Nuget. Package management

1. **Nuget** – infrastructure for using external libraries in your code
2. How to **add**:   
   - right click on the solution in Solution explorer  
   - Manage Nuget Packages for Solution…  
   - Browse tab  
   - search the package name and select it  
   - on the right side (projects list) check the project(s) that you want the package to install to  
   - select required version in the dropdown  
   - hit install
3. Nuget manager will execute the following:   
   - create in the project packages.config descriptor that lists all package(s) (with their version(s)) added to the project  
   - create in file system **packages** folder under solution folder and **downloads** the actual package (dlls) from <https://nuget.org>  
   - in the project add references to the required package dlls (so you can start using them)
4. When you build the project (with packages.config file) MSBuild checks if there is packages folder and inside there are appropriate dlls. In case it is missing – MSBuild automatically runs Nuget manager that redownloads all the missing dlls. So you always can freely delete the packages folder because it will be recreated at the beginning of the build.
5. **Important!** **There is no need to checkin the packages folder itself or any of the dlls inside of it!** The whole **idea** of package managers is the ability to download from online required dlls **ONLY** when you need them (when building).   
   You only checkin packages.config
6. Note: in Post build event add **PackageDLLs**.exe event that will copy the appropriate dlls to NGCSOutput\bin
7. Global Nuget configuration location - %USERPROFILE%\.nuget\Nuget.config (proxy settings etc)
8. Note: in older versions of VS MSBuild sometimes not automatically runs Nuget to download missing dlls (require additional configuration in the project)
9. **Advanced**:  
   - main Microsoft Nuget repository is at <https://nuget.org>  
   - but you can manage your own local Nuget server (and it will be transparent to your VS)  
   - you can distribute your own code as Nuget package
10. **Don’t mix** compilation and deployment phases (Nuget only relevant and handles the compilation phase). You need to handle separately every dll (or other code) imported as Nuget package how you install it on the destination server(s).