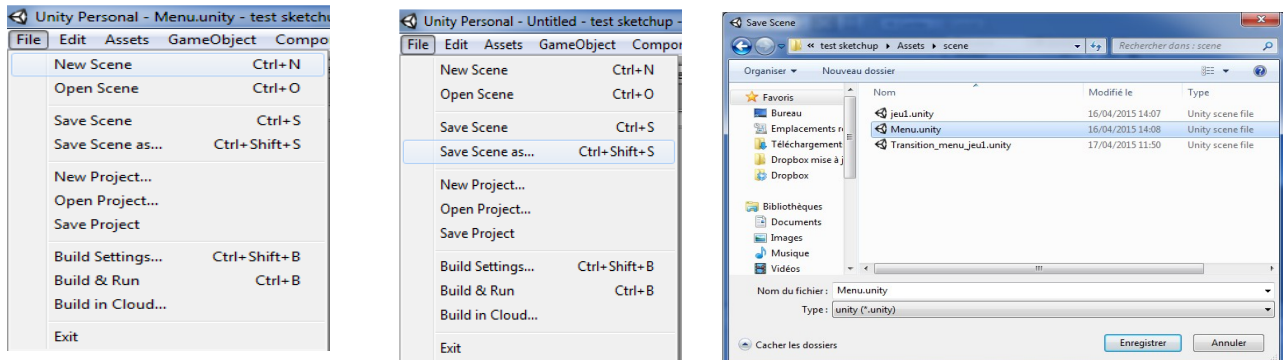
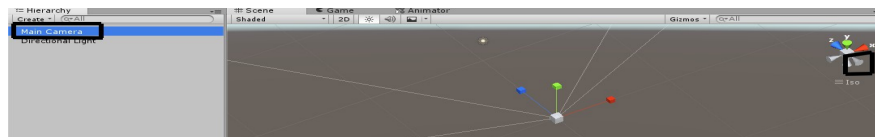


Comment faire un menu de base avec unity3D

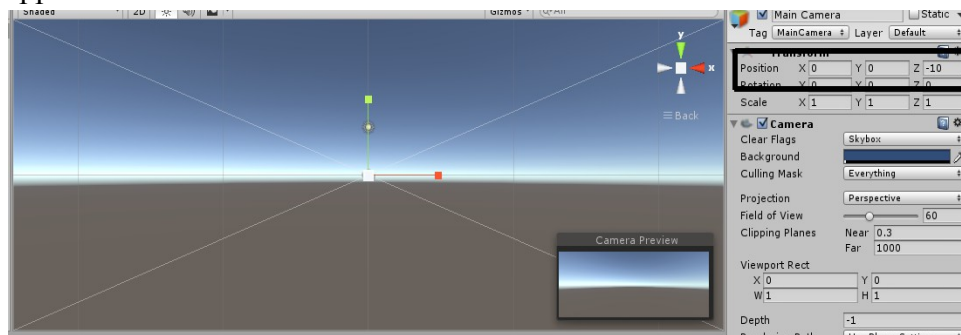
Etape 1: Réaliser une nouvelle scène de menu et l'enregistrer dans le répertoire des scènes :



Etape 2: Adapter votre vue de manière à ne pas avoir de difficultés pour visualiser la scène :

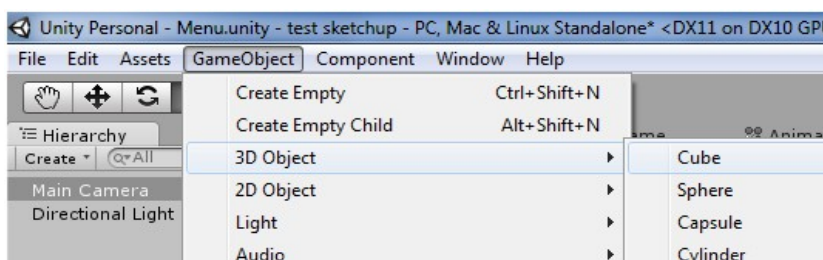


Double cliquer sur la caméra dans "Hierarchy" pour centrer la caméra au milieu de l'écran puis cliquer sur la flèche en opposition à l'axe des Z dans la visualisation des axes en haut à droite de la fenêtre.

