

Workshop Menus de jeu dans Unity

Version d'Unity utilisée pour ce workshop: Unity 2020.2.2f1

1.Intro

Dans un jeu, il y a des Menus : Menu principal, Menu de pause, etc...

Dans ce workshop vous allez apprendre à faire un menu principal de jeu simple ainsi qu'un menu de pause.

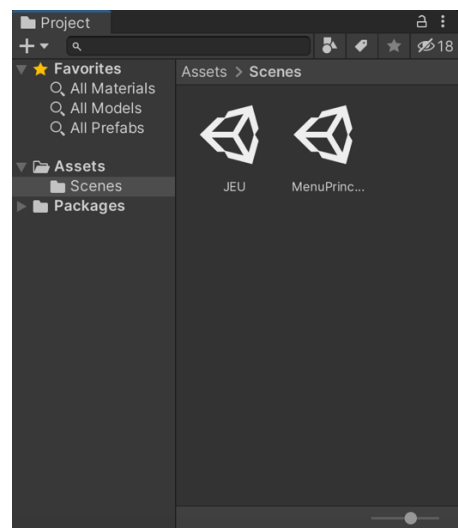
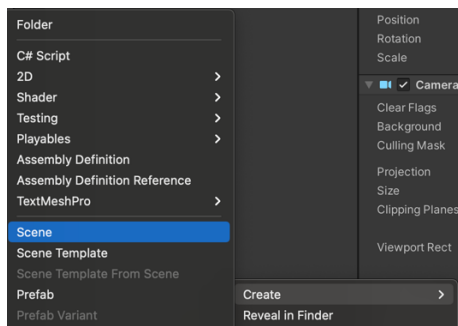
2.Menu Principal

Étape 1 :

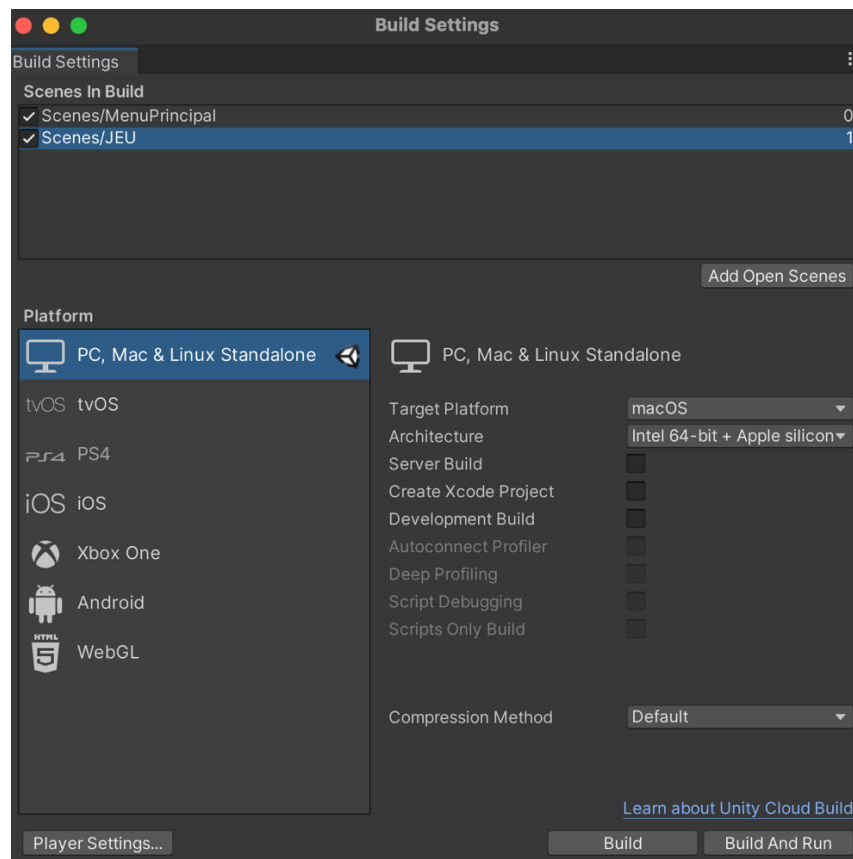
Faites un clic droit dans le panneau « Hierarchy » -> Create Empty

Faites un clic droit dans le panneau « Project » puis créer une Scene

// Cette scene sera notre « jeu »



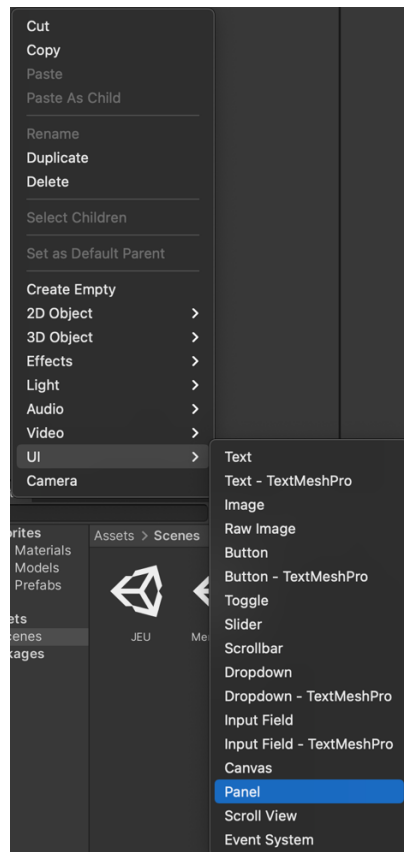
En haut de la fenêtre, dans File, sélectionner « Build Settings » et glisser déposer la scene « JEU » que l'on vient de créer.



