How to list all installed NuGet Packages?

Asked 5 years, 8 months ago   Active 2 months ago   Viewed 42k times

How does one list all locally installed NuGet packages? Is there a NuGet equivalent of `RPM -qa` ? Within chocolatey there is the `chocolatey list -localonly` but for the life of me I cannot find the NuGet equivalent of that command.

75

`c#` `asp.net` `visual-studio` `nuget`

★

10

edited Oct 13 '16 at 7:26	asked Mar 12 '14 at 21:13
student	**Rhodope**
13.7k 10 89 148	**376** 1 3 3

8 Answers

In the *NuGet Package Manager Console*, enter the following command:

82

```
Get-Package
```

This will either print out a list of installed packages, or if none are present write the following line to the console:

```
PM> Get-Package
No packages installed.
```

For more details, have a look at the [NuGet Powershell Reference](#).

edited Mar 12 '14 at 21:35	answered Mar 12 '14 at 21:29
	Philip Allgaier
	2,437 18 46

11   Is there a way to do this using nuget.exe? – bitbonk Dec 21 '16 at 8:25

2   After running the command `Get-Package` , [packages' names are trimmed](#) when they have more than 32 characteres. Is it possible to increase the

package name column so that the output show the full names without trimming them? — Ulysses Alves Mar 16 '17 at 12:44 ✎

8   `Get-Package | Format-Table -AutoSize` will do it — Mant101 May 12 '17 at 15:05 ✎

If you just do

41

```
Get-Package
```

it will list the packages and where they are referenced. It will list the same packages over and over again if you have them referenced many times. If you want to get a clean list of all packages installed in the solution you can do

```
Get-Package | select -Unique Id, Versions
```

answered Oct 12 '16 at 8:59

bitbonk
25.4k   28   160   256

```
Get-Package -ProjectName "Your.Project.Name"
```

15

Will show the packages for the specified project.

See also: Package Manager Console PowerShell Reference

Note that each project will have a packages.config file which is used to track installed packages. If this is altered (specifically if you alter it backwards), the projects may not automatically download the correct package version. In that case, make a note of the packages required and do a uninstall-package, followed by a install-package for each.

Also, backups are your friend! ;)

answered Apr 18 '16 at 11:25

neilhighley
605   7   8

In Visual Studio,

11

- go to the **Project** or **Solution** in question
- right click, *Manage NuGet Packages*...
- on the left, you will see '*Installed Packages*'
- click on this and you will see the list

