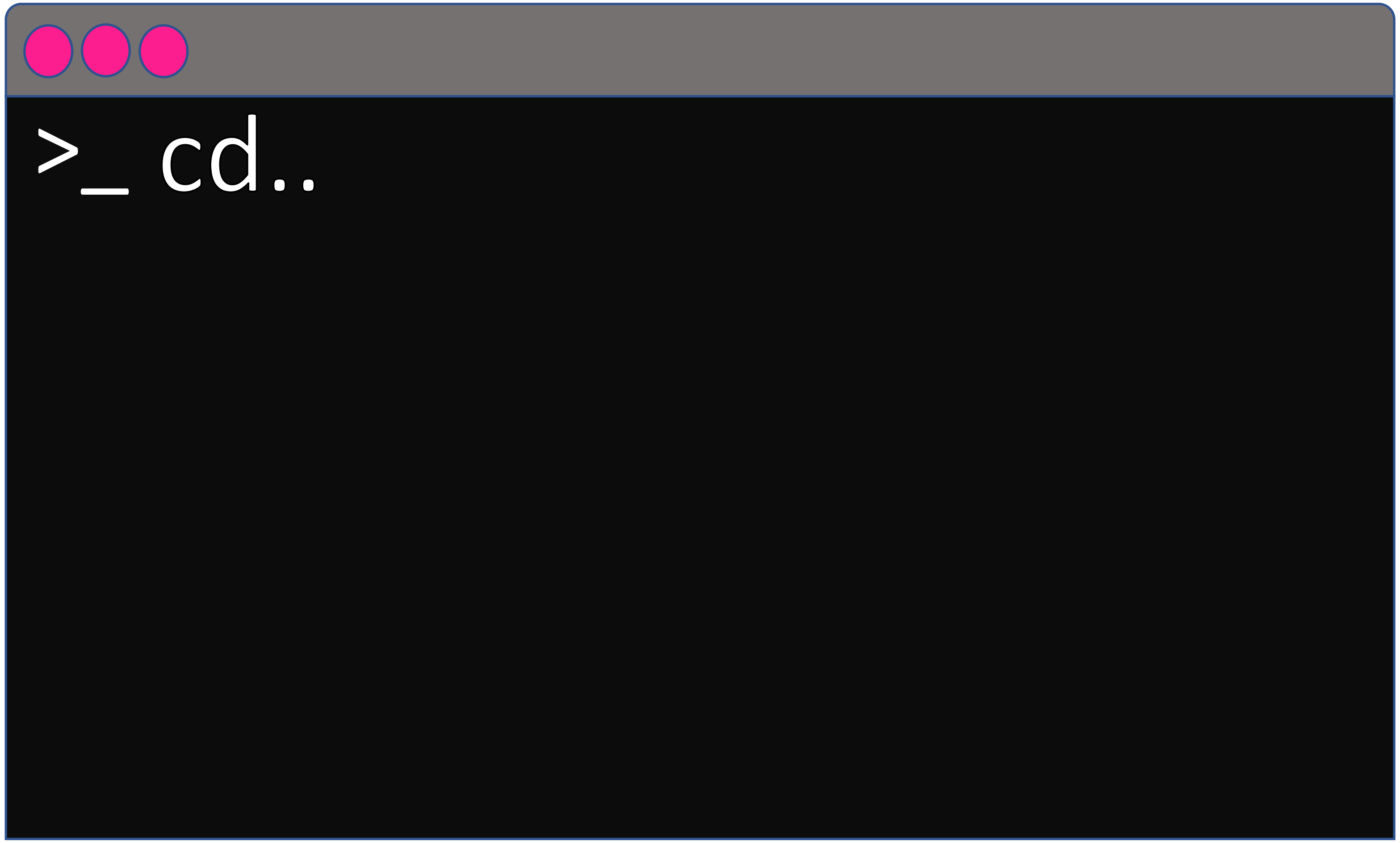
```
>_ ls

    Directory: C:\Code


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-r---       07/02/2023     10:37                Databases
d-r---       16/02/2023     11:19                Demo
d-r---       06/02/2023     12:07                Emails
d-r---       08/02/2023     12:05                GitHub
d-r---       13/02/2023     10:20                LocalPackages
d-r---       02/02/2023     17:01                Training
d-r---       09/02/2023     12:21                Umbraco Versions
d-r---       15/02/2023     08:39                Work
```

Navigate to a directory



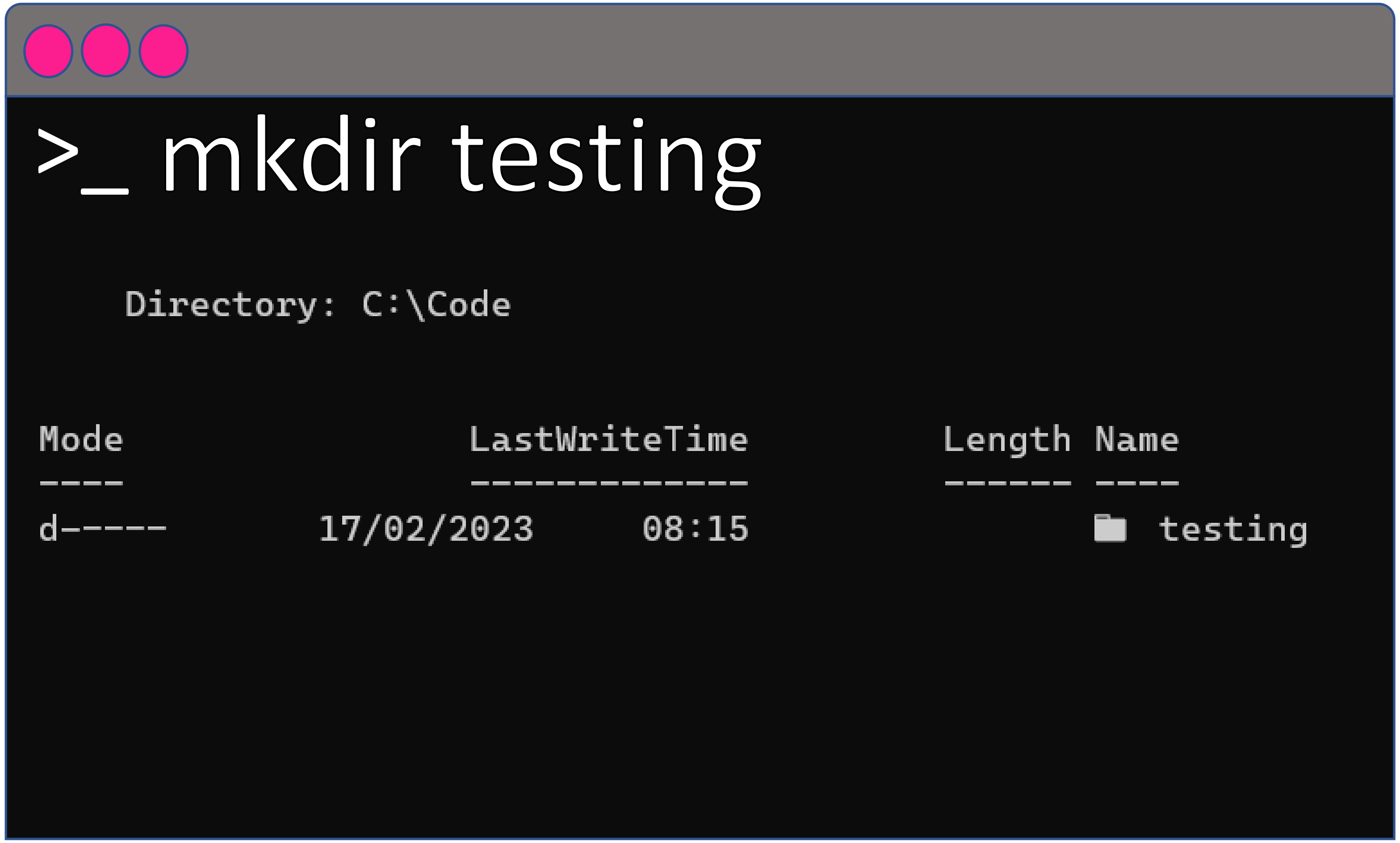
```
>_ cd demo
```

Go back up a level



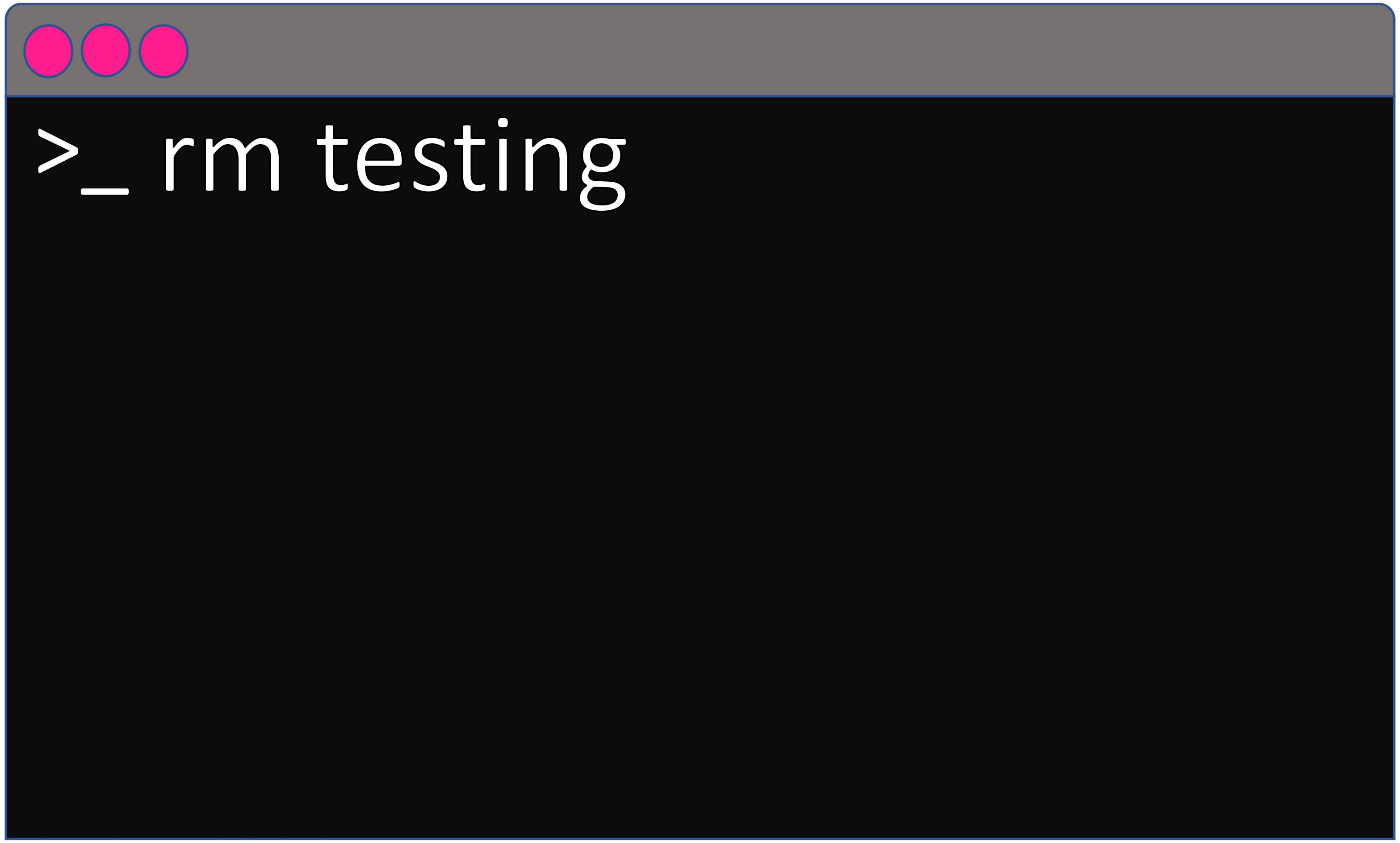
```
>_ cd..
```

Create a directory



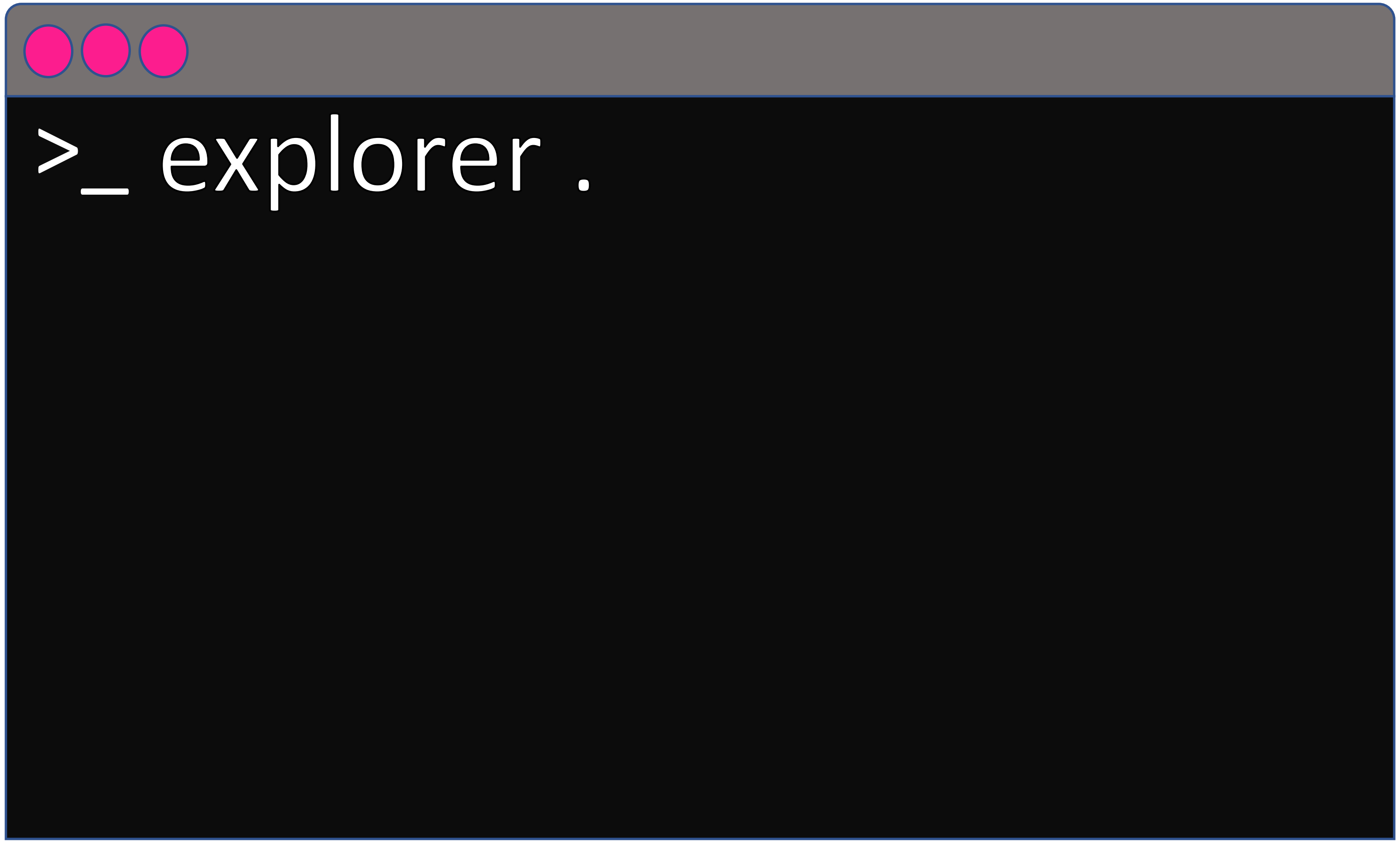
```
>_ mkdir testing

    Directory: C:\Code


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        17/02/2023     08:15                📁 testing
```