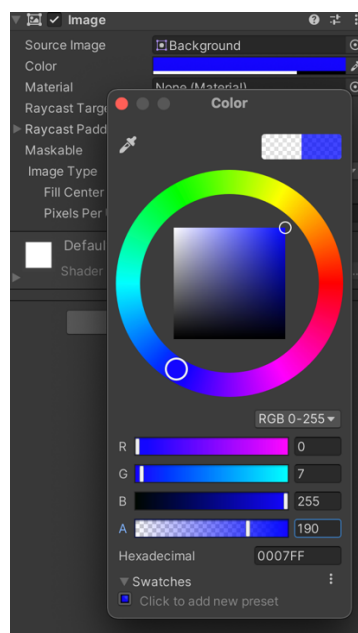
// Cette manipulation est importante car sans cela, le jeu ne reconnaîtra pas l'existence des Scene et provoquera une « error build ».

Étape 2 :

Clic droit dans le panneau Hierarchy -> dans le menu déroulant par UI : sélectionner « Panel » qui sera notre Background



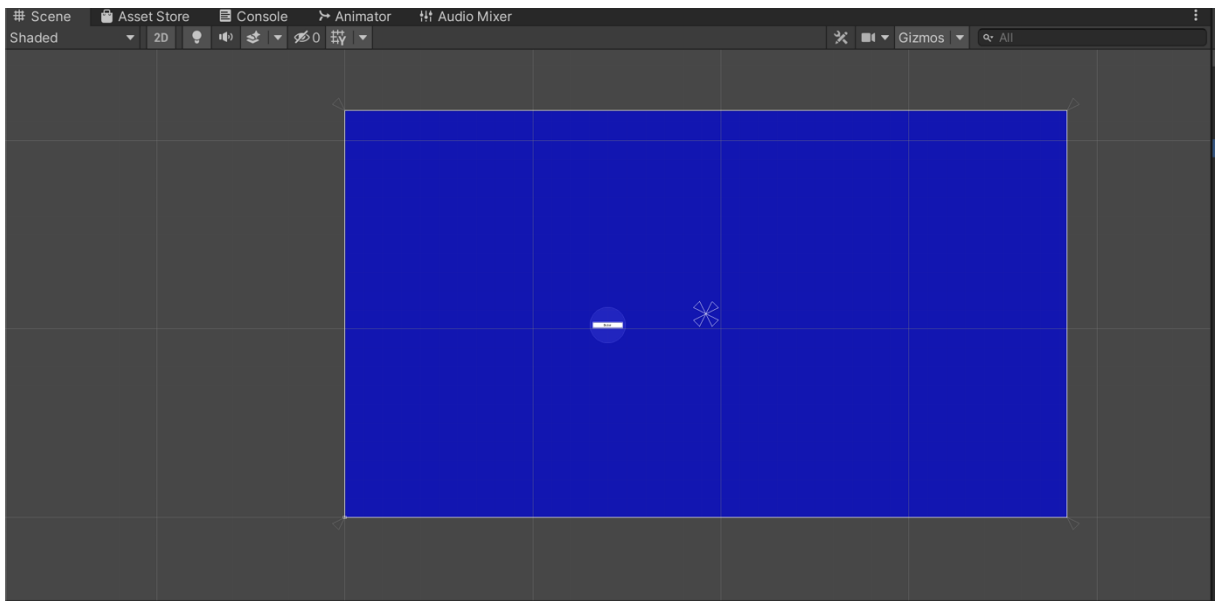
On lui assigne une couleur.



// On peut aussi assigner une image à ce panel. En Sélectionnant « Source image » et en important son image dans le dossier du jeu.

Étape 3 :