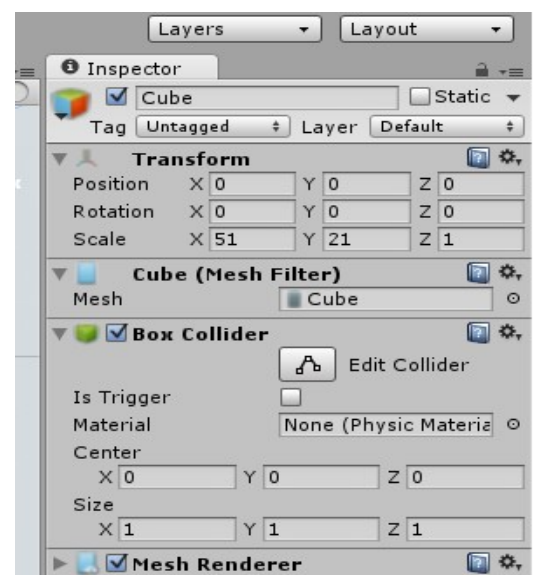


Vous devez obtenir cela. Vérifier que la position de la caméra soit (0;0 ;-10)

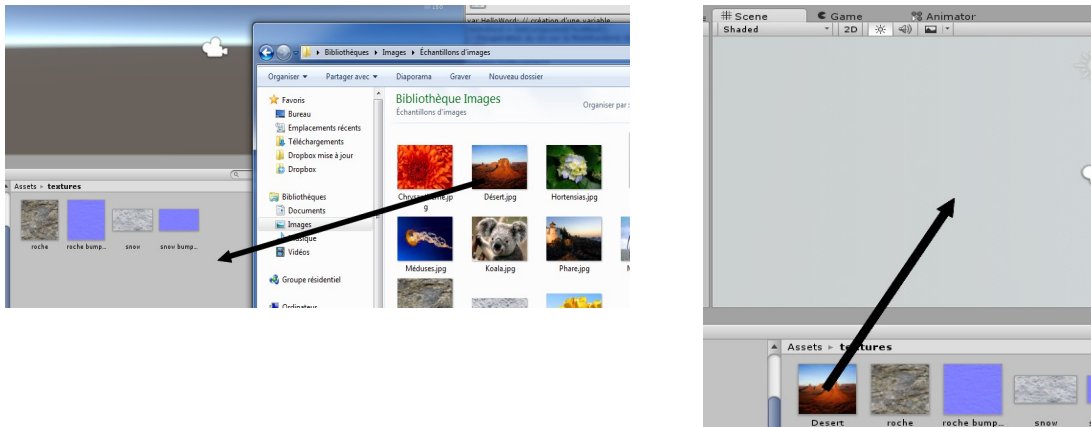
Etape 3: Réaliser le support de l'image de fond (sinon on a l'horizon en fond) :