Delete a directory



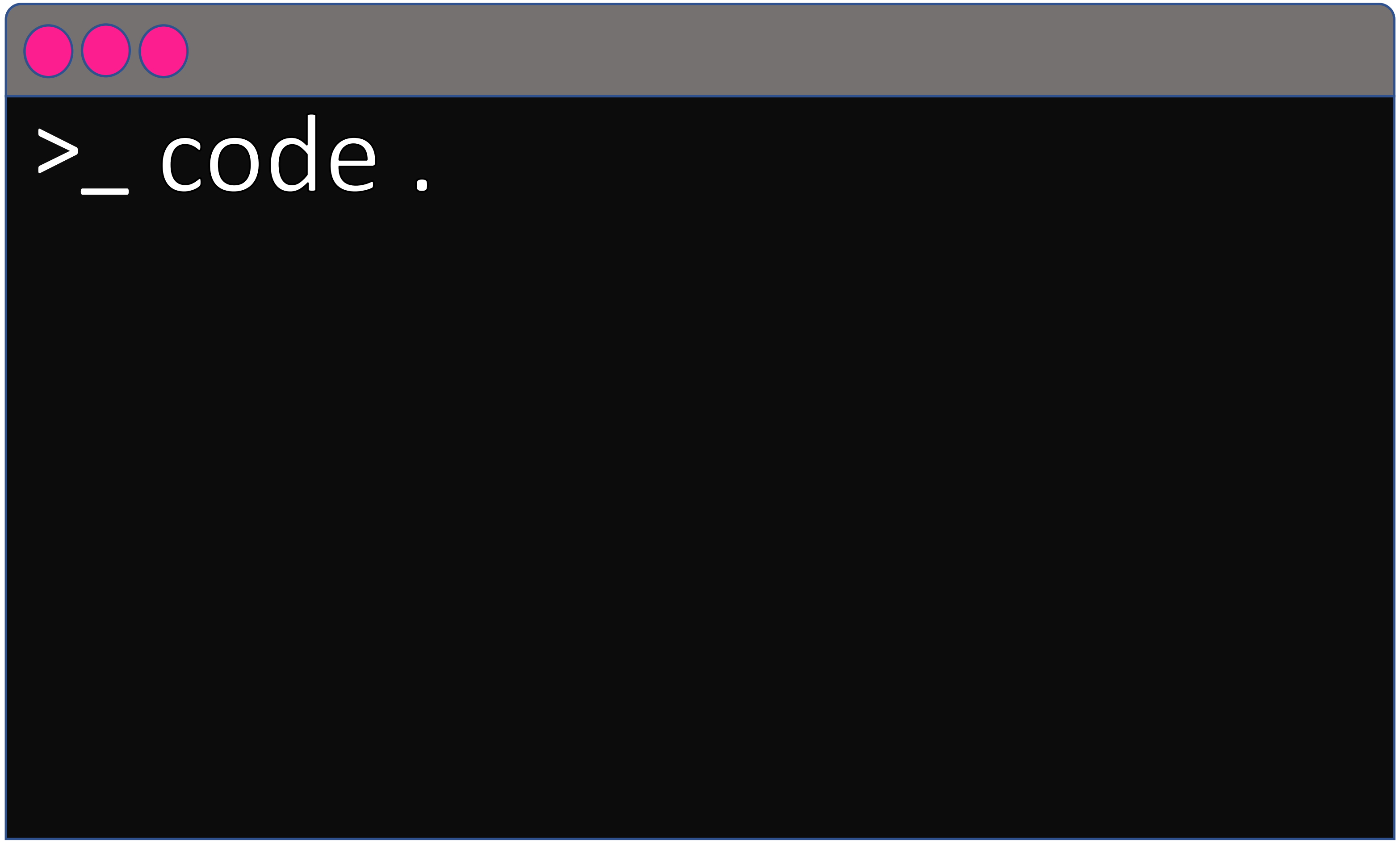
```
>_ rm testing
```

Open the current directory in windows explorer



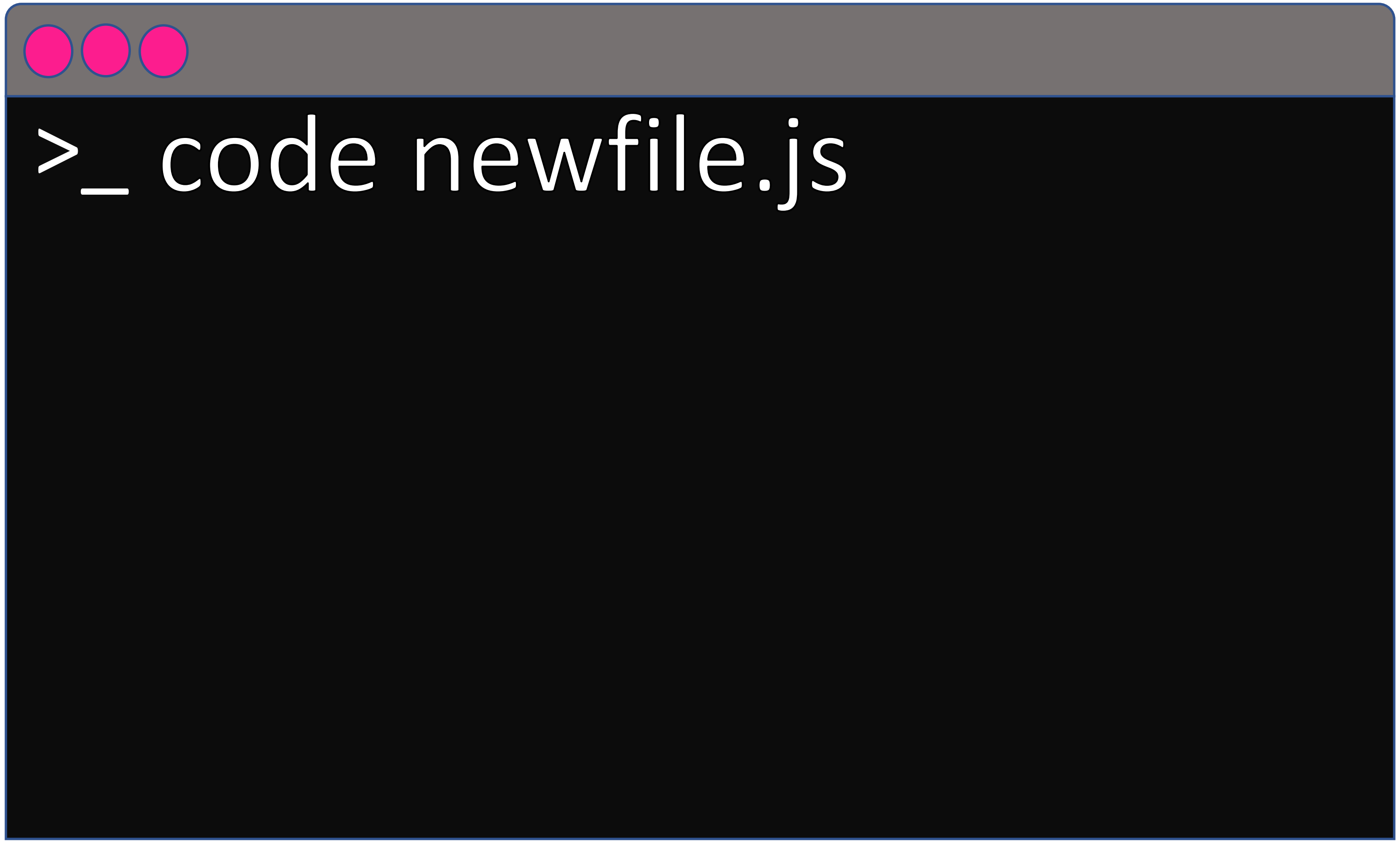
```
>_ explorer .
```

Open the current directory in VS Code



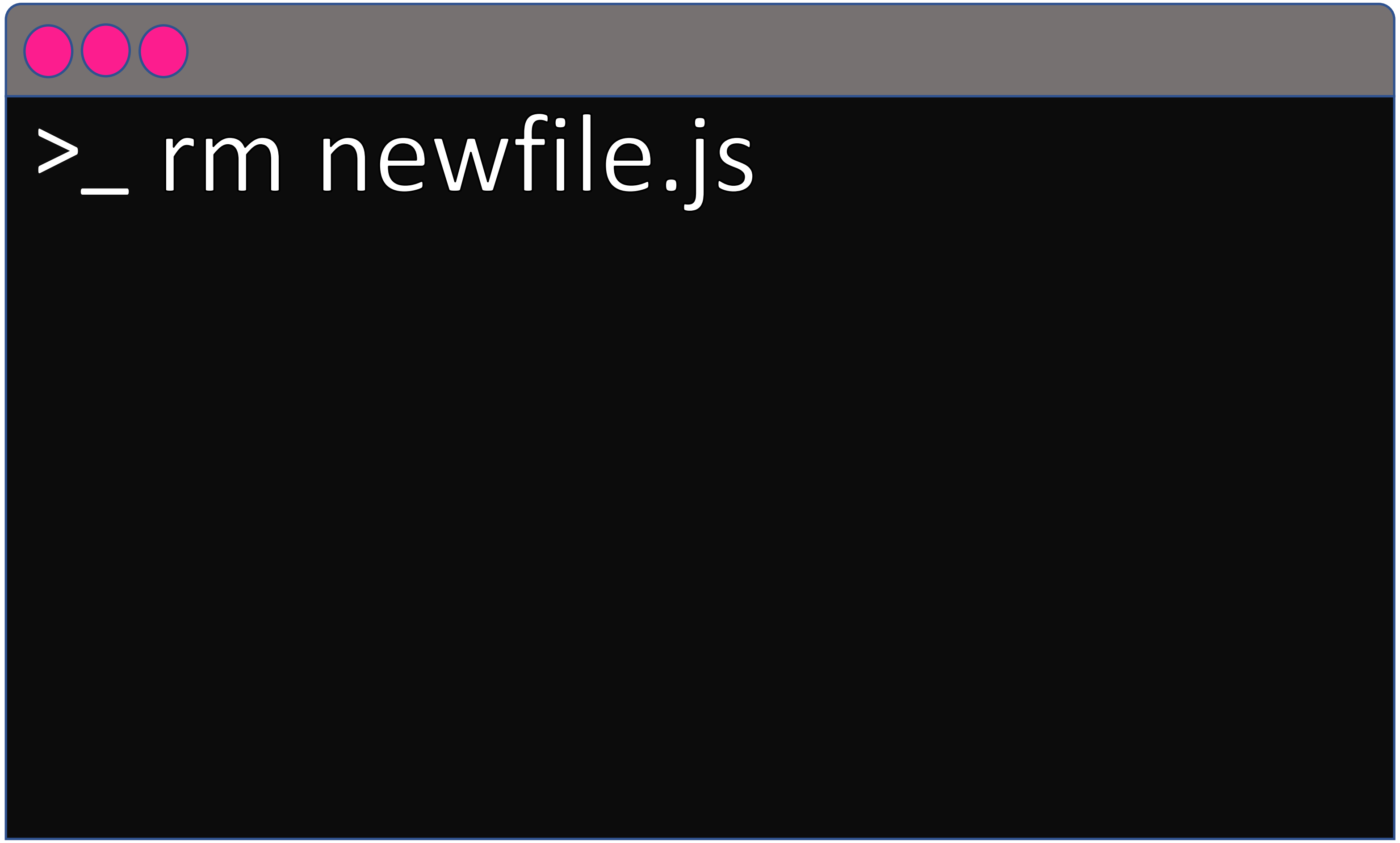
```
>_ code .
```

Create a new file and edit it with VS Code



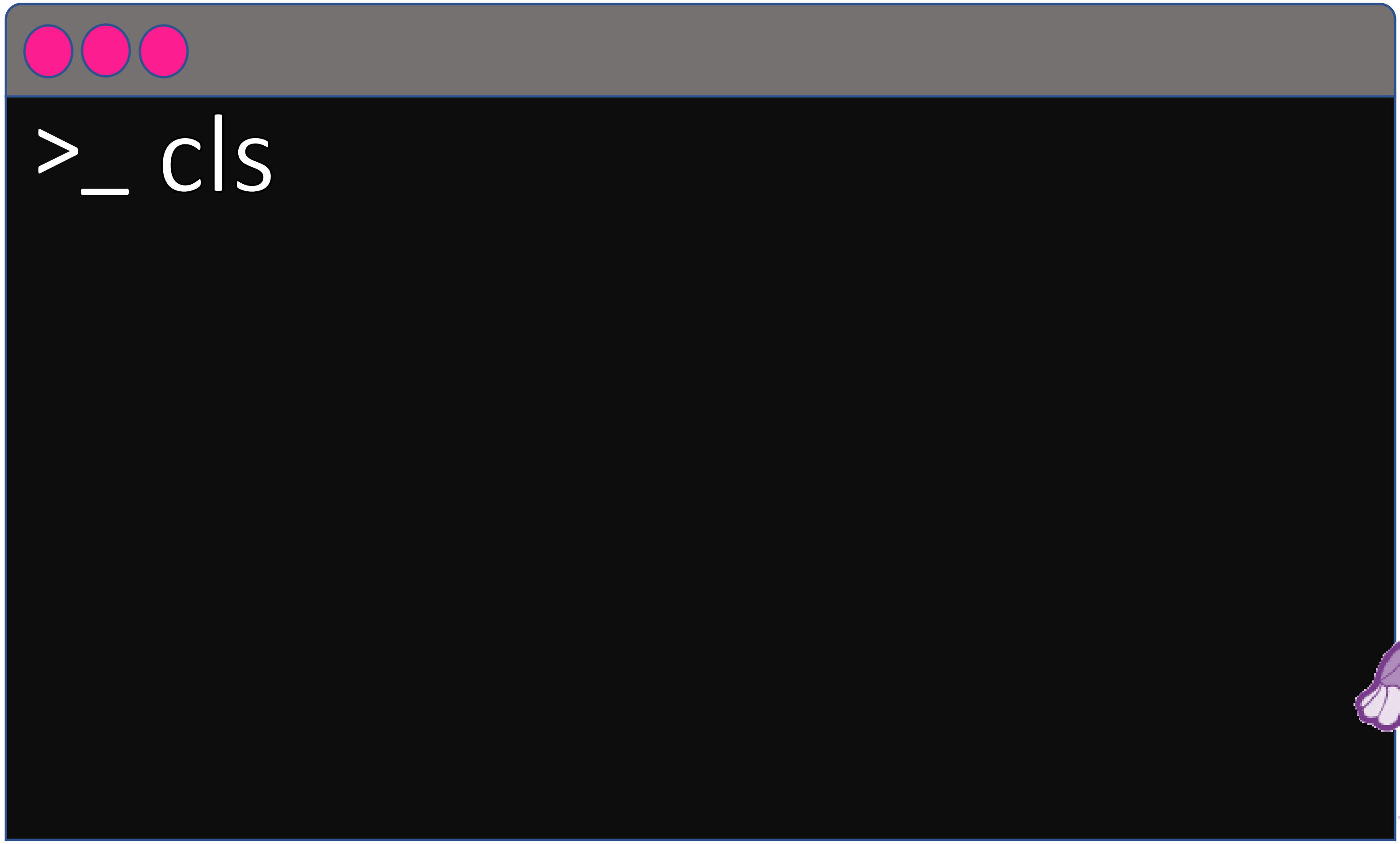
```
>_ code newfile.js
```

Delete a file

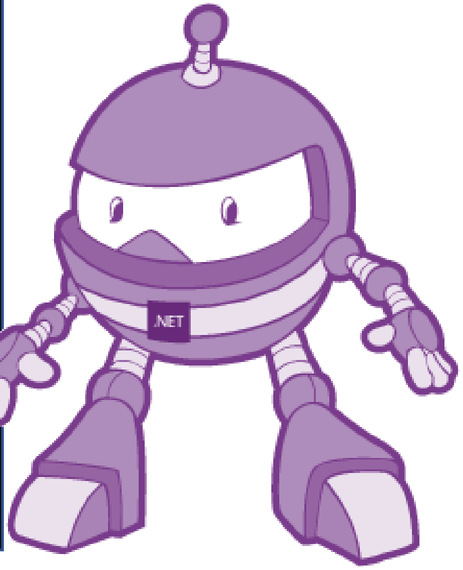


```
>_ rm newfile.js
```

Clear the terminal



```
>_ cls
```



That’s the basics covered

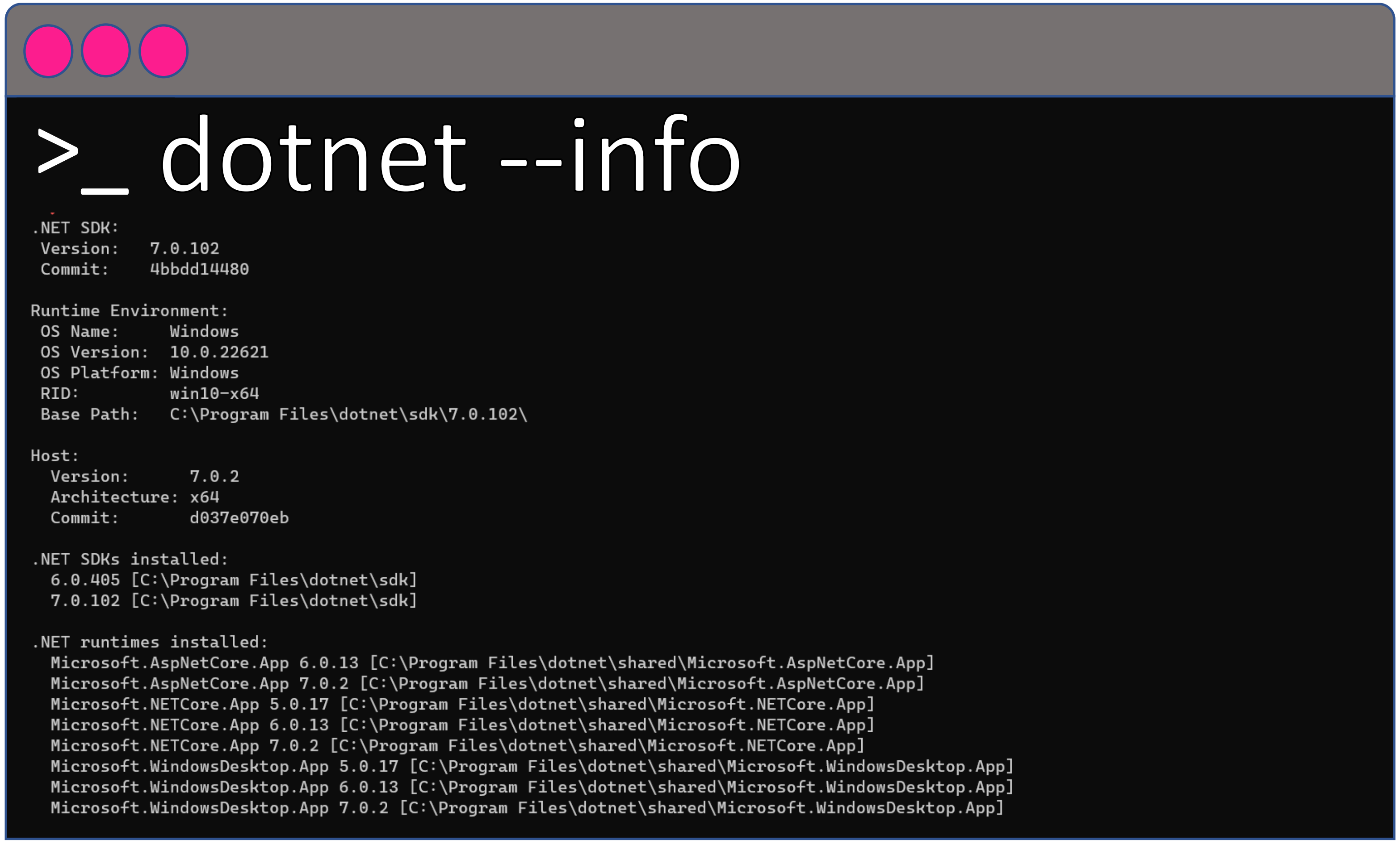
Now lets look at the dotnet CLI

Basics

dotnet

Umbrac

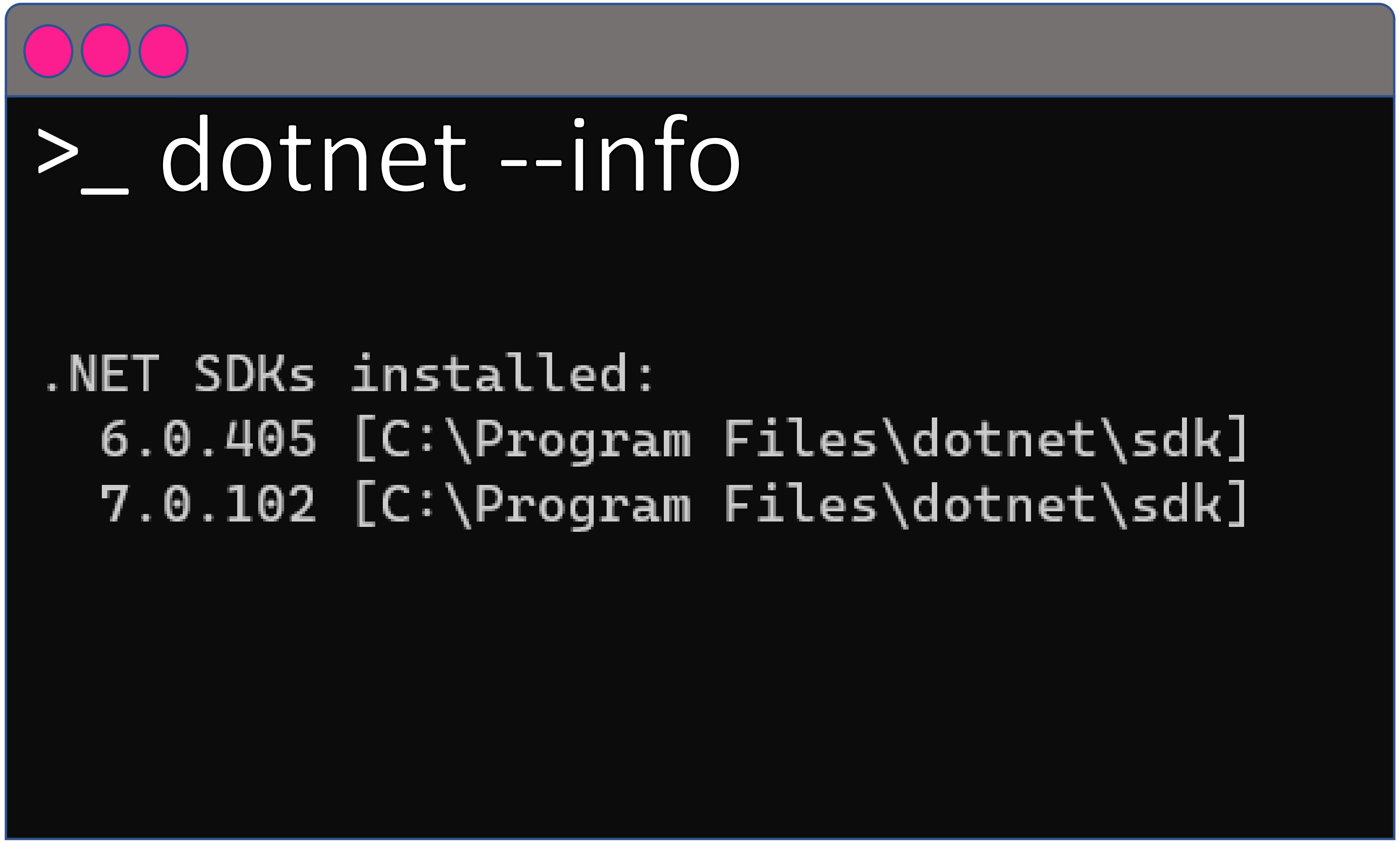
The dotnet CLI



```
>_ dotnet --info
```

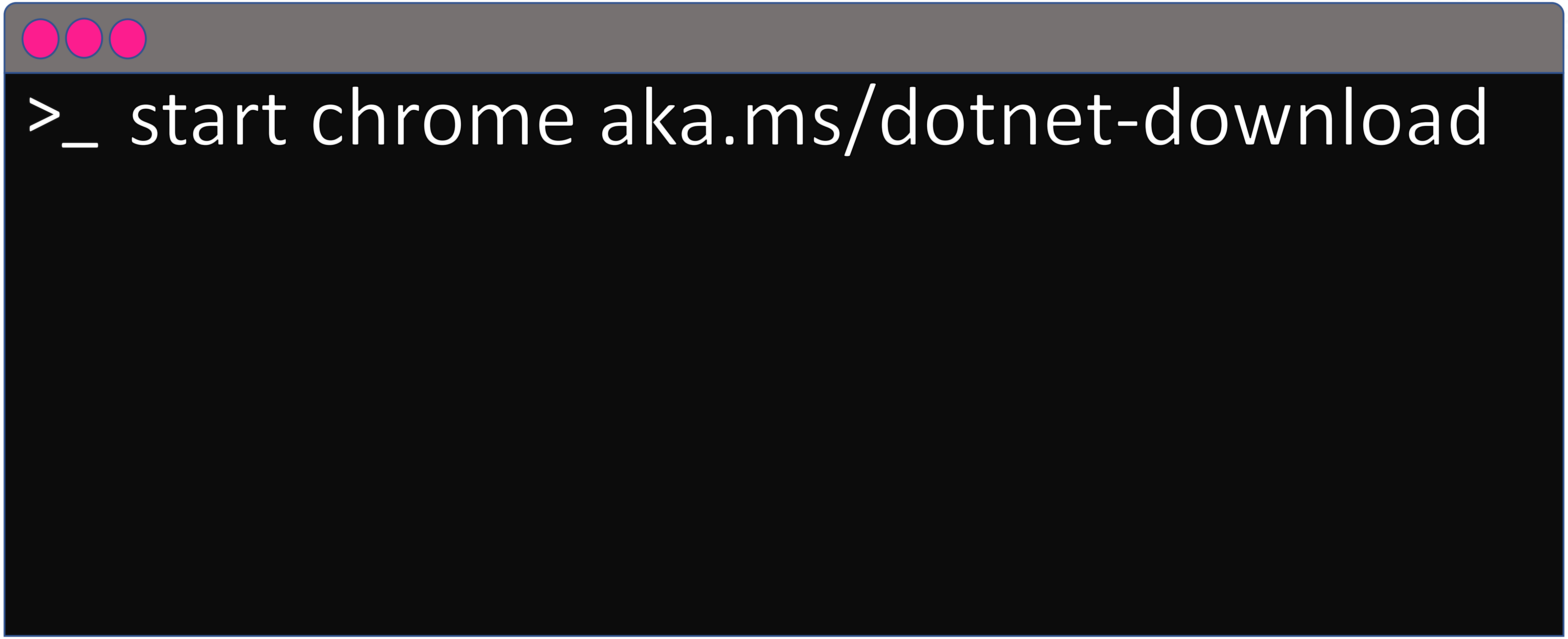
```
.NET SDK:
 Version:   7.0.102
 Commit:    4bbdd14480

Runtime Environment:
 OS Name:     Windows
 OS Version:  10.0.22621
 OS Platform: Windows
 RID:         win10-x64
 Base Path:   C:\Program Files\dotnet\sdk\7.0.102\

Host:
  Version:      7.0.2
  Architecture: x64
  Commit:       d037e070eb

.NET SDKs installed:
  6.0.405 [C:\Program Files\dotnet\sdk]
  7.0.102 [C:\Program Files\dotnet\sdk]

.NET runtimes installed:
  Microsoft.AspNetCore.App 6.0.13 [C:\Program Files\dotnet\shared\Microsoft.AspNetCore.App]
  Microsoft.AspNetCore.App 7.0.2 [C:\Program Files\dotnet\shared\Microsoft.AspNetCore.App]
  Microsoft.NETCore.App 5.0.17 [C:\Program Files\dotnet\shared\Microsoft.NETCore.App]
  Microsoft.NETCore.App 6.0.13 [C:\Program Files\dotnet\shared\Microsoft.NETCore.App]
  Microsoft.NETCore.App 7.0.2 [C:\Program Files\dotnet\shared\Microsoft.NETCore.App]
  Microsoft.WindowsDesktop.App 5.0.17 [C:\Program Files\dotnet\shared\Microsoft.WindowsDesktop.App]
  Microsoft.WindowsDesktop.App 6.0.13 [C:\Program Files\dotnet\shared\Microsoft.WindowsDesktop.App]
  Microsoft.WindowsDesktop.App 7.0.2 [C:\Program Files\dotnet\shared\Microsoft.WindowsDesktop.App]
```

The dotnet CLI



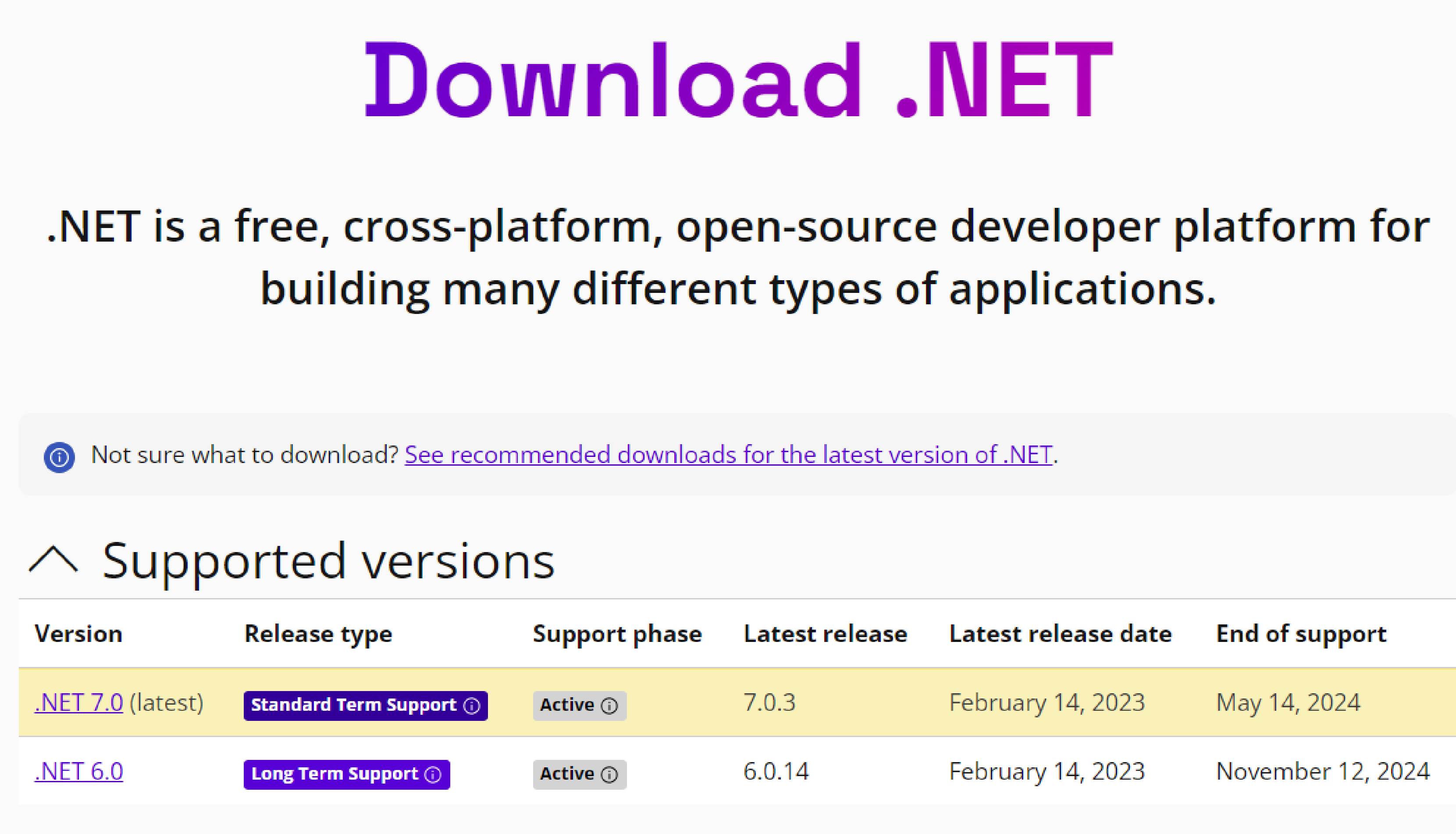
```
>_ dotnet --info


.NET SDKs installed:
  6.0.405 [C:\Program Files\dotnet\sdk]
  7.0.102 [C:\Program Files\dotnet\sdk]
```

Download the dotnet SDK



```
>_ start chrome aka.ms/dotnet-download
```

Download the dotnet SDK



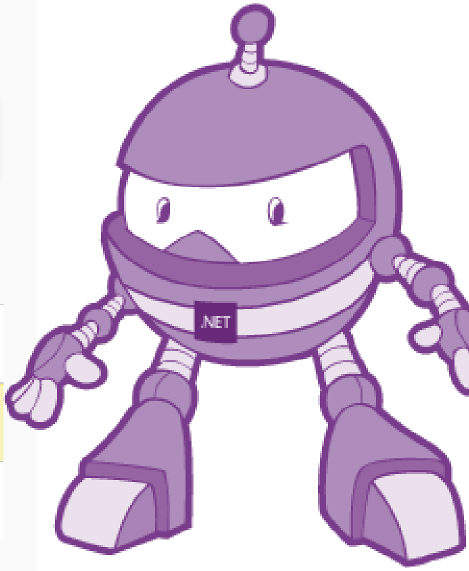
Download .NET

.NET is a free, cross-platform, open-source developer platform for building many different types of applications.

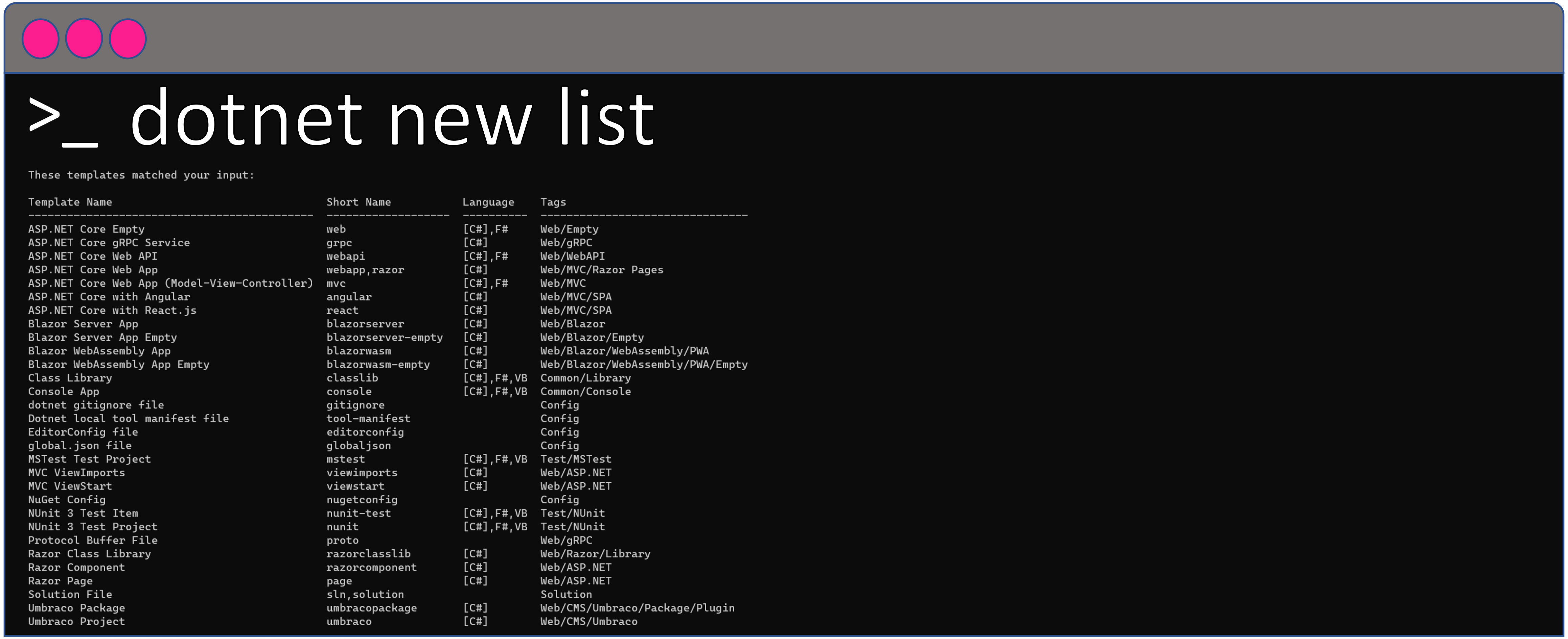
ⓘ Not sure what to download? See recommended downloads for the latest version of .NET.

Supported versions

Version	Release type	Support phase	Latest release	Latest release date	End of support
.NET 7.0 (latest)	Standard Term Support ⓘ	Active ⓘ	7.0.3	February 14, 2023	May 14, 2024
.NET 6.0	Long Term Support ⓘ	Active ⓘ	6.0.14	February 14, 2023	November 12, 2024

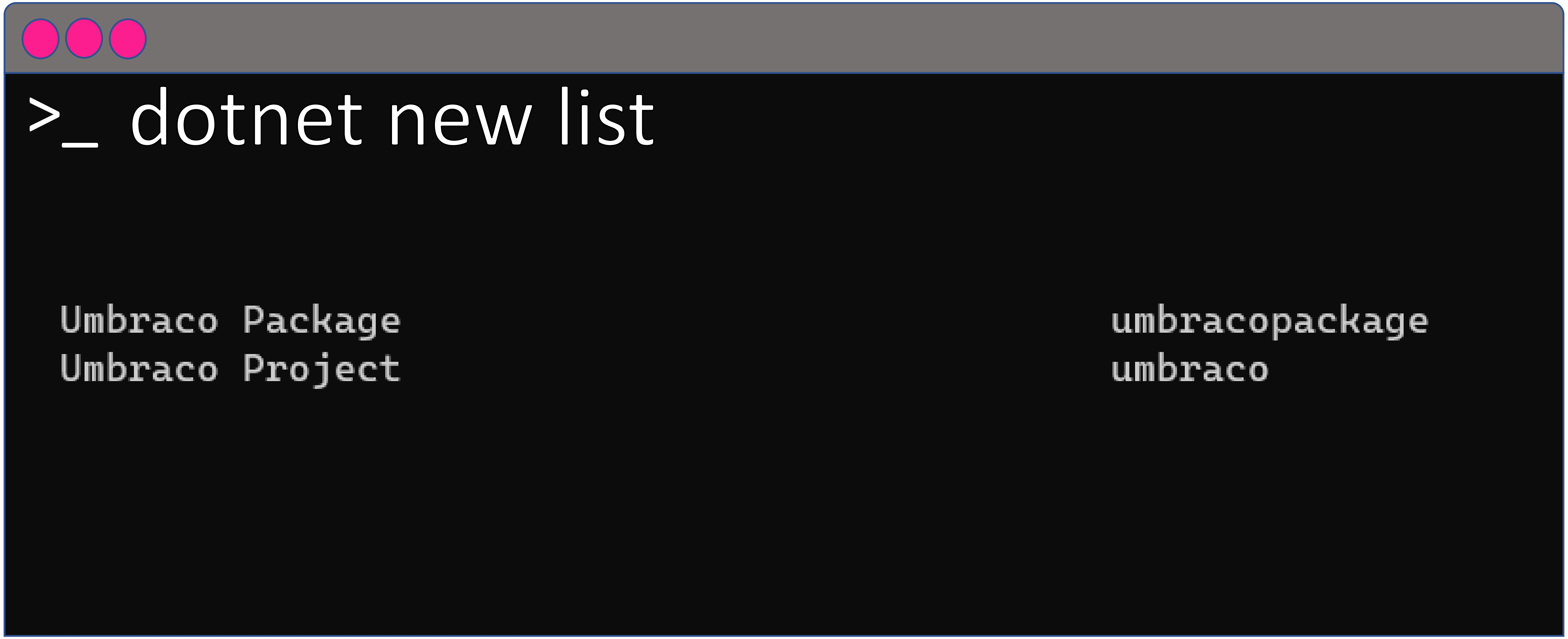


Lets see which templates are installed



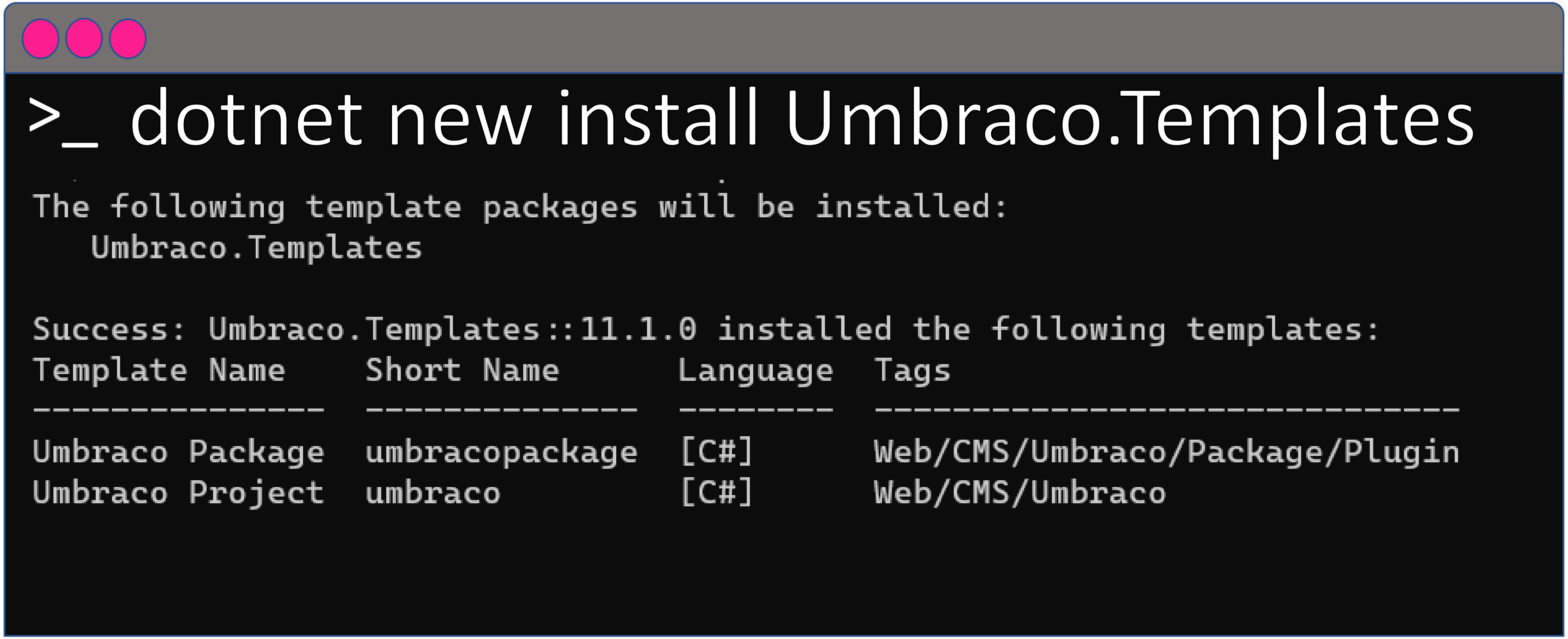
```
>_ dotnet new list

These templates matched your input:

Template Name                                 Short Name           Language    Tags
--------------------------------------------  -------------------  ----------  --------------------------------
ASP.NET Core Empty                            web                  [C#],F#     Web/Empty
ASP.NET Core gRPC Service                     grpc                 [C#]        Web/gRPC
ASP.NET Core Web API                          webapi               [C#],F#     Web/WebAPI
ASP.NET Core Web App                          webapp,razor         [C#]        Web/MVC/Razor Pages
ASP.NET Core Web App (Model-View-Controller)  mvc                  [C#],F#     Web/MVC
ASP.NET Core with Angular                     angular              [C#]        Web/MVC/SPA
ASP.NET Core with React.js                    react                [C#]        Web/MVC/SPA
Blazor Server App                             blazorserver         [C#]        Web/Blazor
Blazor Server App Empty                       blazorserver-empty   [C#]        Web/Blazor/Empty
Blazor WebAssembly App                        blazorwasm           [C#]        Web/Blazor/WebAssembly/PWA
Blazor WebAssembly App Empty                  blazorwasm-empty     [C#]        Web/Blazor/WebAssembly/PWA/Empty
Class Library                                 classlib             [C#],F#,VB  Common/Library
Console App                                   console              [C#],F#,VB  Common/Console
dotnet gitignore file                         gitignore                        Config
Dotnet local tool manifest file               tool-manifest                    Config
EditorConfig file                             editorconfig                     Config
global.json file                              globaljson                       Config
MSTest Test Project                           mstest               [C#],F#,VB  Test/MSTest
MVC ViewImports                               viewimports          [C#]        Web/ASP.NET
MVC ViewStart                                 viewstart            [C#]        Web/ASP.NET
NuGet Config                                  nugetconfig                      Config
NUnit 3 Test Item                             nunit-test           [C#],F#,VB  Test/NUnit
NUnit 3 Test Project                          nunit                [C#],F#,VB  Test/NUnit
Protocol Buffer File                          proto                            Web/gRPC
Razor Class Library                           razorclasslib        [C#]        Web/Razor/Library
Razor Component                               razorcomponent       [C#]        Web/ASP.NET
Razor Page                                    page                 [C#]        Web/ASP.NET
Solution File                                 sln,solution                     Solution
Umbraco Package                               umbracopackage       [C#]        Web/CMS/Umbraco/Package/Plugin
Umbraco Project                               umbraco              [C#]        Web/CMS/Umbraco
```

