

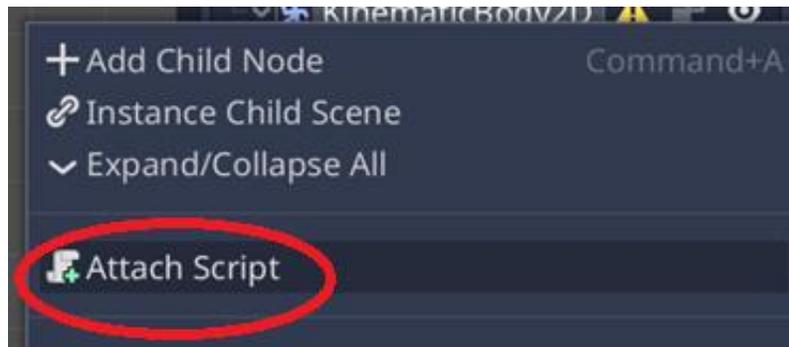
## MODULE 2 : CRÉER LE CODE

Maintenant, ajoutons du code! Un script est le code qui va avoir un impact sur l'objet.

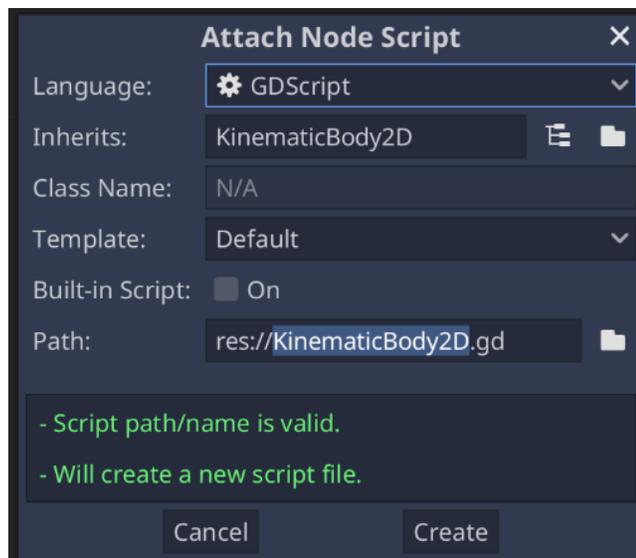
### AJOUTER UN SCRIPT

D'abord, on va ajouter un script pour le nœud parent « KinematicBody2D », puis on va écrire le code.

1. Dans le widget « Scene », fais un clic droit sur « KinematicBody2D ».
2. Clique sur « Attach Script ».



3. Pas besoin de modifier les paramètres, tu peux simplement cliquer sur « Create ».



4. Tu verras alors le script à l'écran. Supprime tout le code en-dessous de la ligne 2. Le seul code que tu dois garder est la ligne 1 : « extends KinematicBody2D ».

5. C'est le moment d'écrire du code! Tape le code exactement comme dans l'image ci-dessous. Si tu te trompes, souviens-toi que tu peux toujours annuler en appuyant sur CTRL + Z.

```

1  extends KinematicBody2D
2
3  var motion = Vector2()
4
5  func _physics_process(delta):
6  >|
7  >|     if Input.is_action_pressed("ui_right"):
8  >| >|     motion.x = 100
9  >| >|
10 >|     elif Input.is_action_pressed("ui_left"):
11 >| >|     motion.x=-100
12 >| >|
13 >|     else:
14 >| >|     motion.x=0
15 >|     move_and_slide(motion)
16 >|     pass
17

```

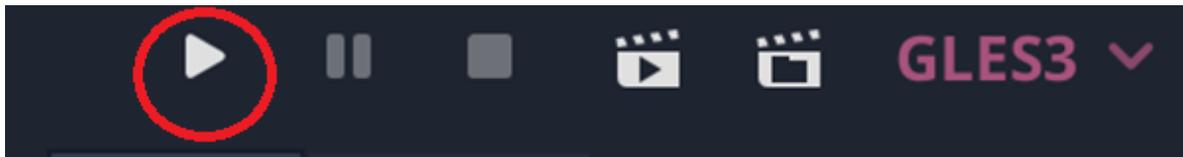
### QUE SIGNIFIE CE CODE?

Le code que tu viens d'écrire dit au logiciel que l'objet est lié par un mouvement 2D. Un moteur physique a été intégré et le personnage peut être déplacé vers la gauche ou vers la droite par un mouvement de glissement.

## TESTONS LE CODE

Et si on testait notre jeu?

