"# MSLearnBuildDotNetAppsWithCSharp"

Module: Introduction to .NET

Unit 3 of 7

Tip

Fortunately, you can programmatically prevent ugly error messages from reaching your users. Learn more by searching for tutorials and documentation about "structured exception handling."

How .NET works at runtime

Dúvida:

\*"Locates the program's entry point and begins running each instruction in the proper sequence."?

Unit 4 of 7

Note

In this module, you use a built-in version of Try .NET. If you want to experiment with the application after you finish this exercise, go to https://try.dot.net. Try .NET is a great way to try out small code examples without installing anything on your local computer.

Unit 5 of 7

Choose .NET for cloud and AI applications

The Azure SDK for .NET allows developers to provision and manage Azure resources. Alternatively, Azure App Service and Azure Functions can host applications that are built by using .NET languages.

ML.NET is a free machine learning library for .NET languages. It enables model-based capabilities for machine learning analysis and prediction.

Module: Create a new .NET project and work with dependencies

Unit 2 of 7

Evaluate a package

You can learn more about a package before installing it by going to https://www.nuget.org/packages/<package name>. This URL will take you to a detailed page for the package. Select the Dependencies drop-down list to see which packages it relies on to function.

Dúvida

The number of listed dependencies might not tell the whole truth. If you download a package, you might end up with a package dependency that contains dozens of packages. Why is that? Every package has a list of dependencies. To ensure that you can use a package, all dependencies are crawled and downloaded when you run the dotnet add package <package name> command.

NuGet registry and dotnet tool

When you run dotnet add package <name of dependency>, .NET goes to a global registry called the NuGet.org registry and looks for the code to download. It's located at https://nuget.org. You can also browse through this page for packages, if you visit it by using a browser. Every package has a dedicated website that you can go to.

Unit 3 of 7

Unit 7 of 7

**Additional resources**

* Review the official [NuGet documentation](https://docs.microsoft.com/en-us/nuget/).
* Review the official [.NET Core CLI documentation](https://docs.microsoft.com/en-us/dotnet/core/tools/).
* Browse packages available on [NuGet](https://www.nuget.org/).
* Check the [.NET website](https://dot.net/) for all things .NET.
* Try [Visual Studio Code](https://code.visualstudio.com/) for editing text and code files.

Obs.: Incluido o commando

dotnet new gitignore

[**Interactively debug .NET apps with the Visual Studio Code debugger**](https://docs.microsoft.com/en-us/learn/modules/dotnet-debug/)