answered Mar 12 '14 at 21:39

christiandev
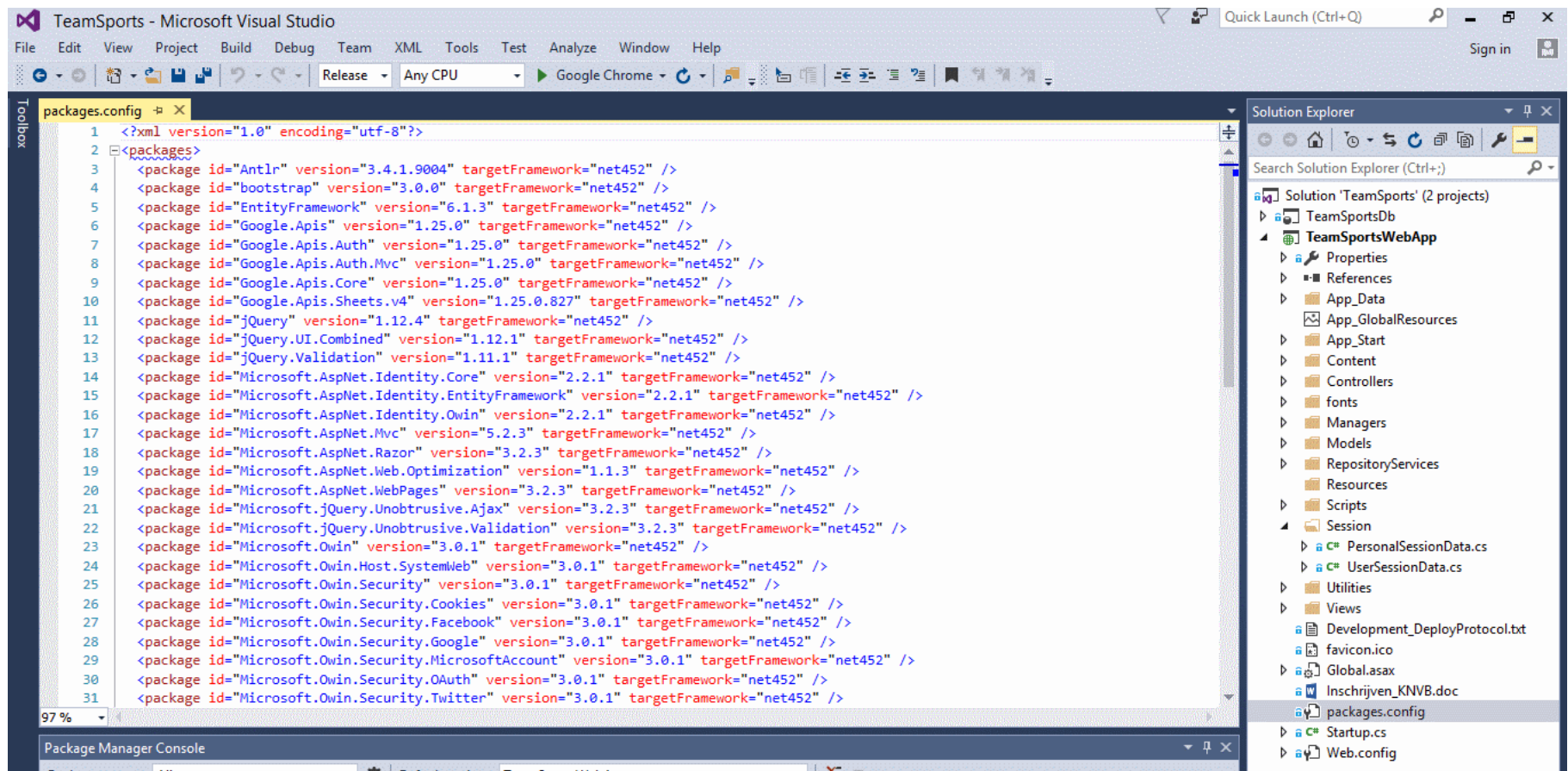**14.6k**   8   39   70

In addition to all of the given answers, there is also a clean listing in xml format of all installed packages in your VisualStudio project root folder: packages.config:

