

dotnet new

<code>dotnet new sln</code>	Solution file
<code>dotnet new console</code>	Console application
<code>dotnet new classlib</code>	Class library
<code>dotnet new mvc</code>	ASP.NET Core Web App (Model-View-Controller)
<code>dotnet new xunit</code>	xUnit test project
<code>dotnet new -l</code>	Obtain a list of the available templates

dotnet sln

<code>dotnet sln list</code>	List all projects in a solution file
<code>dotnet sln todo.sln add todo-a pp/ tod o-a pp.c sproj</code>	Add a C# project to a solution
<code>dotnet sln todo.sln remove todo-a pp/ tod o-a pp.c sproj</code>	Remove a C# project from a solution
<code>dotnet sln todo.sln add **/*.c sproj</code>	Add multiple C# projects to a solution using a globbing pattern

dotnet add

dotnet build

<code>dotnet build</code>	Build a project and all of its dependencies
<code>dotnet build --configuration Release</code>	Build a project and its dependencies using Release configuration

The `dotnet publish` command's output is ready for deployment to a hosting system (for example, a server, PC, Mac, laptop) for execution.

Starting with .NET Core 2.0, you don't have to run `dotnet restore`, because it's run implicitly.

dotnet run

<code>dotnet run</code>	Run the project in the current directory
<code>dotnet run --project .\project s\proj 1\proj 1.c sproj</code>	Run the specified project
<code>dotnet myapp.dll</code>	Run a framework-dependent app named <code>myapp.dll</code>

dotnet clean

```
dotnet add package Newtonsoft.Json dotnet clean
```

Add
Newton-
soft.Json
NuGet
package

```
dotnet ef migrations add
```

Clean the
output
of a
project

```
dotnet clean --configuration Release
```

to a
project

```
dotnet ef migrations remove
```

Clean a
project
built

```
dotnet add reference lib1/lib1.csproj lib2/lib2.csproj
```

Add
multiple
project
references
to the
project in

```
dotnet ef migrations script
```

using
the
Release
config-
uration

```
dotnet ef database update
```

Update
the
data
to a
speci
migrat

```
dotnet add app/app.p.csproj reference lib1/lib1.csproj
```

Only the outputs created during the build
are cleaned. Both intermediate (*obj*) and
final output (*bin*) folders are cleaned.

multiple
project
references
using a
globbing
pattern on
Linux/Unix

```
dotnet ef database drop
```

Drop
data

```
dotnet ef dbcontext list
```

List
availa
DbCo
types

```
dotnet ef dbcontext info
```

Get
infor
ation
about
DbCo
type

```
dotnet ef dbcontext scaffold
```

Scaff
a
DbCo
and e
types
data



dotnet pack

```
dotnet pack
```

```
dotnet pack --no-build --output nupkgs
```

```
dotnet pack /p:PackageVersion=2.1.0
```

dotnet nuget

dotnet remove (cont)

```
dotnet remove app/app.csproj references  
roj project  
and  
create  
NuGet  
packages  
Pack the  
project in  
the  
current  
directory  
into the  
nupkgs  
folder  
and skip  
the build  
step
```

```
Set the  
package  
version to  
2.1.0 with  
the  
PackageVersion  
MSBuild  
property
```

etc.

```
dotnet /help sp  
Remove Show more  
multiple detailed  
project docume-  
references tation  
using a online for  
glob the  
pattern oncommand
```

```
dotnet migrate  
Unix/Linux Migrate a  
Preview 2  
.NET Core  
project to a  
.NET Core  
SDK 1.0  
project
```

```
dotnet msbuild  
Provides  
access to a  
fully  
functional  
MSBuild
```

```
dotnet test  
Run the  
tests in the  
project in  
the current  
directory
```

```
dotnet list reference  
List the  
project  
references  
for the  
project in  
the current  
directory
```

Environment variables

```
dotnet nuget locals -l all
```

Display the paths of all the local cache directories
DOTNET_PACKAGES
The primary package cache.
DOTNET_SERVICING
Specifies the location of the servicing index to use by the shared host when loading the runtime.

```
dotnet nuget push foo.nupkg -k 4003d7 86- cc3 7-4 004 -bf df- c4f 3e8 ef9b3a
```

Push *foo.nupkg* to the default source, specifying an API key
DOTNET_CLI_TELEMETRY_OPTOUT
Specifies whether data about the .NET Core tools usage is collected and sent to default Microsoft.
DOTNET_MULTILEVEL_LOOKUP
Specifies whether .NET Core runtime, shared framework, or SDK are resolved from the global location.

```
dotnet nuget delete Micros oft.As pNe tCo re.Mvc 1.0 --non- interactive
```

Delete version 1.0 of package *Microsoft.AspNet.Mvc*, not prompting user for credentials or other input
DOTNET_ROLL_FORWARD_ON_NO_CANDIDATE_FX
Disables minor version roll forward. For more information, see Roll forward.

dotnet remove

```
dotnet remove package Newton soft t.Json
```

Remove *Newtonsoft.Json* NuGet package from a project in the current directory

```
dotnet remove reference lib/li b.c spr oj
```

Remove a project reference from the current project