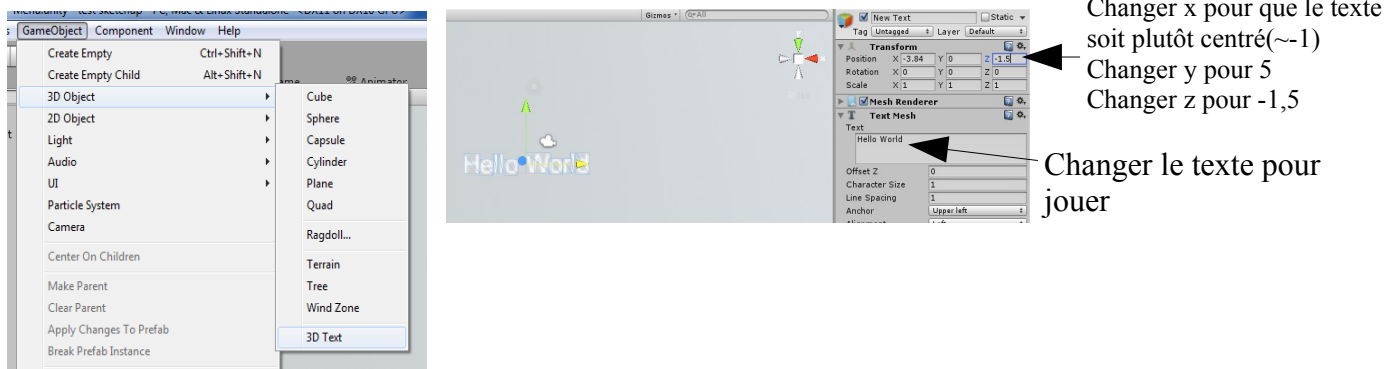
Position : (0;0;0)
Rotation : (0;0;0)
Scale : (51;21;0)



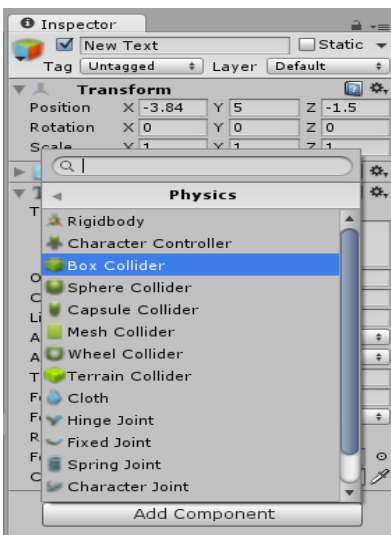
Etape 4: Placer l'image de fond dans le dossier texture puis sur le cube support :



Etape 5: Placer les textes jouer, quitter et options (si il y en a) :



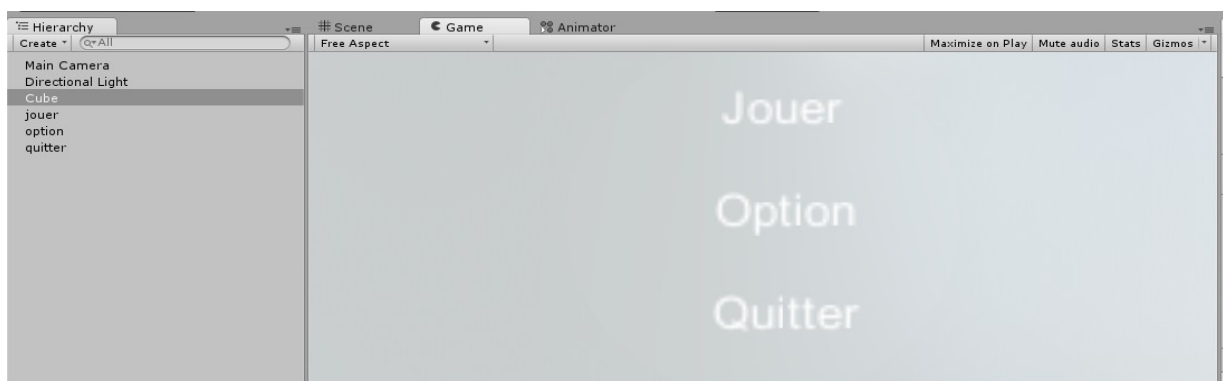
En bas de l'inspector, cliquer sur Add Component puis physics et Box Collider