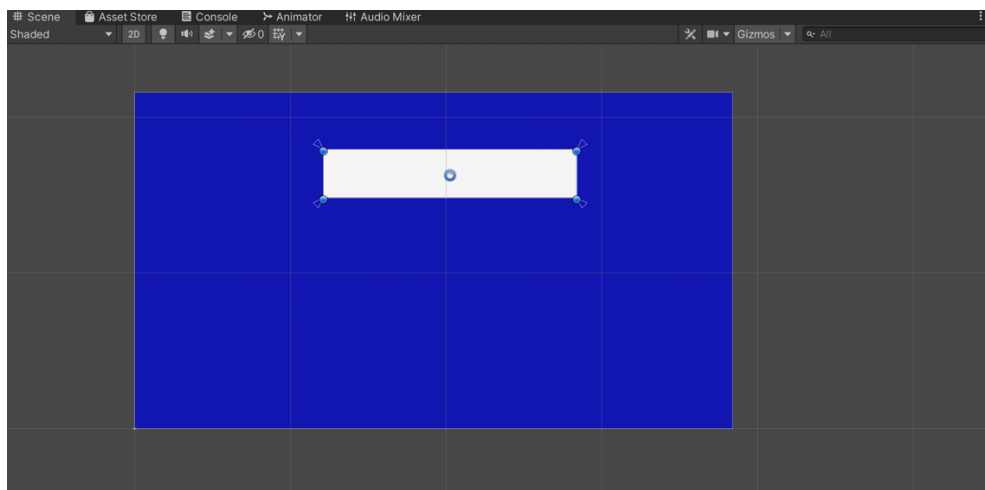
Clic droit dans le panneau Hierarchy -> dans le menu déroulant par UI : sélectionner Button



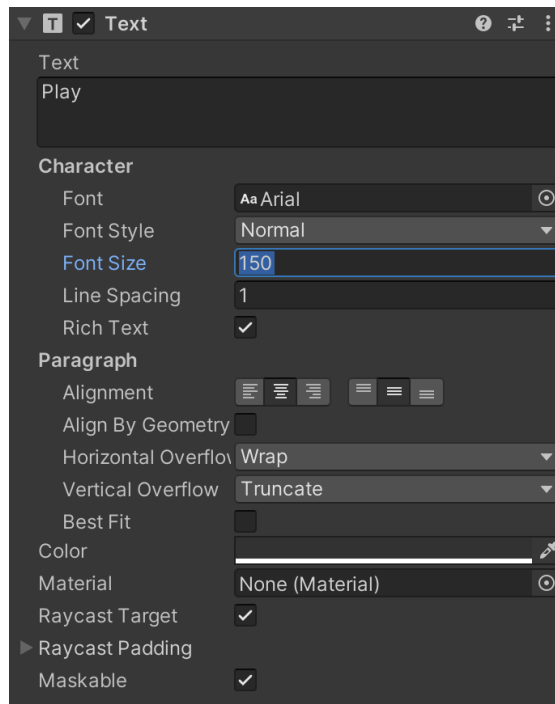
// Un bouton est créé avec un texte : « Button »

Étape 4 :

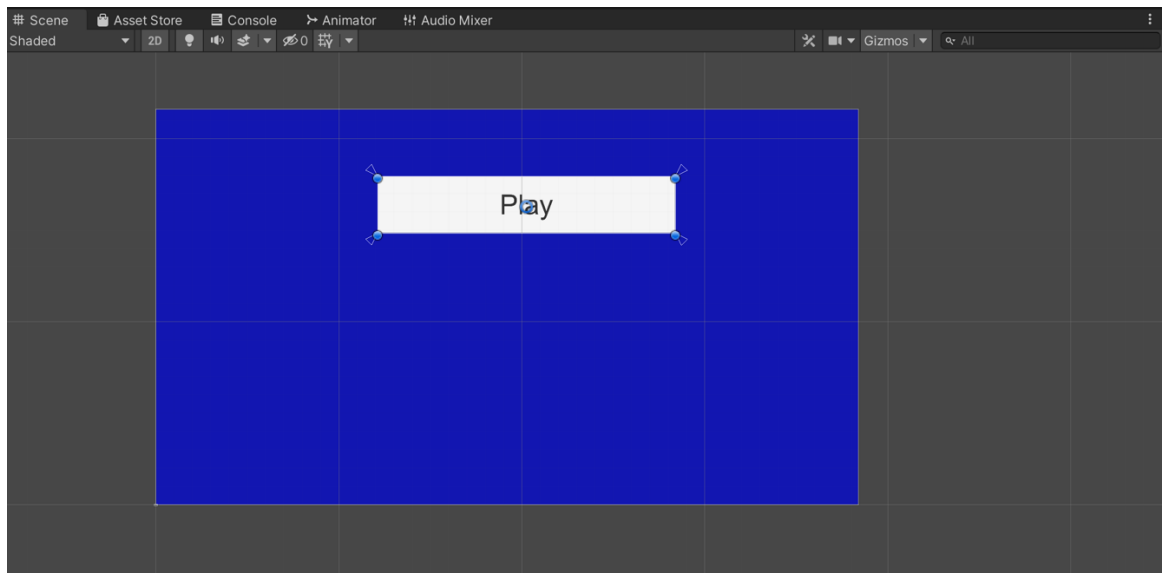
On réajuste la taille du bouton.



Via le panneau Inspector on change le texte par « Play » (ou Jouer).