5

answered Nov 9 '17 at 10:59

wecky
324   4   6
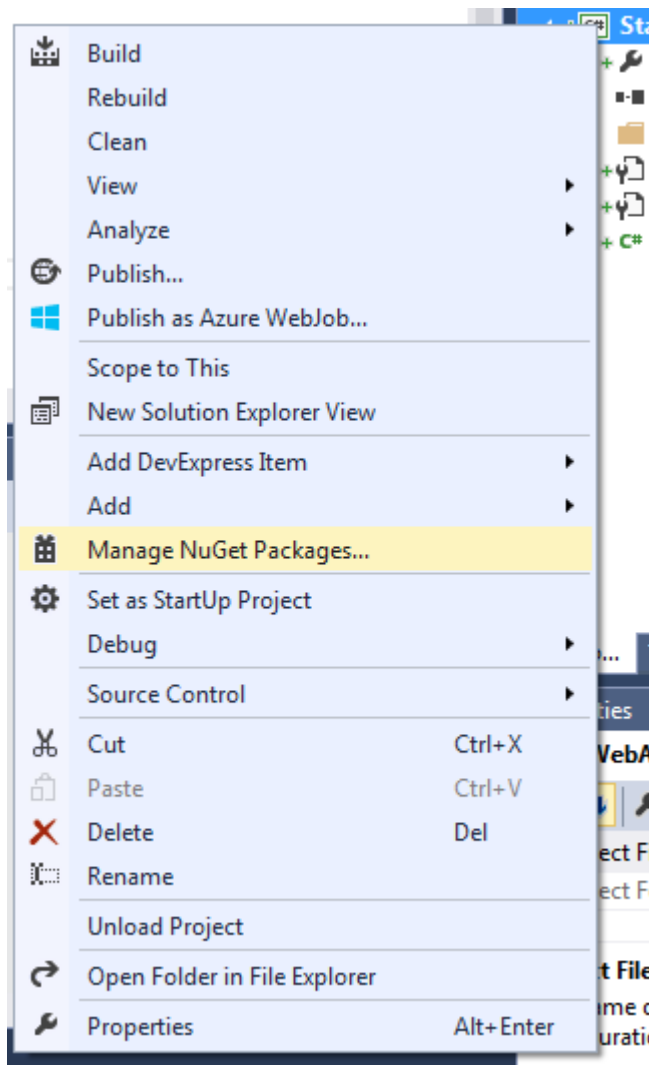
In my opinion, the best response. – etiennejcharles Aug 16 '18 at 19:07
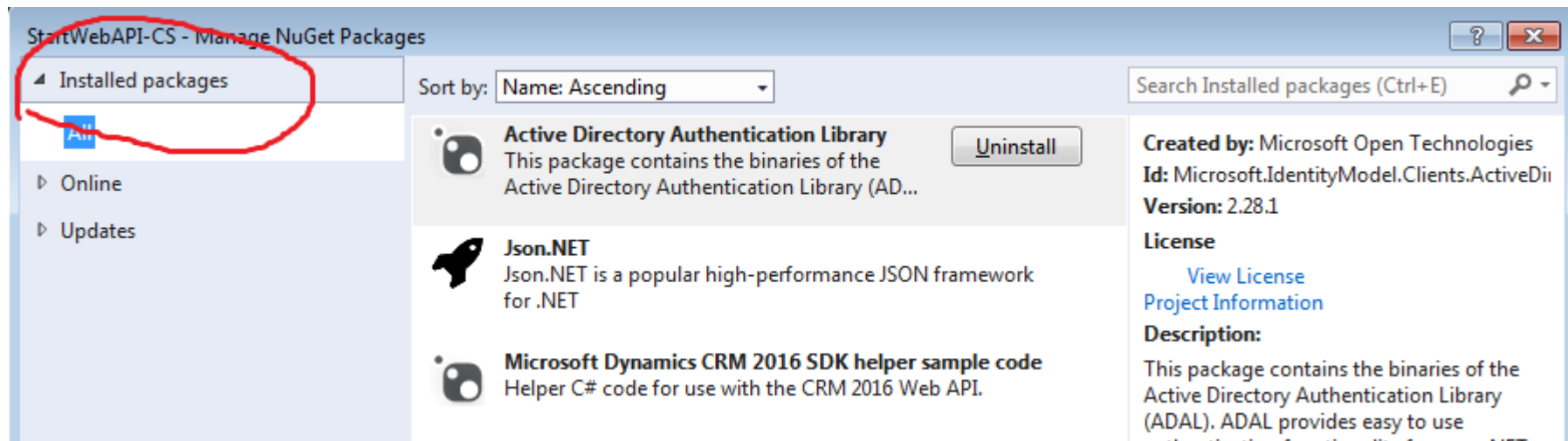
How to list all installed NuGet Packages?

5

Assuming that Nuget is properly installed

Right click **Project** node and click **Manage NuGet Packages**

Build	
Rebuild	
Clean	
View	▶
Analyze	▶
Publish...	
Publish as Azure WebJob...	
Scope to This	
New Solution Explorer View	
Add DevExpress Item	▶
Add	▶
Manage NuGet Packages...	
Set as StartUp Project	
Debug	▶
Source Control	▶
Cut	Ctrl+X
Paste	Ctrl+V
Delete	Del
Rename	
Unload Project	
Open Folder in File Explorer	
Properties	Alt+Enter

See **installed packages list**

answered Oct 13 '16 at 7:24

student
13.7k   10   89   148

Answer to "Is there a way to do this using nuget.exe? – bitbonk"

2

nuget list -Source c:/packages

Where c:/packages is a path to your local repository.

edited Aug 7 at 11:25   answered Apr 24 at 6:33

jstar
472   6   9

Answer to "Is there a way to do this using nuget.exe?" – bitbonk

0

Based on the answer from jstar above. I used `\` instead of `/` which fits more to the Windows environment where nuget is at home. My edit of the answer was rejected so I post my own.

```
nuget list -Source c:\code\packages
```

Where `c:\code` is a path to your local code-repository. The `packages` folder is on the same level like your solution-file (*.sln).

answered Aug 29 at 20:13

KargWare
31   1   8