* Restore back TaskMgr.bak or run script TaskMgrScript.sql on SQL Server/Express to create TaskMgr database. If you don’t have SQL Server license you may use SQL Express which is free license. Development was done on SQL Server 2014. So if you have lower version you may create database manually then run script to generate remaining objects like tables, relations etc.
* There are 3 solutions provided: TaskMgr.sln, TaskMgrConsole.sln, and DynamicStepClasses.sln. Depending your need and available resources you can deploy applications or directly run from Visual Studio 2017.
  + TaskMgr.sln
    - This is a web interface solution, you can host on web server like IIS or you can just run from VS2017.
    - Before deploy/run need to set these settings in appsettings.json. TaskMgrConnectionString: set correct connection string to access TaskMgr database in read write mode. DynamicStepClassesDllWithFullPath: set location of dynamic steps DLL (output of solution DynamicStepClasses.sln).
  + TaskMgrConsole.sln
    - This is console application targeting windows environment. It may change to run on different environment.
    - This can be run from VS2017 or built DLL can run using “dotnet” command.
    - Before build/run open appsettings.json and make sure to change all settings accordingly.
      * TaskMgrConnectionString: this is connection string to TaskMgr database in read write mode.
      * DynamicStepClassesDllWithFullPath: set location of dynamic steps DLL (output of solution DynamicStepClasses.sln).
      * TaskMgrInfoFile: File path of SMTP email password file. Sending email is optional you can comment application code accordingly. If sending email then please make sure to create this text file and set password in this file. You may change this implementation as per your security requirements.
      * ScheduleIntervalInSeconds: Delay between triggers that run the main process to execute steps.
      * EmailIntervalInSeconds: Email queue execution triggers delay in seconds.
      * SmtpServer: SMTP server address.
      * SmtpPortNumber: SMTP port number.
      * FromEmailId: Sender’s email id.
      * FromEmailTitle: Sender’s title.
  + DynamicStepClasses.sln
    - This is DLL output solution. Other applications use this DLL and execute steps.
    - Make sure the output folder of this solution matches with the other 2 applications “DynamicStepClassesDllWithFullPath” setting.
    - At this moment there was an issue with unloading dynamic loaded DLLs using “System.Runtime.Loader.AssemblyLoadContext”. So compilation may give sharing violation if running on windows desktop. It may run on windows server, not tested. You may have to stop other applications to compile this solution.