On change la taille du texte.
// On obtient ceci



Dans le panneau Inspector du bouton, on va apporter ces modifications pour donner l'animation de bouton :

Changer en couleur Noir

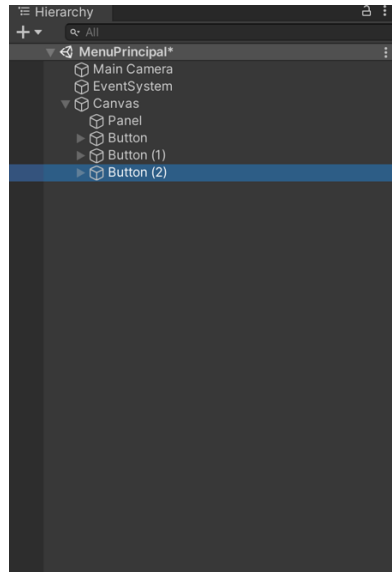
Cette valeur

- Normal Color = 0
- Highlighted Color = 61
- Pressed Color = 117

/*Le font par défaut d'Unity est Arial, mais il est possible de changer de Font. Soit en important un font personnel ou aussi importer un font via l'Asset Store d'Unity.*/

Étape 5 :

Faire un copier-coller du 2 fois du bouton crée (vous pouvez utiliser le raccourci clavier **Ctrl+D** pour dupliquer)

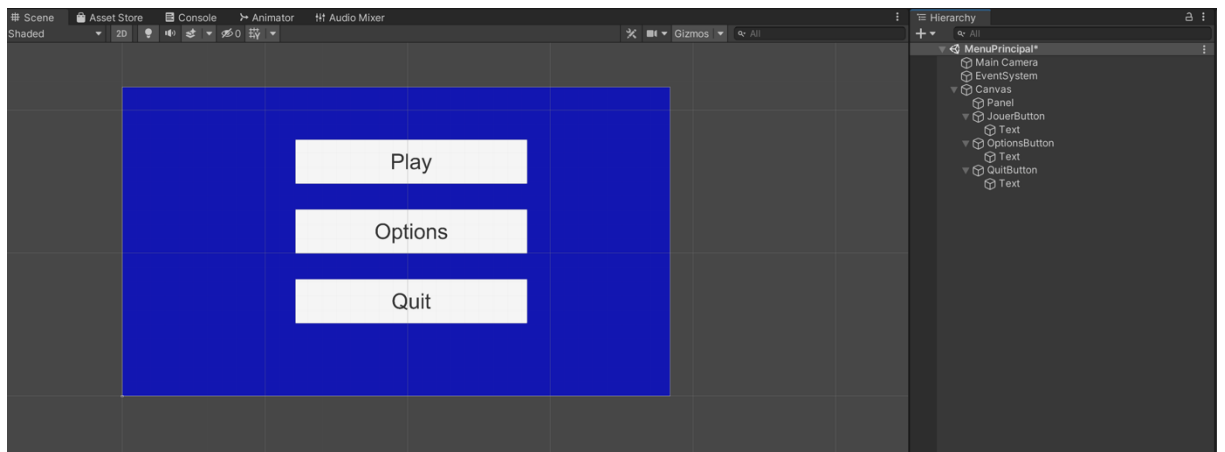


// Ces deux autres copies seront nos boutons Options et Quitter.

On change les Text des deux copies par « Options » et « Quitter ».

On repositionne et réajuste les boutons.

// On obtient alors ceci



Étape conseil :

Pour ne pas se perdre, renommer les Objects Buttons par les noms : « JouerButton », « OptionsButton », « QuitButton ».

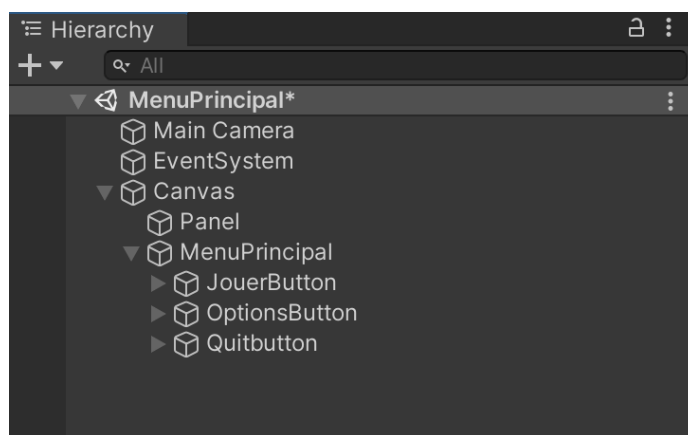
À ce stade, vous avez trois boutons, si vous testez le jeu vous verrez que les trois boutons font une animation de clic : une animation de bouton.

Cependant, ces boutons ne font rien.

Avant de continuer, pour éviter de tout laisser en bazar, on va « ranger » nos éléments.

Glisser et déposer les trois boutons dans le « GameObject » créé au début du workshop.

Glisser ensuite ce GameObject dans le Canvas.

