CHEATSHEET FOR
DOTNET CLI

CREATE A NEW PROJECT

Create a new project in the current folder
```
dotnet new [TEMPLATE]
dotnet new web
```

Create a new project in a subfolder
```
dotnet new [TEMPLATE] -o [NAME]
dotnet new web -o Backend
dotnet new mvc -o MyApp
dotnet new console -o MyApp
dotnet new classlib -o MyLib
dotnet new mstest -o MyLib.Test
dotnet new nunit -o MyLib.Test
```

Get the list of installed templates
```
dotnet new
```

MANAGE SOLUTIONS FILES (.SLN)

Create a new .sln file
```
dotnet new sln
```

Create a new .sln file in a subfolder
```
dotnet new sln -o MySolution
```

List all projects in the solution
```
dotnet sln list
```

Add a project to the solution
```
dotnet sln add [PROJECT_PATH]
dotnet sln add MyLib\MyLib.csproj
```

Remove a project from the solution
```
dotnet sln remove [PROJECT_PATH]
dotnet sln remove MyLib/MyLib.csproj
```

RUN

Run an application
```
dotnet run
```

MANAGE PROJECT DEPENDENCIES

Add a reference to another project
```
dotnet add reference [PROJECT_PATH]
dotnet add reference ../MyLib/MyLib.csproj
```

Remove a project reference
```
dotnet remove reference [PROJECT_PATH]
dotnet remove reference ../MyLib/MyLib.csproj
```

List all project references
```
dotnet list reference
```

Add a Nuget package
```
dotnet add package [PACKAGE]
dotnet add NewtonSoft.Json
```

Remove a Nuget package
```
dotnet remove package [PACKAGE]
dotnet remove NewtonSoft.Json
```

List all project references
```
dotnet list package
```

BUILD

Build the project or solution in the current folder
```
dotnet build
```

Build the Release configuration
```
dotnet build -c Release
```

Build the project or solution
```
dotnet build [PROJECT/SOLUTION]
dotnet build MySolution.sln
```

NUGET

Create a nuget package
```
dotnet pack
```

Publish a nuget package
```
dotnet nuget push
```

TEST

Run the tests
```
dotnet test
```

Run the tests and create test report
```
dotnet test --logger "trx;LogFileName=results.trx"
```

List all tests without running them
```
dotnet test -t
```

Run specific tests
```
dotnet test --filter "[FILTER]"
dotnet test --filter Unit
dotnet test --filter "TestCategory=Database"
dotnet test --filter "TestCategory!=Slow"
dotnet test --filter "Unit&(TestCategory=Cat1)"
```

PUBLISH

Publish the project or solution from the current folder
```
dotnet publish
```

Publish specific project or solution
```
dotnet publish [FOLDER]
dotnet publish MyApp
```

Build a self-contained executable for Windows
```
dotnet publish -c Release -r win-x64 -f netcoreapp2.2
```

Build a self-contained executable for Linux
```
dotnet publish -c Release -r linux-x64 -f netcoreapp2.2
```

GLOBAL TOOLS

Install a global tool
```
dotnet tool install -g [TOOLNAME]
```

Uninstall a global tool
```
dotnet tool uninstall -g [TOOLNAME]
```

List all globally installed tools
```
dotnet tool list -g
```