

# TP00 – Ma première fenêtre

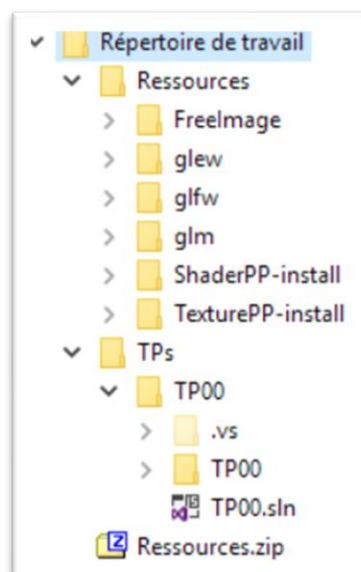


Ce TP est primordial ! Il est votre point de départ pour tous vos développements futurs dans ce cours. Le fruit de votre travail d'aujourd'hui doit absolument être conservé afin de servir de base aux travaux suivants.

## Exercice 0 Préparation de votre environnement de travail

### 0.1 Arborescence de répertoires

- 📄 Téléchargez le fichier **Ressources.zip** sur l'ENT et décompressez-le dans votre répertoire de travail.
  - ➡ Cela crée un répertoire **Ressources** contenant notamment les répertoires de bibliothèques non encore prises en charge par [conan](#).
- 📄 Dans votre répertoire de travail, à côté du répertoire **Ressources**, créez un répertoire **TPs** où vous stockerez tous vos futurs TPs.



### 0.2 Solution Visual Studio


- 📄 Ouvrez Visual Studio puis créez une solution de type console que vous nommerez **TP00**, stockée dans le répertoire **TPs**
  - ➡ Gardez cochée la case « **Créer un répertoire pour la solution** », de façon à créer un répertoire **TP00** dans le répertoire **TPs**.
  - ➡ Votre arborescence de répertoire doit alors ressembler à celle représentée sur l'image ci-dessus (les sous-répertoires de [Ressources](#) ne sont pas nécessairement ceux représentés sur cette capture).
  - ➡ En utilisant [conan](#), ajoutez à votre projet les bibliothèques [glew](#) en version supérieure ou égale à [2.1.0](#) et [glfw](#) en version supérieure ou égale à [3.3.2](#). Référez-vous au document « [Prise en main de conan](#) »
- 📄 Vous devriez être en mesure de compiler le programme (inutile) suivant :


```
#include <GL\glew.h>
#include <GLFW\glfw3.h>

int main(int argc, char* argv[])
{
    glfwInit();           //Fonction de la bibliothèque GLFW
    glewInit();           //Fonction de la bibliothèque GLEW
    glGetString(GL_VERSION); //Fonction de la bibliothèque OpenGL du système
    return 0;
}
```






## Exercice 1 Création d'une fenêtre

 Cette partie permet de recréer le code d'exemple **FirstWindow** du cours

 En suivant les instructions du cours, mettez en place le code minimal pour créer une fenêtre à l'aide de la bibliothèque **GLFW**.


 Si vous êtes en train de travailler sur un ordinateur portable muni de deux contrôleurs vidéo (un pas cher Intel et un de performance nVidia) alors le programme s'exécutera par défaut avec le GPU Intel, empêchant peut-être l'utilisation d'OpenGL 3.3. Pour que votre programme s'exécute automatiquement avec le processeur nVidia, vous devez ajouter la ligne magique suivante en début de votre fichier source :

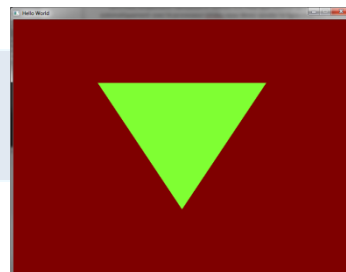
```
extern "C" _declspec(dllexport) unsigned long NvOptimusEnablement = 0x00000001
```


-  Faites en sorte que la couleur de fond soit rouge.
-  Faites en sorte que la position en **x** de la souris influence la saturation de la couleur, et que la position en **y** de la souris influence la valeur de la couleur.
  -  Pour la gestion de la couleur dans l'espace **HSV**, vous pouvez utiliser la classe **CColor** fournie avec l'exemple **FirstWindow** du cours.
  -  Vous devez mettre en place une fonction de rappel pour les événements souris et passer à cette fonction un pointeur vers l'objet **CColor** utilisé pour la couleur de fond.
-  Faites en sorte que l'appuie sur les touches **↑** et **↓** fasse incrémenter ou décrémenter la valeur de la teinte.

## Exercice 2 Mon premier triangle

 Cette partie permet de recréer le code d'exemple **FirstTriangle** du cours


 En suivant les instructions du cours, modifiez le programme de façon à afficher un triangle vert, une pointe vers le bas.

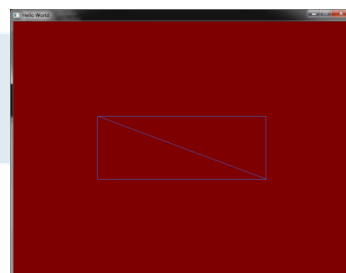


 Lors de l'édition des codes sources des shaders, il est agréable de bénéficier de la coloration syntaxique. Vous trouverez sur l'ENT deux fichiers vous permettant de bénéficier de cette fonctionnalité sous Visual Studio 2017 (NShader) et sous Notepad++ (fichier XML)

## Exercice 3 Mon premier rectangle

 Cette partie permet de recréer le code d'exemple **FirstRectangle** du cours

 En suivant les instructions du cours, modifiez le programme de façon à afficher un rectangle bleu en fil de fer et utilisant un EBO.



## Exercice 4 Mon premier travail

🖱️ Cette partie vous permet de valider si vous avez compris les précédentes

- 💻 Modifiez le programme pour afficher deux rectangles l'un au-dessus de l'autre.
- ➡️ Le premier doit être affiché en orange, le second en bleu.
  - ➡️ Le premier doit être affiché en fil de fer, le second en polygones pleins
    - ➡️ A l'appuie sur la touche **W**, les modes d'affichage des rectangles doivent être inversés.

