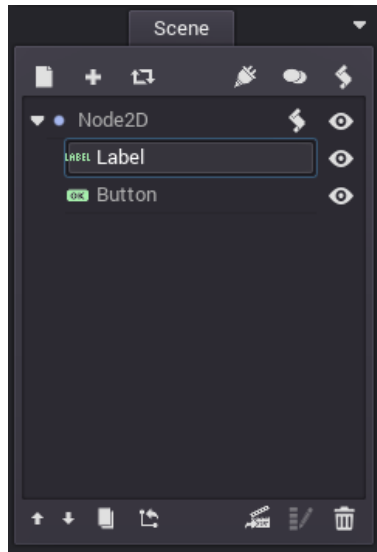
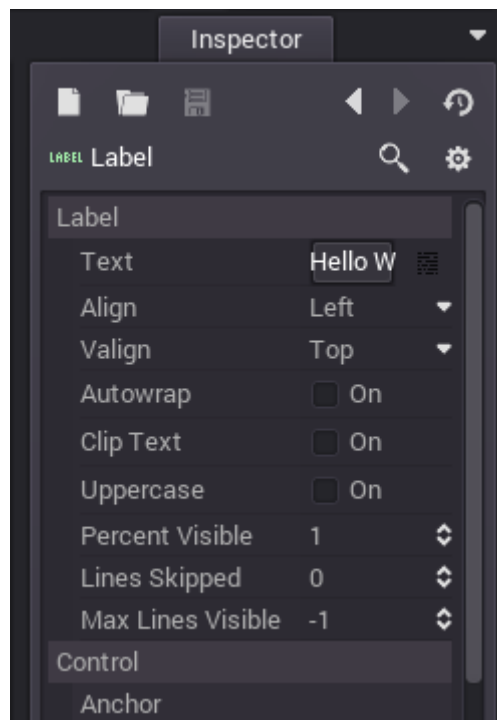


سلام . پس از یک وقفه ی نسبتاً طولانی با ادامه ی آموزش در خدمت شما هستیم . می خواهیم با چند مطلب ساده و حیاتی شما رو برای پروژه ی Pong کنیم . بهتره به سراغ اصل مطلب بریم .
یک پروژه ی جدید بسازید . گره هایی مطابق تصویر زیر به صحنه اضافه کنید و سپس اسکریپتی به گره ریشه وصل کنید.

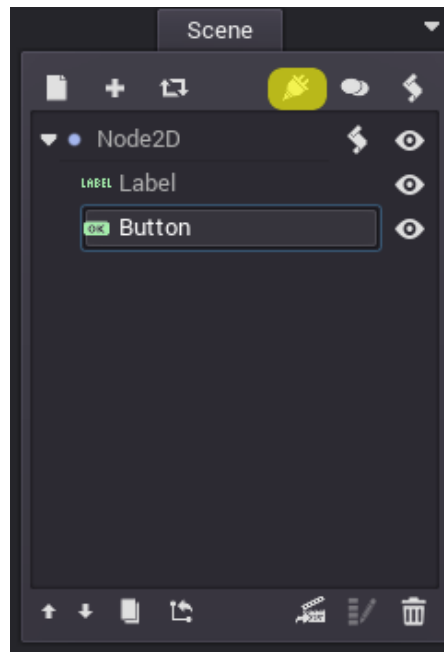


حالا با انتخاب گره Label در ویژگی Text آن متنی مثل Hello World را اضافه کنید.

