

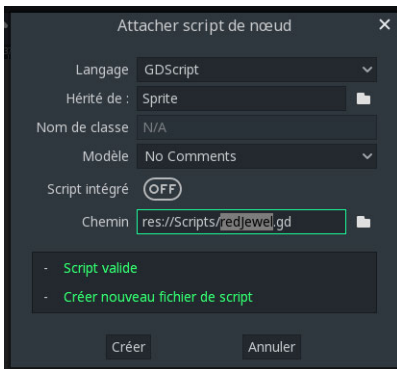
Notre personnage ainsi que notre pierre sont configurés comme il faut ! Nous pouvons maintenant programmer les interactions.

13.2. Création du script de ramassage

Penchons-nous maintenant sur l'écriture du script qui permettra au personnage de ramasser les pierres rouges lorsque celui-ci les touchera. Ce script pourra parfaitement être adapté au ramassage de pièces ou de n'importe quel autre type d'objets. Vous pourrez donc l'ajuster selon vos besoins.

Retournez sur la scène de la pierre précieuse et cliquez sur celle-ci dans la fenêtre Scène. Cliquez ensuite sur l'icône de création de scripts. Donnez un nom à votre script, choisissez NO COMMENTS et enregistrez-le dans le dossier des scripts.

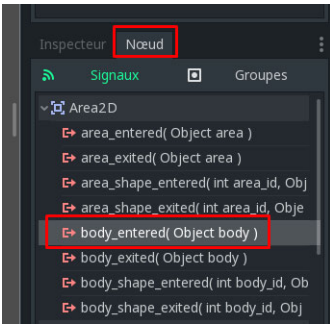
Figure 13.7 : Création du script de ramassage



Par défaut, le script contiendra la fonction `ready`. Supprimez-la, nous n'en n'avons pas besoin. Nous allons plutôt avoir besoin d'une fonction qui se déclenchera lors d'une collision.

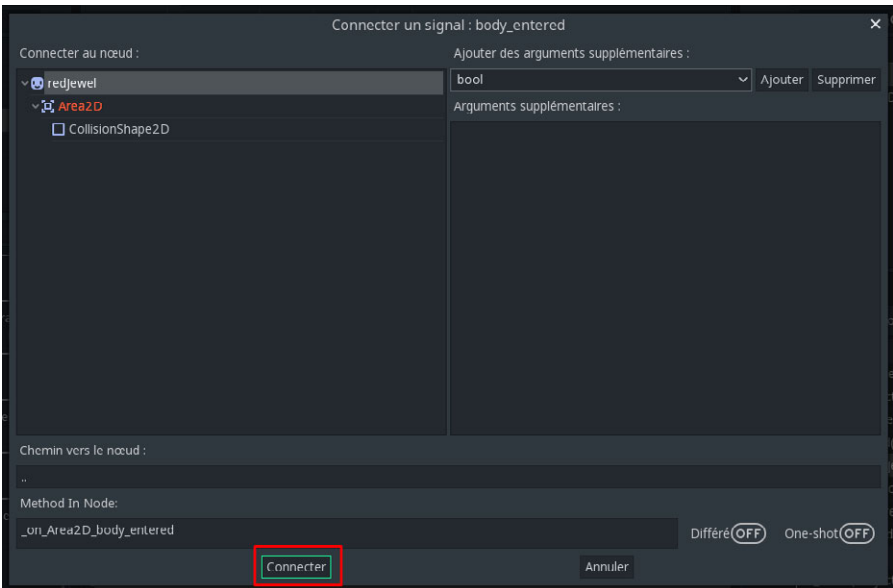
Lorsque vous ajoutez un composant Area2D à un objet (ce qui est le cas de notre bijou), vous avez accès à un certain nombre de fonctions spécifiques à ce composant, dont une fonction `body_entered`. Cette fonction se déclenche lorsqu'un Body entre dans l'Area2D. Pour retrouver cette fonction, sortez de votre script pour retourner sur la scène de votre pierre précieuse et cliquez sur son Area2D. En bas, au niveau de l'inspecteur, vous trouverez un onglet NŒUD dans lequel sont regroupées toutes les fonctions propres au Area2D, et en particulier la fonction `body_entered`.