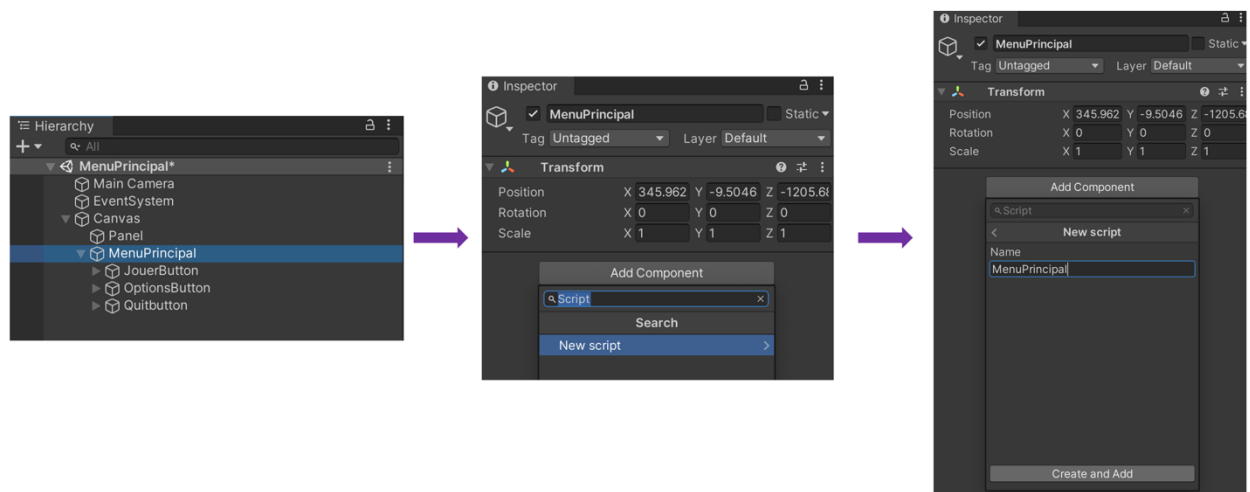


// Et voilà c'est rangé !

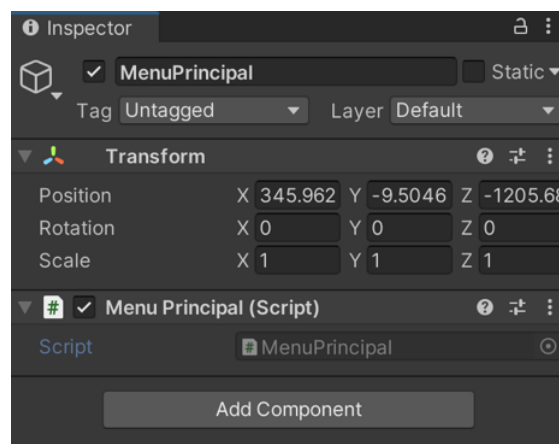
Maintenant, on voudrait que nos boutons fassent ce qu'on leur demande de faire, c'est-à-dire être un menu. La façon de faire est de créer un script qui va donner à chacun de nos composants un comportement.

Étape 6 :

Dans le panneau « Hierarchy », sélectionner le GameObject MenuPrincipal, dans le panneau « Inspector » : « Add Component », chercher « New script » puis « Create and Add ».



En faisant ceci, vous avez assigné un script qui sera responsable du comportement de tous les composants du GameObject.



Double cliquer sur le titre du Script, votre IDE va alors s'ouvrir.
Remplacez tout ce qu'il y a dans le fichier par ceci :

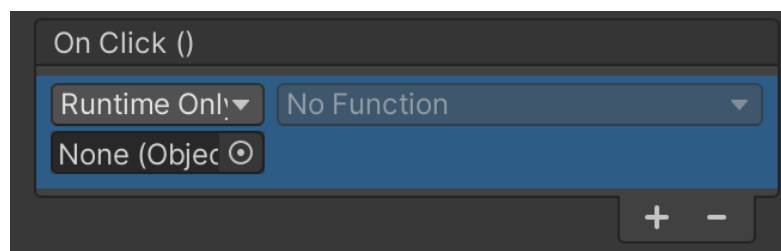
```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.SceneManagement;
5
6  public class MenuPrincipal : MonoBehaviour
7  {
8      void Start()
9      {
10         Application.targetFrameRate = 60;
11     }
12
13     public void GoToGame()
14     {
15         SceneManager.LoadScene("JEU");
16         //Cette fonction va nous permettre de charger la scene du jeu
17     }
18
19     public void QuitGame()
20     {
21         Application.Quit();
22         //cette fonction permet de quitter le jeu.
23     }
24 }

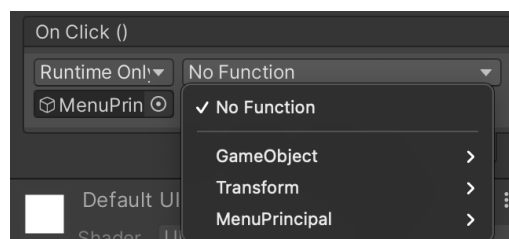
```

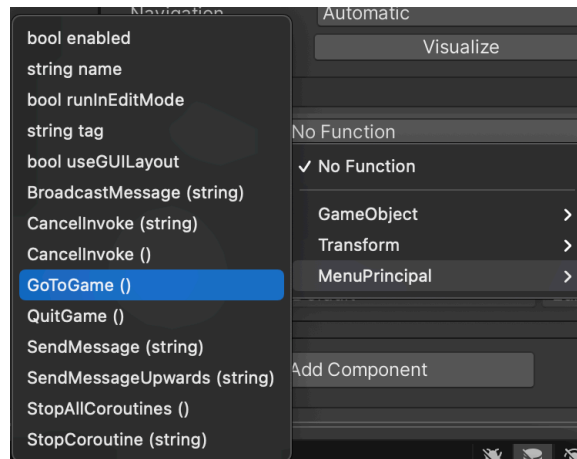
Étape 7 :

Sélectionner le bouton Jouer, dans son Inspector, cliquez sur le '+' de « On Click() ».