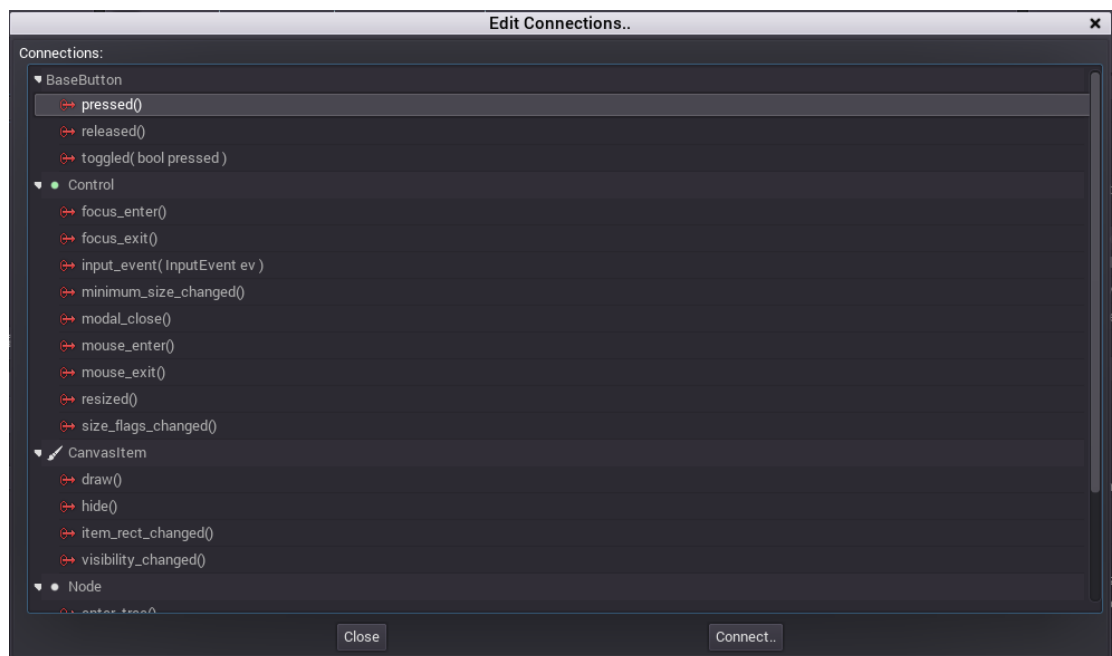


برای اضافه کردن عملکرد به دکمه ی Button یا اصطلاح وصل کردن (Connect کردن) آن دو راه وجود دارد : ۱- به صورت گرافیکی
۲- مستقیماً در اسکریپت .

برای اضافه کردن عملکرد به دکمه‌ها به صورت گرافیکی، ابتدا گره `Button` را انتخاب کنید و بعد بر روی آیکن دوشاخه مانند (آیکن مشخص شده در تصویر زیر) کلیک کنید.

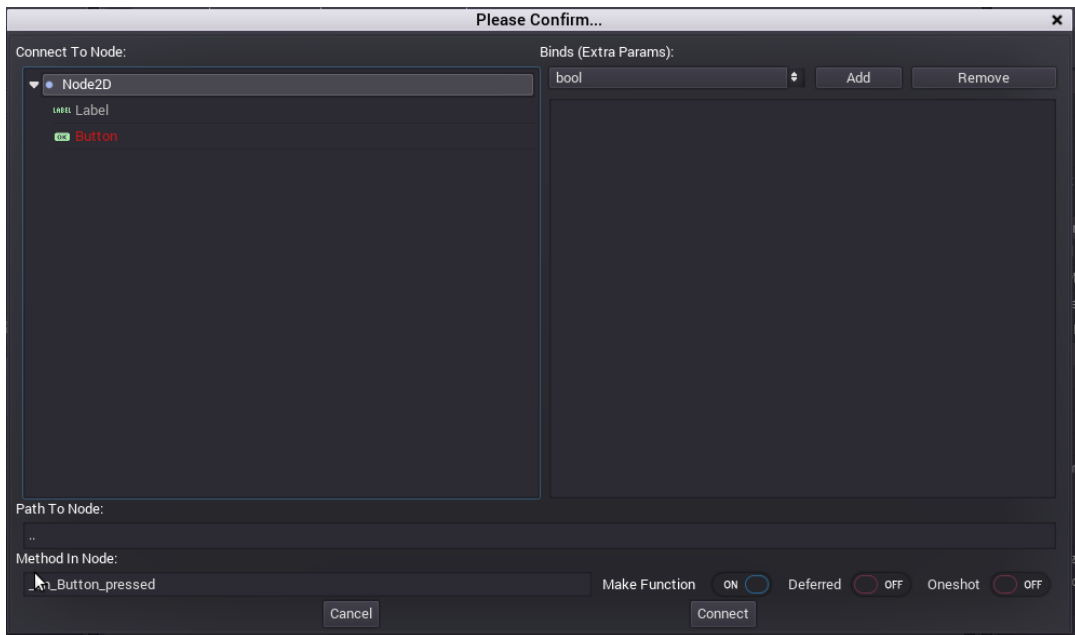


پنجره `Edit Connections` باز می‌شود که همانطور که می‌بینید می‌توانیم با استفاده از آن انواع مختلفی از ارتباط‌ها را انتخاب کنیم. در اینجا ما با انتخاب گزینه `pressed()` بر روی کلید `connect` کلیک می‌کنیم.