1. Enregistre ton jeu. (Ctrl + S) ou dans le menu, Scene > Save Scene.
2. Clique sur le bouton de lecture en haut à droite du logiciel, au-dessus du widget « Scene » (tu peux aussi appuyer sur le raccourci F5).
3. Clique sur « Select » pour confirmer. Si tu vois s'afficher la fenêtre « Pick a main scene », sélectionne le fichier que tu as enregistré et clique sur « Open ».
4. S'il n'y a pas d'erreurs dans le code, le jeu s'ouvrira dans une nouvelle fenêtre.
5. C'est le moment de jouer! Si tu appuies sur les flèches gauche et droite de ton clavier, ton personnage devrait bouger vers la gauche ou vers la droite.



### CONSEIL DE PRO :

- S'il y a des erreurs, le jeu ne s'ouvrira pas correctement. Le code doit être exact pour que tout fonctionne bien.
- S'il y a des erreurs dans le code, elles seront mises en évidence en rouge, avec une explication dans le bas sur la manière de les corriger.
- S'il y en a, corrige toutes les erreurs et clique à nouveau sur le bouton de lecture. Le jeu devrait maintenant s'ouvrir et fonctionner correctement.
- Tu peux ignorer les avertissements en jaune. Ils indiquent des erreurs mineures qui n'empêcheront pas le code de fonctionner.
- Un autre problème que tu pourrais rencontrer, c'est si tu as choisi l'option 3.0 mais que ton ordinateur ne prend pas en charge ce niveau de graphisme. Pour résoudre ce problème, tu devras copier le code et recréer le projet en choisissant l'option 2.0.
- Si la fenêtre de ton jeu est juste un fond gris (sans joueur), essaie de faire un zoom sur la fenêtre.

## GRAVITÉ

Ajoutons maintenant de la gravité! On a déjà indiqué au jeu comment se déplacer le long de l'axe X (horizontal) en 2D. Maintenant, on veut lui dire comment se déplacer le long de l'axe Y (vertical) pour que le personnage puisse se déplacer vers le haut et vers le bas.

1. Dans le widget « Scene », retourne dans ton script sous « KinematicBody2D ».
2. Ajoute une nouvelle ligne de code à la ligne 6 : « motion.y+=10 ».
3. Enregistre le fichier.
4. Clique sur le bouton de lecture.

```
5 v func _physics_process(delta):
6   >|   motion.y+=10
7 v >|   if Input.is_action_pressed("ui_right"):
```

## QU'EST-CE QUI VIENT DE SE PASSER?

Notre personnage vient juste de tomber dans le vide! Créons un mur pour arrêter sa chute.

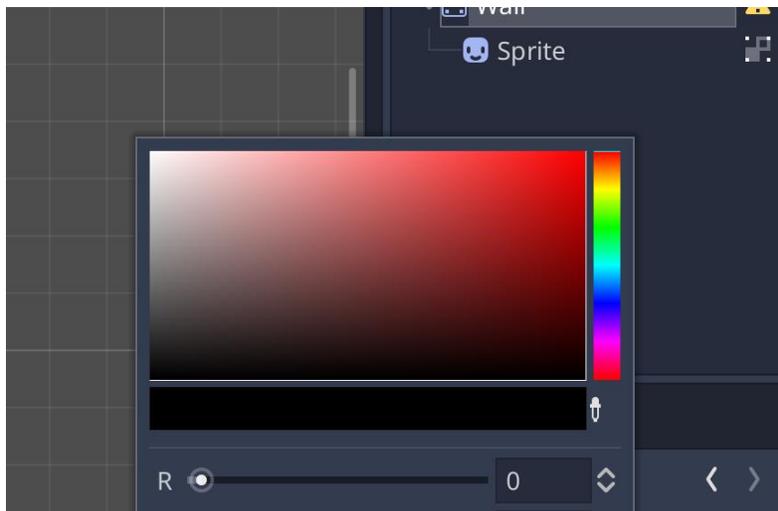
1. Ajoute un nouvel élément « StaticBody2D », exactement comme tu as ajouté le sprite :
  - a. Va dans « Scene » et clique sur le « + ».
  - b. Tape « StaticBody2D » dans la barre de recherche et clique sur « Create ».
  - c. Dans le widget « Scene », double-clique sur « StaticBody2D » et renomme-le « Wall » (ou « Mur »).
2. Ajoute un deuxième sprite, cette fois-ci sous le « StaticBody2D » :
  - a. Va dans « Scene ».
  - b. Sélectionne « Wall », puis clique sur le « + ». Tape « Sprite » dans la barre de recherche et clique sur « Create ».
  - c. Après avoir créé le deuxième sprite, sélectionne-le.
  - d. Fais glisser l'image « icon.png » sous le widget « FileSystem » jusque dans l'espace à droite de « Texture » dans le widget « Inspector » (exactement comme tu as fait pour créer le premier sprite au module 1). Tu devrais maintenant voir deux images de Godot sur ton écran. Pour transformer une des images en mur, on va changer sa couleur sur noir.