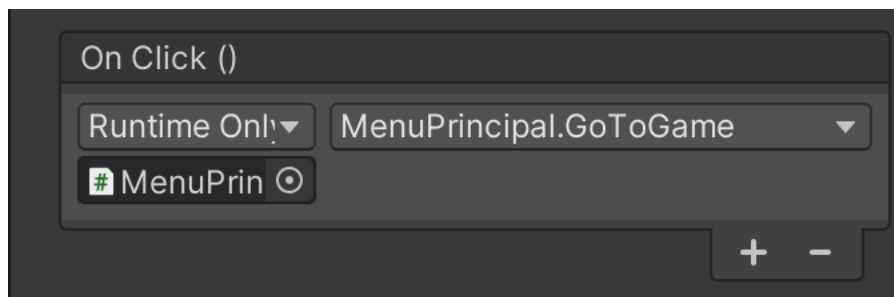


Glisser et déposer le GameObject MenuPrincipal à la place de la case « None (Object) ».





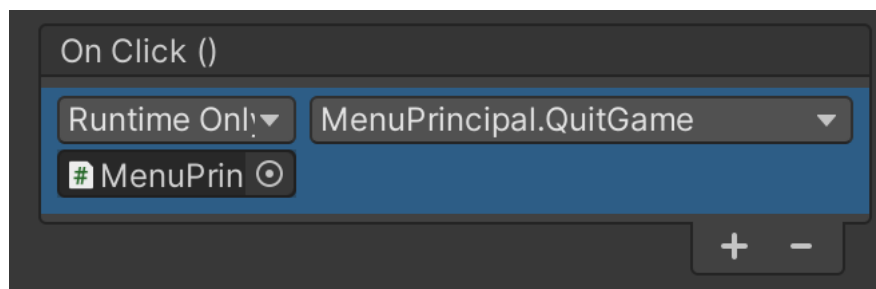
Vous Obtenez ceci :



Si vous testez le jeu, vous verrez que lorsque vous appuyez sur le bouton Jouer, vous avez bien la Scene de jeu qui s'affiche.

// Bravo vous avez un bouton Play fonctionnel

Pour rendre votre bouton Quitter fonctionnel, vous réitérez la même manipulation que le bouton Play mais en sélectionnant comme fonction : `QuitGame()`.

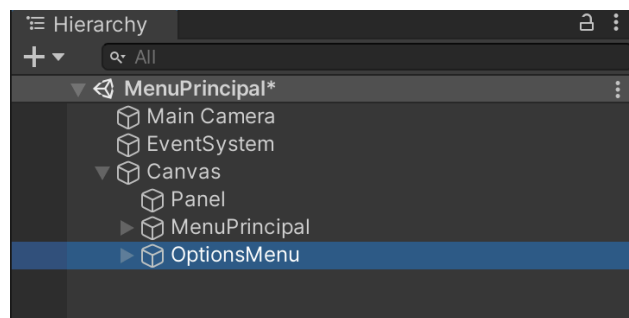


// Si vous testez le jeu et appuyer sur Quit, rien ne se passera, c'est normal, mais ce bouton Quitter marche vraiment. À la fin du développement de votre jeu, via Build Settings, vous créez un exécutable en cliquant sur Build. Cet exécutable sera votre jeu. Lorsque vous lancerez votre jeu via l'exécutable, en cliquant le bouton Quitter, le jeu se fermera. 😊

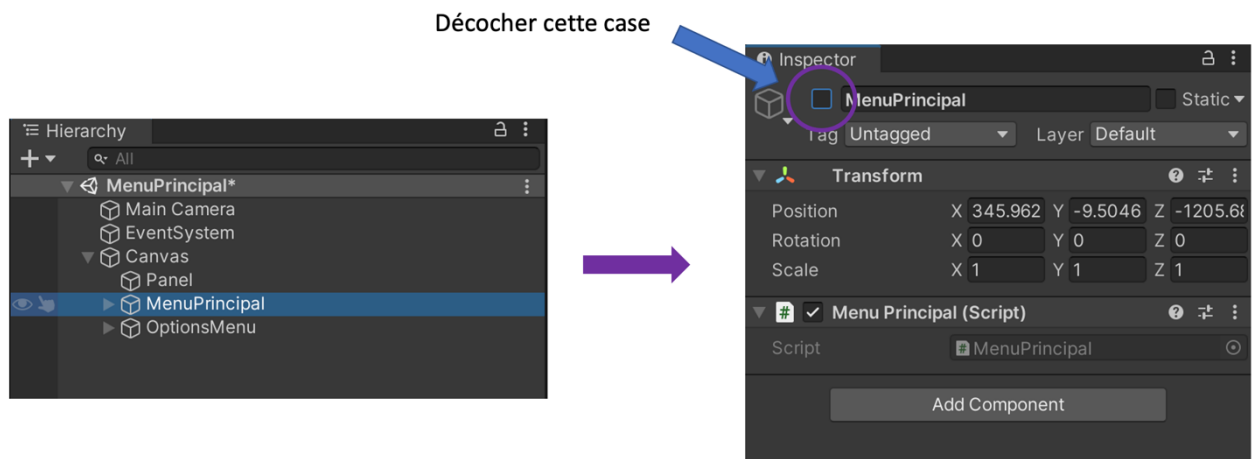
Maintenant que nous avons les boutons Play et Quit, on s'attaque au bouton Options.