Umbraco Package and Umbraco Project



```
>_ dotnet new list


  Umbraco Package                                            umbracopackage
  Umbraco Project                                            umbraco
```

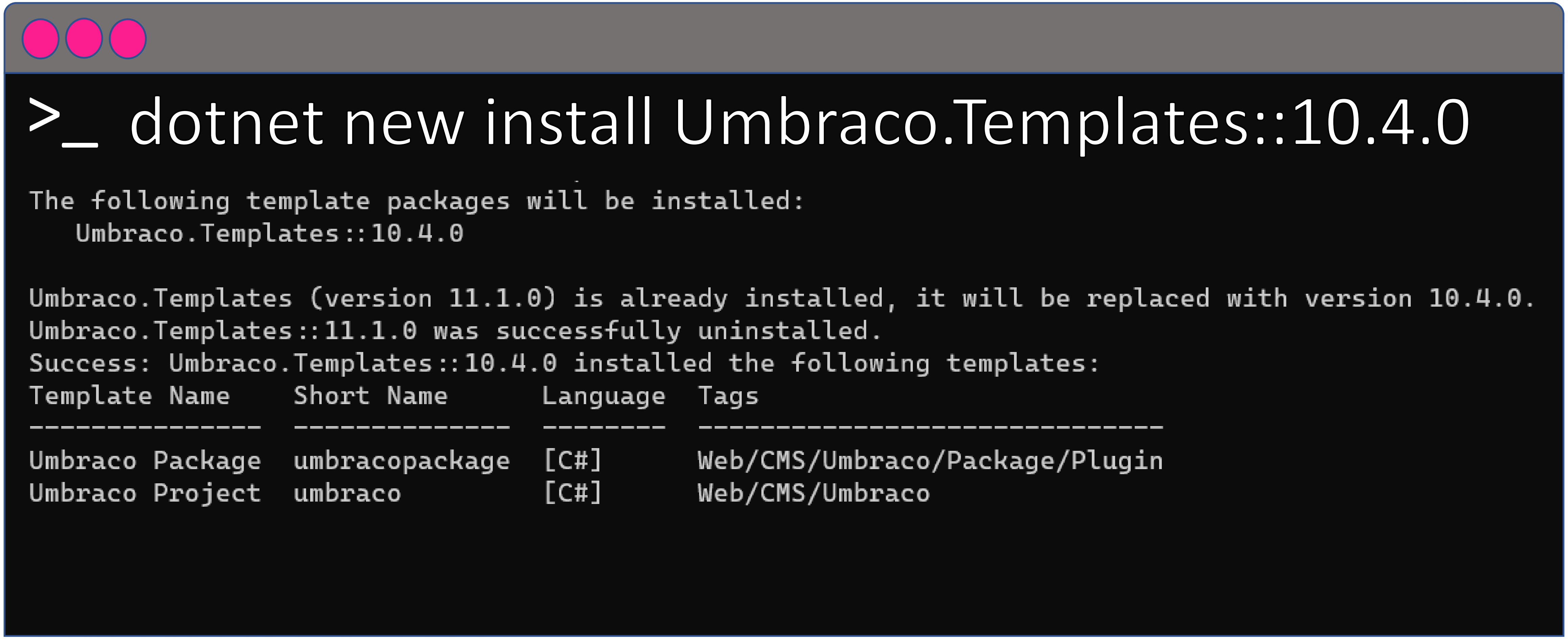
If you don’t have them already



```
>_ dotnet new install Umbraco.Templates
```

```
The following template packages will be installed:
   Umbraco.Templates

Success: Umbraco.Templates::11.1.0 installed the following templates:
Template Name     Short Name       Language  Tags
---------------   ---------------  --------  ------------------------------
Umbraco Package   umbracopackage   [C#]      Web/CMS/Umbraco/Package/Plugin
Umbraco Project   umbraco          [C#]      Web/CMS/Umbraco
```

You can specify the version



```
>_ dotnet new install Umbraco.Templates::10.4.0

The following template packages will be installed:
   Umbraco.Templates::10.4.0

Umbraco.Templates (version 11.1.0) is already installed, it will be replaced with version 10.4.0.
Umbraco.Templates::11.1.0 was successfully uninstalled.
Success: Umbraco.Templates::10.4.0 installed the following templates:
Template Name    Short Name      Language  Tags
---------------  --------------  --------  ------------------------------
Umbraco Package  umbracopackage  [C#]      Web/CMS/Umbraco/Package/Plugin
Umbraco Project  umbraco         [C#]      Web/CMS/Umbraco
```

We will do more dotnet CLI later

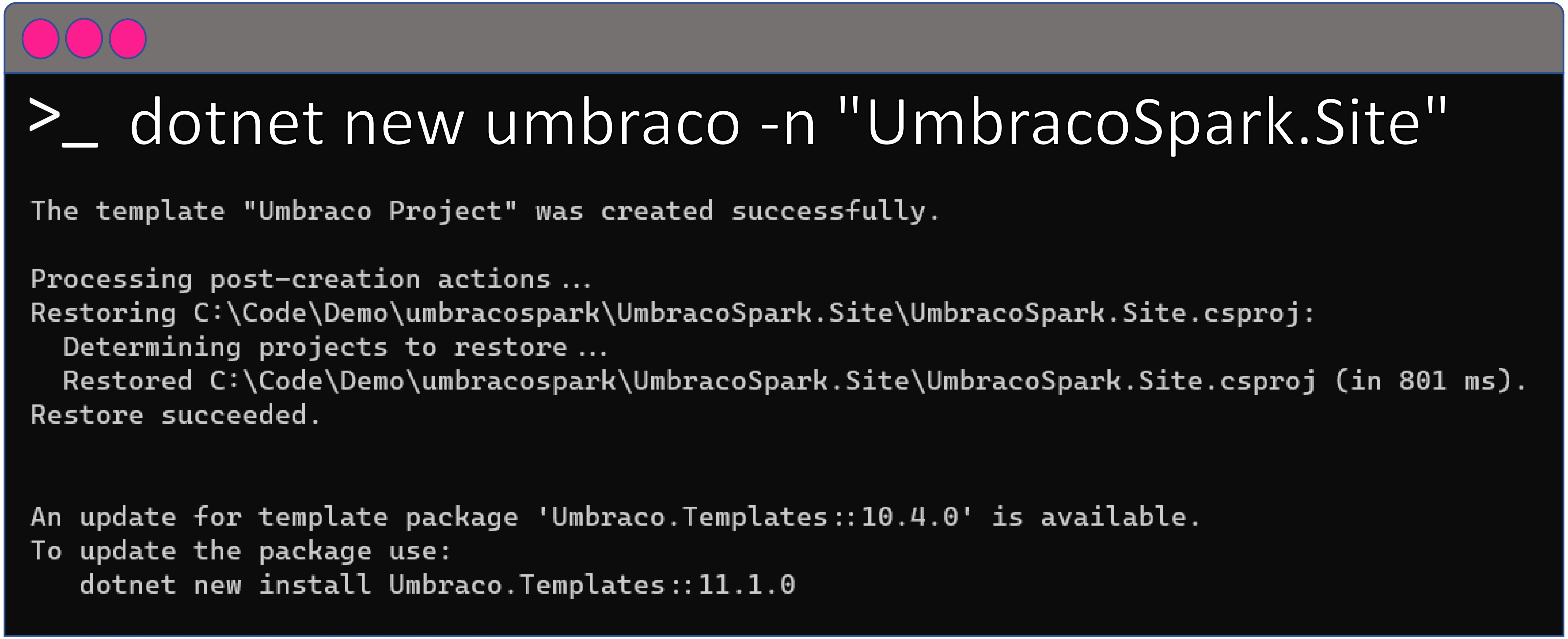
Let’s move onto Umbraco

ics

dotnet

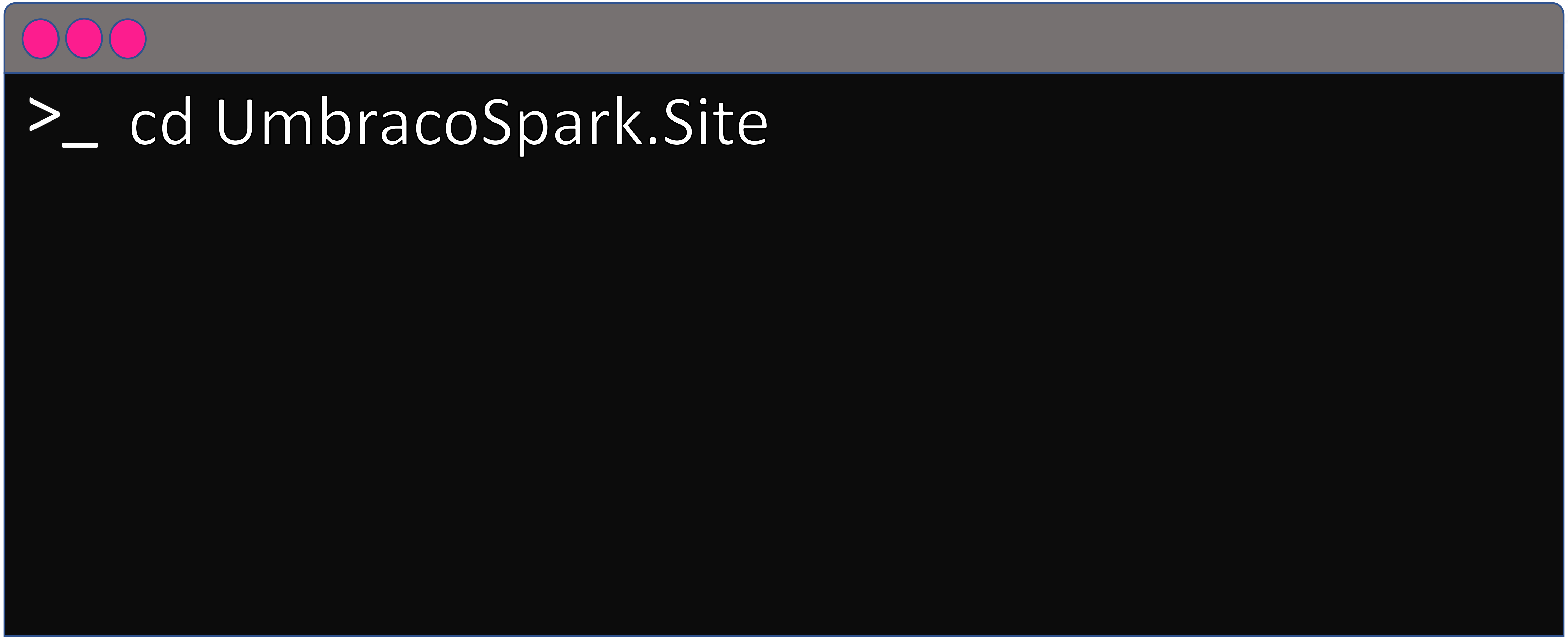
Umbraco

Lets create an Umbraco site



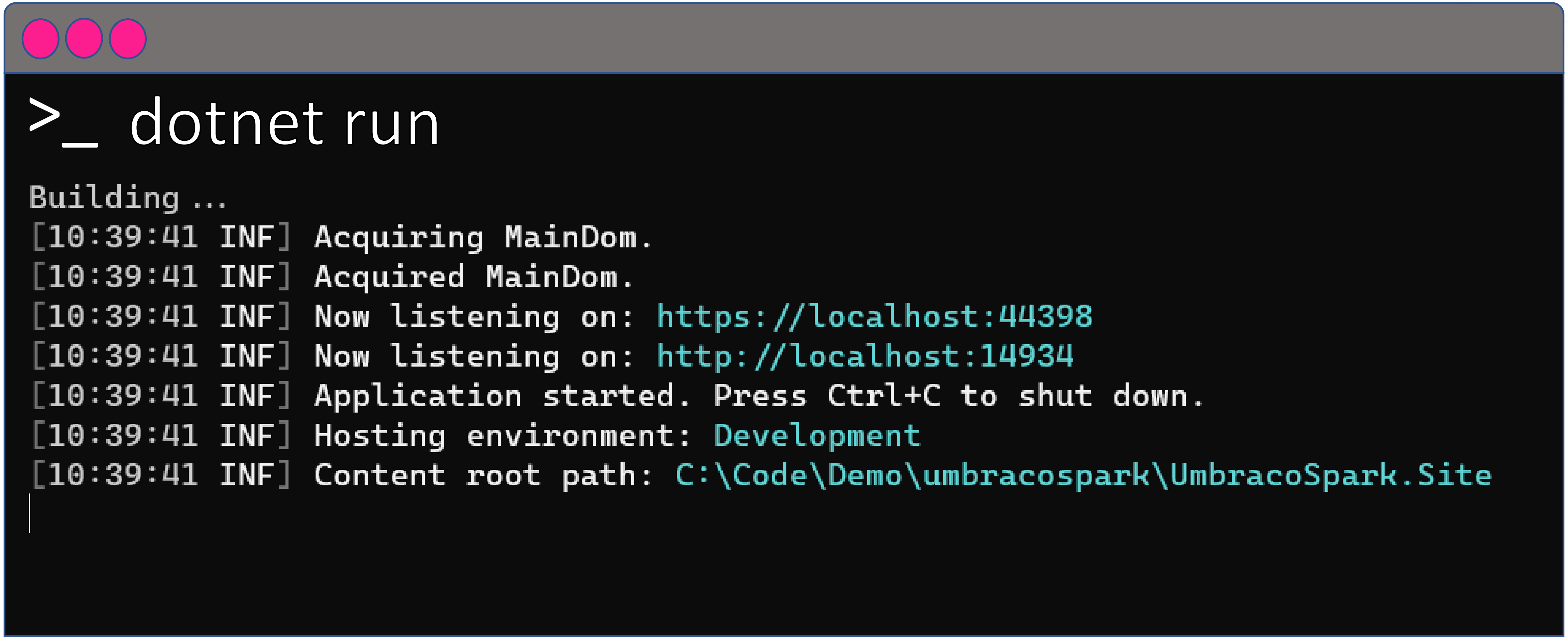
```
>_ dotnet new umbraco -n "UmbracoSpark.Site"

The template "Umbraco Project" was created successfully.

Processing post-creation actions...
Restoring C:\Code\Demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj:
  Determining projects to restore...
  Restored C:\Code\Demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj (in 801 ms).
Restore succeeded.


An update for template package 'Umbraco.Templates::10.4.0' is available.
To update the package use:
   dotnet new install Umbraco.Templates::11.1.0
```

Navigate to the newly created folder



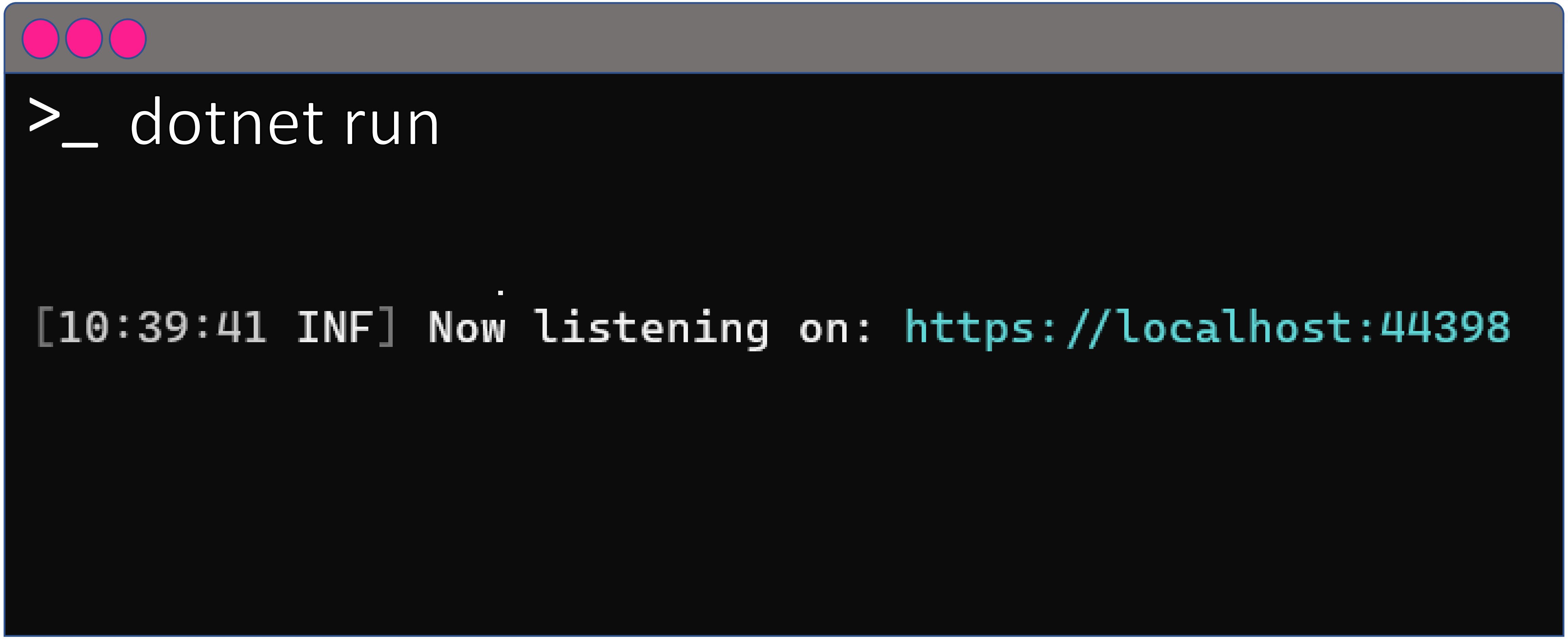
```
>_ cd UmbracoSpark.Site
```

And run the project



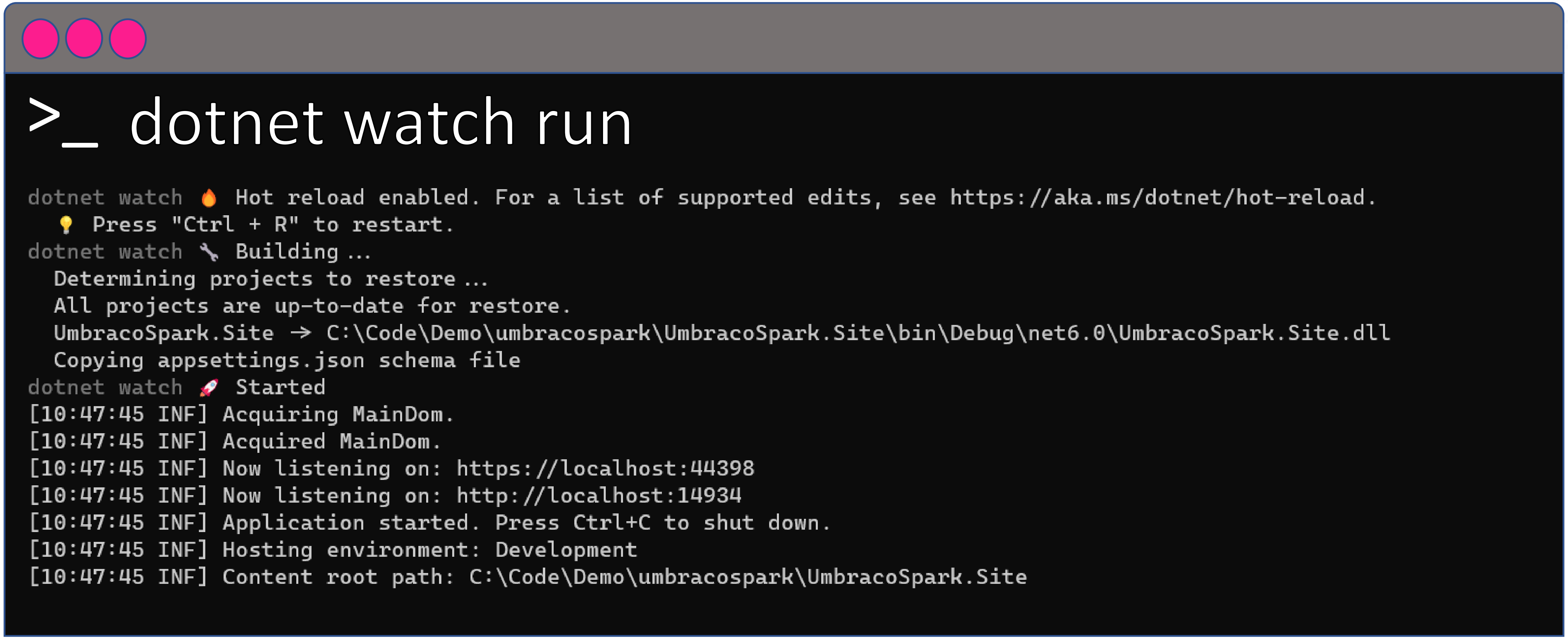
```
>_ dotnet run

Building ...
[10:39:41 INF] Acquiring MainDom.
[10:39:41 INF] Acquired MainDom.
[10:39:41 INF] Now listening on: https://localhost:44398
[10:39:41 INF] Now listening on: http://localhost:14934
[10:39:41 INF] Application started. Press Ctrl+C to shut down.
[10:39:41 INF] Hosting environment: Development
[10:39:41 INF] Content root path: C:\Code\Demo\umbracospark\UmbracoSpark.Site
```

And run the project



```
>_ dotnet run

[10:39:41 INF] Now listening on: https://localhost:44398
```

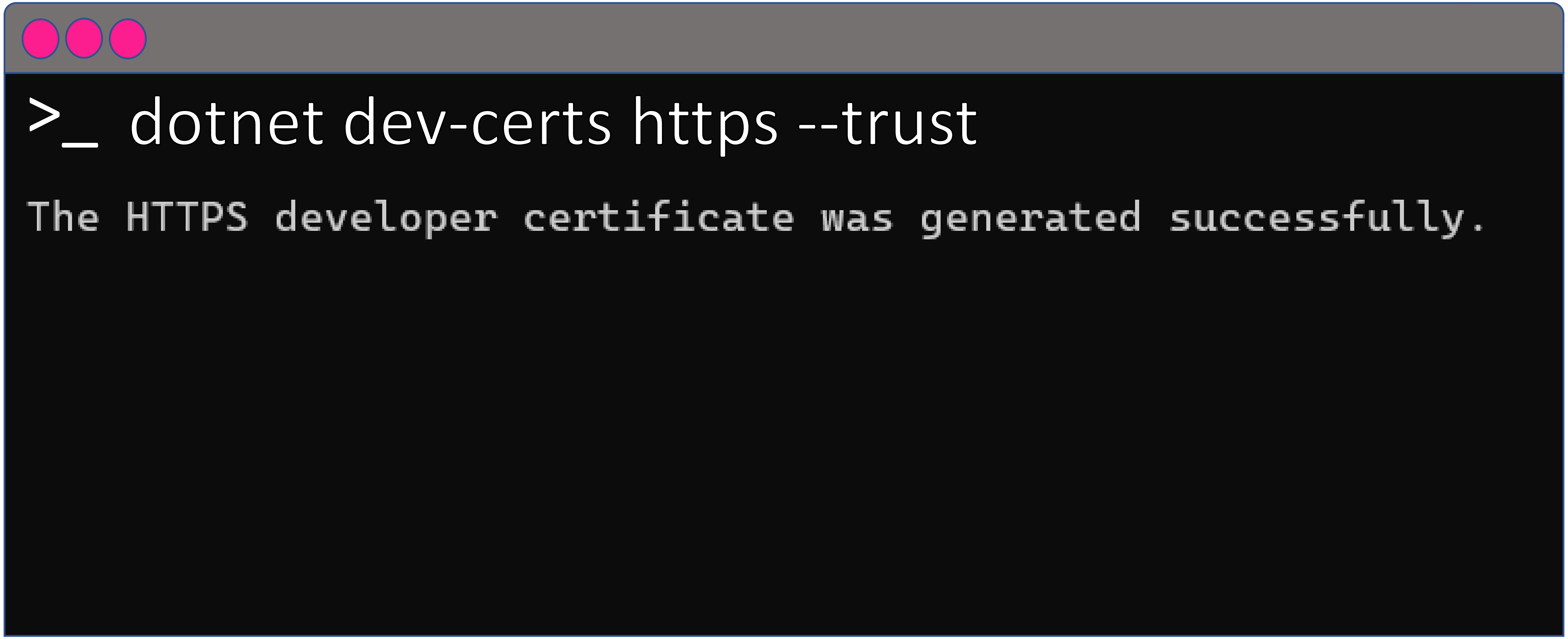
Hot reload



```
>_ dotnet watch run

dotnet watch 🔥 Hot reload enabled. For a list of supported edits, see https://aka.ms/dotnet/hot-reload.
  💡 Press "Ctrl + R" to restart.
dotnet watch 🔧 Building...
  Determining projects to restore...
  All projects are up-to-date for restore.
  UmbracoSpark.Site -> C:\Code\Demo\umbracospark\UmbracoSpark.Site\bin\Debug\net6.0\UmbracoSpark.Site.dll
  Copying appsettings.json schema file
dotnet watch 🚀 Started
[10:47:45 INF] Acquiring MainDom.
[10:47:45 INF] Acquired MainDom.
[10:47:45 INF] Now listening on: https://localhost:44398
[10:47:45 INF] Now listening on: http://localhost:14934
[10:47:45 INF] Application started. Press Ctrl+C to shut down.
[10:47:45 INF] Hosting environment: Development
[10:47:45 INF] Content root path: C:\Code\Demo\umbracospark\UmbracoSpark.Site
```

