

# Premiers pas avec Visual Studio Community

Préparé par  
M.G. BELKASMI

# Installation VSC

- Lancer l'exé

 Documents

 Images

 Musique

 Sans titre 1

 vs\_community\_ENU

- attendre

 Visual Studio



Community 2015  
with Updates

Choose your installation location

C:\Program Files (x86)\Microsoft Visual Studio 14.0

Setup requires up to 8 GB across all drives.

Choose the type of installation



Default

Includes C#/VB, Web and Desktop features



Custom

Allows you to customize features for your installation

You can add or remove additional features at any time after setup via Programs and Features in the Control Panel.

By clicking the "Install" button, I acknowledge that I accept the [License Terms](#) and [Privacy Statement](#).

Cancel



Install

# Installation VSC

- Par défaut on disposera du langage C#
- Cliquer sur install
  - Veiller à enlever la vérification des comptes ou à lancer en tant qu'admin



Community 2015  
with Updates

Acquiring:



Microsoft Visual Studio Enterprise 2015

Applying:



Microsoft Visual Studio Preparation



Cancel

# Installation VSC

- Le téléchargement et l'installation des composants est en cours
- Patience



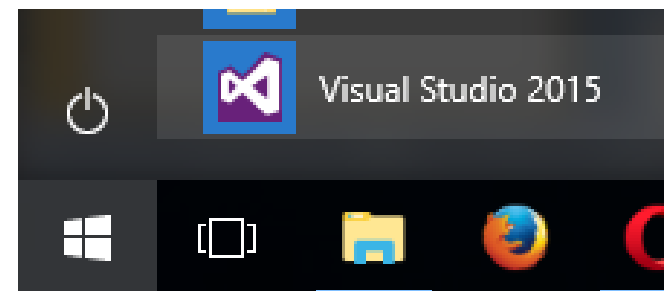
Community 2015  
with Updates

Setup Completed!  
All specified components have been installed successfully.

LAUNCH

# Installation VSC

- Voilà c'est fini
- Prêt à lancer
- On peut lancer aussi via menu





Community 2015

This program is protected by U.S. and international copyright laws as described in Help/About.

© 2016 Microsoft Corporation.  
All rights reserved.

# Installation VSC

- C'est lancé



Welcome!

Connect to all your developer services.

Sign in to start using your Azure credits, publish code to a private Git repository, sync your settings, and unlock the IDE.

[Learn more](#)

Sign in

Don't have an account? [Sign up](#)

[Not now, maybe later.](#)

x

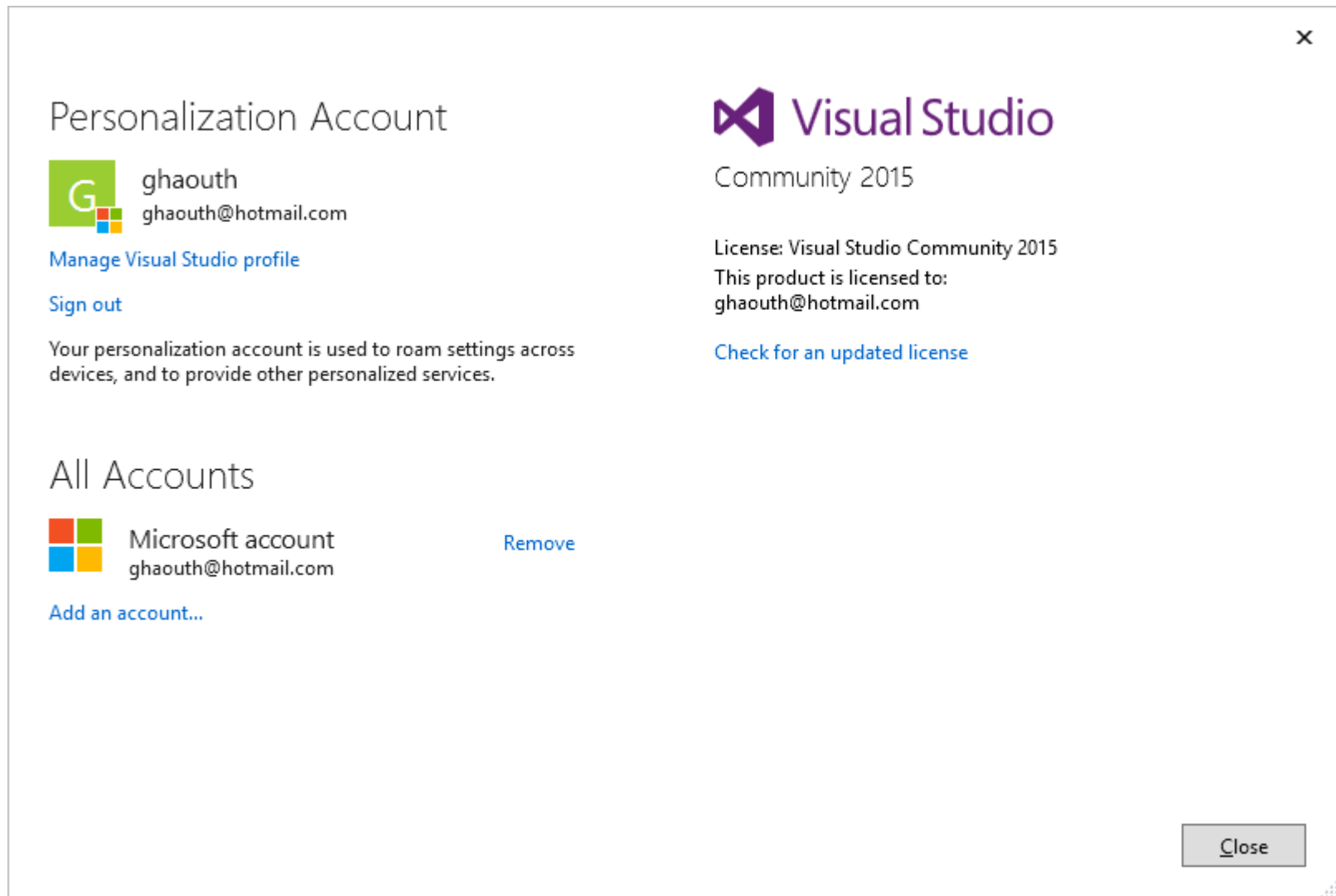
# Installation VSC

- On peut utiliser un compte Microsoft pour travailler on équipe via Git ou sur la plateforme cloud Azure