**CONSEIL DE PRO :**

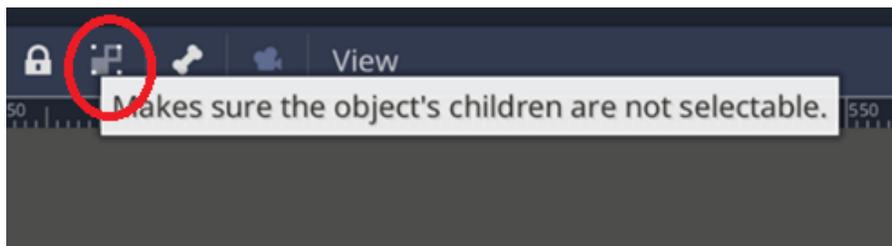
- Tu as sélectionné « StaticBody2D » plutôt que « KinematicBody2 » car cet objet restera statique (il ne bougera pas).

## 3. Change la couleur du mur :

- Clique sur « Wall ».
- Va dans le widget « Inspector ».
- Va dans Inspector > CanvasItem > Visibility > Modulate.
- Change la couleur sur noir (code #000000 au lieu de #FFFFFF).

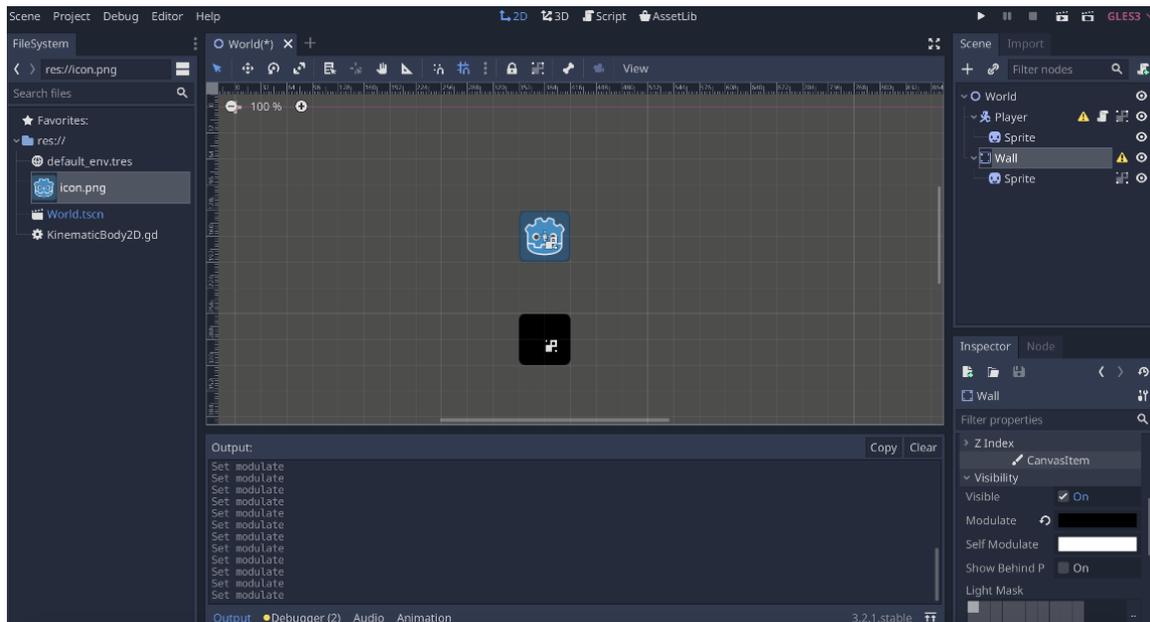


## 4. Relie le mur et le sprite.



## 5. Double-clique sur « KinematicBody2D » et renomme-le « Player » (ou « Joueur »).

6. Glisse le mur qui est maintenant noir pour le déposer sous le personnage, et clique sur le bouton de lecture.



Qu'est-ce qui s'est passé? Notre joueur est passé à travers le mur!

**Découvrir plus de modules pour Développement de jeu avec Godot!**

**Découvrir le tutoriel vidéo sur Godot!**

**Retourner à l'Apprentissage en ligne Académie Geek Squad!**

**Suggestions, questions ou commentaires? Envoyez-nous un courriel à [academy@geeksquad.ca](mailto:academy@geeksquad.ca)**