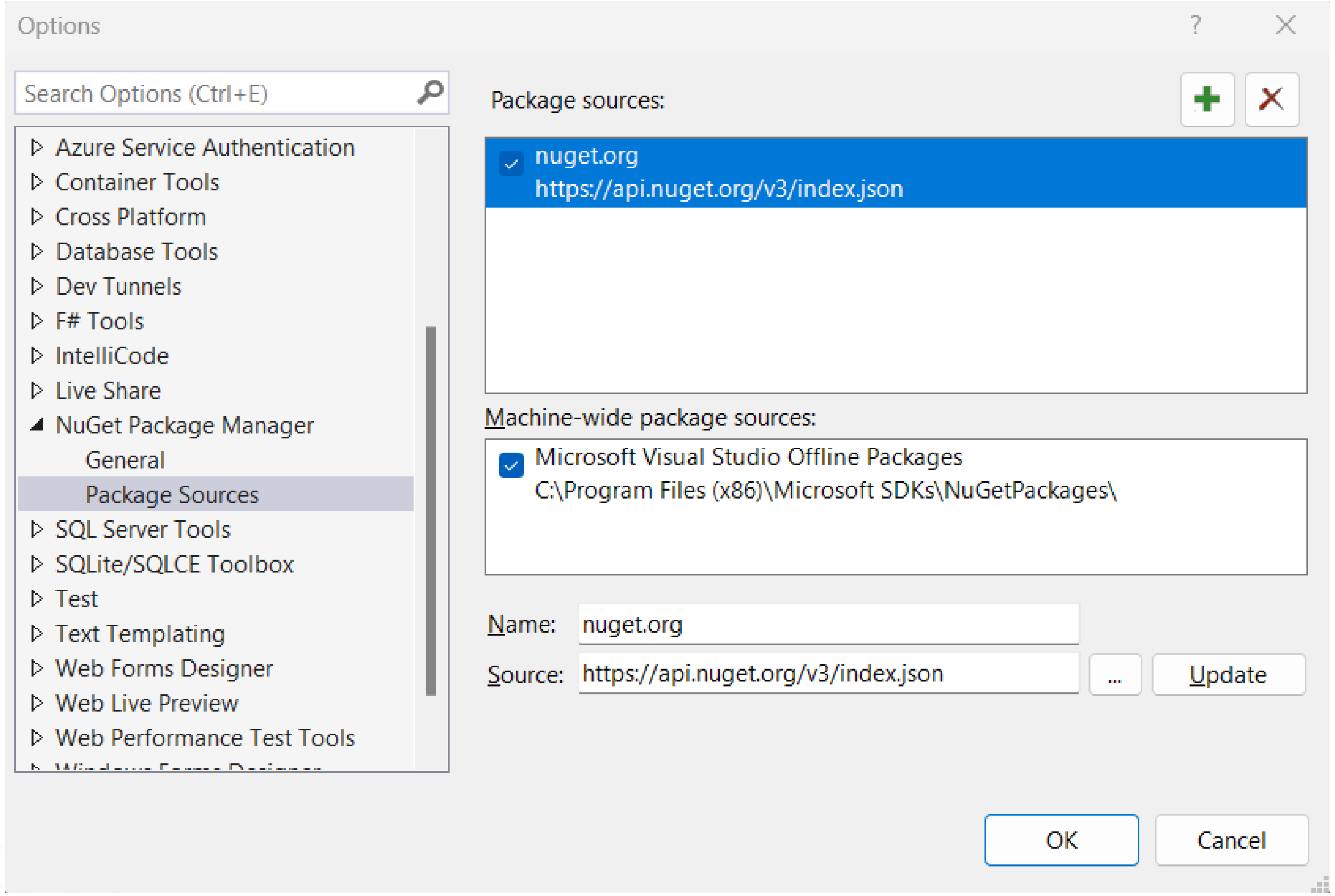
#SPARKQA

If your certificate isn’t valid



```
>_ dotnet dev-certs https --trust

The HTTPS developer certificate was generated successfully.
```

Managing package sources in VS



Options

Package sources:

☑ nuget.org
https://api.nuget.org/v3/index.json

Machine-wide package sources:

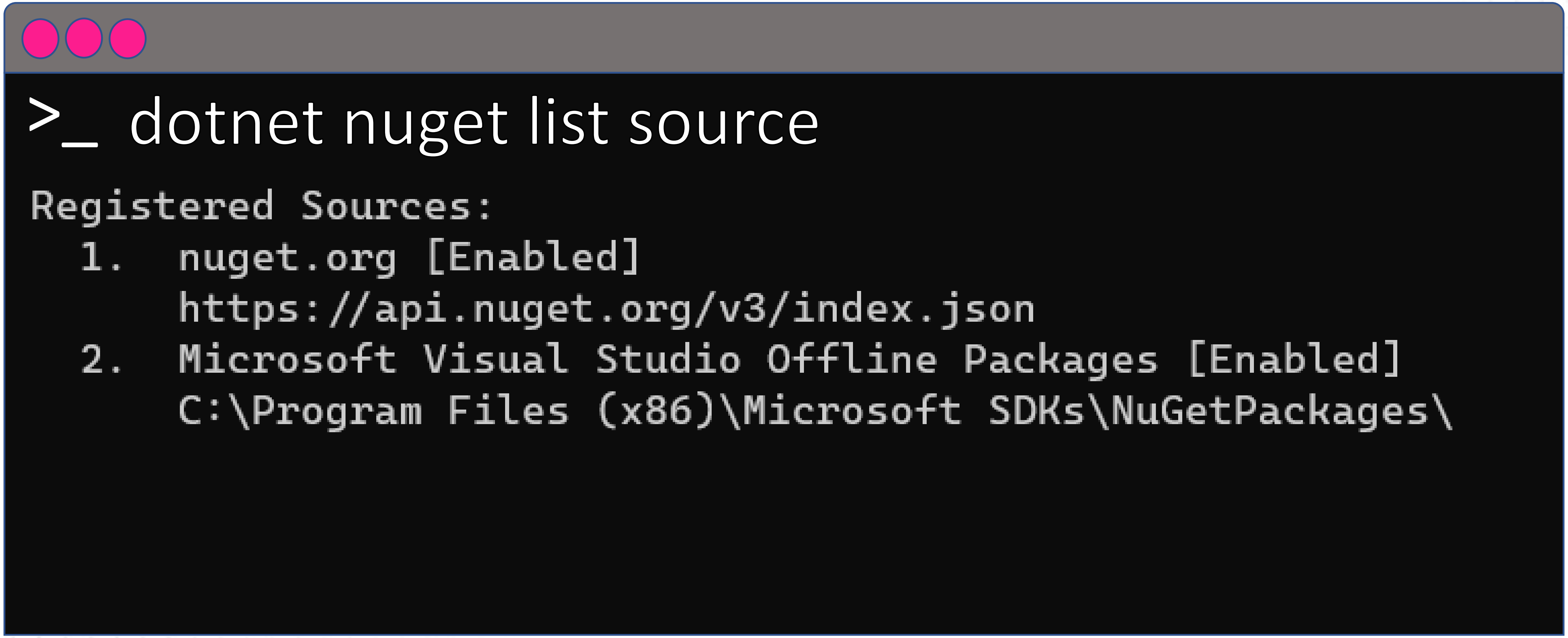
☑ Microsoft Visual Studio Offline Packages
C:\Program Files (x86)\Microsoft SDKs\NuGetPackages\

Name: nuget.org

Source: https://api.nuget.org/v3/index.json

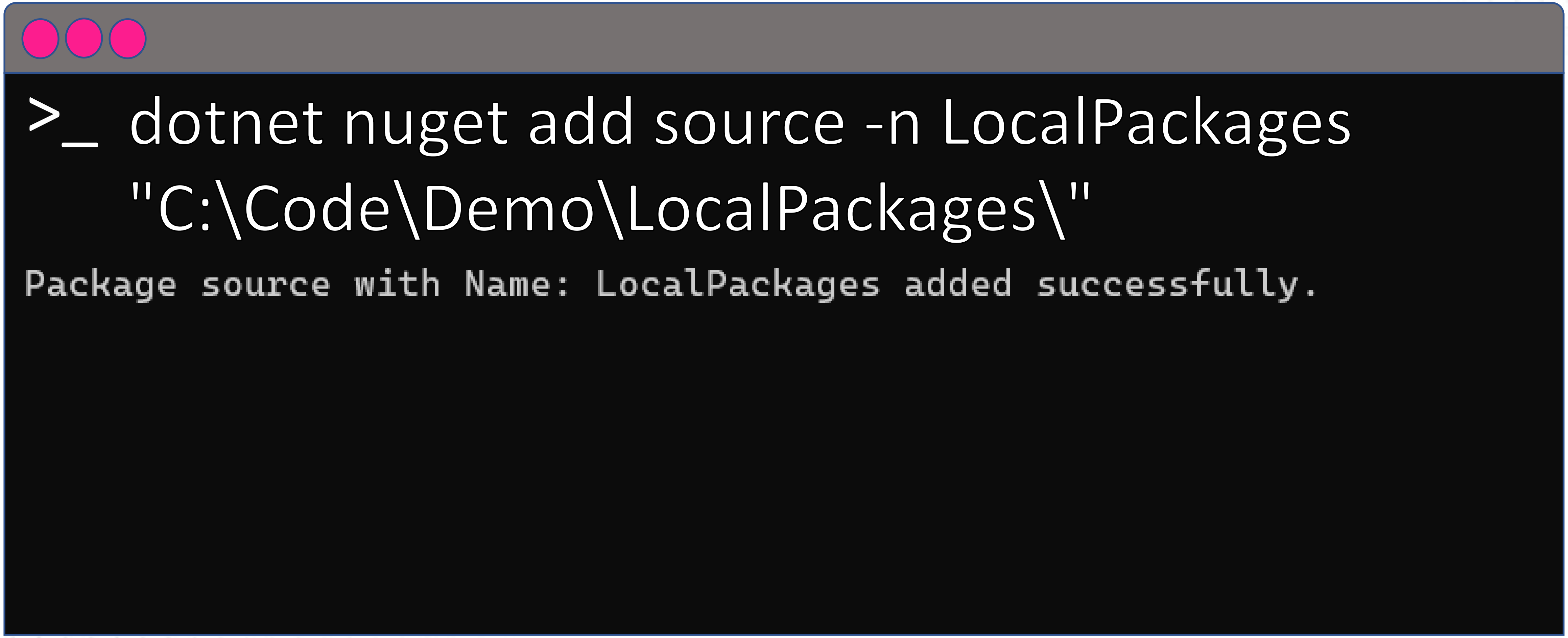
#SPARKQA

Managing them in the CLI



```
>_ dotnet nuget list source

Registered Sources:
  1.  nuget.org [Enabled]
      https://api.nuget.org/v3/index.json
  2.  Microsoft Visual Studio Offline Packages [Enabled]
      C:\Program Files (x86)\Microsoft SDKs\NuGetPackages\
```

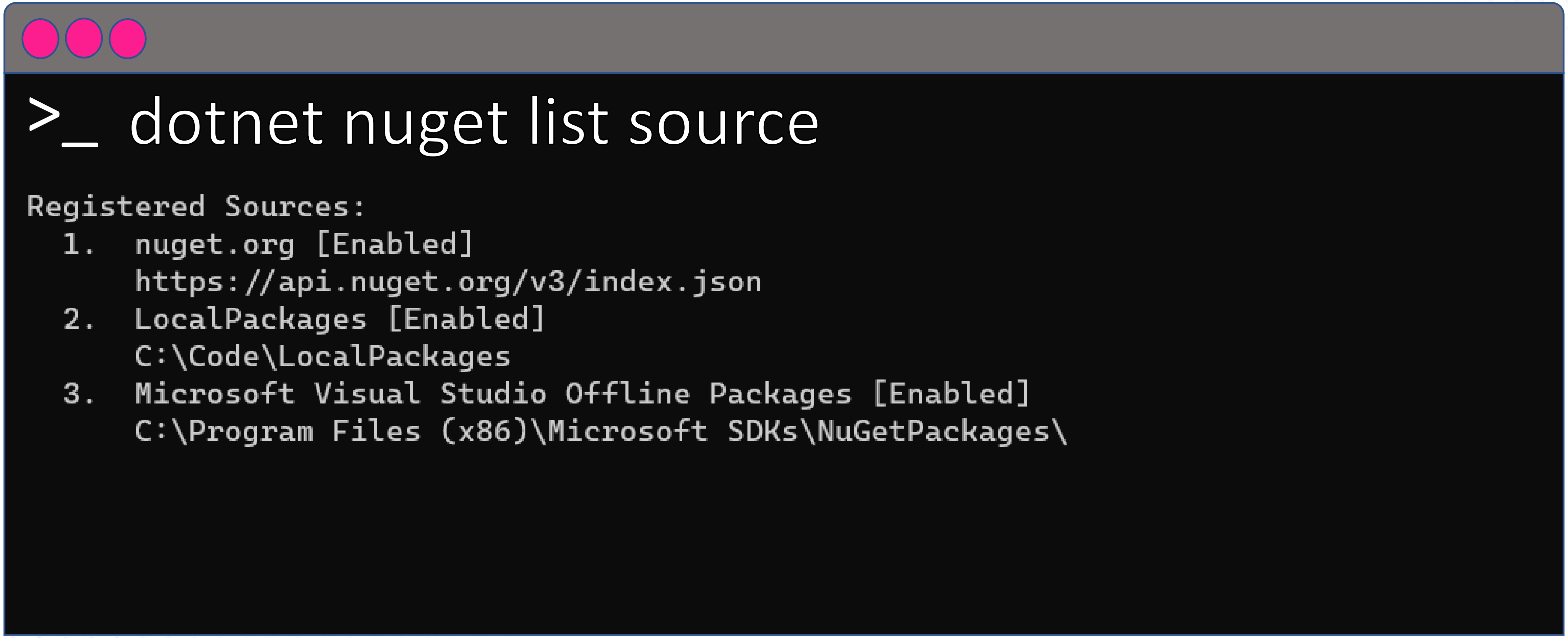
Adding a new nuget source



```
>_ dotnet nuget add source -n LocalPackages
   "C:\Code\Demo\LocalPackages\"
Package source with Name: LocalPackages added successfully.
```

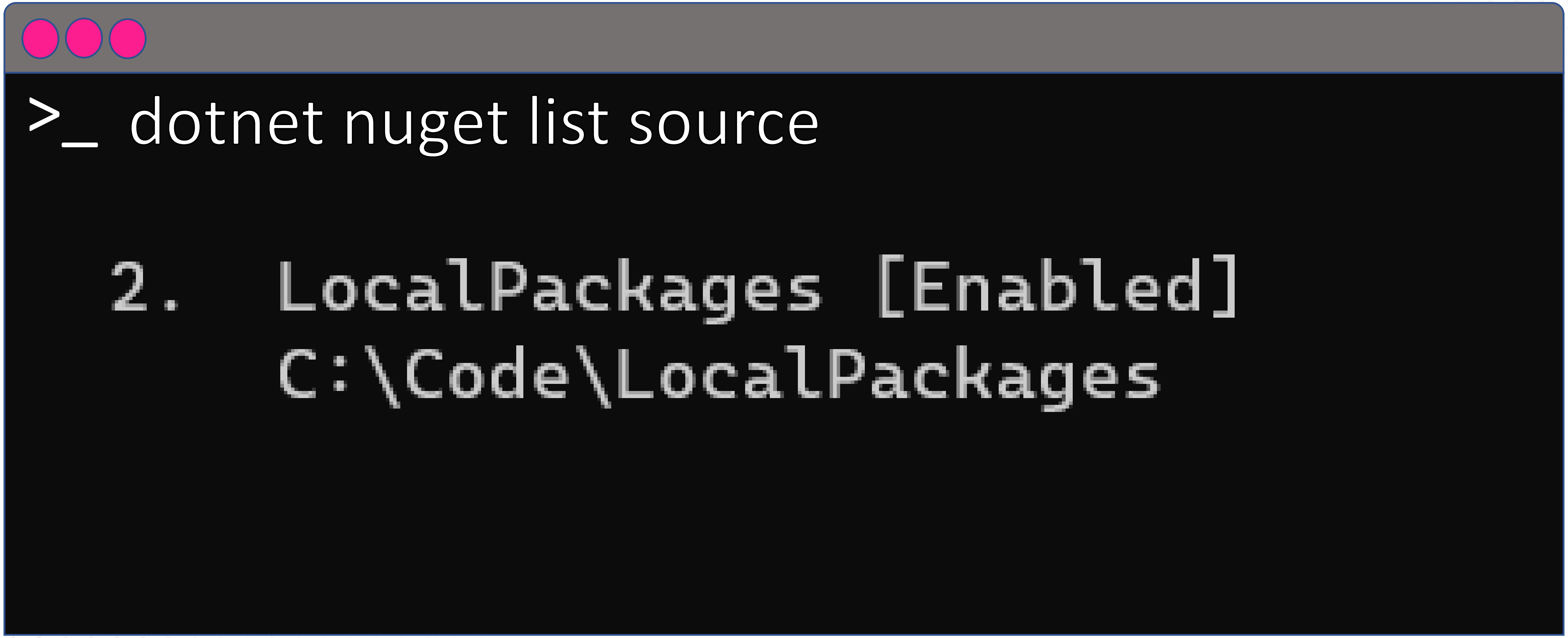
#SPARKQA

Check the list again



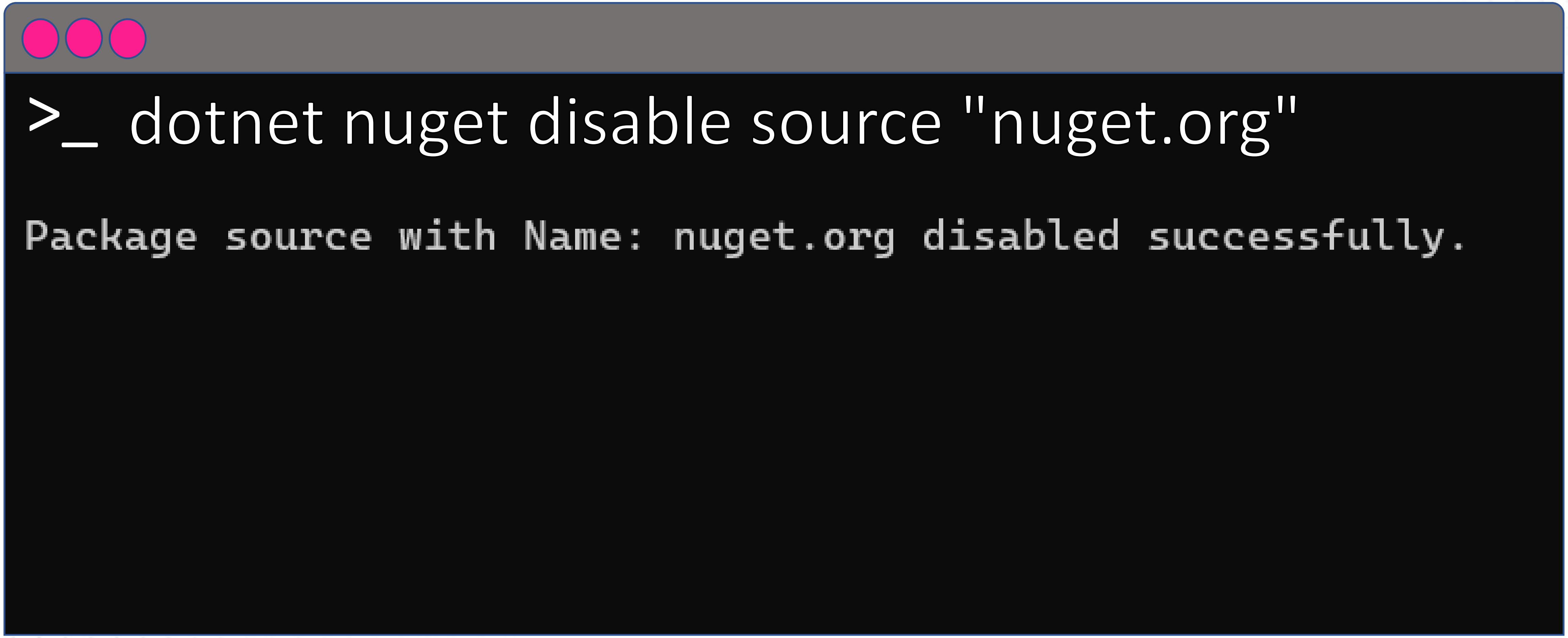
```
>_ dotnet nuget list source

Registered Sources:
  1.  nuget.org [Enabled]
      https://api.nuget.org/v3/index.json
  2.  LocalPackages [Enabled]
      C:\Code\LocalPackages
  3.  Microsoft Visual Studio Offline Packages [Enabled]
      C:\Program Files (x86)\Microsoft SDKs\NuGetPackages\
```

Check the list again



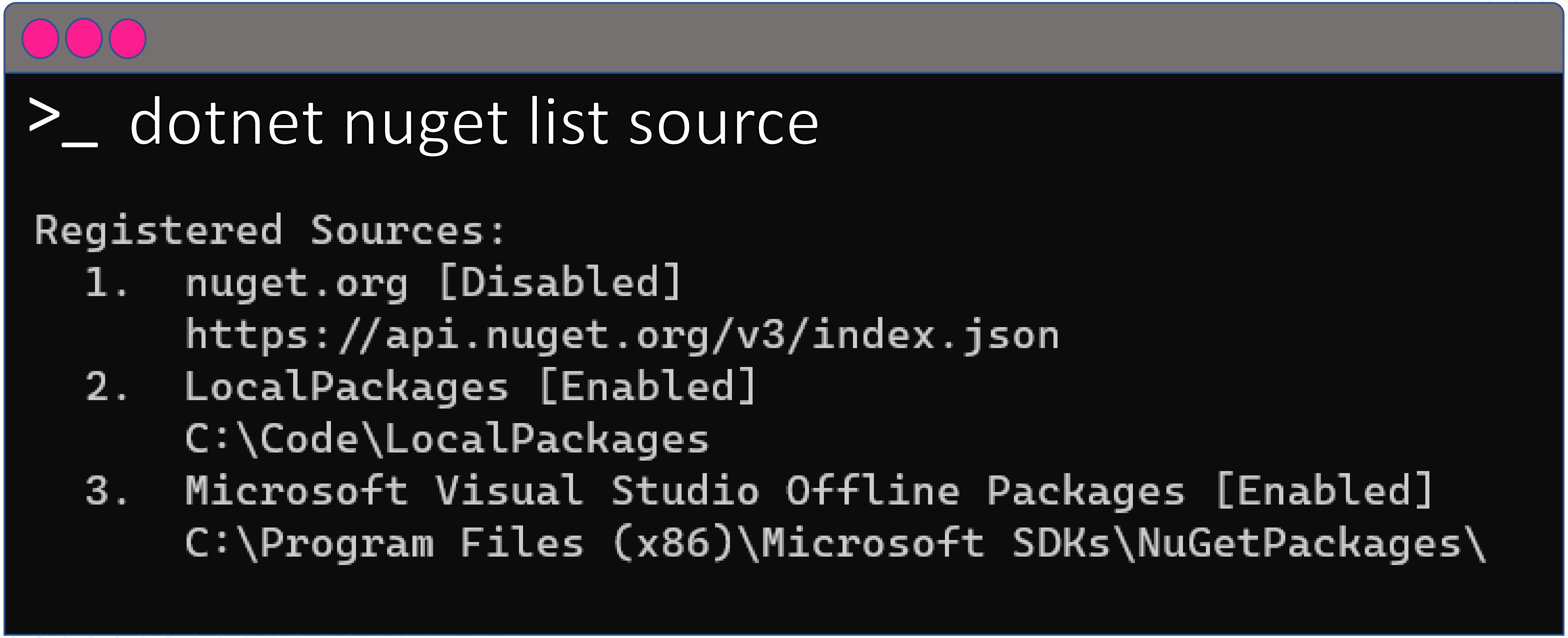
```
>_ dotnet nuget list source

   2.   LocalPackages [Enabled]
        C:\Code\LocalPackages
```

Disabling a nuget source



```
>_ dotnet nuget disable source "nuget.org"

Package source with Name: nuget.org disabled successfully.
```

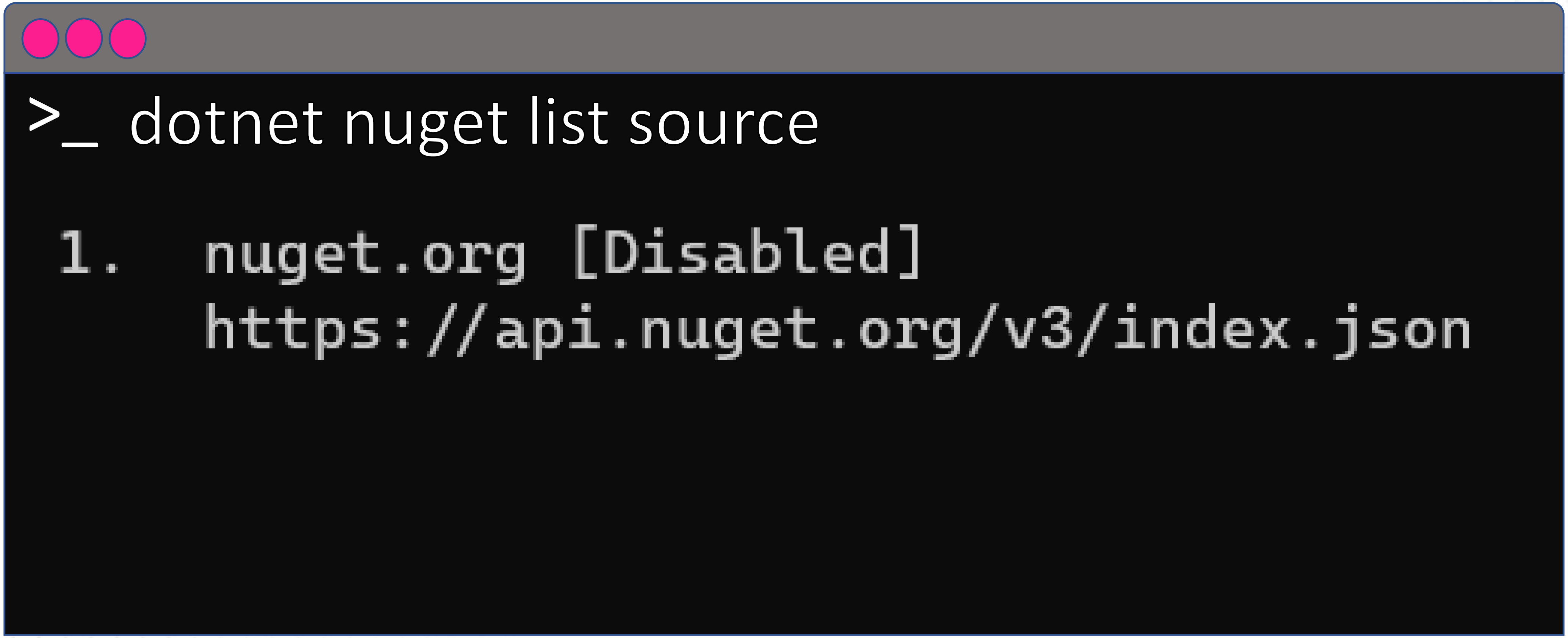
Check the list again



```
>_ dotnet nuget list source

Registered Sources:
  1.  nuget.org [Disabled]
      https://api.nuget.org/v3/index.json
  2.  LocalPackages [Enabled]
      C:\Code\LocalPackages
  3.  Microsoft Visual Studio Offline Packages [Enabled]
      C:\Program Files (x86)\Microsoft SDKs\NuGetPackages\
```