حالا وارد پنجره `Please Confirm` می‌شویم. در اینجا می‌توانیم از قسمت `Connect To Node`، گره‌ای را که می‌خواهیم این عمل‌کرد به آن وصل شود را انتخاب کنیم. ما گره `Node2D` را انتخاب می‌کنیم که به آن اسکریپتی وصل هست. حالا از قسمت

ما method in node می‌توانیم نام متدی که قرار است به صورت خودکار در اسکریپت گره Node2D ساخته شود را مشخص کنیم . بدون تغییر این نام ، بر روی دکمه ی Connect کلیک می‌کنیم .



بلافاصله برنامه ما را وارد اسکریپت گره هدف می‌کند و متدی با نام مشخص شده در آن می‌سازد .

```

01
02 extends Node2D
03
04
05 func _ready():
06 > # Called every time the node is added to the scene.
07 > # Initialization here
08 > pass
09
10
11
12 s://First.gd
13 func _on_Button_pressed():
14 > pass |
15

```

حالا متد مورد نظر را به این صورت تغییر می‌دهیم .

```

func _on_Button_pressed():
    self.get_node("Label").set_text("Text was changed !")

```

توضیح کد بالا :

در اینجا با استفاده از کلمه ی کلیدی self مشخص کردیم که به گره فعلی اشاره می‌کنیم ، سپس با استفاده از متد get_node() گره Label را گرفتیم . حالا با استفاده از متد set_text متن Label را تغییر دادیم .

صحنه را ذخیره کرده و بازی را اجرا کنید . حالا می بینید که با کلیک بر روی دکمه ی Button متن تغییر می کند که یعنی دکمه یا کلید به درستی به گره ی مورد نظر وصل شده و عملکرد مورد نظر را دارد .
حالا می خواهیم روش دوم و بهتر را به شما نشان بدهیم . برای اینکار ابتدا به صحنه یک دکمه ی دیگر به نام Button 2 اضافه کنید .
حالا به اسکریپت گره ریشه بیاید و گد زیر را به متد ready() اضافه کنید.

```
self.get_node("Button 2").connect("pressed",self,"_button_pressed")
```

توضیح کد بالا :

در اینجا ابتدا گره ی مورد نظر را انتخاب کردیم ، حالا با استفاده از متد connect آن را به گره ی دلخواه ما وصل می کنیم . اولین آرگومان مشخص کننده ی رویداد و یا به عبارت بهتر و درست تر سیگنالی است که به این گره وصل می شود . در اینجا ما سیگنال pressed را مشخص کردیم . در آرگومان دوم این متد ، باید شی هدف را مشخص کنیم ، چون می خواهیم گد مورد نظر را بنویسیم را در همین اسکریپت بنویسیم ، از کلمه ی کلیدی self استفاده کردیم که نشان دهنده ی شی فعلی است . در آرگومان سوم هم نام متد هدفی که قرار است به رویداد و یا سیگنال فشار داده شدن این دکمه واکنش نشان دهد را نوشتیم . توجه کنید که این متد هنوز ساخته نشده برای همین در انتهای این اسکریپت این گد را می نویسیم .

```
func _button_pressed():  
    self.get_node("Label").set_text("This is Button 2!")
```

در انتها اسکریپت ما به این صورت می شود .

```
01 extends Node2D  
02  
03 func _ready():  
04 > self.get_node("Button 2").connect("pressed",self,"_button_pressed")  
05 > pass  
06  
07  
08 func _on_Button_pressed():  
09 > self.get_node("Label").set_text("Text was changed !")  
10  
11 func _button_pressed():  
12 > self.get_node("Label").set_text("This is Button 2!")
```

حالا بازی را اجرا کنید و نتیجه را ببینید .

توجه داشته باشید که متد Connect می تواند آرگومان های دیگری را بپذیرد که فعلاً با آنها صحبت نمی کنیم ، برای کسب اطلاعات بیشتر در این مورد می توانید به اسناد Godot مراجعه کنید.

نویسنده : رضا پویا

تاریخ : ۸ مرداد ۱۳۹۵ هجری شمسی