

Tutoriel phase 2

But

- Faire une application console de A-Z
- Lire un fichier et exploiter ces données

Initialisation du projet

Avant d'utiliser votre IDE, ouvrez votre terminal et tapez cette commande :

Dotnet new console --name PenduProject

Elle va créer :

- Un PenduProject.csproj. Ce type de fichier stocke la liste des fichiers inclus dans le projet, les assemblies du projet, le GUID du projet et d'autres informations comme la version du projet.
- Un Program.cs le point d'entrée de votre programme.
- Un dossier obj utilisé pour stocker des fichiers de manière temporaires durant le processus de compilation.

Le but de ce projet va d'être de créer un pendu. Pour plus de facilité, on travaillera avec la langue anglaise car l'encodage des accents n'aura pas à être géré.

Pour vous aider à lire ce fichier, je vous conseille d'utiliser la classe **StreamReader** (<https://docs.microsoft.com/fr-fr/dotnet/api/system.io.streamreader?view=netframework-4.7.2>) elle permet d'utiliser des stream de lectures pour lire les flux ici un fichier texte.

Pour vous aider je vous conseille de faire en deux étapes :

- Choisir un mot du fichier et le mettre en mémoire
- Faire la partie où vous devinez le mot

Pour pouvoir gérer l'interaction entre le joueur et le programme il faut utiliser les méthodes de la classe static Console :

- WriteLine pour écrire un texte
- ReadLine pour récupérer la réponse du joueur

Voici les règles du jeu :

- Dans la console le jeu doit choisir un mot
- Seuls les mots ayant 6 lettres minimums peuvent être choisis
- Au début de chaque tour la console affiche le mot composé de * pour les lettres non devinées et de lettres pour le reste
- Vous avez 11 chances pour deviner le mot avant de perdre

Pour initialiser le projet :

- Faire un git clone de ce projet github :
 - git clone <https://github.com/plouiserre/DevouxPenduDnetCore.git>

- L'initialisation du projet se trouve sur la branche master
- Il y a un fichier dico.txt listant des mots de la langue anglaise.

Pour votre exercice vous allez devoir utiliser la classe Random du framework .NET. Pour cela voici la documentation correspondante :

- <https://docs.microsoft.com/fr-fr/dotnet/api/system.random?view=netframework-4.7.2>
- <https://stackoverflow.com/questions/2706500/how-do-i-generate-a-random-int-number>

Vous avez 30 minutes pour réaliser ce projet en solo, vous pouvez le tenter seul ou suivre la solution détaillée juste après.