Étape 8 :

Copier-Coller le GameObject « MenuPrincipal ».
Renommer la copie en « OptionsMenu ».



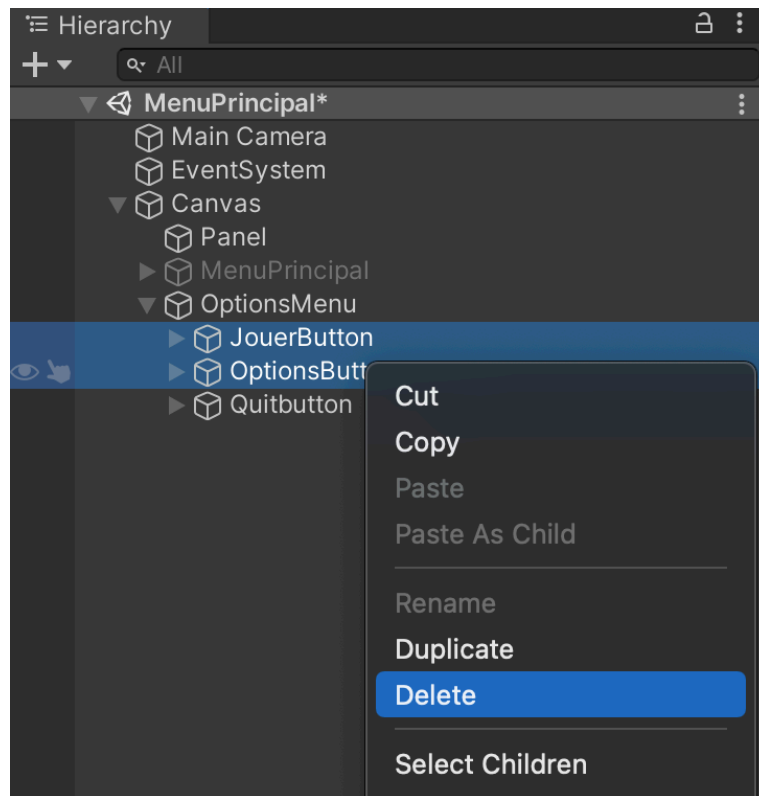
// Ce sera notre menu Options



L'action de décocher la case permet de ne pas afficher l'élément sélectionné.

Étape 9 :

Supprimez les boutons Play et Options (*On n'en a pas besoin*).

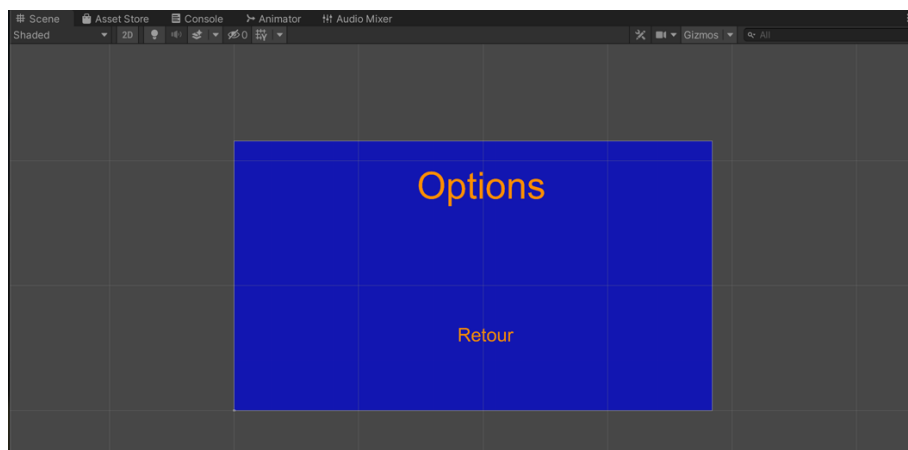


Ajouter au GameObject un Text.

Changer ce texte en « OPTIONS ».

Réajuster le texte, pour que cela apparaisse comme titre.

Et changer le texte « Quit » en « Retour ».