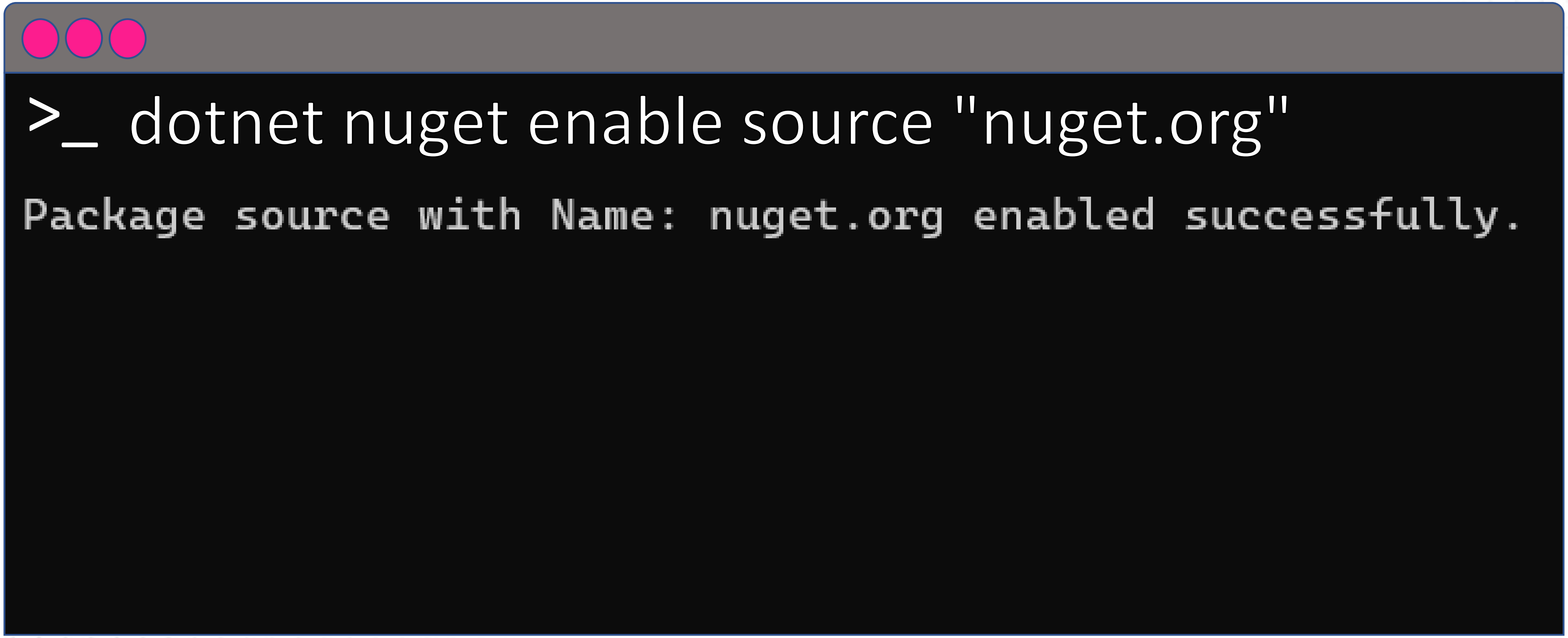
#SPARKQA

Check the list again



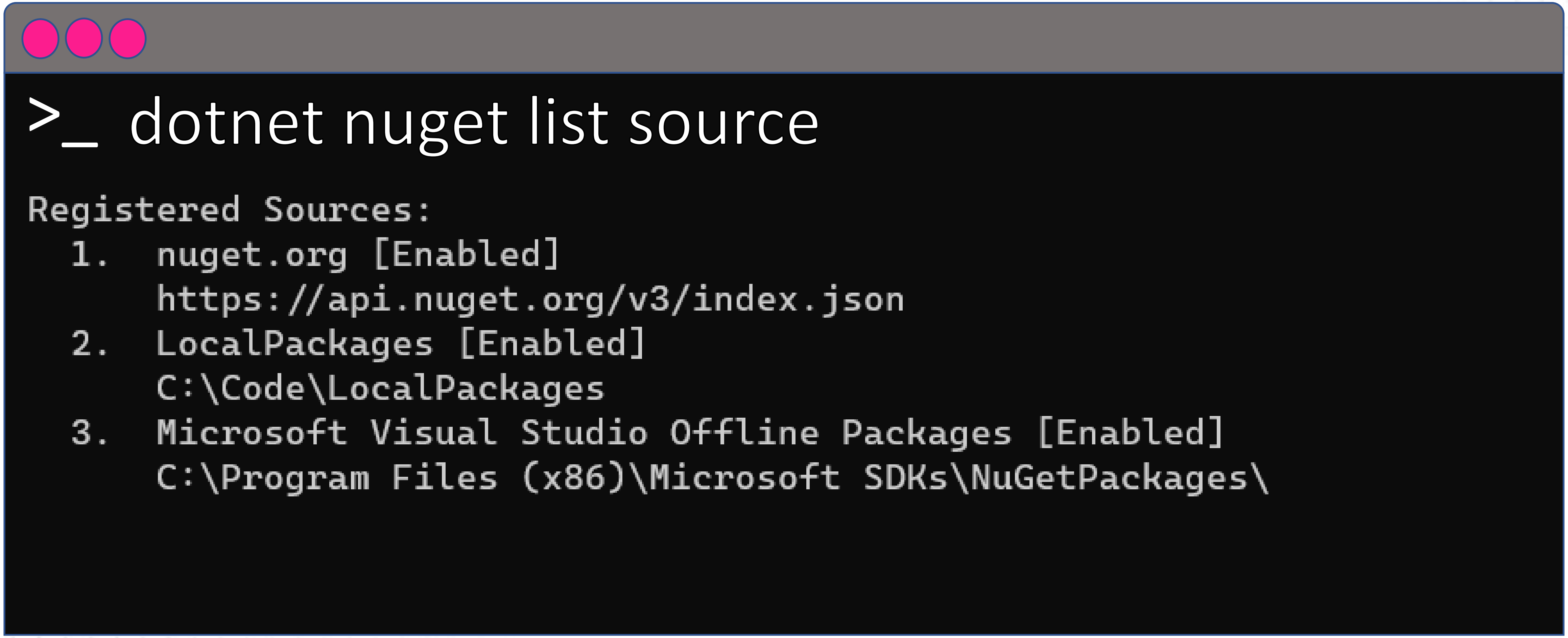
```
>_ dotnet nuget list source

 1.   nuget.org [Disabled]
      https://api.nuget.org/v3/index.json
```

Enabling a nuget source



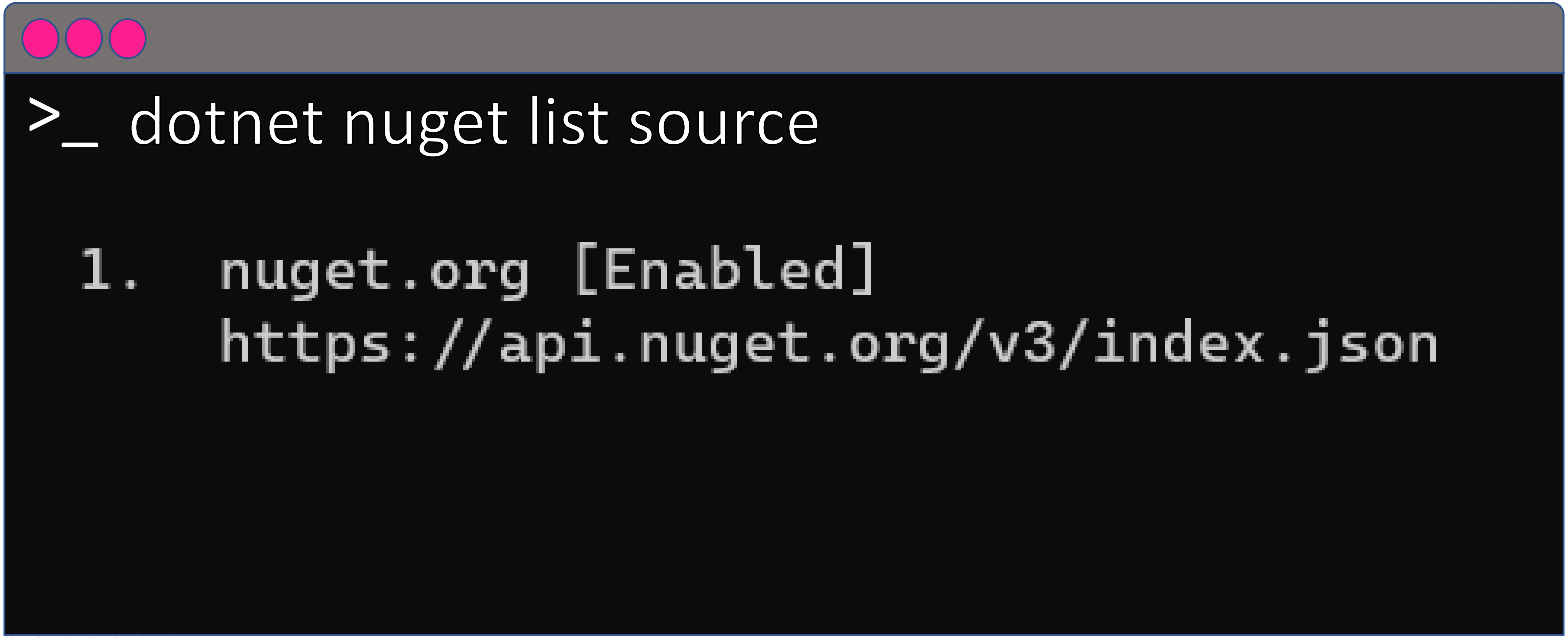
```
>_ dotnet nuget enable source "nuget.org"

Package source with Name: nuget.org enabled successfully.
```

Check the list again



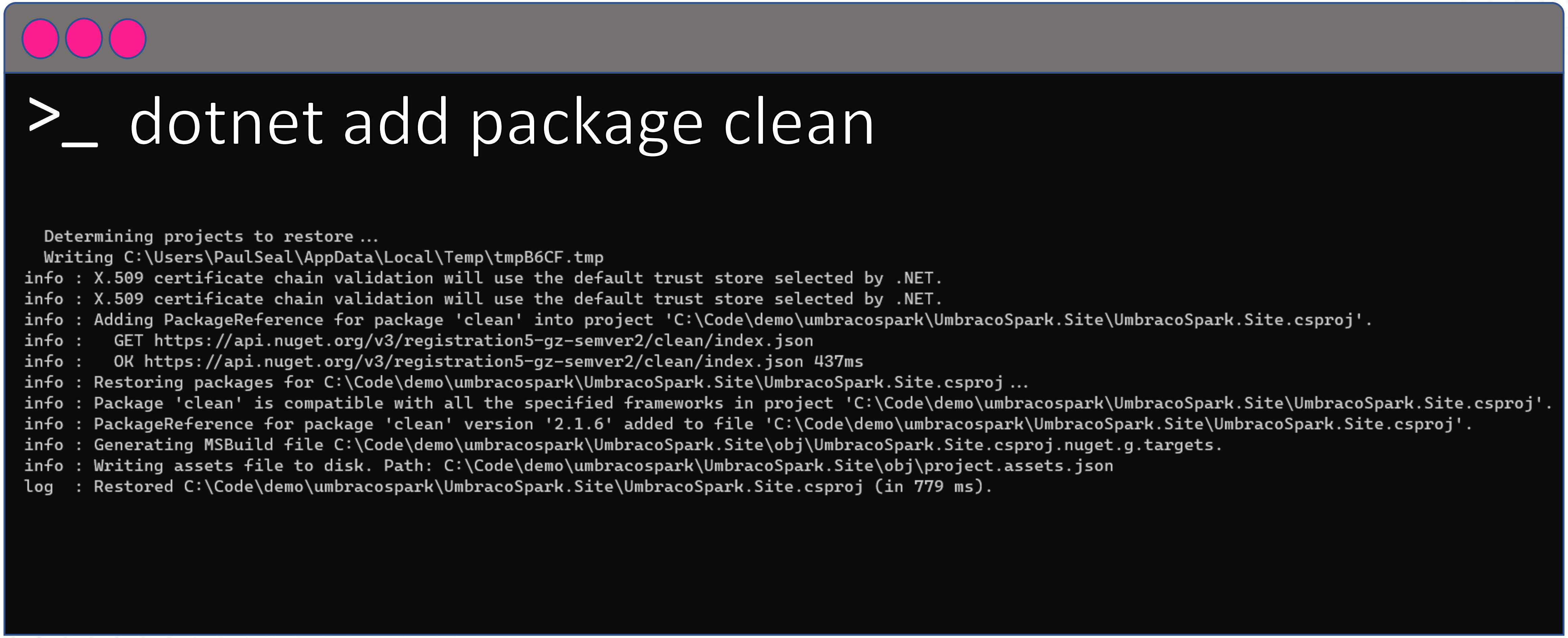
```
>_ dotnet nuget list source

Registered Sources:
  1.   nuget.org [Enabled]
       https://api.nuget.org/v3/index.json
  2.   LocalPackages [Enabled]
       C:\Code\LocalPackages
  3.   Microsoft Visual Studio Offline Packages [Enabled]
       C:\Program Files (x86)\Microsoft SDKs\NuGetPackages\
```

Check the list again



```
>_ dotnet nuget list source

  1.   nuget.org [Enabled]
       https://api.nuget.org/v3/index.json
```

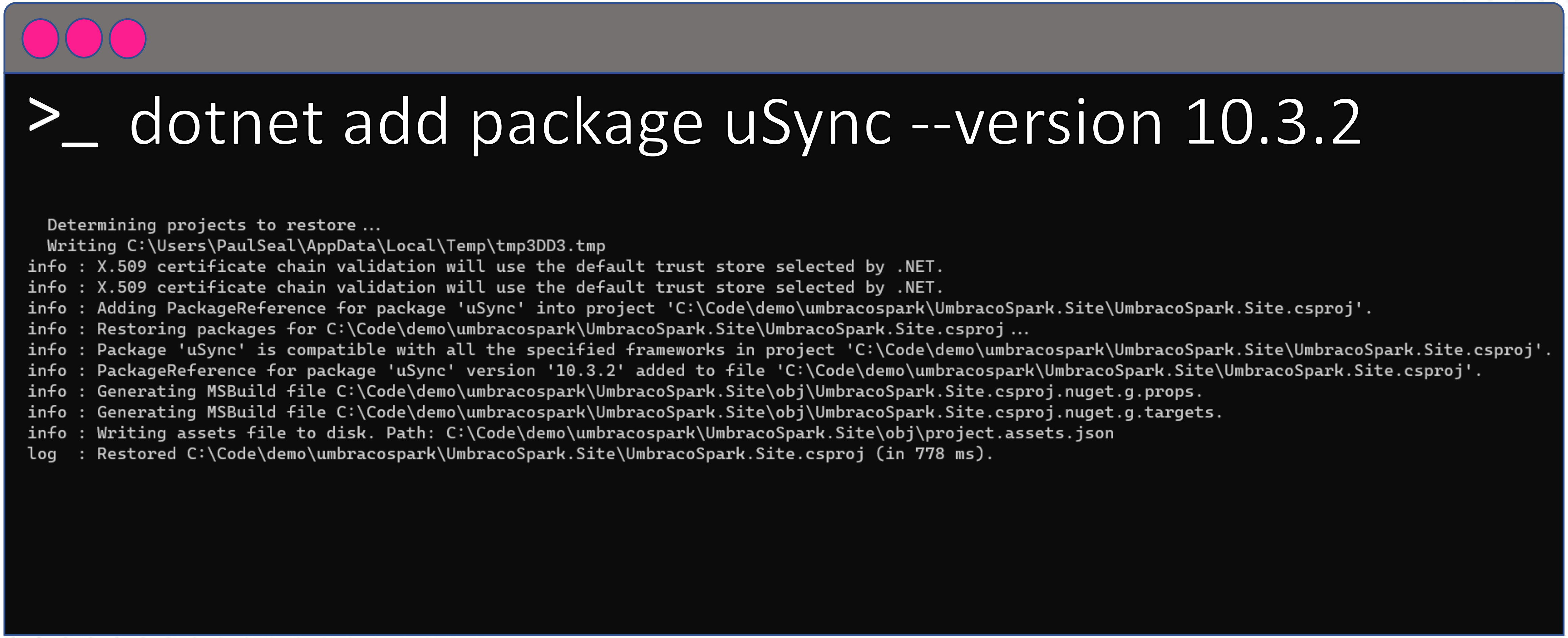
Adding a package



```
>_ dotnet add package clean
```

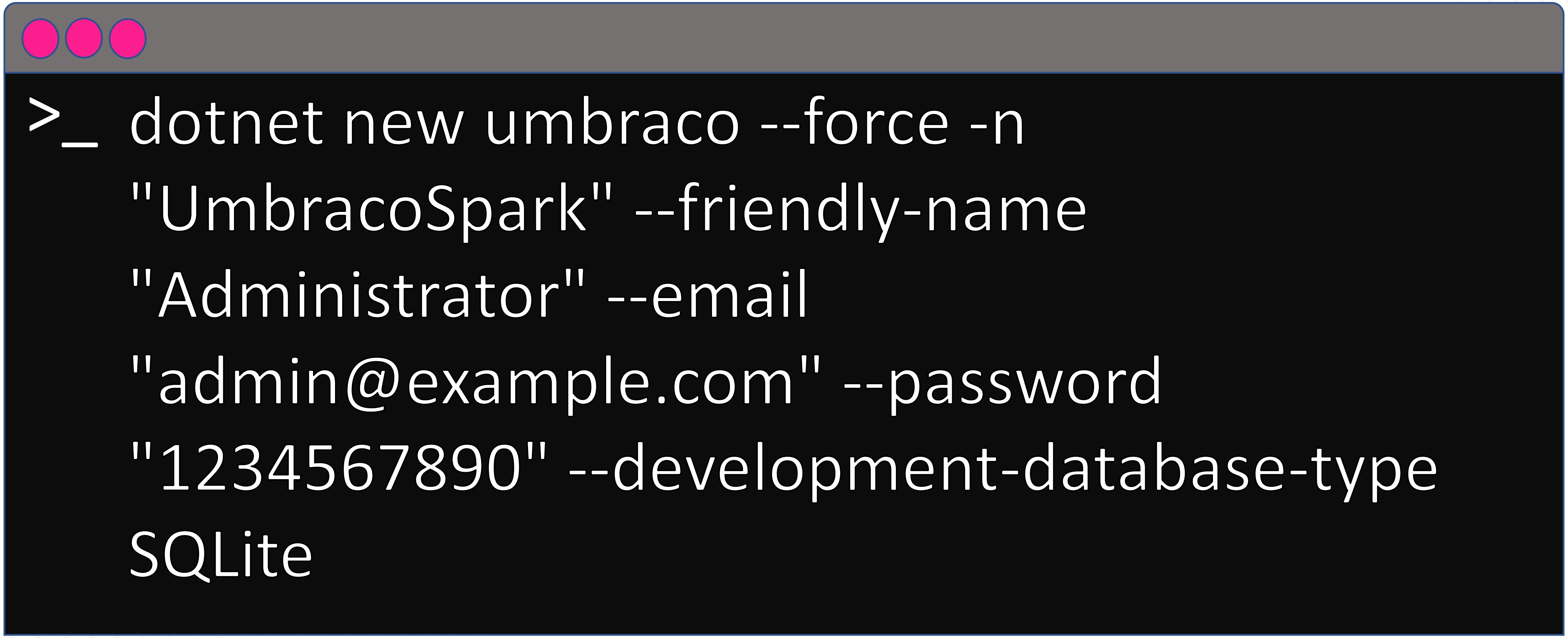
```
  Determining projects to restore...
  Writing C:\Users\PaulSeal\AppData\Local\Temp\tmpB6CF.tmp
info : X.509 certificate chain validation will use the default trust store selected by .NET.
info : X.509 certificate chain validation will use the default trust store selected by .NET.
info : Adding PackageReference for package 'clean' into project 'C:\Code\demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj'.
info :   GET https://api.nuget.org/v3/registration5-gz-semver2/clean/index.json
info :   OK https://api.nuget.org/v3/registration5-gz-semver2/clean/index.json 437ms
info : Restoring packages for C:\Code\demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj...
info : Package 'clean' is compatible with all the specified frameworks in project 'C:\Code\demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj'.
info : PackageReference for package 'clean' version '2.1.6' added to file 'C:\Code\demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj'.
info : Generating MSBuild file C:\Code\demo\umbracospark\UmbracoSpark.Site\obj\UmbracoSpark.Site.csproj.nuget.g.targets.
info : Writing assets file to disk. Path: C:\Code\demo\umbracospark\UmbracoSpark.Site\obj\project.assets.json
log  : Restored C:\Code\demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj (in 779 ms).
```

Specifying a package version



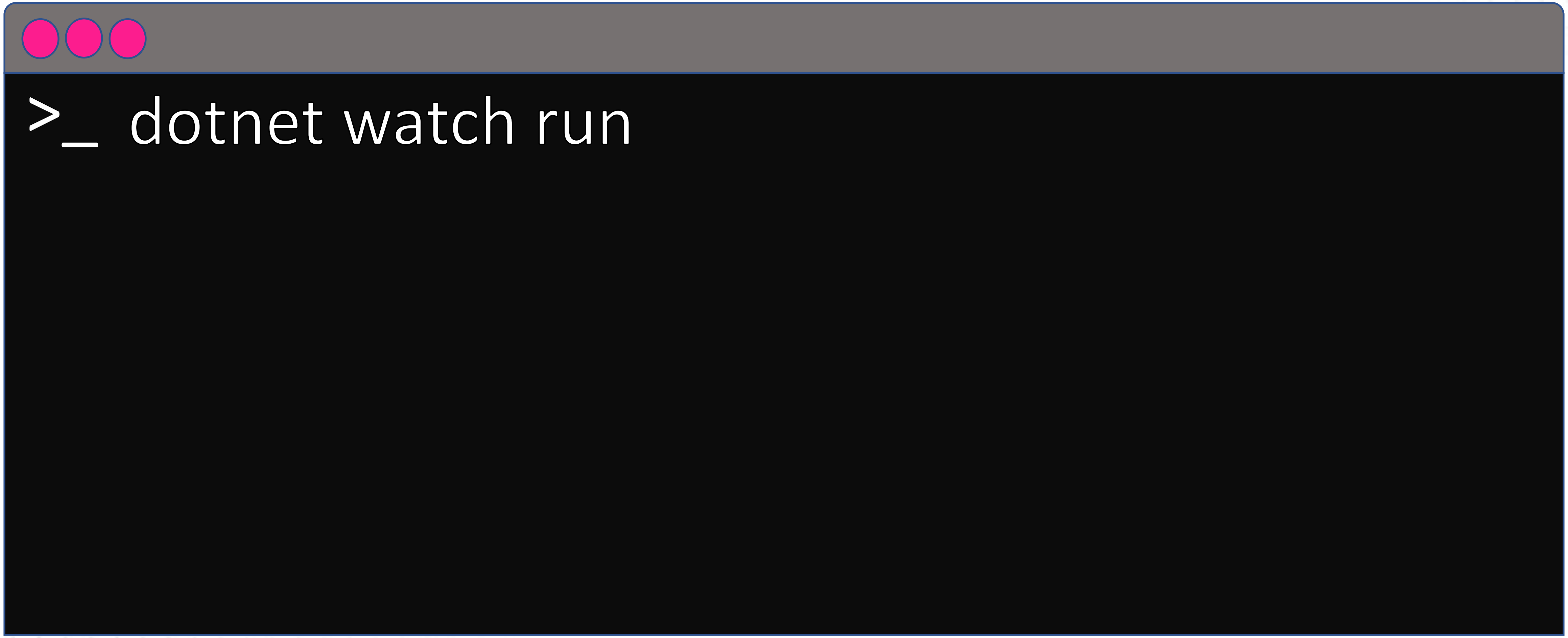
```
>_ dotnet add package uSync --version 10.3.2

  Determining projects to restore...
  Writing C:\Users\PaulSeal\AppData\Local\Temp\tmp3DD3.tmp
info : X.509 certificate chain validation will use the default trust store selected by .NET.
info : X.509 certificate chain validation will use the default trust store selected by .NET.
info : Adding PackageReference for package 'uSync' into project 'C:\Code\demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj'.
info : Restoring packages for C:\Code\demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj...
info : Package 'uSync' is compatible with all the specified frameworks in project 'C:\Code\demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj'.
info : PackageReference for package 'uSync' version '10.3.2' added to file 'C:\Code\demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj'.
info : Generating MSBuild file C:\Code\demo\umbracospark\UmbracoSpark.Site\obj\UmbracoSpark.Site.csproj.nuget.g.props.
info : Generating MSBuild file C:\Code\demo\umbracospark\UmbracoSpark.Site\obj\UmbracoSpark.Site.csproj.nuget.g.targets.
info : Writing assets file to disk. Path: C:\Code\demo\umbracospark\UmbracoSpark.Site\obj\project.assets.json
log  : Restored C:\Code\demo\umbracospark\UmbracoSpark.Site\UmbracoSpark.Site.csproj (in 778 ms).
```

Unattended install in Umbraco



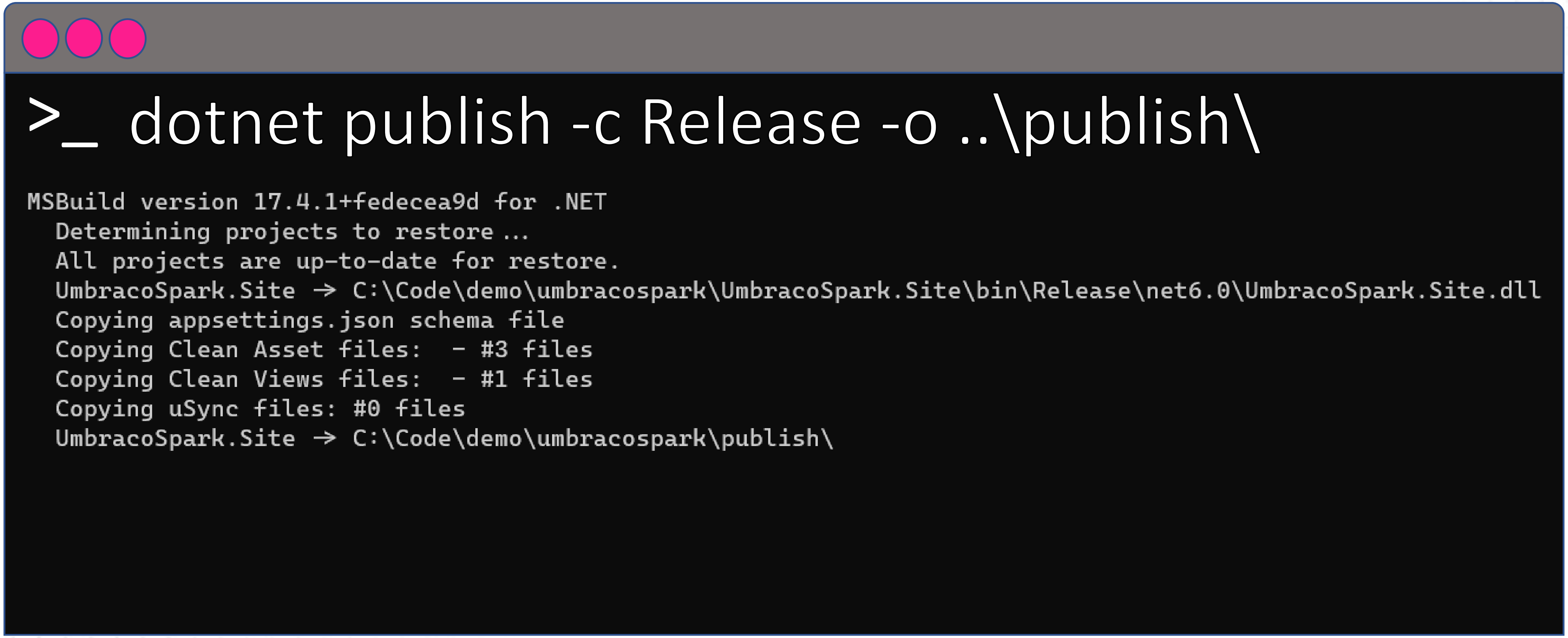
```
>_ dotnet new umbraco --force -n
   "UmbracoSpark" --friendly-name
   "Administrator" --email
   "admin@example.com" --password
   "1234567890" --development-database-type
   SQLite
```

Now build, run and launch the site



```
>_ dotnet watch run
```

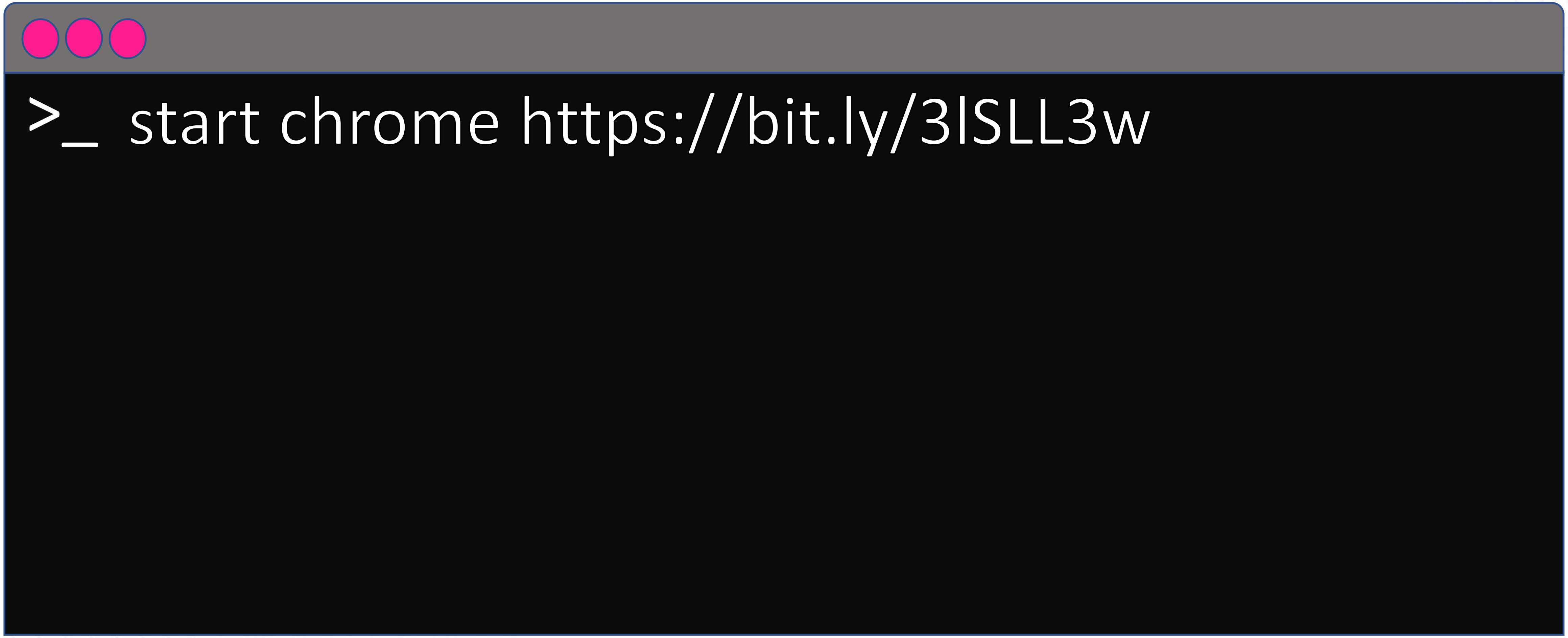
Publishing a site



```
>_ dotnet publish -c Release -o ..\publish\

MSBuild version 17.4.1+fedecea9d for .NET
  Determining projects to restore...
  All projects are up-to-date for restore.
  UmbracoSpark.Site -> C:\Code\demo\umbracospark\UmbracoSpark.Site\bin\Release\net6.0\UmbracoSpark.Site.dll
  Copying appsettings.json schema file
  Copying Clean Asset files:  - #3 files
  Copying Clean Views files:  - #1 files
  Copying uSync files: #0 files
  UmbracoSpark.Site -> C:\Code\demo\umbracospark\publish\
```