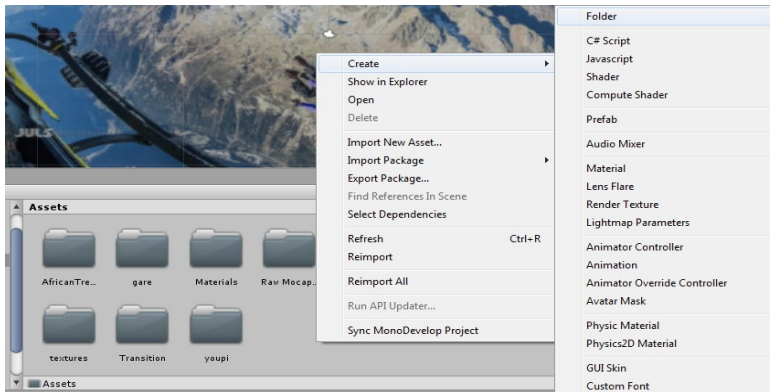
Copier coller le texte "jouer" et modifier les nouveaux textes en "options" et "quitter" .

Options :
Position : (~-1;2;-1,5)
Rotation : (0;0;0)
Scale : (1;1;1)

Quitter :
Position : (~-1;-1;-1,5)
Rotation : (0;0;0)
Scale : (1;1;1)

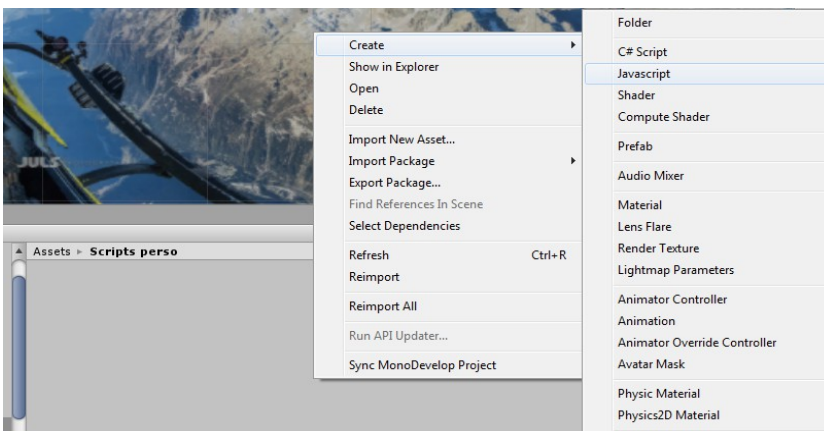


Etape 6: Réaliser un dossier pour vos scripts :



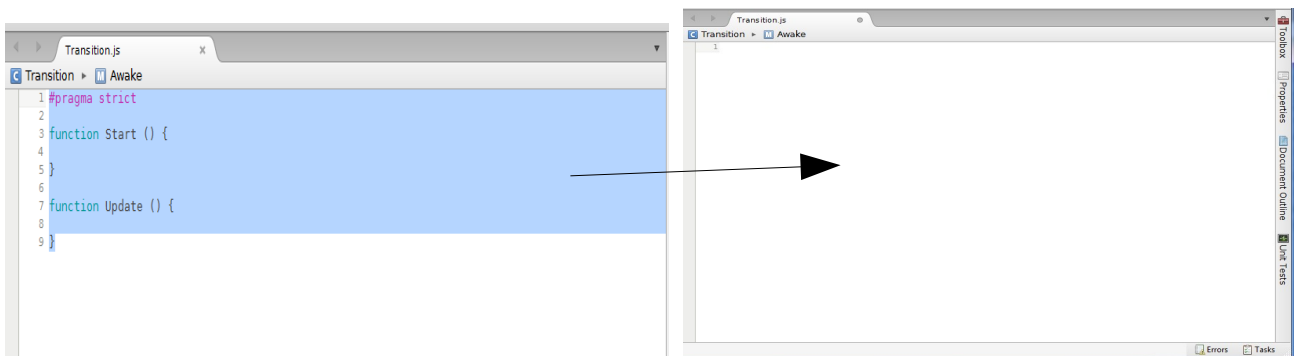
Créer un dossier Scripts perso, pour les séparer des scripts du standard assets

Etape 7: Réaliser un script en Java pour animer les textes et créer le passage de scène:



Etape 8: Editer le script :

Double cliquer sur le script et retirer les portions de code par défaut :



Taper ensuite le code suivant :

Le code va permettre d'affecter les modifications sur la zone de texte jouer.