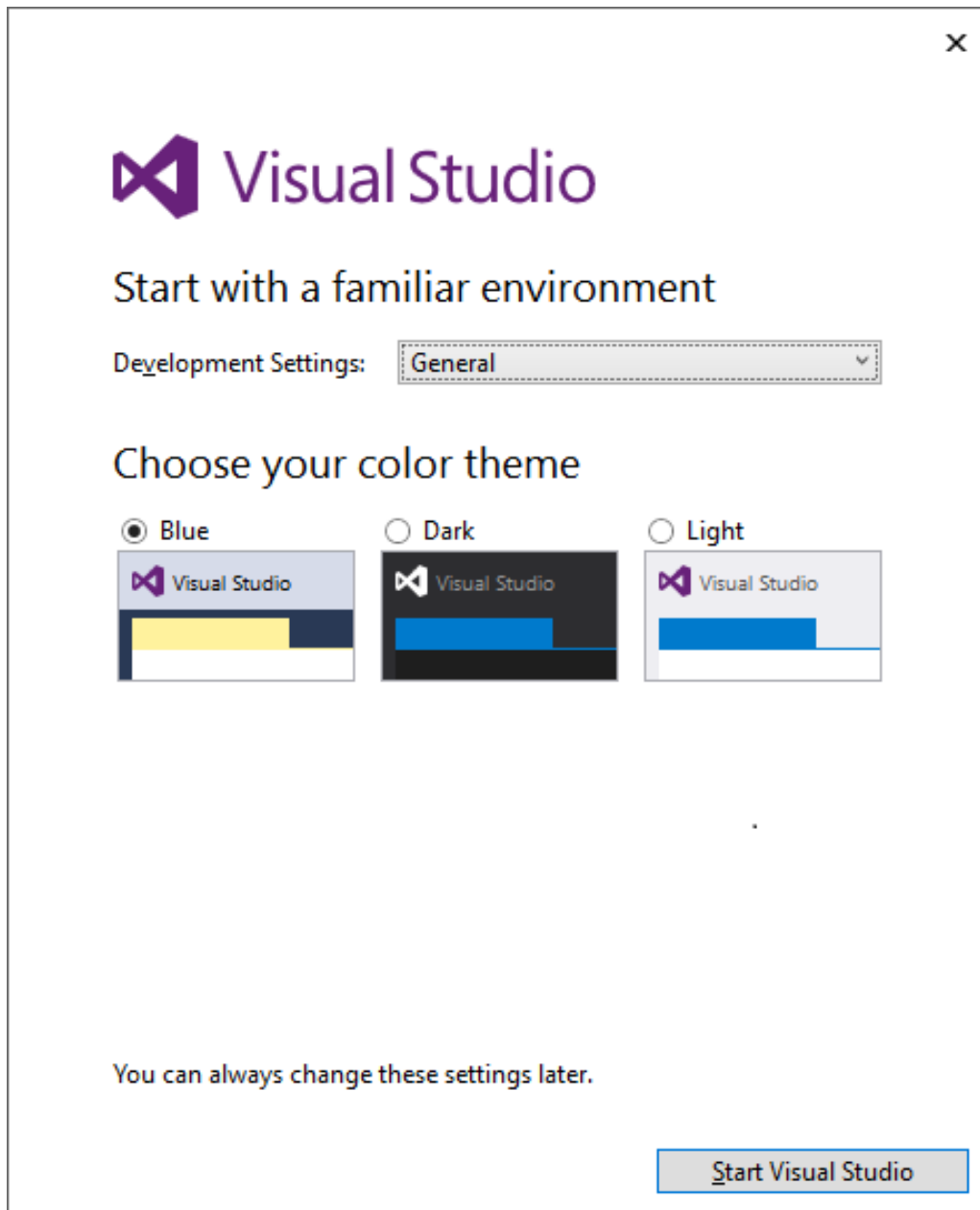
ou

- Not now
  - 30 jours d'essai

# Installation VSC







# Installation VSC

- Choisir le thème et une config pour l'IDE
- Cliquer sur start



We're preparing for first use

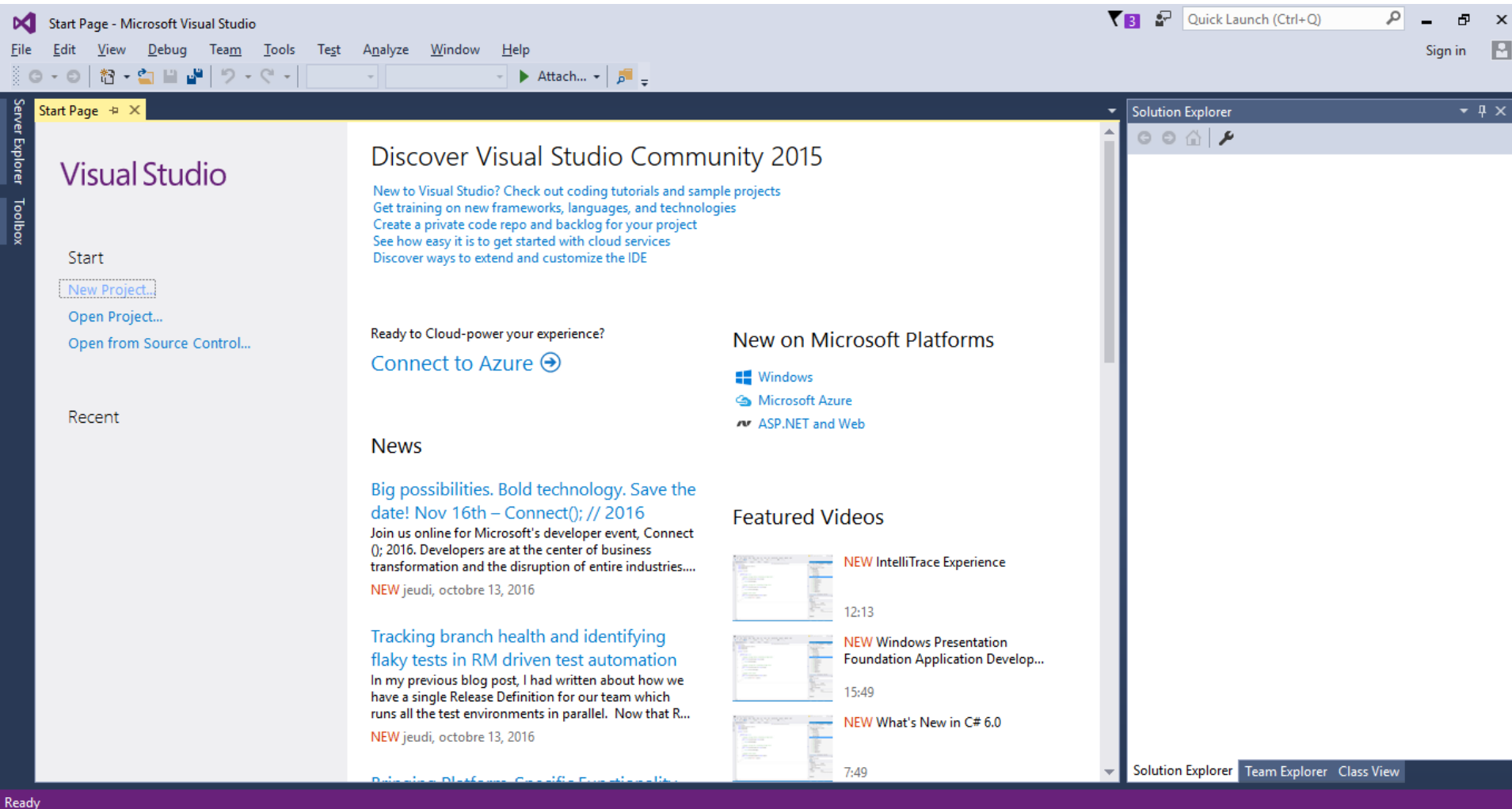
This may take a few minutes.