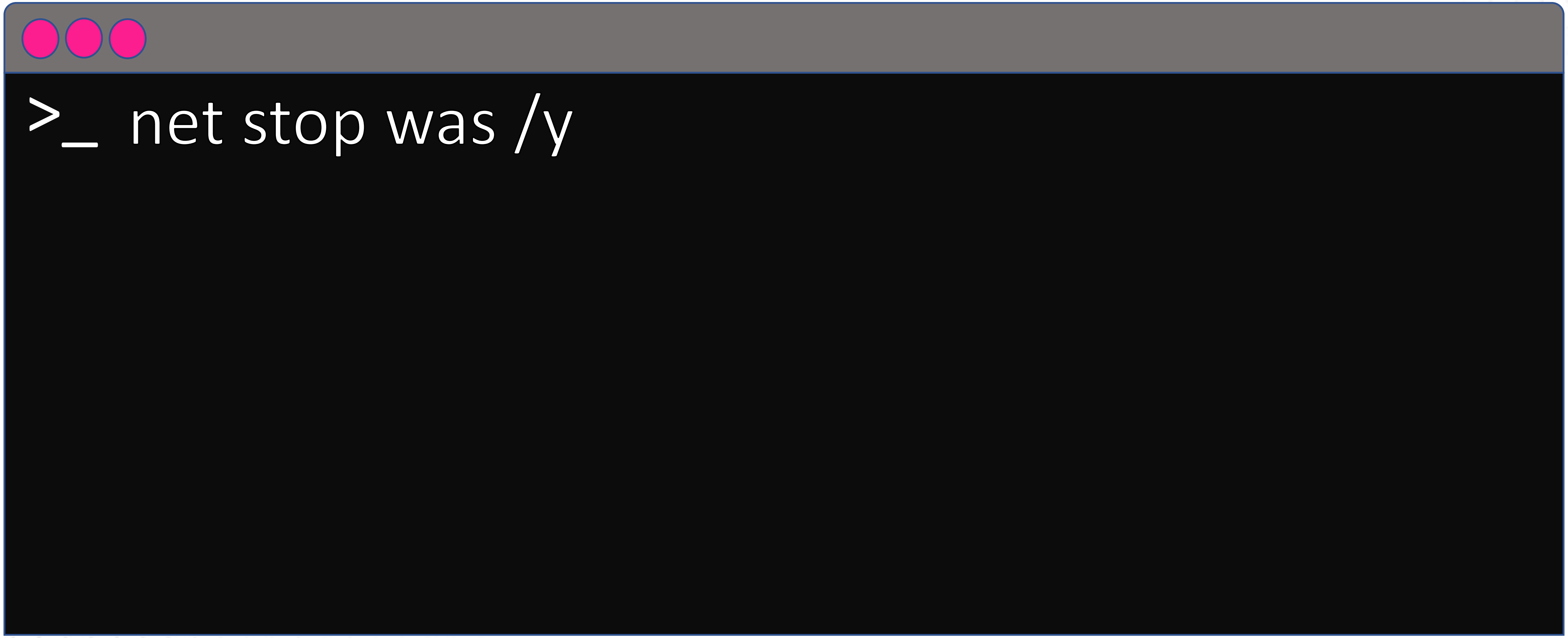
#SPARKQA

Download the hosting bundle



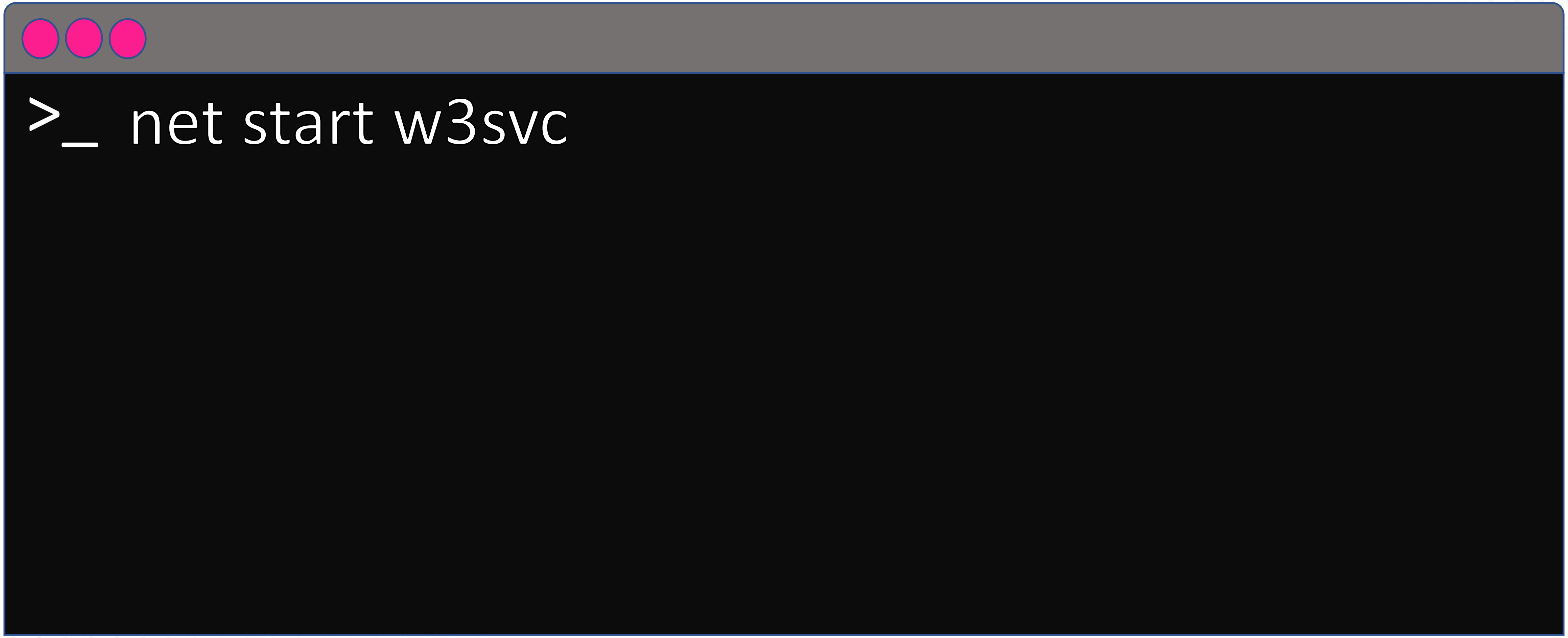
```
>_ start chrome https://bit.ly/3lSLL3w
```

Restart the server or execute



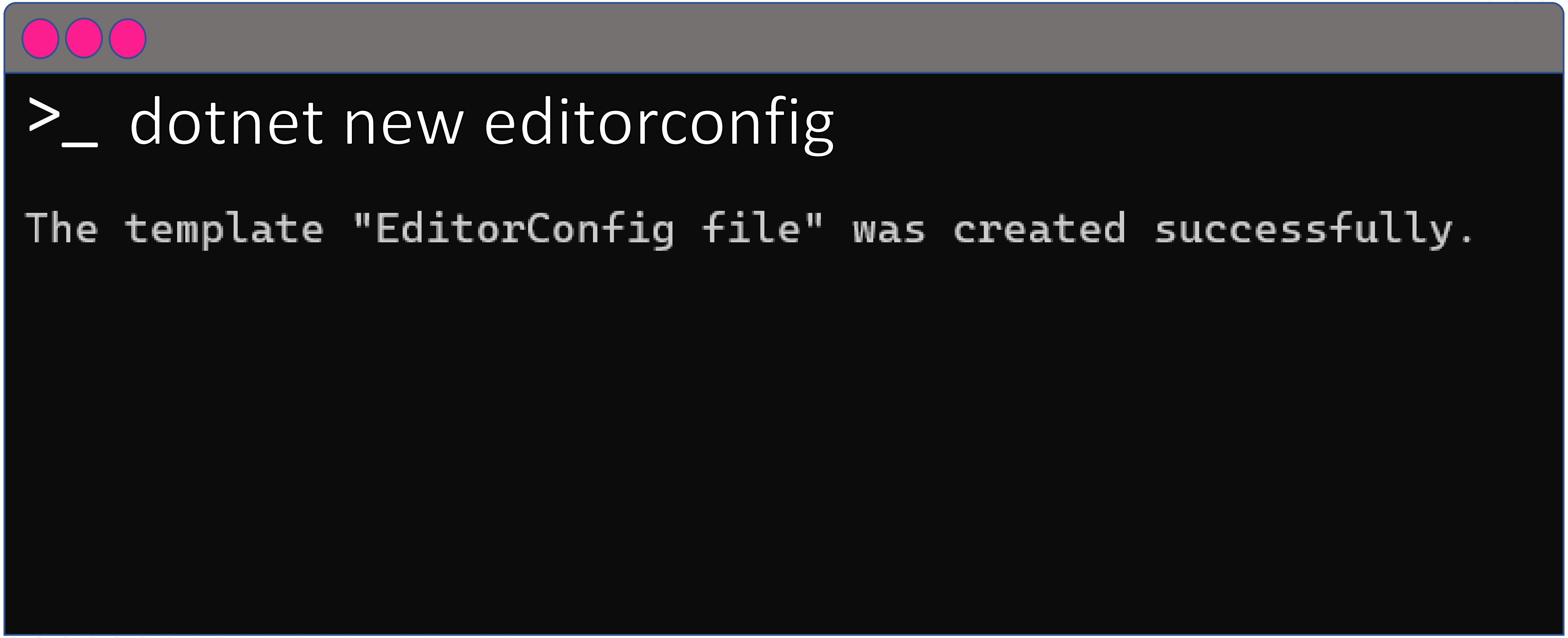
```
>_ net stop was /y
```

Followed by



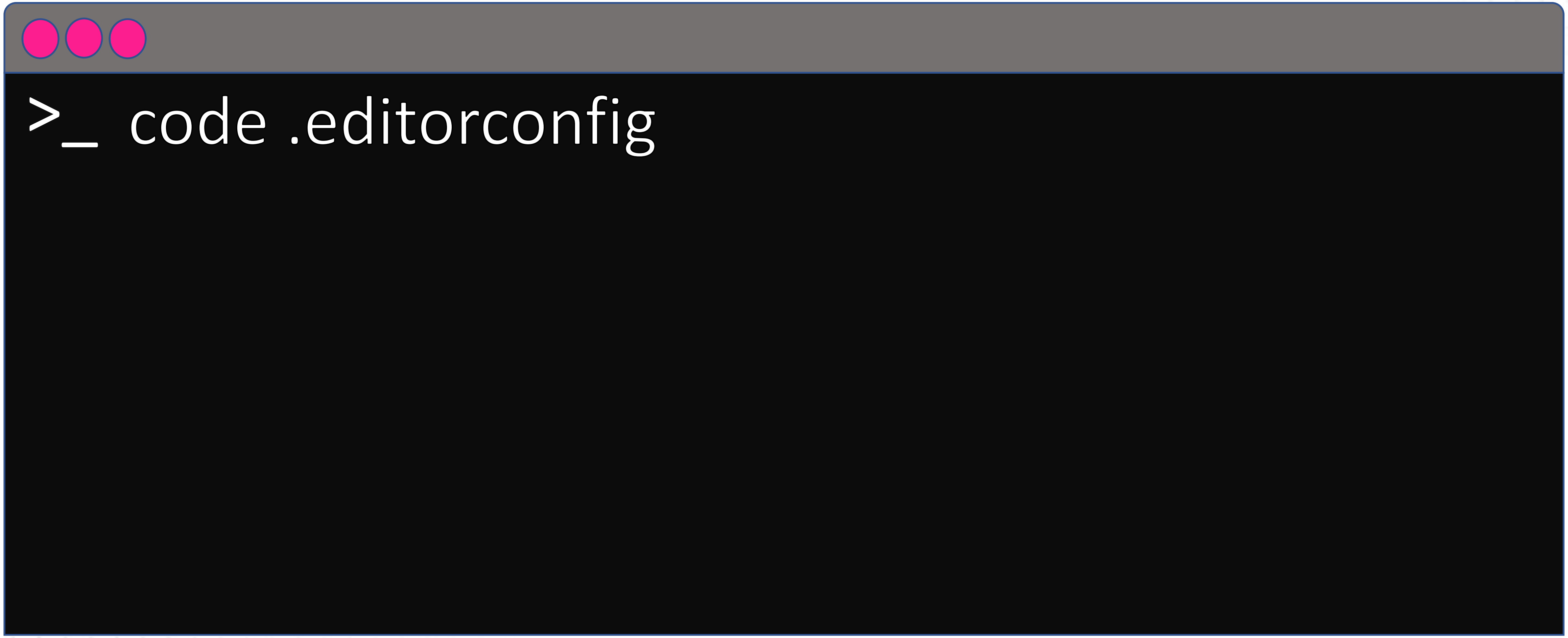
```
>_ net start w3svc
```

Editor Config



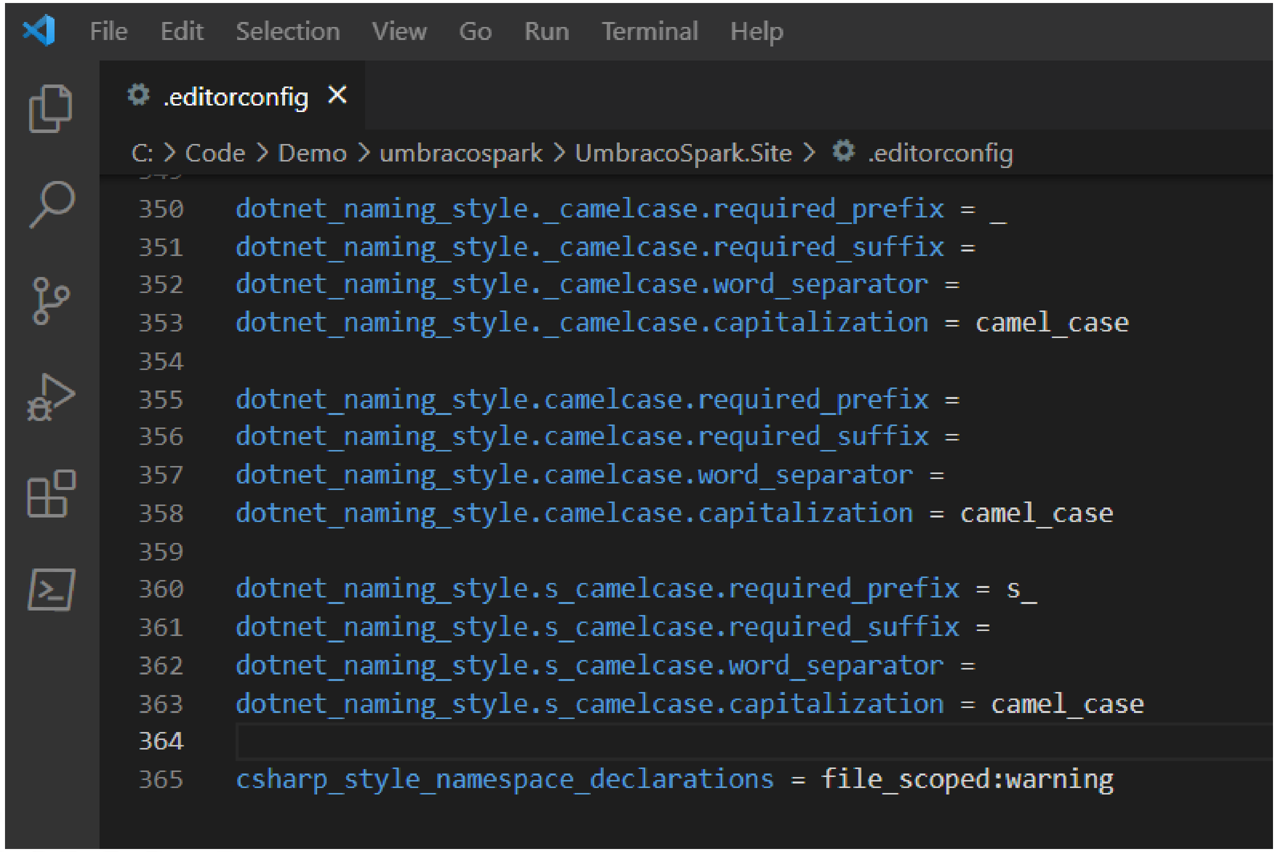
```
>_ dotnet new editorconfig

The template "EditorConfig file" was created successfully.
```

Edit the file in VS Code



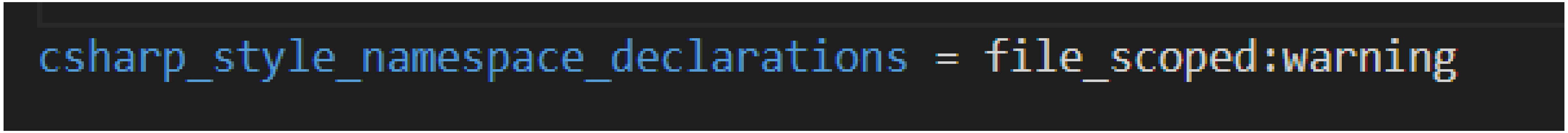
```
>_ code .editorconfig
```

Edit the file in VS Code


File
Edit
Selection
View
Go
Run
Terminal
Help
.editorconfig
C: > Code > Demo > umbracospark > UmbracoSpark.Site > .editorconfig

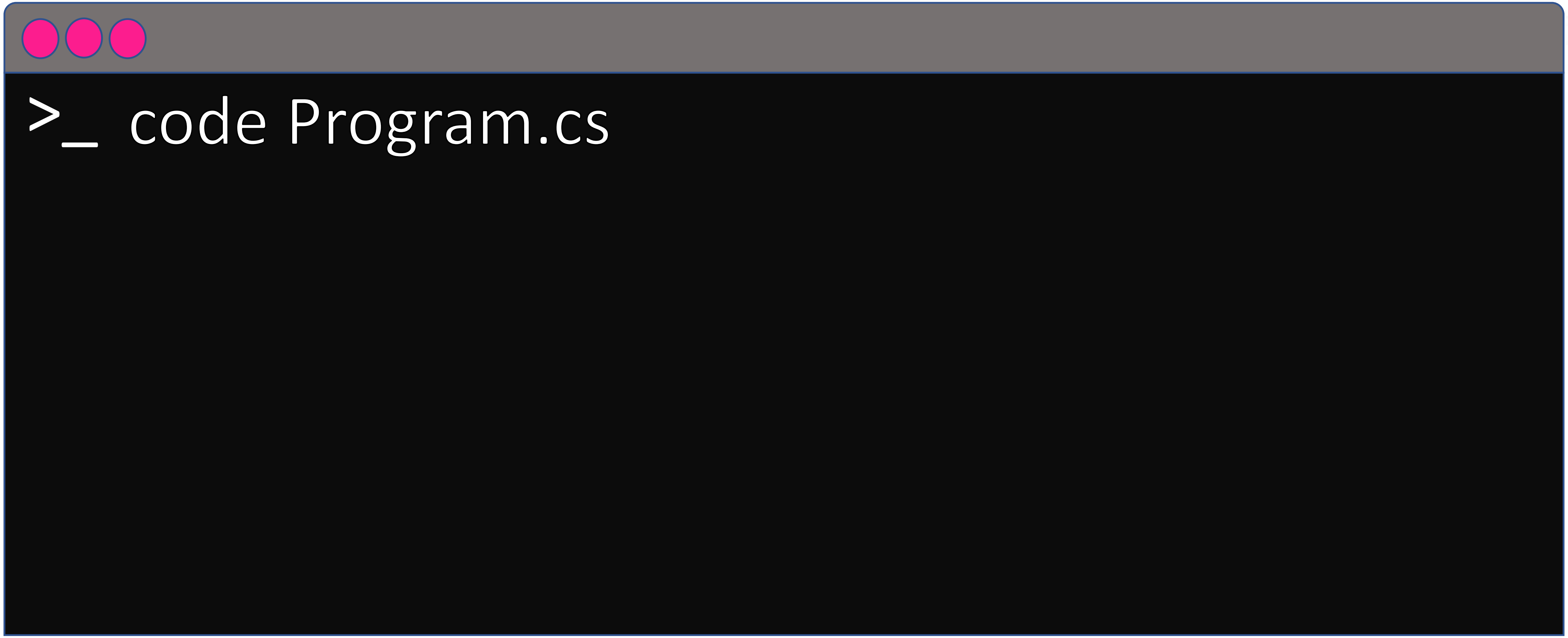
```
dotnet_naming_style._camelcase.required_prefix = _
dotnet_naming_style._camelcase.required_suffix =
dotnet_naming_style._camelcase.word_separator =
dotnet_naming_style._camelcase.capitalization = camel_case

dotnet_naming_style.camelcase.required_prefix =
dotnet_naming_style.camelcase.required_suffix =
dotnet_naming_style.camelcase.word_separator =
dotnet_naming_style.camelcase.capitalization = camel_case

dotnet_naming_style.s_camelcase.required_prefix = s_
dotnet_naming_style.s_camelcase.required_suffix =
dotnet_naming_style.s_camelcase.word_separator =
dotnet_naming_style.s_camelcase.capitalization = camel_case

csharp_style_namespace_declarations = file_scoped:warning
```

Edit the file in VS Code and add this line



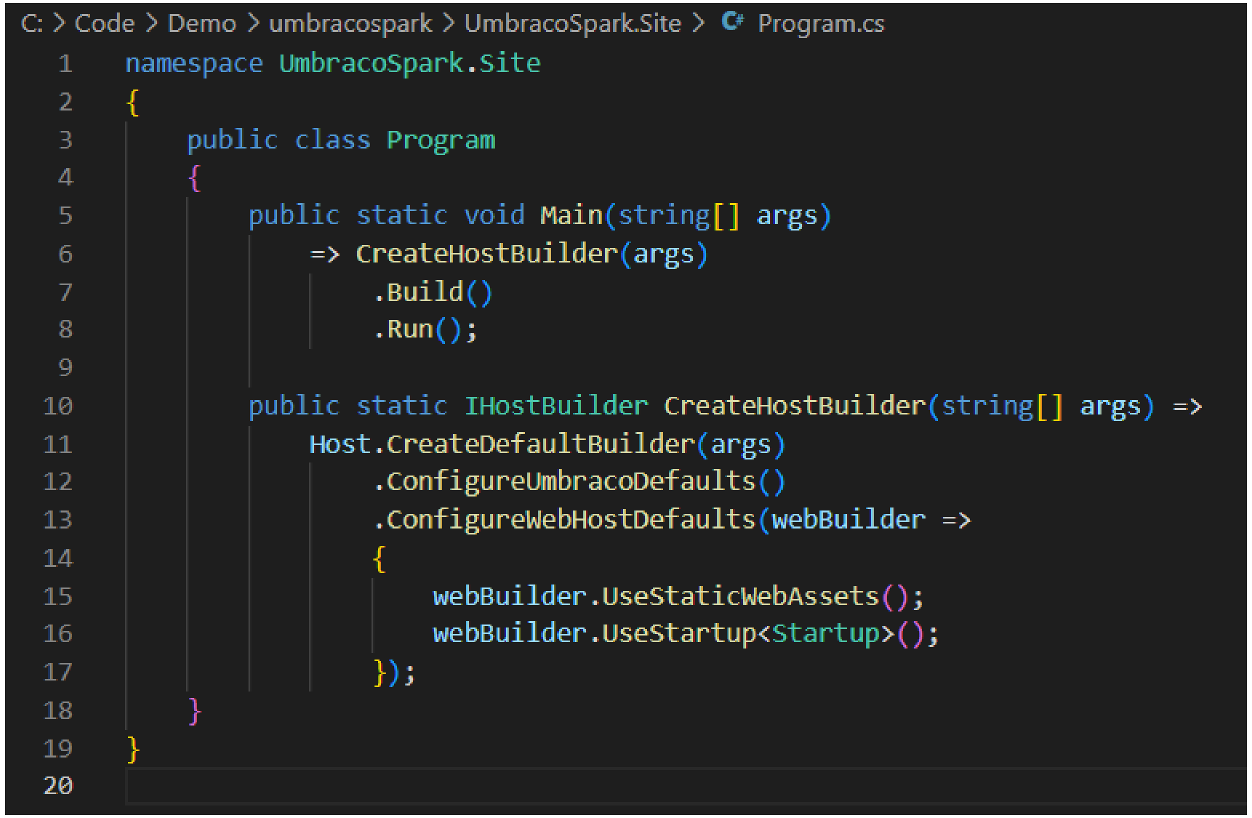
```
csharp_style_namespace_declarations = file_scoped:warning
```

Now open the Program.cs file in VS Code



```
>_ code Program.cs
```

Look at the namespace declaration

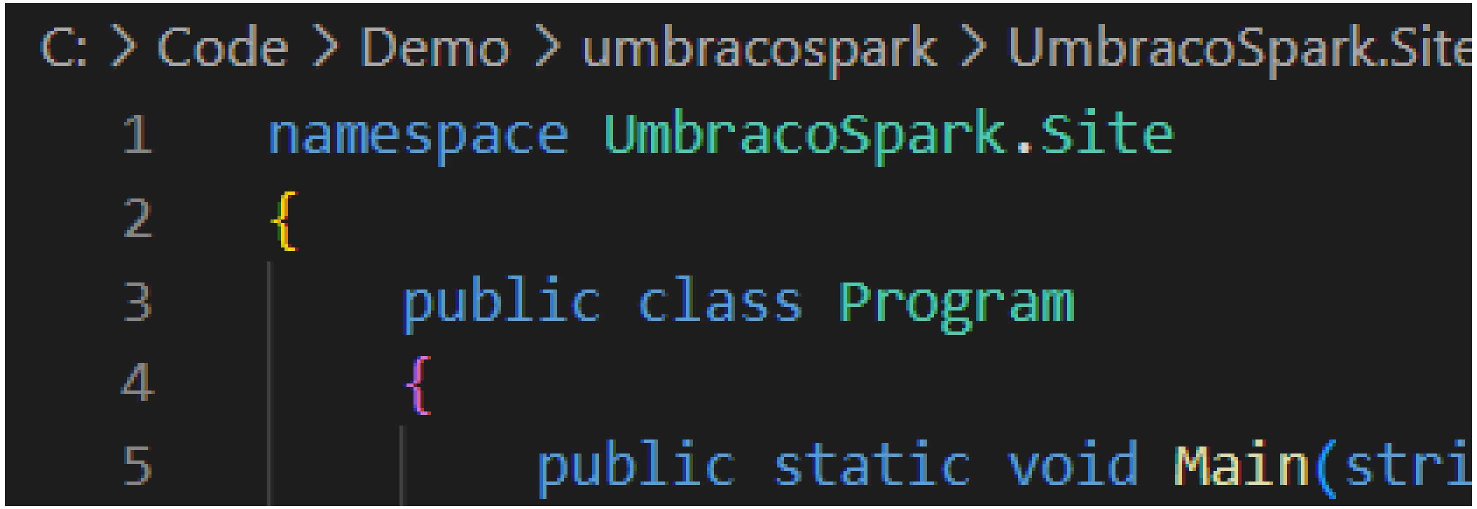


C: > Code > Demo > umbracospark > UmbracoSpark.Site > Program.cs

```csharp
namespace UmbracoSpark.Site
{
    public class Program
    {
        public static void Main(string[] args)
            => CreateHostBuilder(args)
                .Build()
                .Run();

        public static IHostBuilder CreateHostBuilder(string[] args) =>
            Host.CreateDefaultBuilder(args)
                .ConfigureUmbracoDefaults()
                .ConfigureWebHostDefaults(webBuilder =>
                {
                    webBuilder.UseStaticWebAssets();
                    webBuilder.UseStartup<Startup>();
                });
    }
}