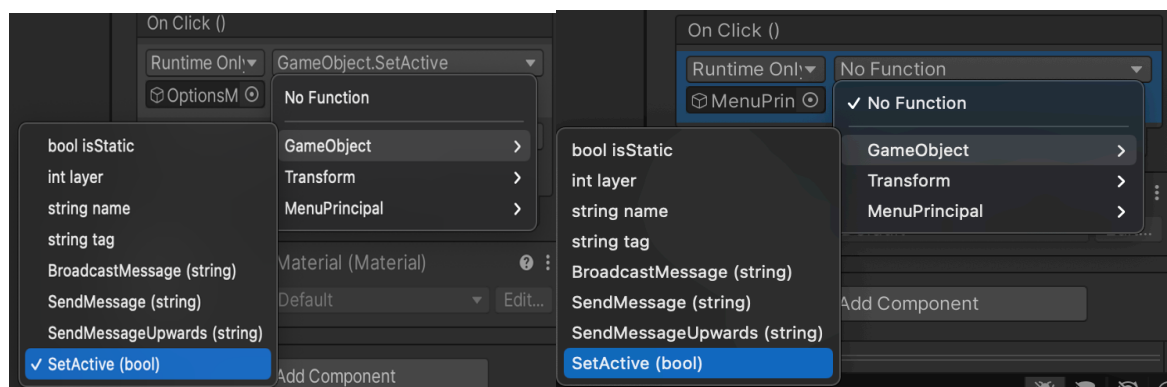


Nous souhaitons que le bouton retour nous fasse retourner au menu principal. Comment faire ?

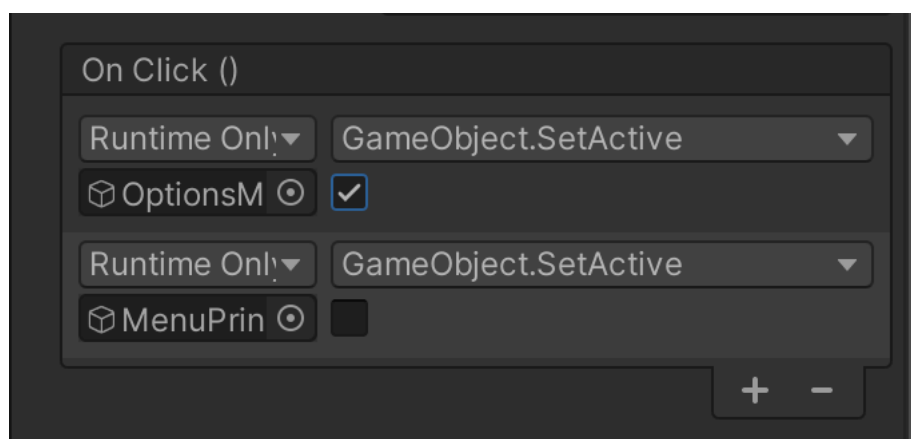
Réponse : Ce n'est pas via un script non. Nous allons utiliser une fonctionnalité d'Unity de contrôle des GameObject : « SetActive ». Qui fonctionne comme un booléen.

Étape 10 :

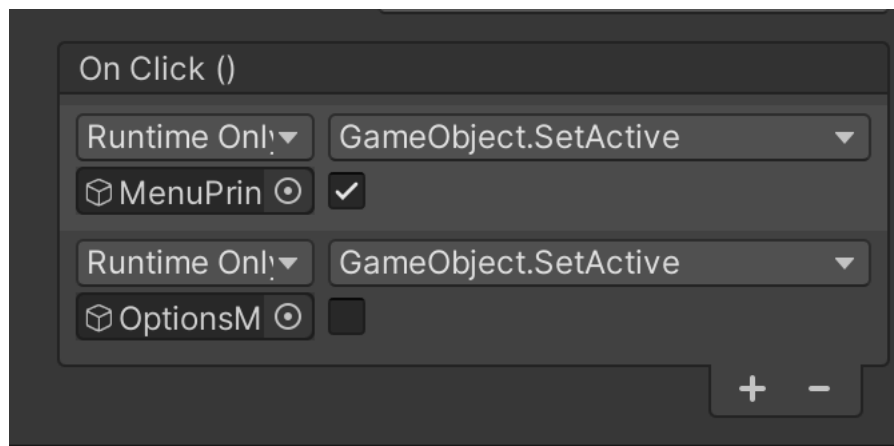
Dans l'Inspector du bouton Options, dans la catégorie OnClick(), appuyer sur le '+' 2 fois, puis sélectionner via GameObject -> SetActive (bool) sur les deux lignes.



Glisser le GameObject « OptionsMenu » dans la première ligne et glisser le GameObject « MenuPrincipal » dans la seconde ligne. Vous devez avoir ce résultat :



Puis dans le GameObject « OptionsMenu », changez les valeurs de la catégorie OnClick() du bouton « Retour », de sorte à ce que vous obteniez ceci :



// Si tout s'est bien passé, lorsque vous lancez votre jeu, que vous cliquez sur Options, le menu Options apparaît et lorsque vous cliquez sur retour vous revenez sur le menu principal et lorsque vous appuyez sur Play, la scene qui correspond au jeu apparaît.

Fin.