.....

# Installation VSC

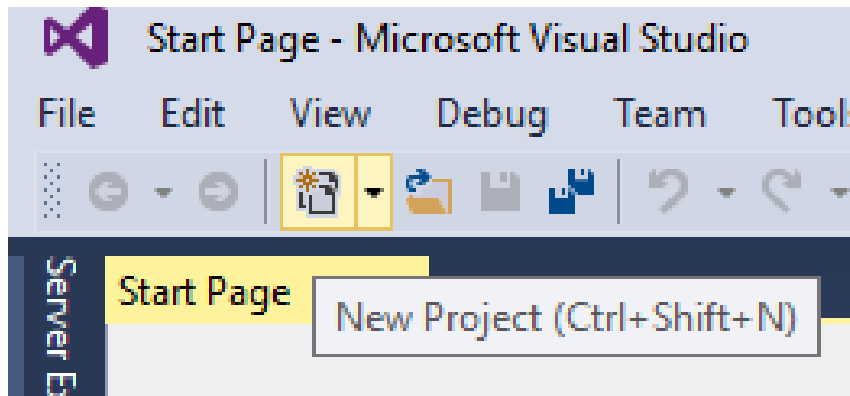
- On y est preque

# Installation VSC

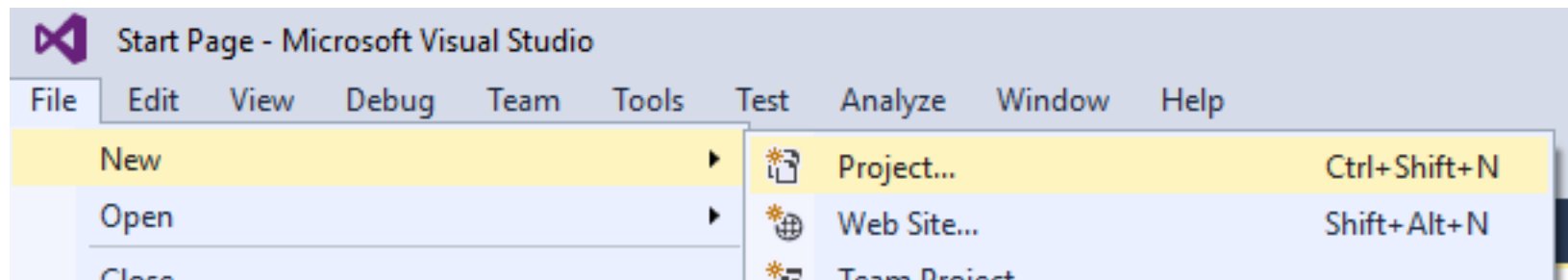
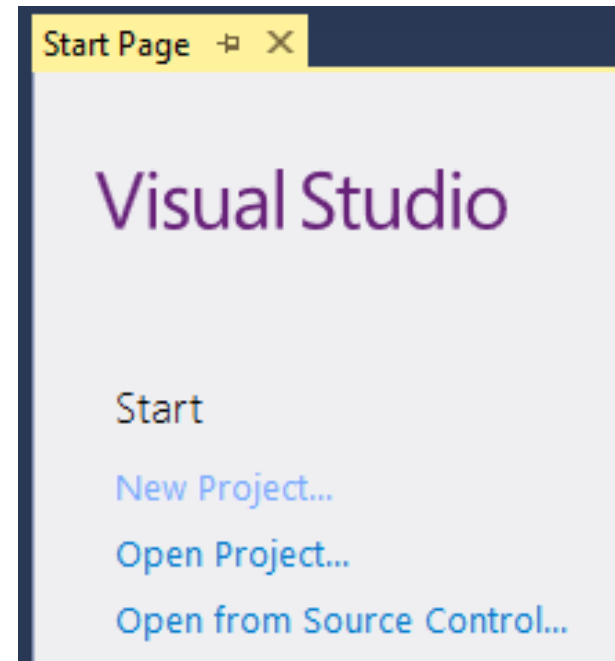