```

#SPARKQA

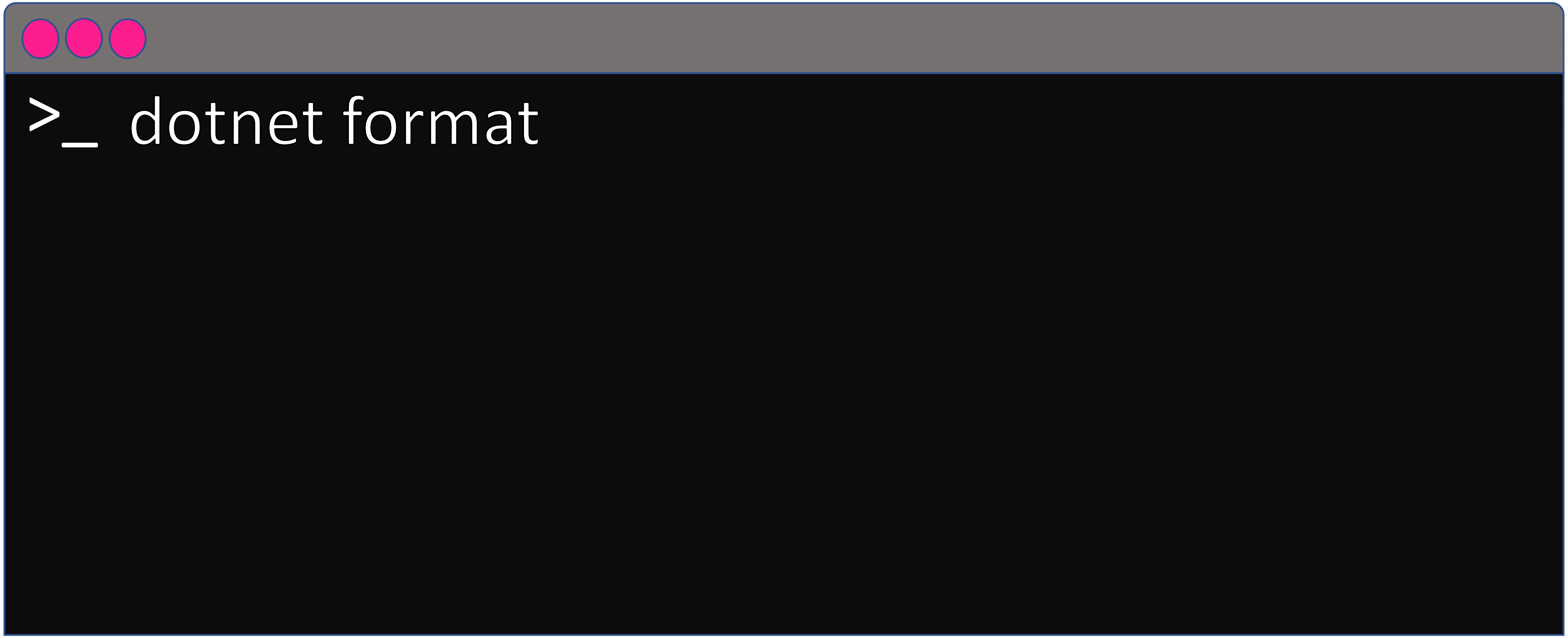
Look at the namespace declaration



C: > Code > Demo > umbracospark > UmbracoSpark.Site

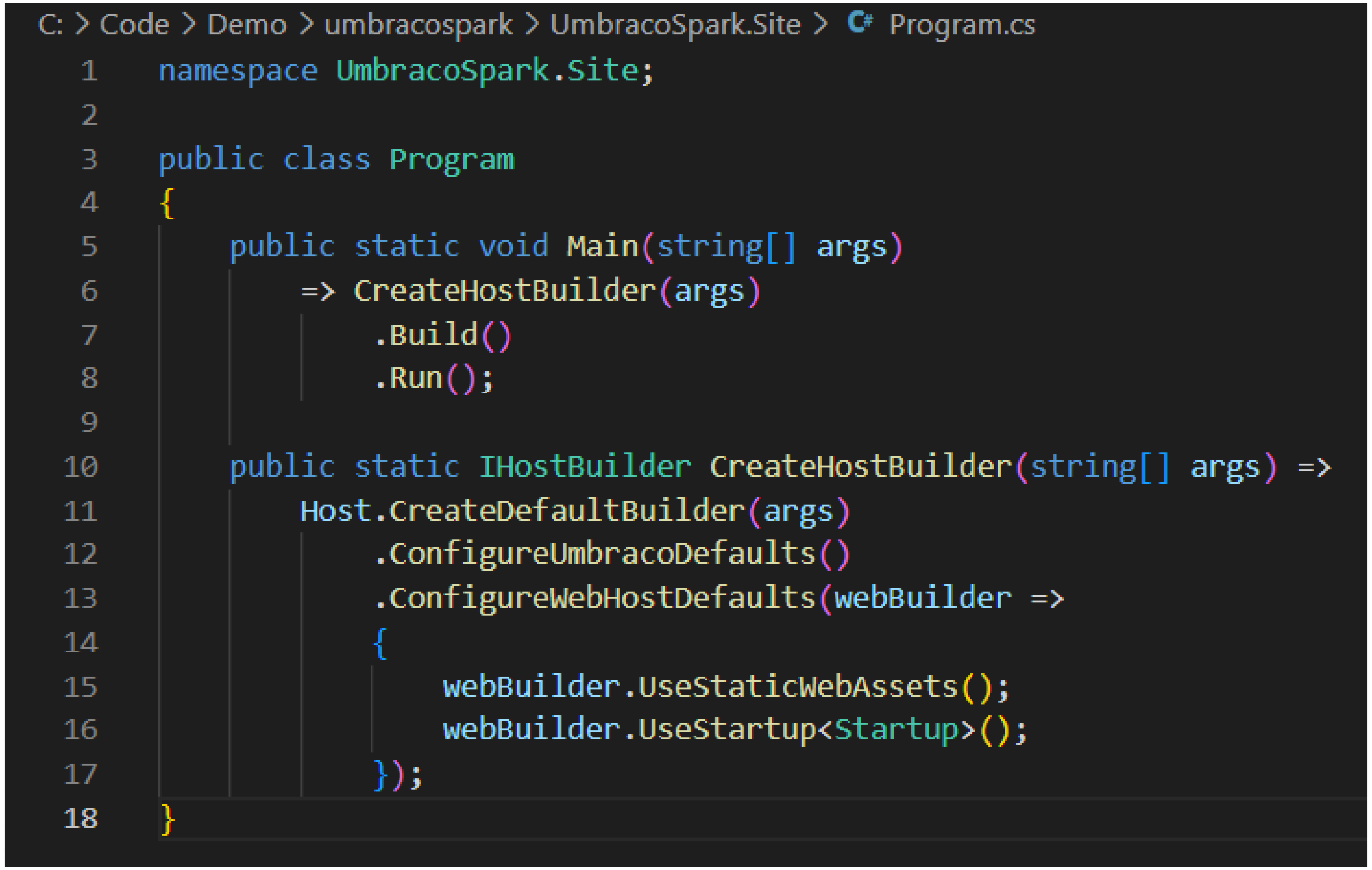
```
namespace UmbracoSpark.Site
{
    public class Program
    {
        public static void Main(stri
```

Run dotnet format



```
>_ dotnet format
```

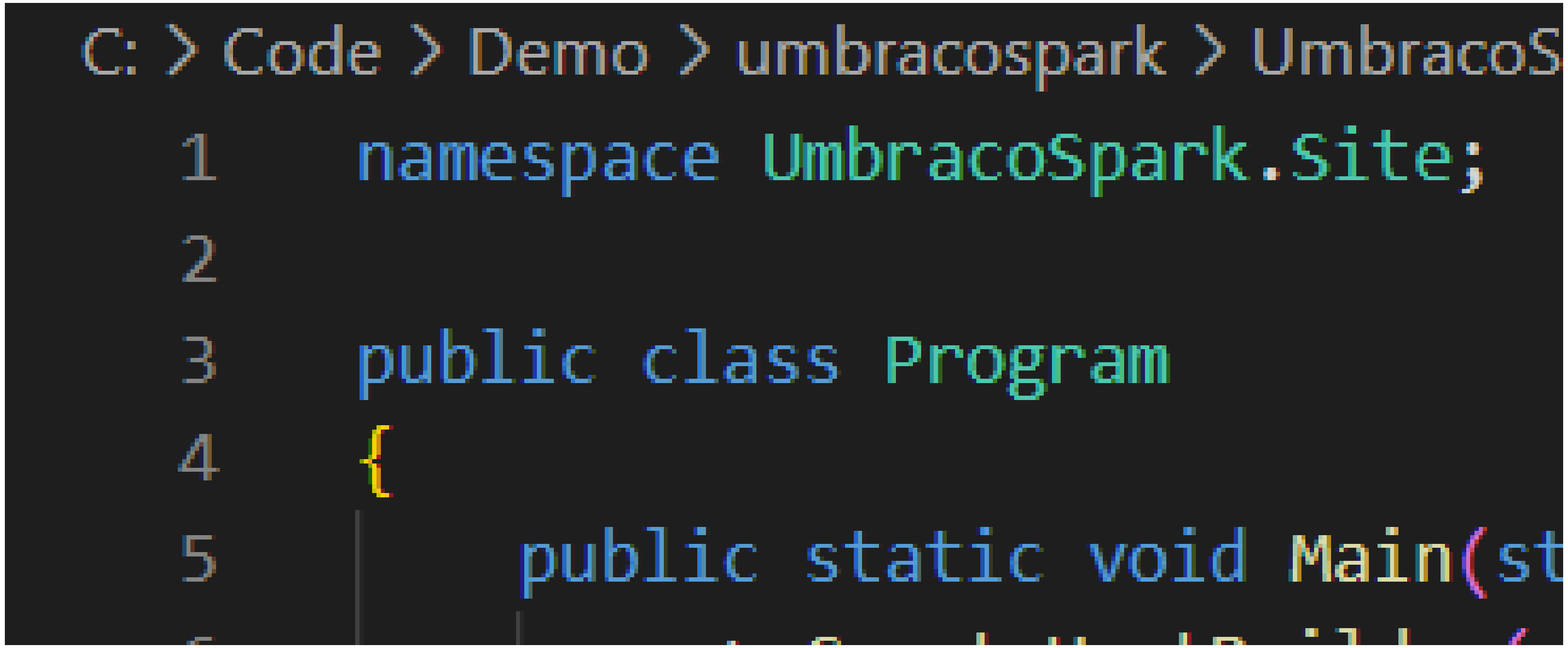
Look at the namespace again



C: > Code > Demo > umbracospark > UmbracoSpark.Site > Program.cs

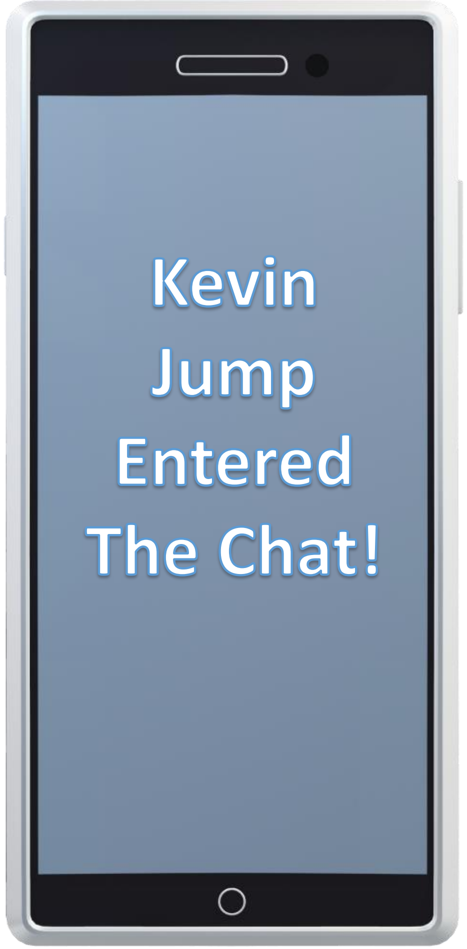
```csharp
namespace UmbracoSpark.Site;

public class Program
{
    public static void Main(string[] args)
        => CreateHostBuilder(args)
            .Build()
            .Run();

    public static IHostBuilder CreateHostBuilder(string[] args) =>
        Host.CreateDefaultBuilder(args)
            .ConfigureUmbracoDefaults()
            .ConfigureWebHostDefaults(webBuilder =>
            {
                webBuilder.UseStaticWebAssets();
                webBuilder.UseStartup<Startup>();
            });
}
```

It is now file scoped



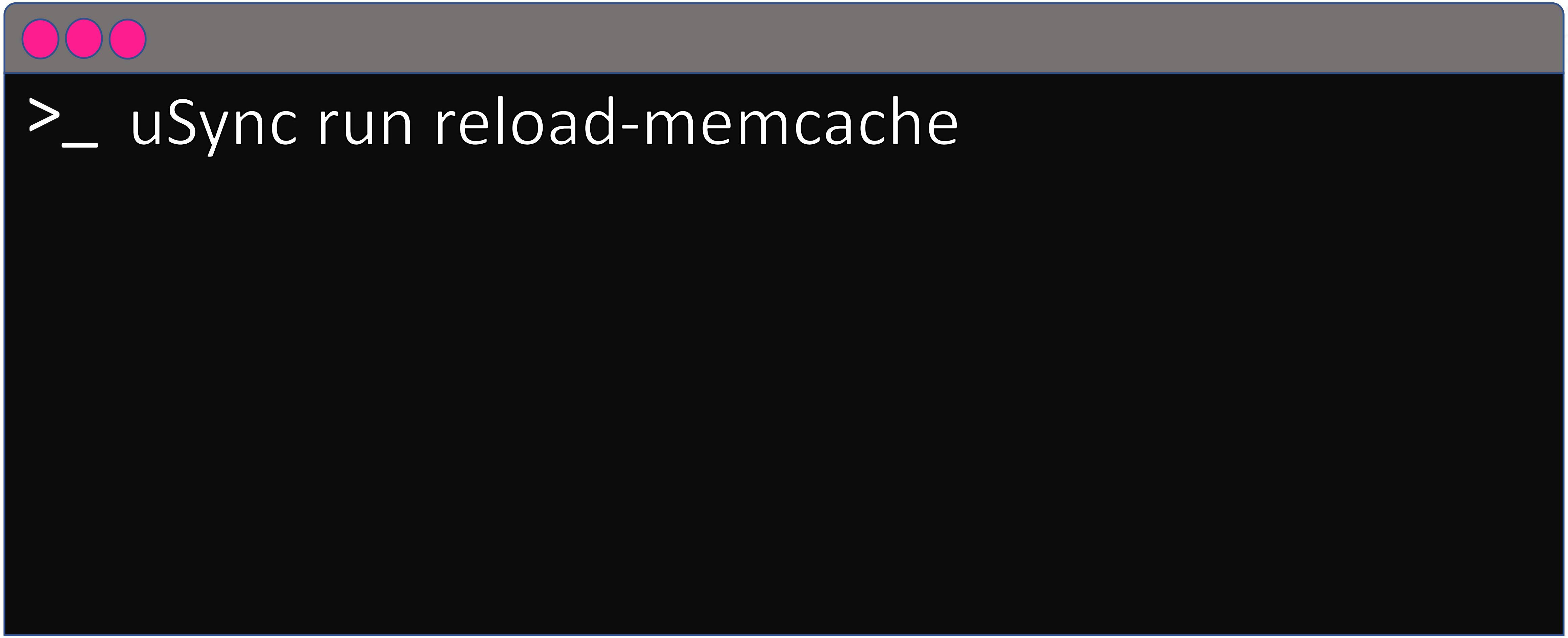
```
C: > Code > Demo > umbracospark > UmbracoS
namespace UmbracoSpark.Site;

public class Program
{
    public static void Main(st
```

What else can we do?



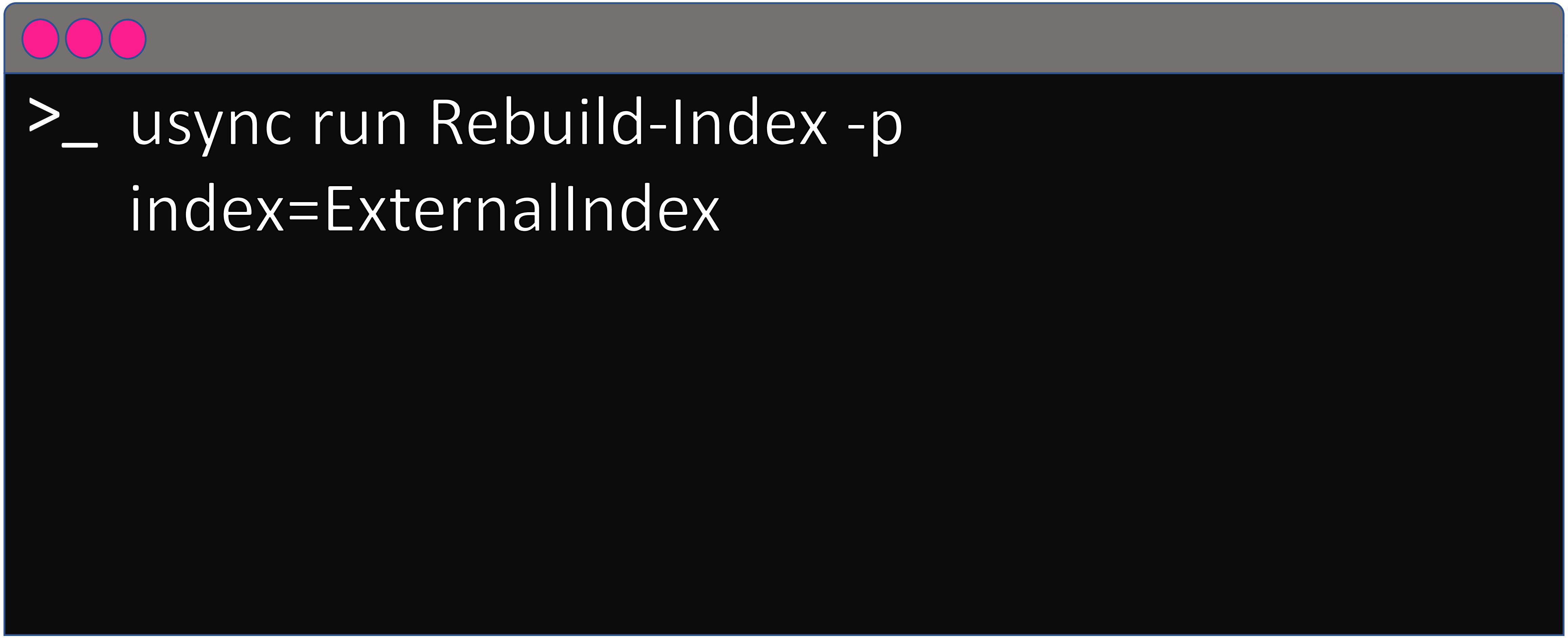
Kevin Jump Entered The Chat!

Lets reload the Umbraco memory cache?



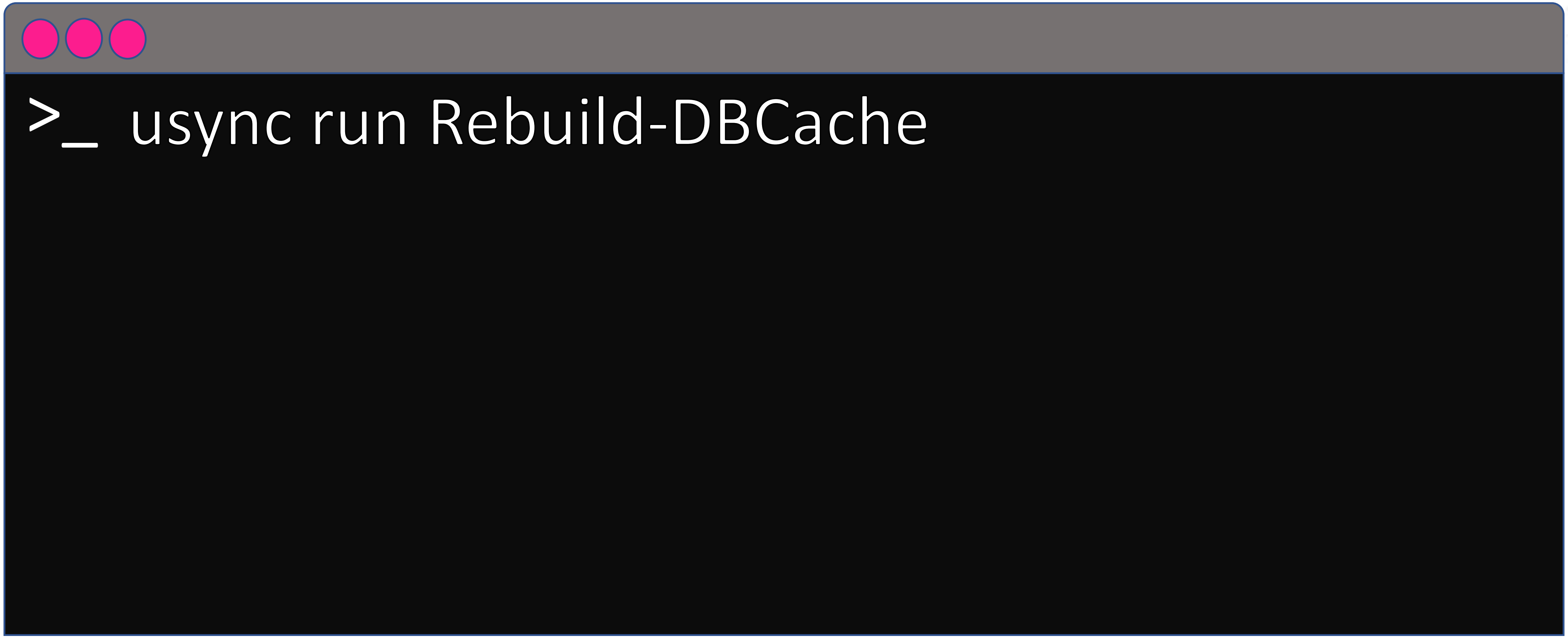
```
>_ uSync run reload-memcache
```

Or rebuild the ExternalIndex?



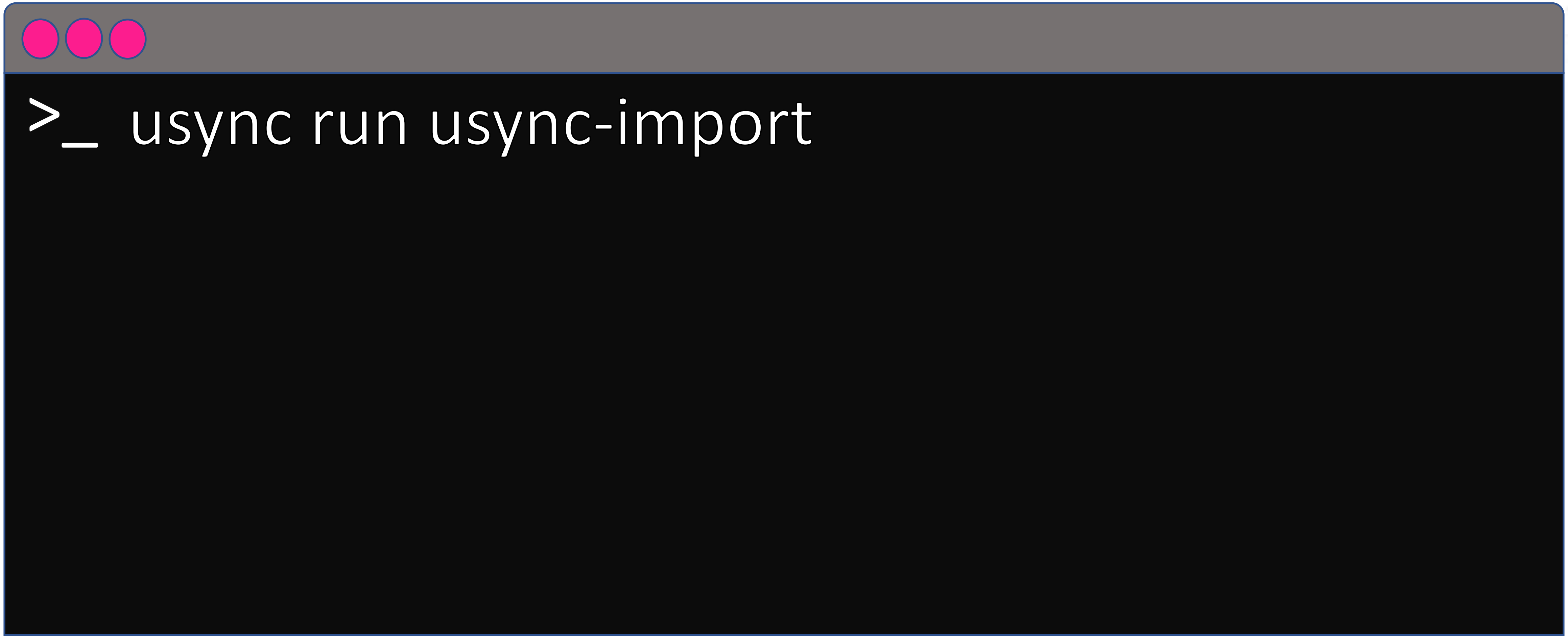
```
>_ usync run Rebuild-Index -p
   index=ExternalIndex
```

Or even rebuild the DB cache?