Si le curseur est au dessus du texte, le texte devient rouge.

Si on relâche le bouton, le niveau 1 est chargé.

Si on retire le curseur d'au dessus du texte, celui-ci redevient blanc.

```
var Jouer; // création d'une variable "Jouer"
```

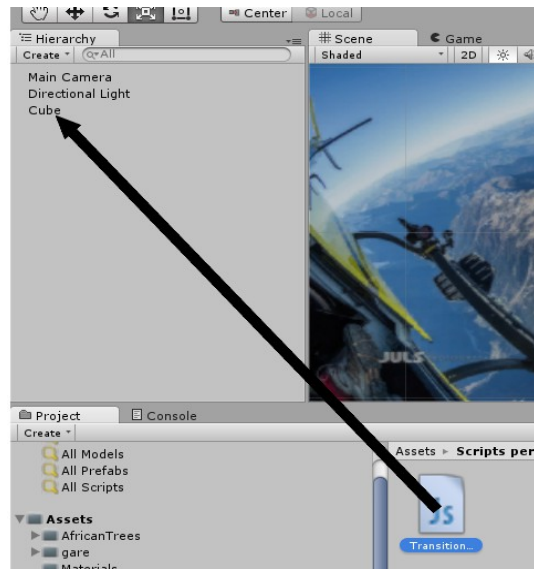
```
Jouer = GetComponent(TextMesh); // Détection du curseur sur le MeshRendered du texte
```

```
function OnMouseUp(){  
Application.LoadLevel("jeu1");  
} // Si le clic remonte (OnMouseUp), alors le niveau "jeu1" est chargé
```

```
function OnMouseEnter(){  
Jouer.color = Color.red;  
} // Si le curseur passe au dessus du texte (OnMouseEnter), alors le MeshRendered du texte passe au rouge
```

```
function OnMouseExit(){  
Jouer.color = Color.white;  
} // Si le curseur sort d'au dessus du texte (OnMouseExit), alors le MeshRendered du texte passe au rouge
```

Etape 9: Placer le script sur l'objet :



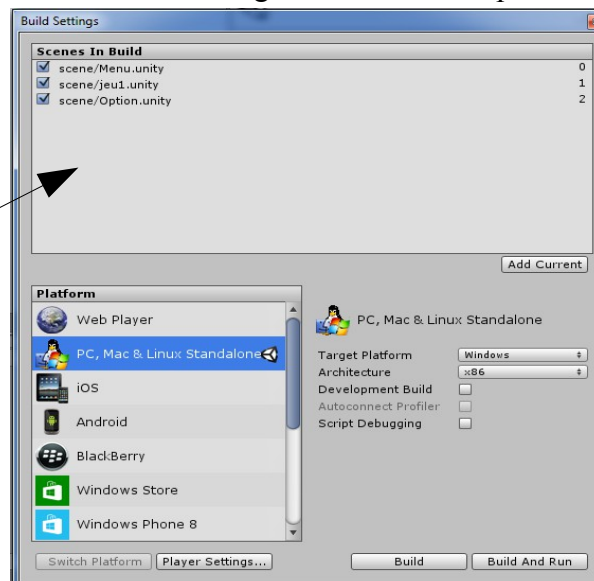
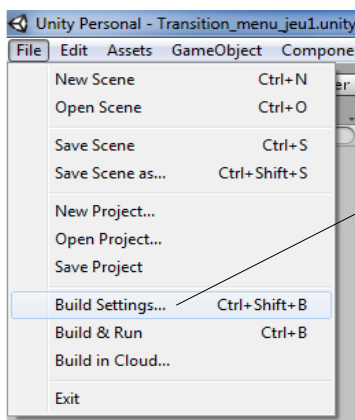
Etape 10: Reproduire le script en modifiant pour "Option" et "Quitter" :