# Première application C#

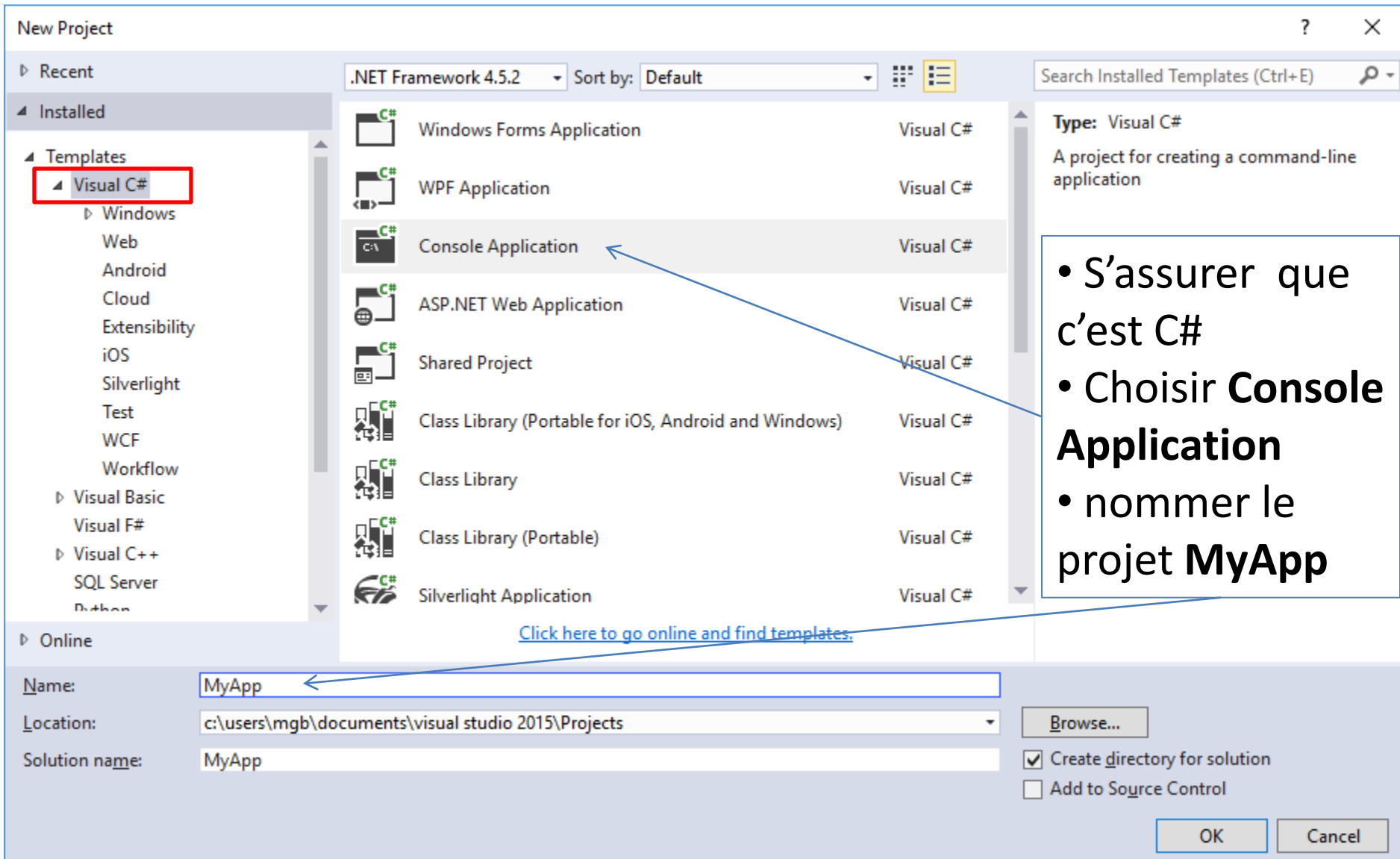
- Créer le projet



Ou

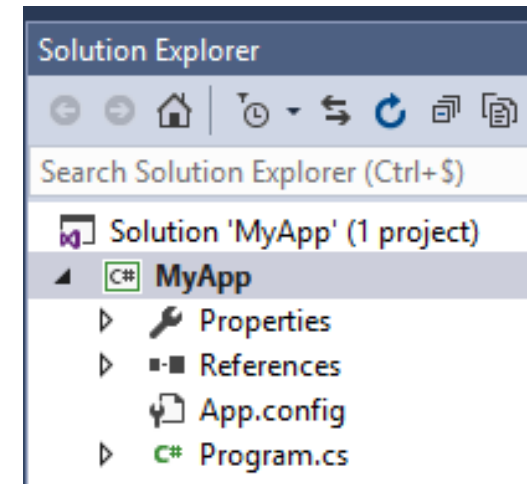


# Première application C#



# Première application C#

```
Program.cs*  [icon] [X]
C# MyApp
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace MyApp
8  {
9      class Program
10     {
11         static void Main(string[] args)
12         {
13             [ ]
14         }
15     }
16 }
17
```



# Première application C#

- Ajouter le corps du main

```
static void Main(string[] args)  
{
```

Console.W

- SetWindowSize
- Title
- TreatControlCAsInput
- WindowHeight
- WindowLeft
- WindowTop
- WindowWidth
- Write
- WriteLine**

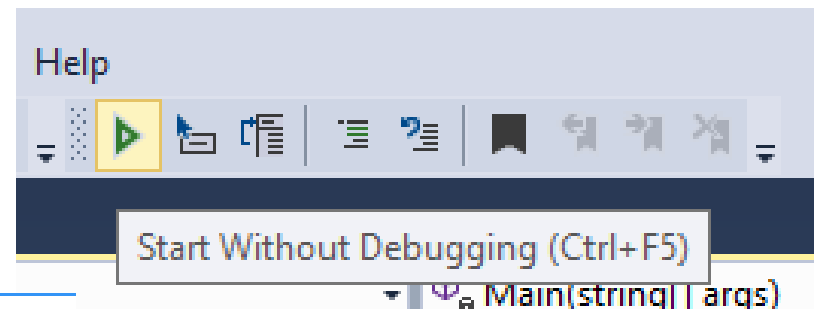
void Console.WriteLine() (+ 18 overloads)  
Writes the current line terminator to the standard output stream.

# Première application C#

- Ajouter le corps du main

```
class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("Hello!");
    }
}
```

- Sauvegarder et exécuter



C:\WINDOWS\system32\cmd.exe

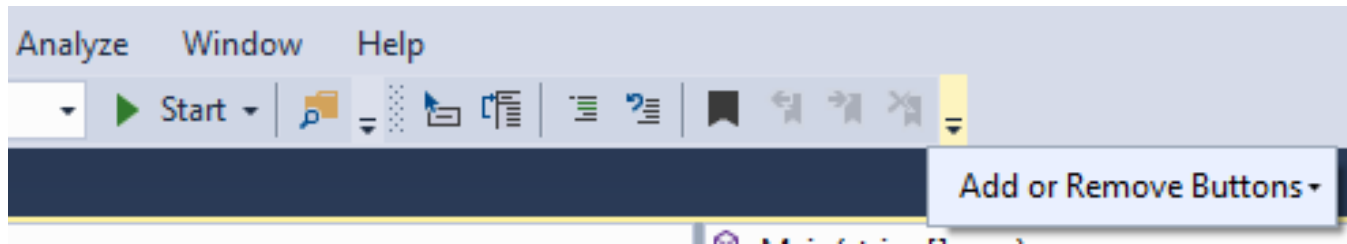
Hello!

Appuyez sur une touche pour continuer...