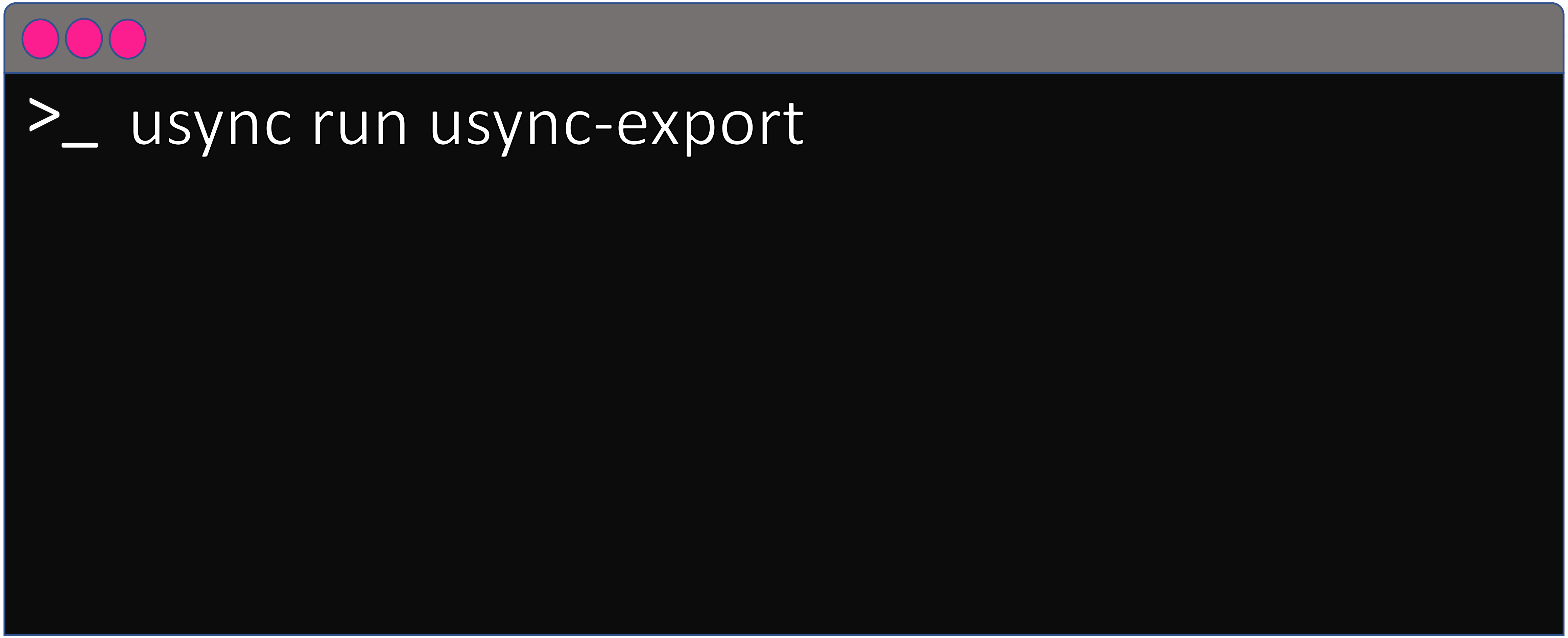
```
>_ usync run Rebuild-DBCache
```

We could import the uSync files after publish



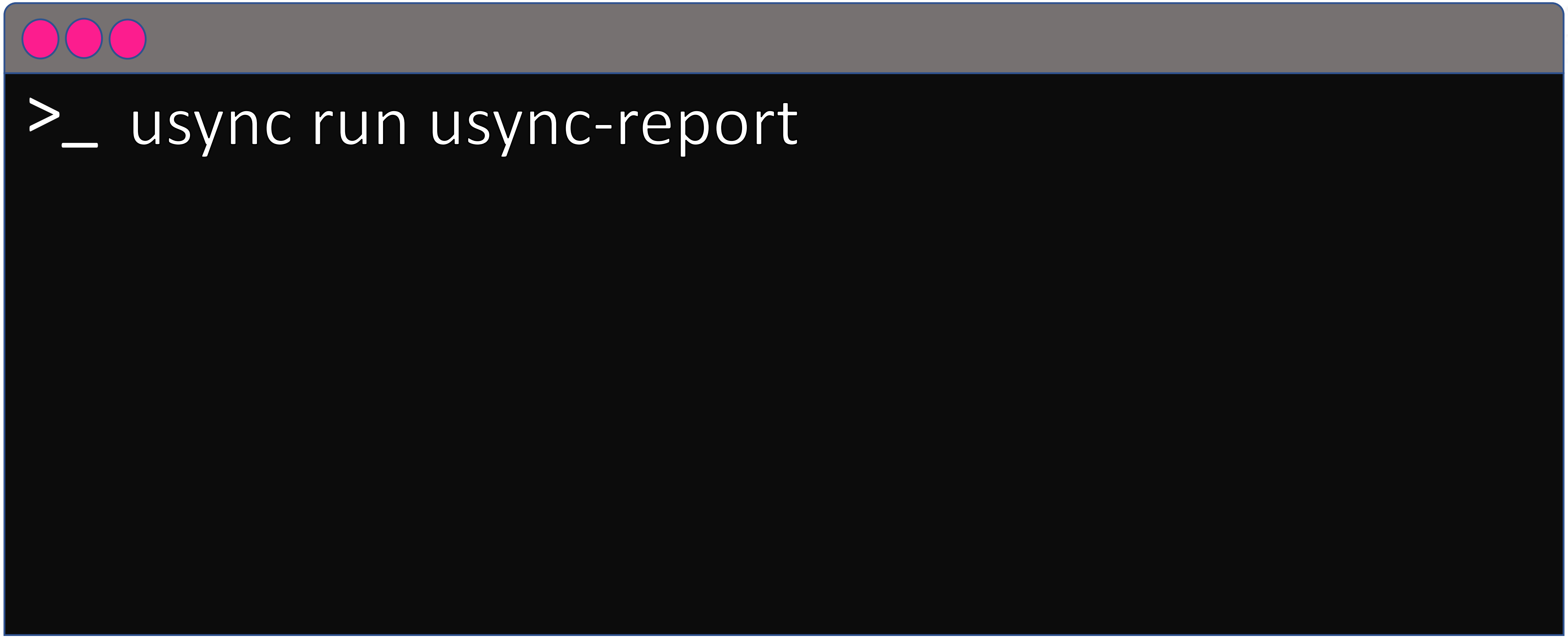
```
>_ usync run usync-import
```

Or export them



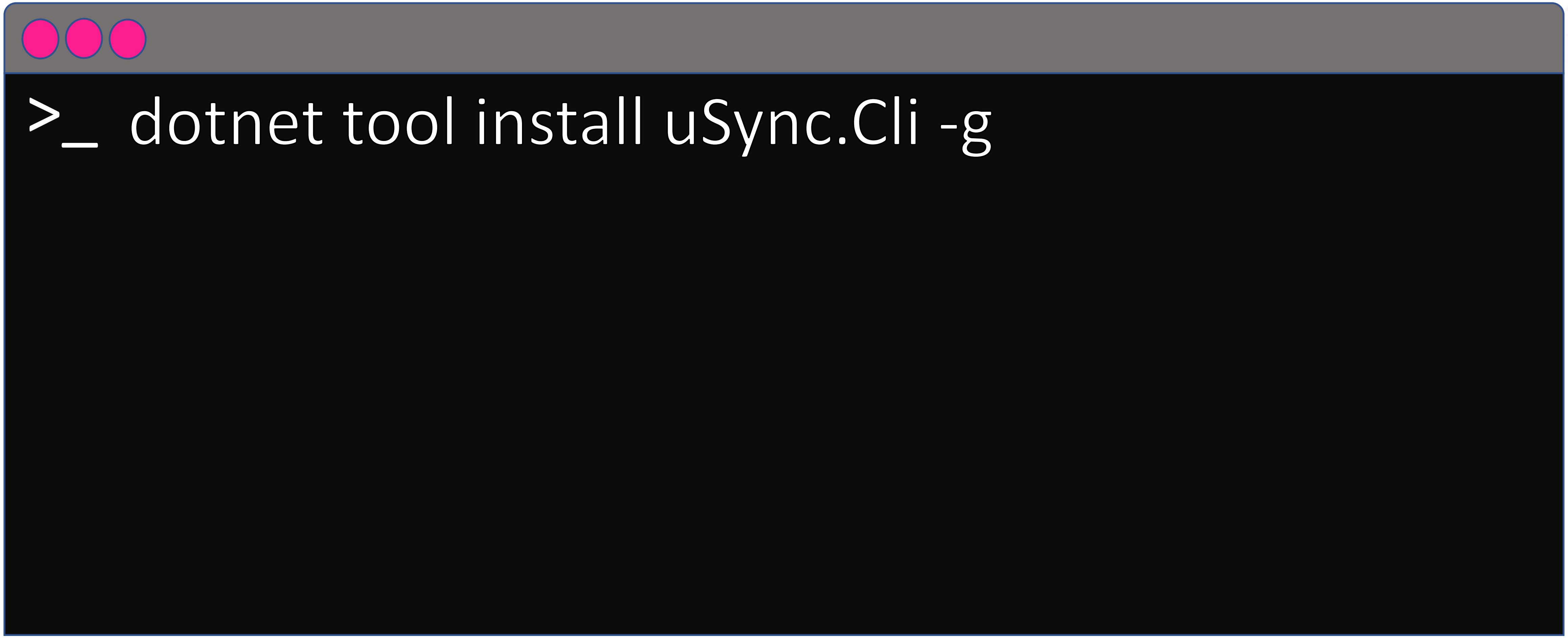
```
>_ usync run usync-export
```

Or just do a report



```
>_ usync run usync-report
```

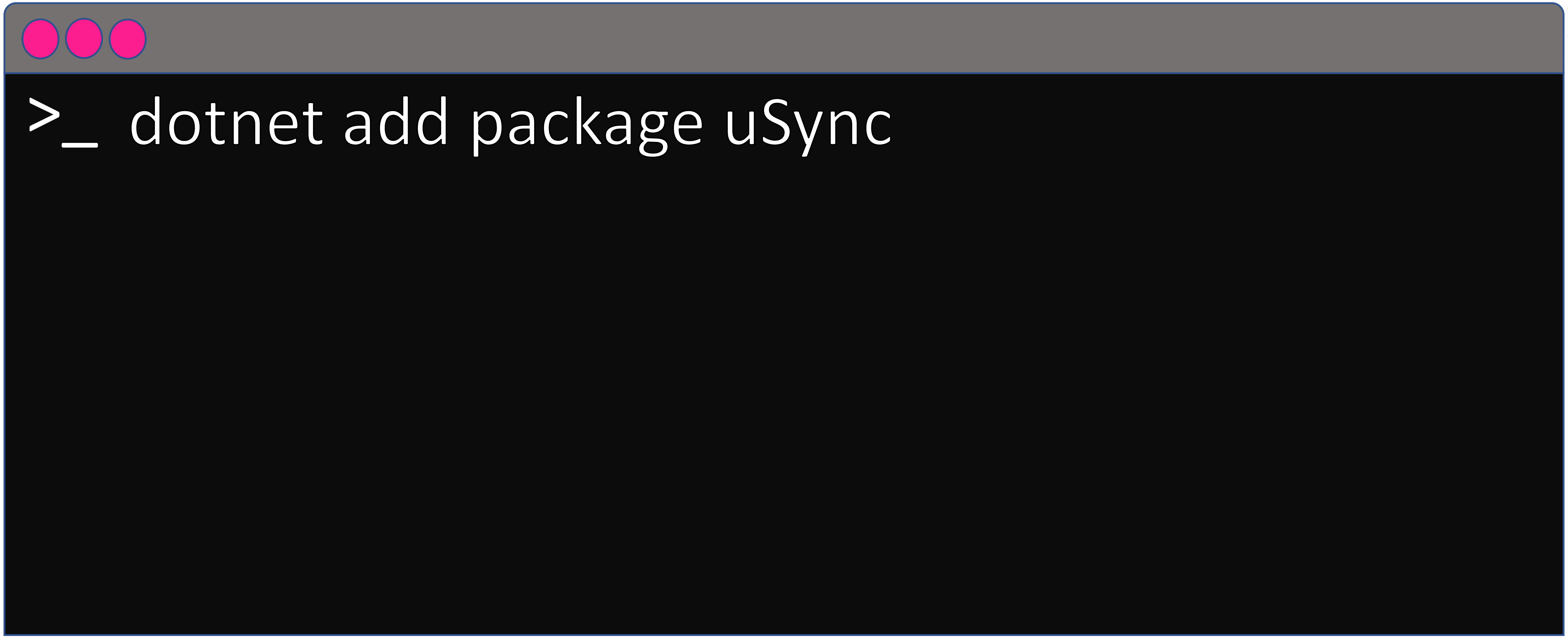
Install the uSync cli tool



```
>_ dotnet tool install uSync.Cli -g
```

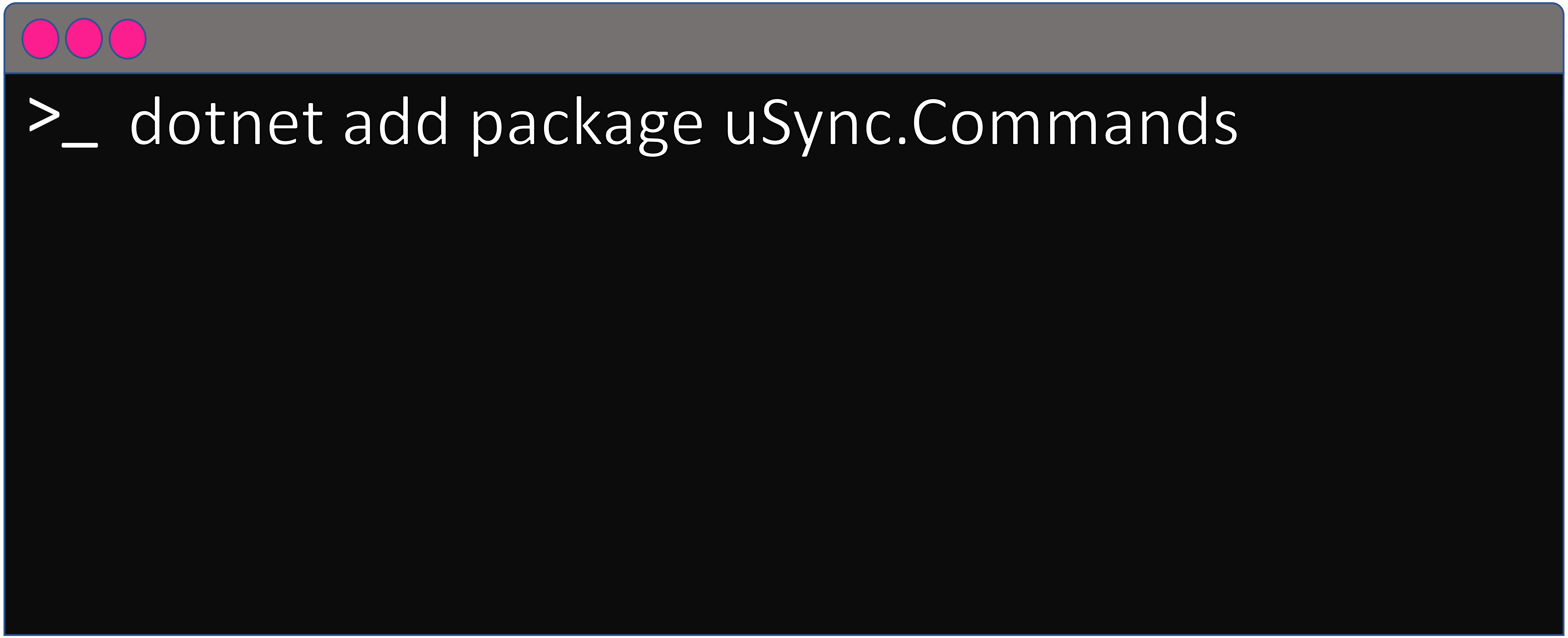
#SPARKQA

Install uSync in your project



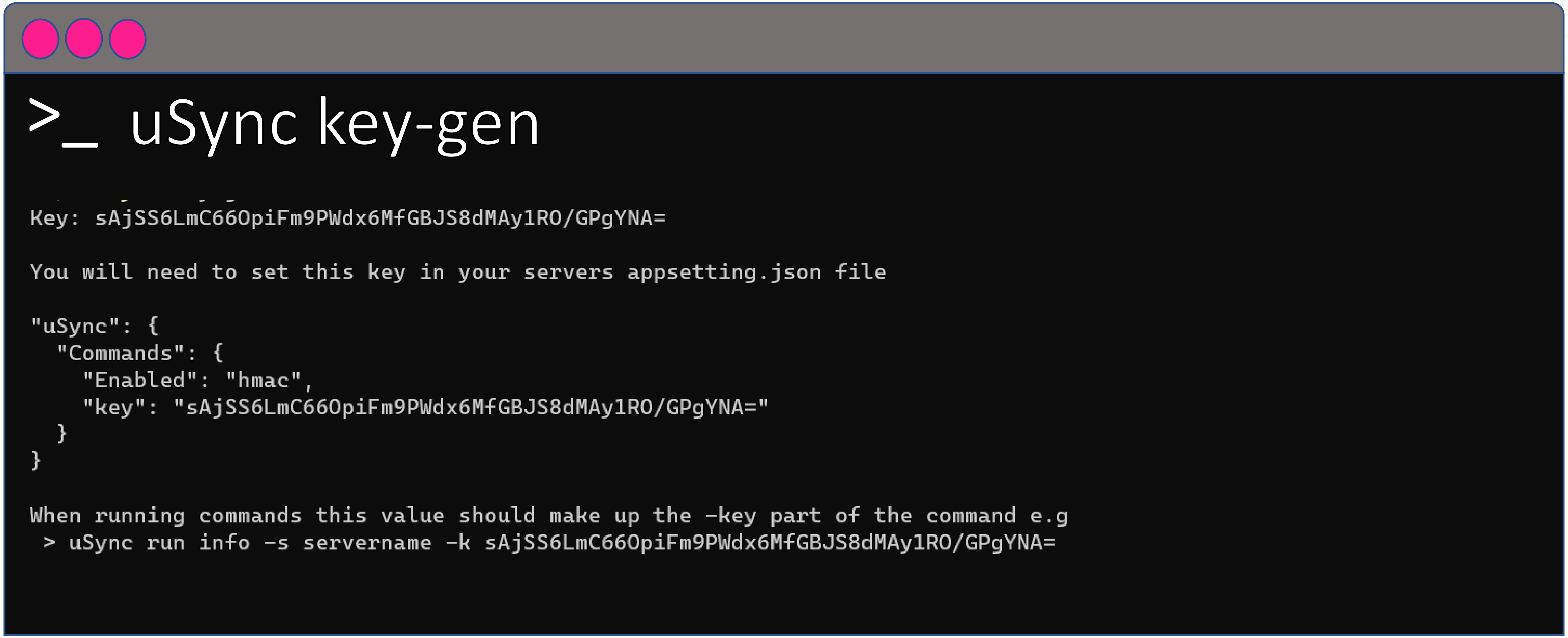
```
>_ dotnet add package uSync
```

Install the uSync Commands package too



```
>_ dotnet add package uSync.Commands
```

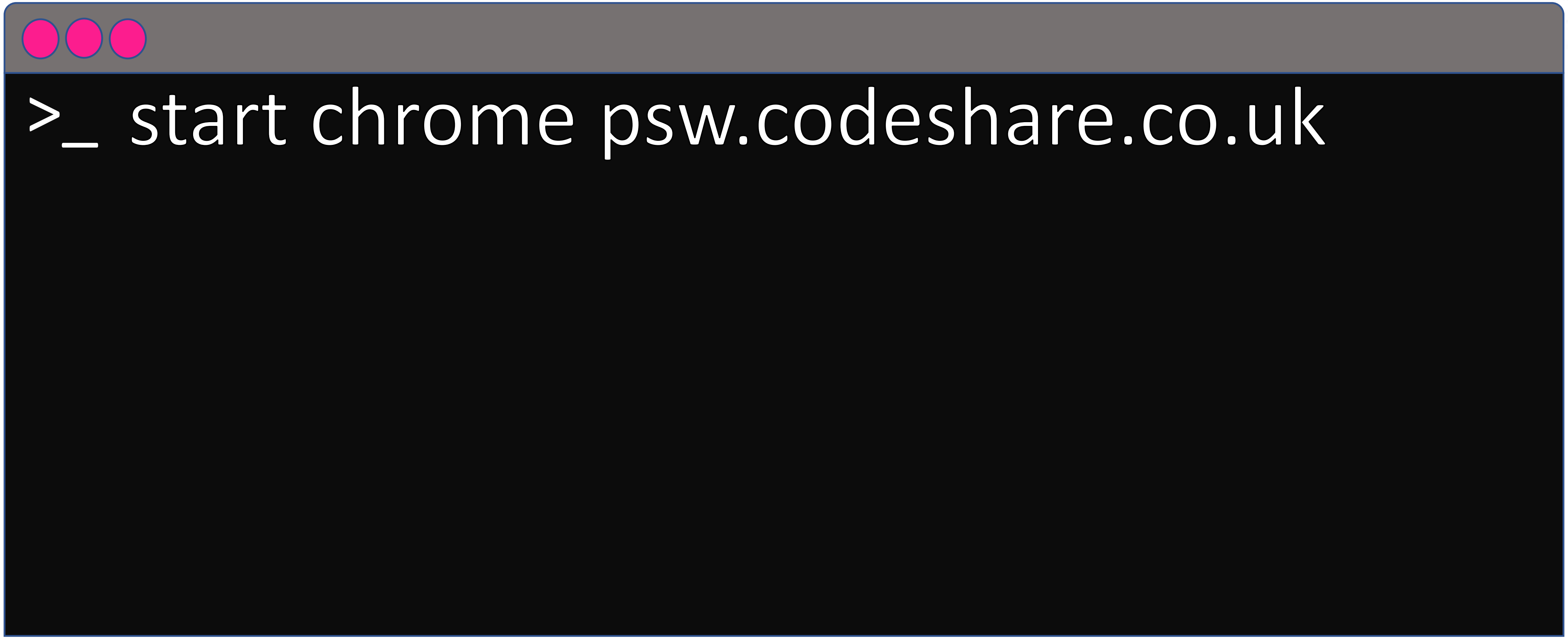
Generate a key to use in the commands



>_ uSync key-gen

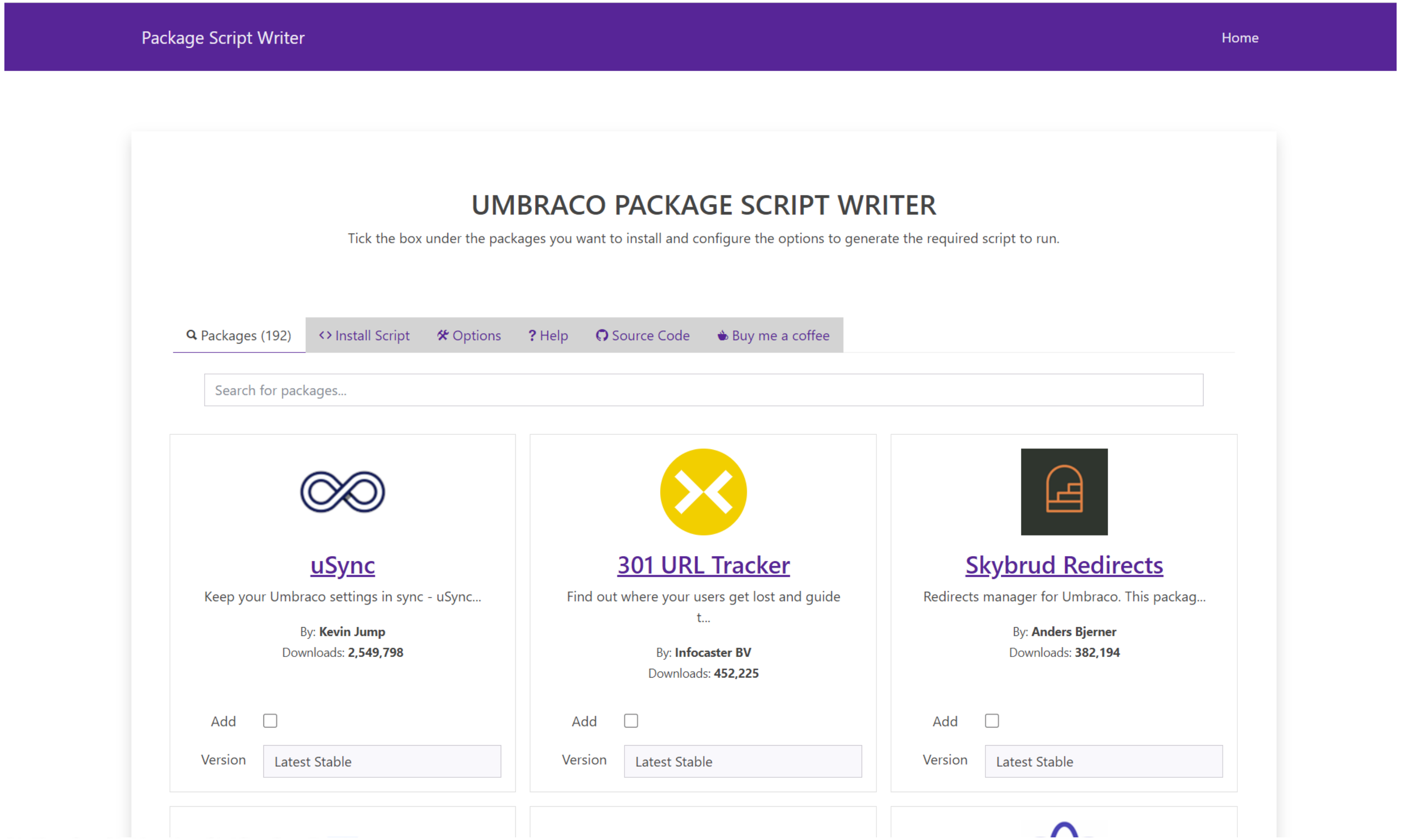
```
Key: sAjSS6LmC66OpiFm9PWdx6MfGBJS8dMAy1RO/GPgYNA=

You will need to set this key in your servers appsetting.json file

"uSync": {
  "Commands": {
    "Enabled": "hmac",
    "key": "sAjSS6LmC66OpiFm9PWdx6MfGBJS8dMAy1RO/GPgYNA="
  }
}

When running commands this value should make up the -key part of the command e.g
 > uSync run info -s servername -k sAjSS6LmC66OpiFm9PWdx6MfGBJS8dMAy1RO/GPgYNA=
```

Help with Umbraco commands



```
>_ start chrome psw.codeshare.co.uk
```

Package Script Writer



Package Script Writer

Home

UMBRACO PACKAGE SCRIPT WRITER

Tick the box under the packages you want to install and configure the options to generate the required script to run.

Packages (192) | <> Install Script | Options | ? Help | Source Code | Buy me a coffee

Search for packages...

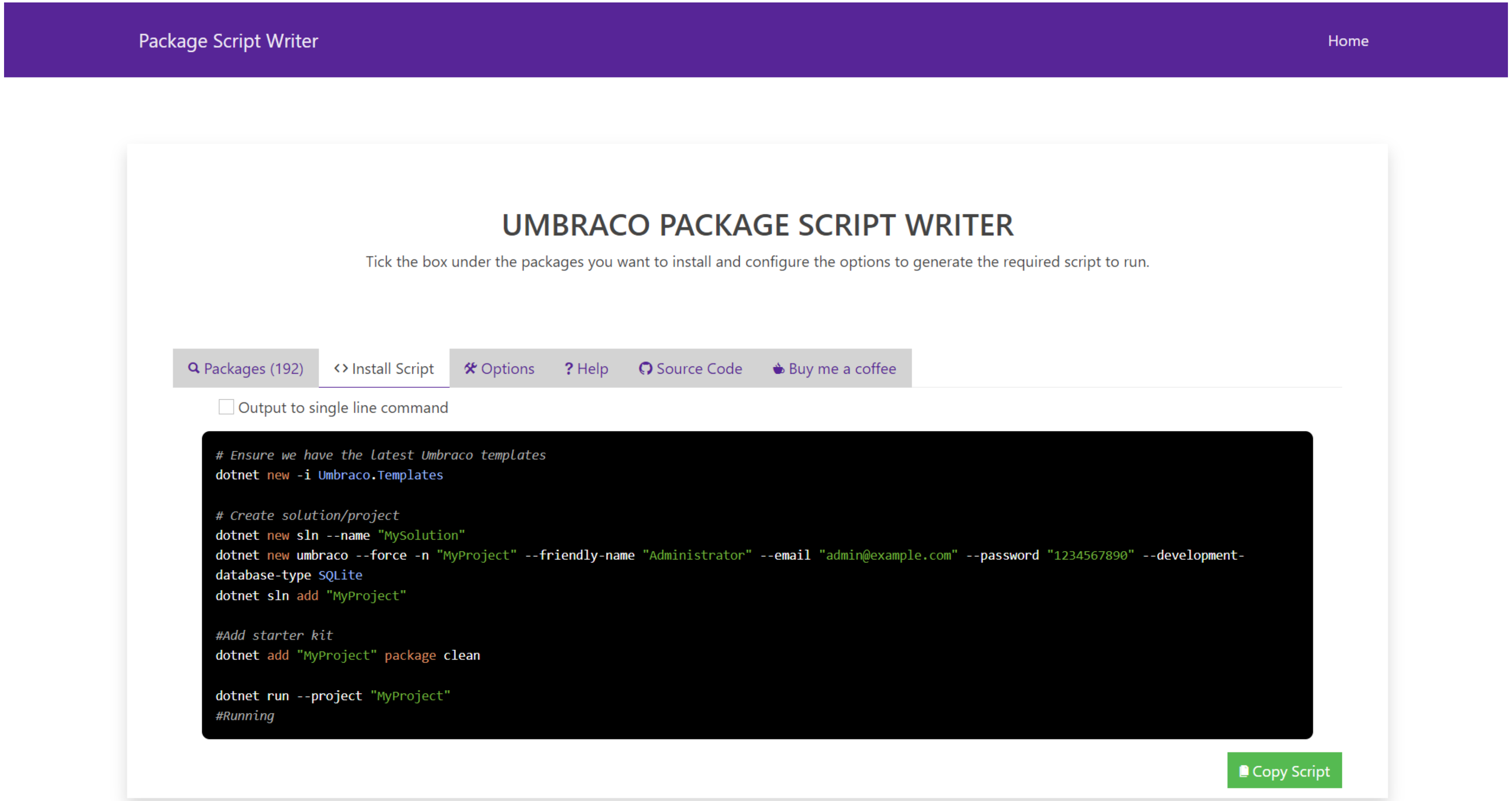
uSync
Keep your Umbraco settings in sync - uSync...
By: **Kevin Jump**
Downloads: **2,549,798**
Add □
Version: Latest Stable

301 URL Tracker
Find out where your users get lost and guide t...
By: **Infocaster BV**
Downloads: **452,225**
Add □
Version: Latest Stable

Skybrud Redirects
Redirects manager for Umbraco. This packag...
By: **Anders Bjerner**
Downloads: **382,194**
Add □
Version: Latest Stable

#SPARKQA

Copy and paste commands



UMBRACO PACKAGE SCRIPT WRITER

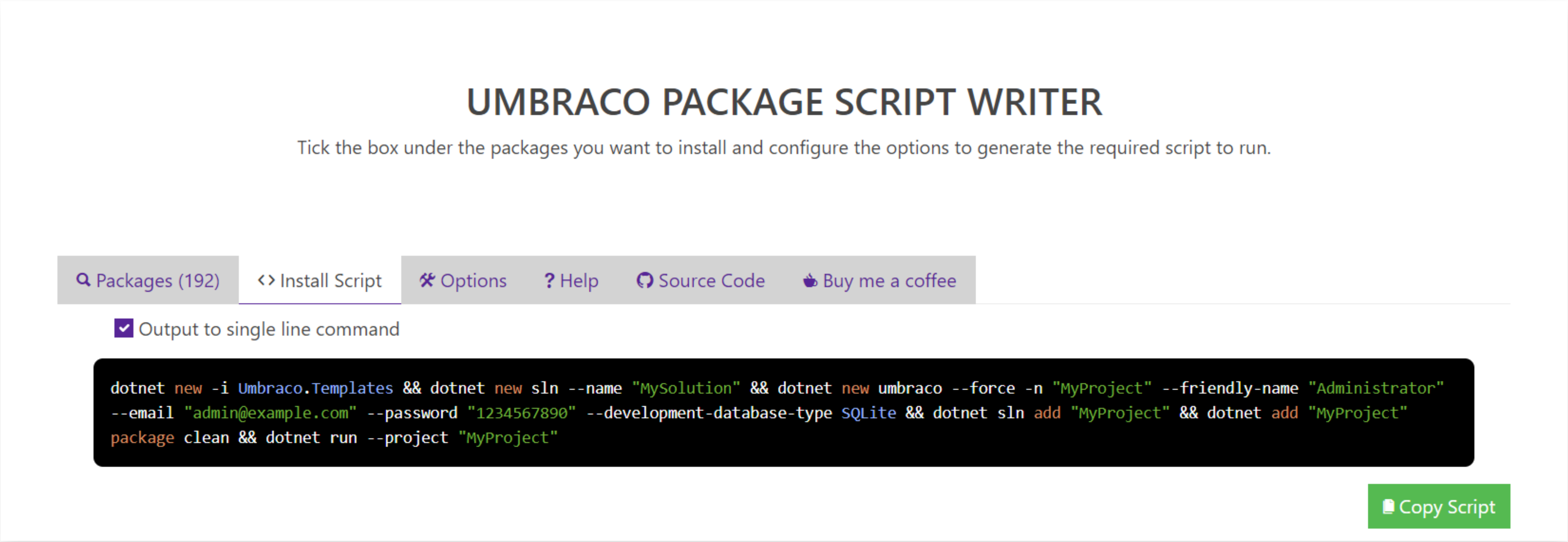
Tick the box under the packages you want to install and configure the options to generate the required script to run.

□ Output to single line command

```
# Ensure we have the latest Umbraco templates
dotnet new -i Umbraco.Templates

# Create solution/project
dotnet new sln --name "MySolution"
dotnet new umbraco --force -n "MyProject" --friendly-name "Administrator" --email "admin@example.com" --password "1234567890" --development-database-type SQLite
dotnet sln add "MyProject"

#Add starter kit
dotnet add "MyProject" package clean

dotnet run --project "MyProject"
#Running
```

#SPARKQA

Single line command



UMBRACO PACKAGE SCRIPT WRITER

Tick the box under the packages you want to install and configure the options to generate the required script to run.

☑ Output to single line command

```
dotnet new -i Umbraco.Templates && dotnet new sln --name "MySolution" && dotnet new umbraco --force -n "MyProject" --friendly-name "Administrator"
--email "admin@example.com" --password "1234567890" --development-database-type SQLite && dotnet sln add "MyProject" && dotnet add "MyProject"
package clean && dotnet run --project "MyProject"
```

#SPARKQA

Thanks for watching

Paul Seal

ClerksWell.

umbraco GIBE Great State MENTOR Rock Solid Knowledge Umbhost

#SPARKQA