Pour "Option" Application.LoadLevel("jeu1") devient Application.LoadLevel("Option")

Pour "Quitter" Application.LoadLevel("jeu1") devient Application.Quit()

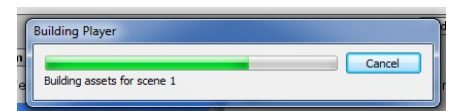
(Penser à Placer les Box Collider et les scripts correspondants sur les textes "Option" et "Quitter")

Etape 11: Placer les scènes dans l'interface de gestion des scènes pour tester :

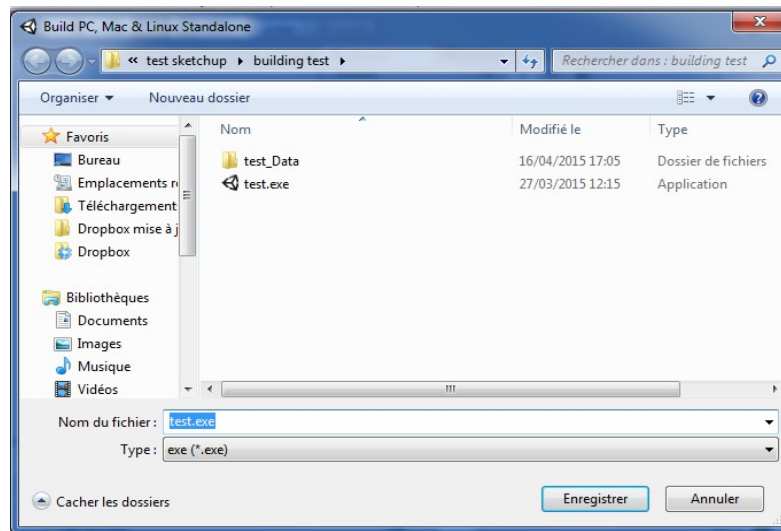


Placer les scènes à tester
(ici notre scène de transition et
le jeu)

Cliquer sur Build pour
compiler la portion de jeu
Cliquer sur Build and Run pour
tester la version finale du jeu



Etape 12: Enregistrer votre compilation dans un dossier de test. Utiliser le même dossier pour enregistrer vos futures essais :



Etape 13: Enregistrer vos scènes et votre projet :

Enregistrer la scène = enregistrer les éléments présents dans votre jeu

Enregistrer le projet = enregistrer les éléments présents dans les assets.

Etape 14: Pour aller plus loin:

L'ensemble des éléments présents dans la partie Inspector est modifiable par le script.

Ainsi, si l'on change

```
function OnMouseEnter(){
Jouer.color = Color.red;
}
```

par

```
function OnMouseEnter(){
Jouer.color = Color.red;
Jouer.transform.localScale += new Vector3(0.5F, 0, 0); // Augmentation de l'échelle x0,5
}
```

alors le texte va être 0,5 fois plus large, mais il va se décaler vers la droite uniquement. Si on lui indique de se déplacer à gauche en même temps de 0,5, alors le texte va rester centré.

```
function OnMouseEnter(){
Jouer.color = Color.red;
Jouer.transform.localScale += new Vector3 (0.5F, 0, 0);
Jouer.transform.position = Vector3(-1.62F, 5F, -1.5F); // Déplacement du texte 0,5unités en x (position d'origine de mon
texte (-1.12 , 5 , -1.5))
}
```

Bien sûr, il faut rétablir les valeurs initiales en sortie du curseur :

```
Jouer.transform.localScale += new Vector3(-0.5F, 0, 0);
Jouer.transform.position = Vector3(-1.12F, 5F, -1.5F);
```