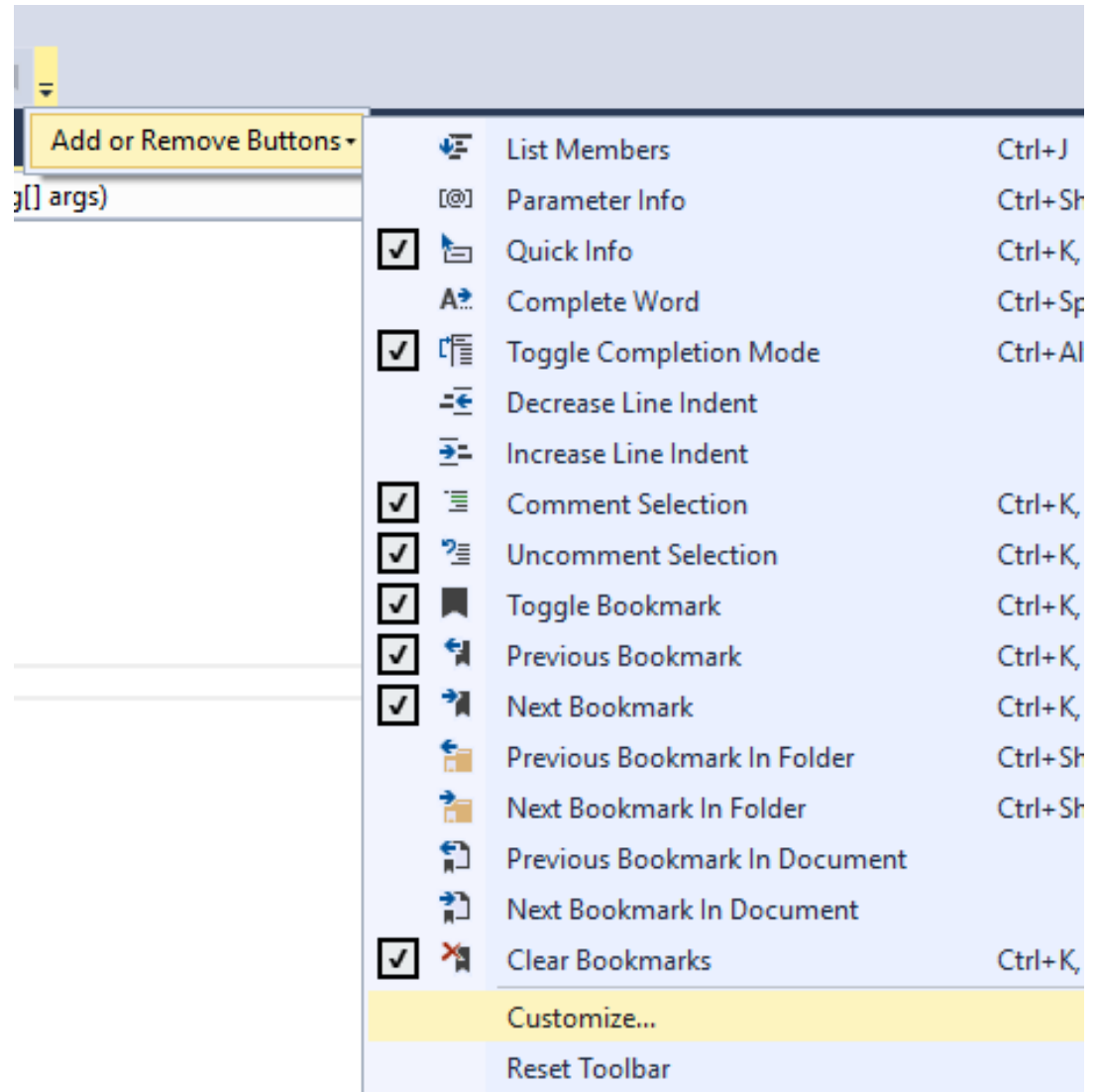
# Première application C#

- Pour ajouter le bouton Exécuter sans débogage



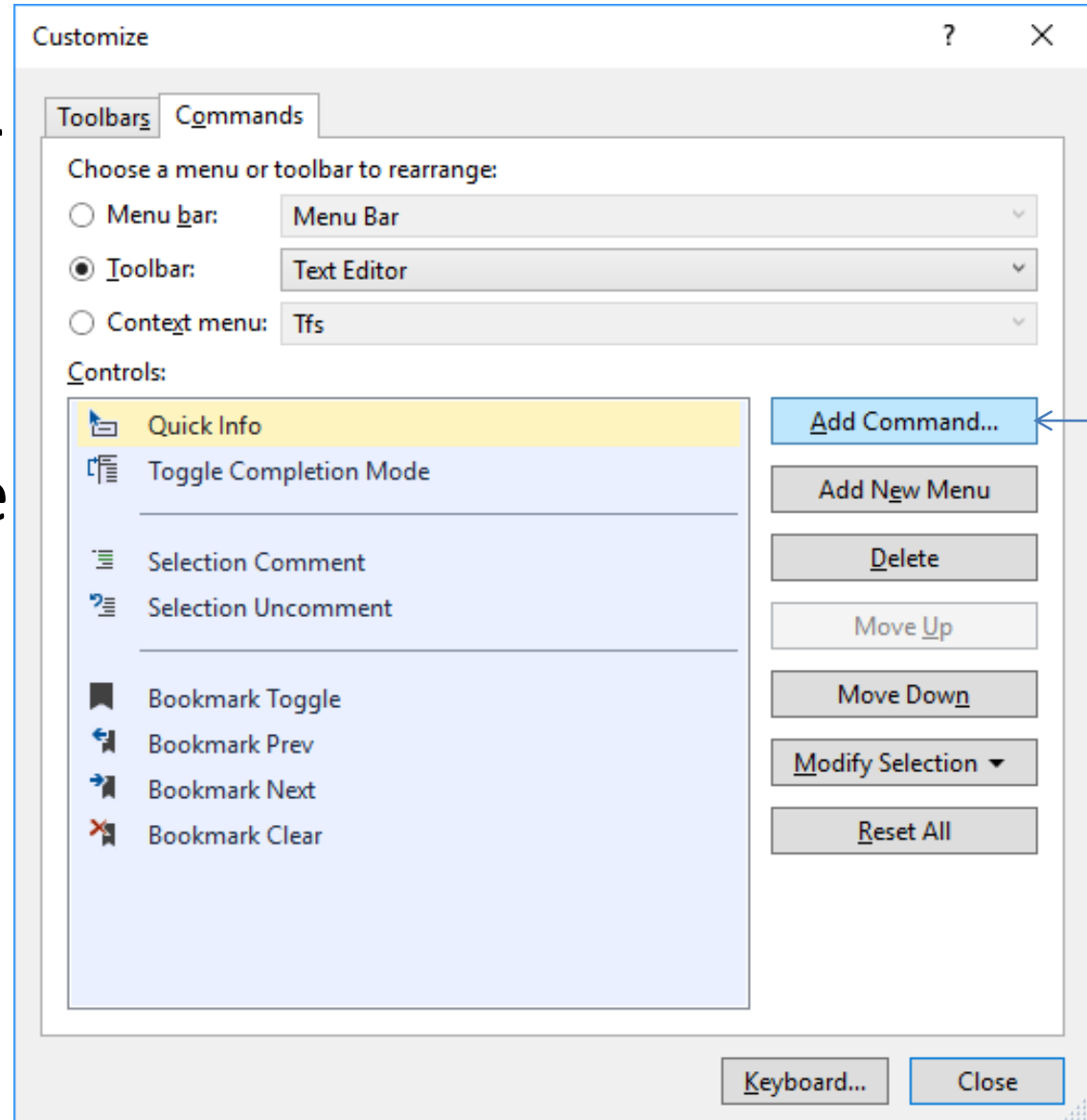
# Première application C#

- Pour ajouter le bouton Exécuter sans débogage



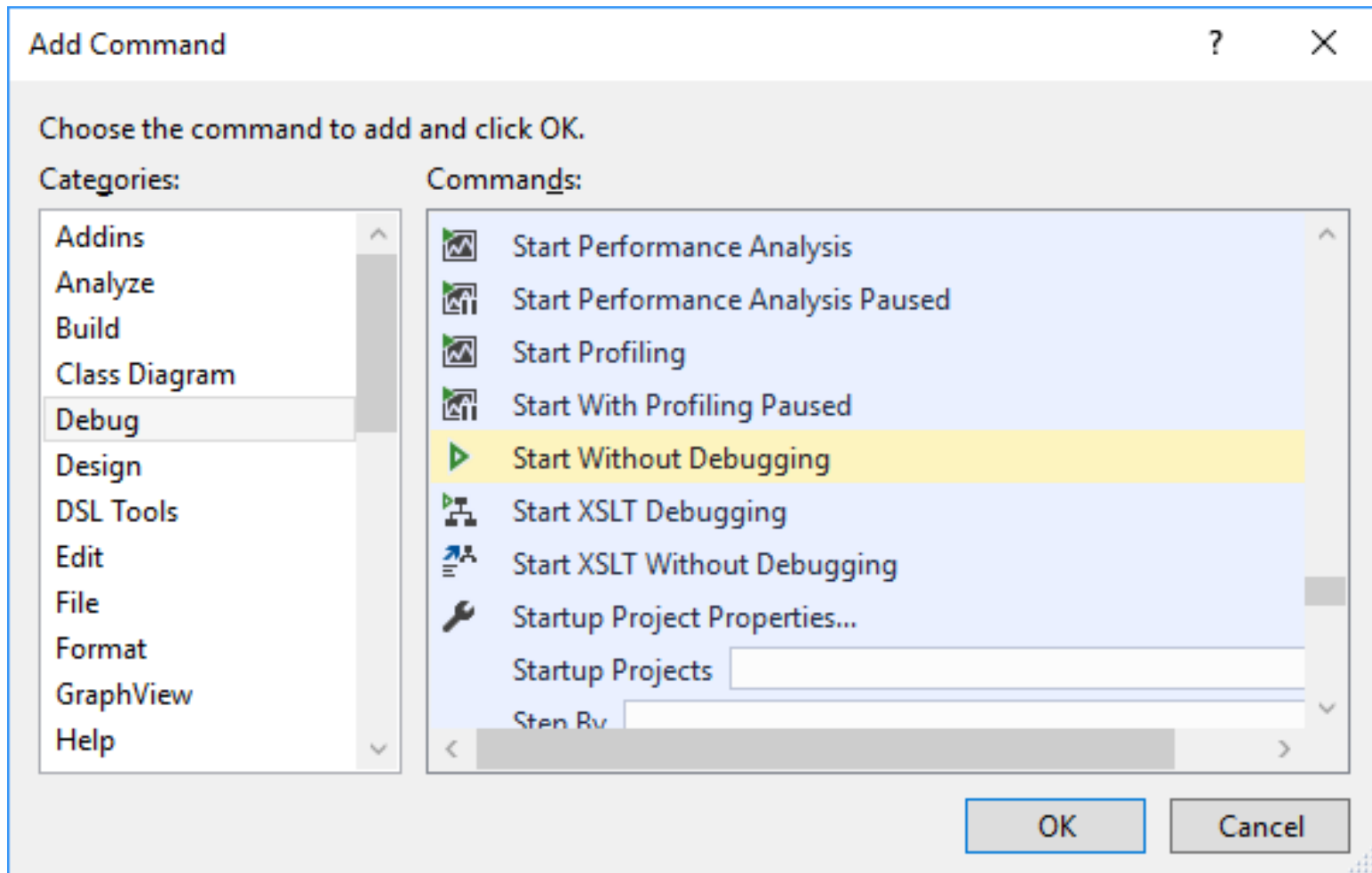
# Première application C#

- Pour ajouter le bouton Exécuter sans débogage



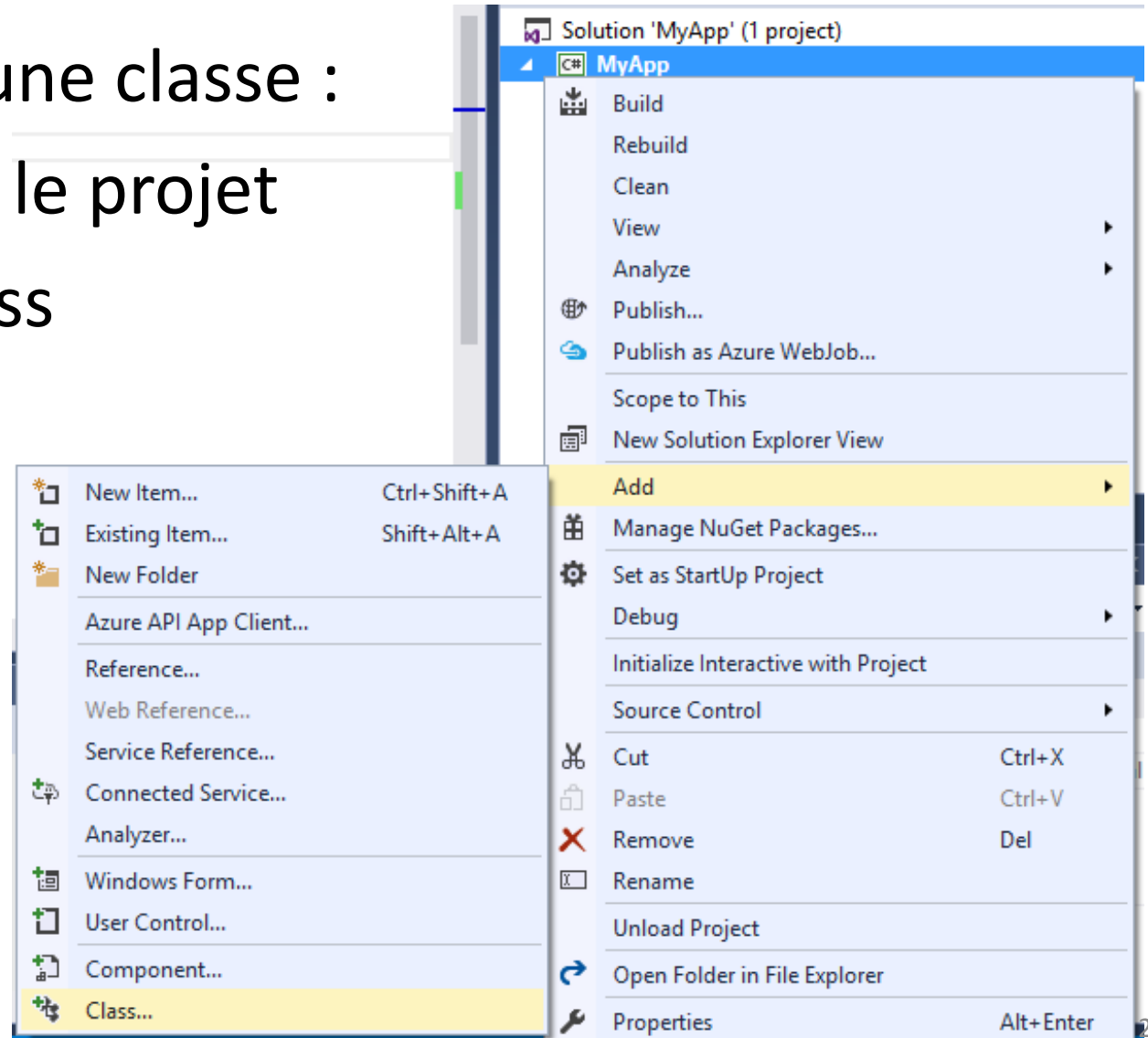