Figure 13.8 : Liste des fonctions disponibles



Cliquez sur cette fonction puis cliquez sur CONNECTER en bas de la fenêtre NœUD. Une nouvelle fenêtre s'ouvre et vous demande de choisir l'objet qui sera connecté. Laissez tout par défaut et cliquez sur CONNECTER.

Figure 13.9 : Connecter la fonction

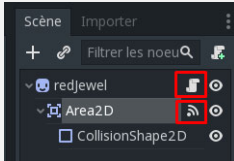


Cette action créera une nouvelle fonction `_on_Area2D_body_entered` dans votre script.

Votre signal est-il bien connecté au nœud ?

Pour connecter un signal à un node, il faut que le node en question possède un script et que celui-ci ait été enregistré. Vous pouvez savoir qu'un script est bien attaché grâce à l'icône Script. Vous pouvez savoir également qu'un signal est correctement connecté grâce à son icône.

Figure 13.10 : Icônes Script et Signal



Si vous essayez de connecter un signal à un nœud n'ayant pas de script, une erreur surviendra.

À l'intérieur de la fonction `_on_Area2D_body_entered`, nous allons décrire ce qui doit se produire lors de la collision entre le personnage et l'objet. Notre objectif est d'augmenter la richesse du personnage de 1 puis de faire disparaître le bijou car il a été ramassé.

Pour le moment, nous n'avons pas mis en place de système d'argent au niveau de notre personnage. Aucune variable ne stocke la quantité de pierres précieuses que le personnage possède. Nous allons rectifier cela en ajoutant une variable `var coins = 0` au script `Player`. Une fois ceci fait, retournez dans le script du bijou.

La première chose à faire est de créer une variable qui sera une référence vers le script du personnage car nous aurons besoin d'y accéder pour modifier la valeur de `coins`. Nous devons renseigner cette variable une fois que le jeu est lancé et que tous les scripts sont prêts. Nous utiliserons pour cela le mot clé `onready`. Notre variable s'appellera `PlayerScript`.

Pour récupérer un script se trouvant sur un objet de la scène, nous devons récupérer l'arbre avec `get_tree`, la racine avec `get_root` puis le nœud avec `find_node`. Vous pouvez en savoir plus sur ces fonctions en faisant un Ctrl+click sur celles-ci afin d'afficher leur documentation. La ligne complète est la suivante.

```
onready var PlayerScript = get_tree().get_root().find_node("Player",
    true, false)
```

Cette ligne nous permet de récupérer le Player et de le stocker dans une variable. Cela nous sera très utile pour accéder rapidement aux variables de Player notamment `coins`.

Dans la fonction `_on_Area2D_body_entered`, nous allons pouvoir déclencher l'ajout d'une pièce (une pierre précieuse) dans l'inventaire du personnage. Pour cela, écrivez :

```
PlayerScript.coins += 1
```

Si vous souhaitez vérifier que tout fonctionne bien, affichez le nombre de pièces dans la console de cette façon :

```
print(PlayerScript.coins)
```

Ainsi vous verrez à chaque collision avec un bijou le nombre de pièces dans l'inventaire du personnage.

Pour terminer, comme le bijou a été récupéré, nous devons le faire disparaître. Nous utilisons la fonction `queue_free()` pour réaliser cela. Le script complet de votre pierre précieuse doit être le suivant :

```
extends Sprite

onready var PlayerScript = get_tree().get_root().find_node("Player",
true, false)

func _on_Area2D_body_entered(body):
    PlayerScript.coins += 1
    print(PlayerScript.coins)
    queue_free()
```