# Première application C#

- Pour ajouter le bouton Exécuter sans débogage



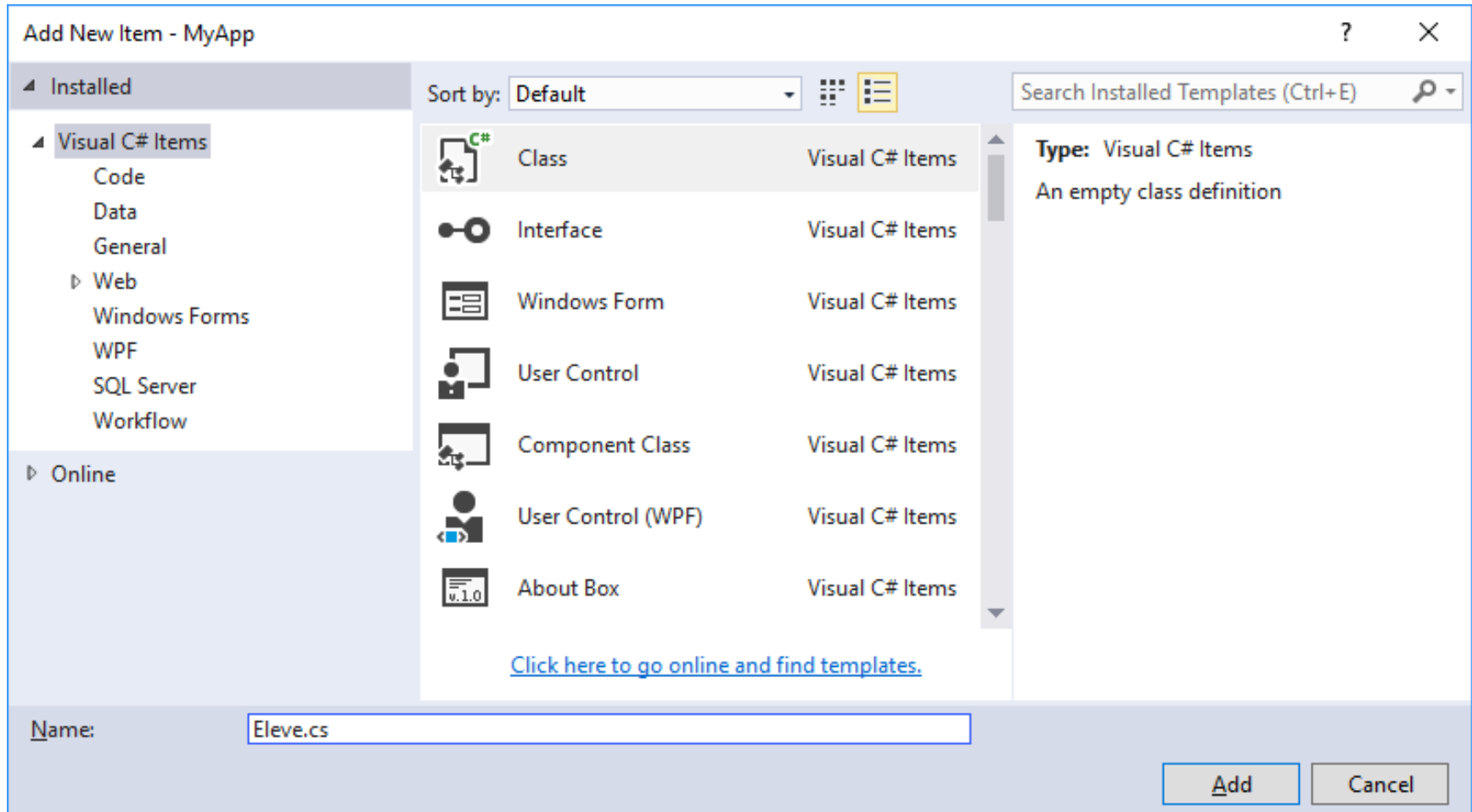
# Première application C#

- Pour créer une classe :  
Click droit sur le projet  
→ Add → Class



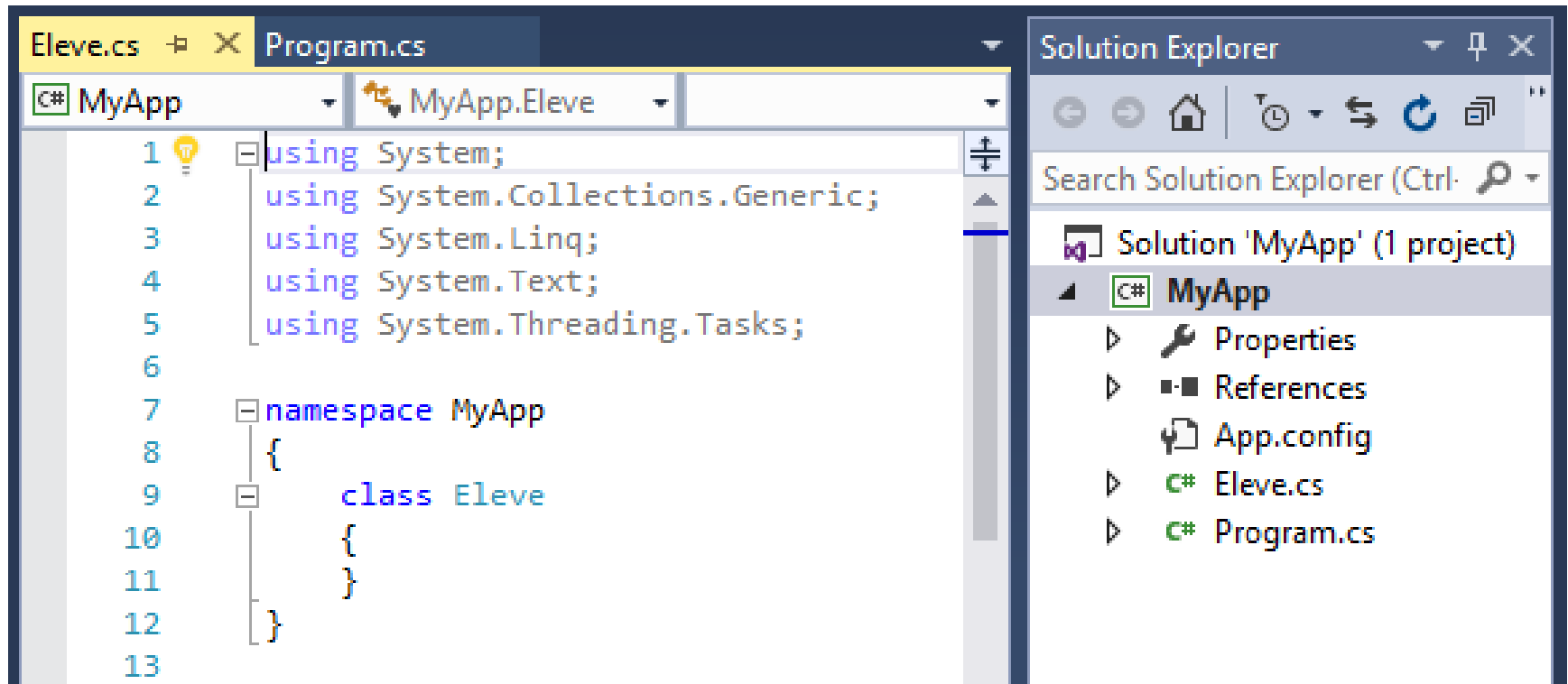
# Première application C#

- Pour créer une classe : nommer la classe Eleve



# Première application C#

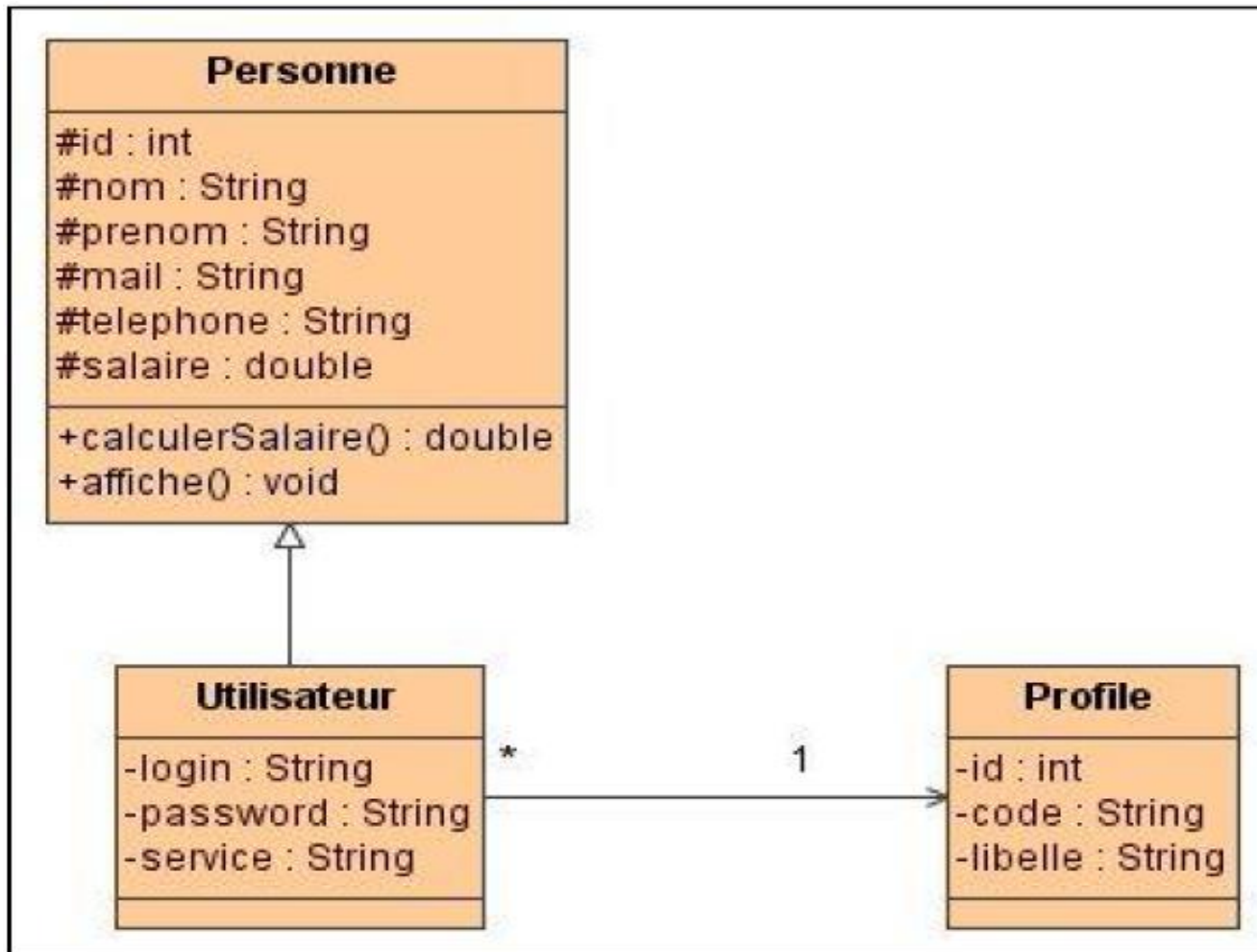
- La classe Eleve est créée



- Compléter Eleve.cs et tester dans Program.cs

# Exercice

On souhaite développer un module pour la gestion des utilisateurs, pour cela le diagramme de classe suivant a été établi :





# Exercice

1. Développer les classes `Personne`, `Utilisateur` et `Profile` en C# dans un namespace de votre choix.
2. Redéfinir la méthode **`calculerSalaire()`** et la méthode **`affiche()`** dans la classe `Utilisateur`, sachant que :
  - Le manager aura une augmentation de **10%** par rapport à son salaire normal,
  - Le directeur général aura une augmentation de **40%** par rapport à son salaire normal.
3. Créer les profiles : Chef de projet (CP), Manager (MN), Directeur de projet (DP), Directeur des ressources humaines (DRH), Directeur général (DG),
4. Créer des utilisateurs avec les différents profiles métiers.
5. Afficher la liste des managers.