

Documentazione Godot Engine

Cos'è Godot

Godot è un game engine 2D e 3D general-purpose pensato per supportare tutti i tipi di progetti immaginabili. Può essere utilizzato per sviluppare giochi o applicazioni che possono essere rilasciate su desktop, mobile oppure in versione web.

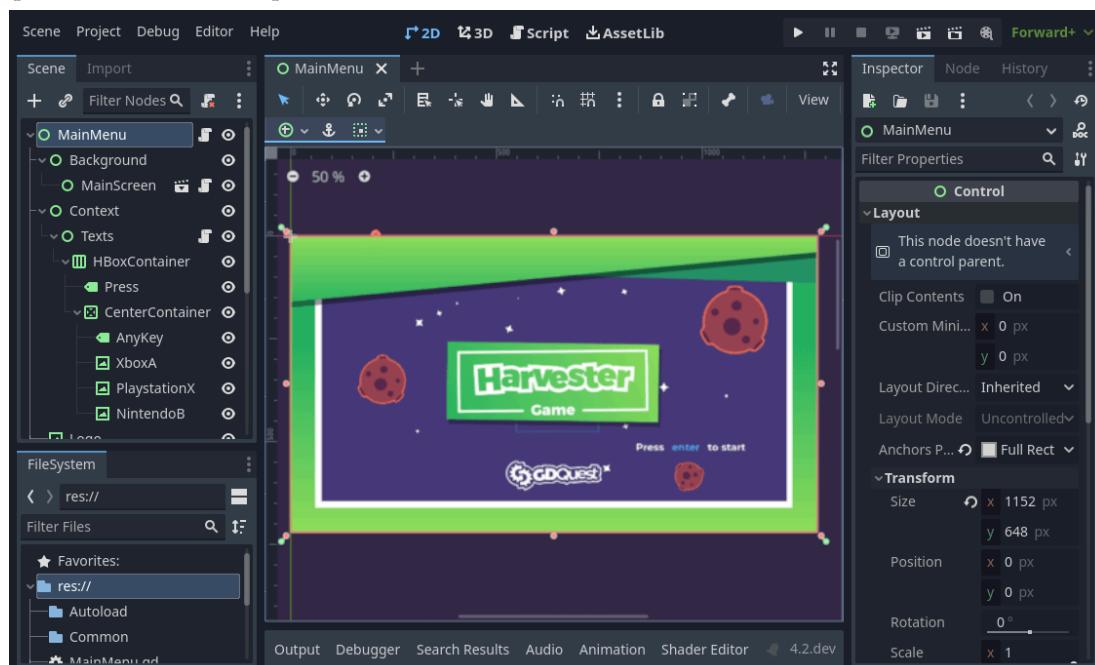
Godot è accompagnato da un game editor integrato con i vari tool necessari: editor di codice, animazioni, luci, shader, un debugger, un profiler e così via. Inoltre è disponibile la possibilità di integrare tool esterni, ad esempio è supportata l'importazione di scene 3D da blender.

Elementi principali di Godot

Come ogni game engine, Godot gira attorno a delle astrazioni. Un'applicazione è composta da un [albero di nodi](#) che vengono raggruppati all'interno di [scene](#). Successivamente tramite dei [segnali](#) è possibile far comunicare tra loro questi nodi.

Scene

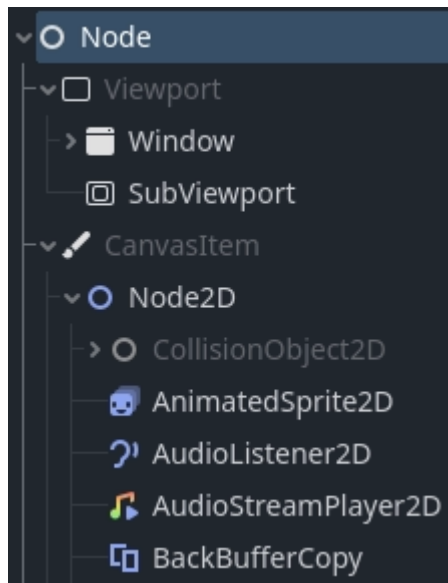
Una **scena** è una collezione di nodi riutilizzabile all'interno della nostra applicazione. Una scena potrebbe essere un personaggio, un menu, un ambiente, un intero livello o molto altro. Le scene in Godot sono quindi molto flessibili e possono anche essere innestate tra di loro.



Nodi

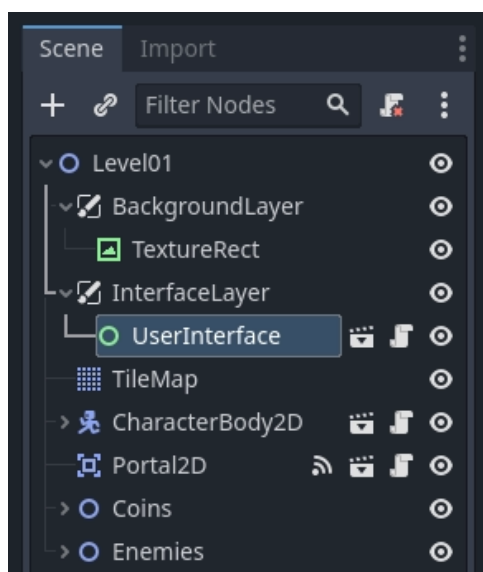
Un nodo è l'elemento più piccolo, un atomo, di Godot che possono essere organizzati in alberi. Ogni nodo ha un proprio tipo e una propria funzionalità ed è possibile 'attaccare' ad ogni nodo del codice per gestire il

suo comportamento. Quando si salva un albero di nodi come una scena, quest'ultima diventa un nodo singolo e la sua struttura interna rimane nascosta nell'editor.



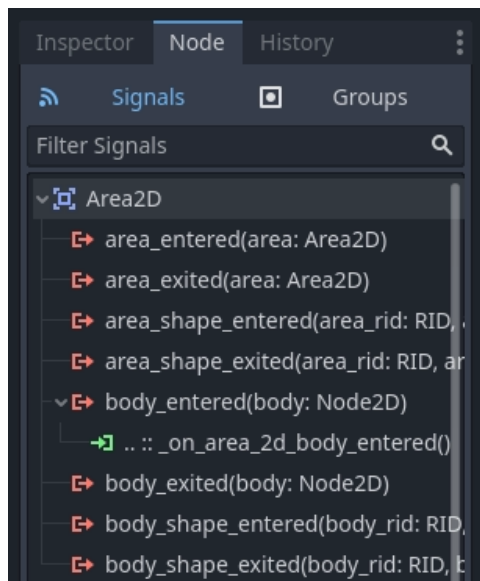
Albero delle scene

Tutte le scene dell'applicazione vengono visualizzate nell'albero delle scene, il quale è a sua volta un albero di nodi.



Segnali

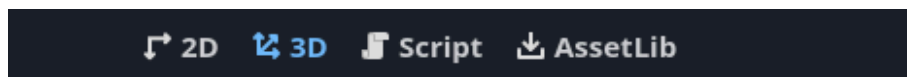
I nodi possono emettere dei segnali specifici al compimento di specifiche azioni. Questa funzionalità permette ai nodi di comunicare tra di loro senza renderli hard-wired. Ad ogni segnale può essere connesso del codice che viene eseguito nel momento in cui il segnale relativo è emesso. Esempi di segnali built-in sono quello per gestire le collisioni tra due corpi oppure quello per verificare quando un corpo entra in una determinata area.



Interfaccia Grafica

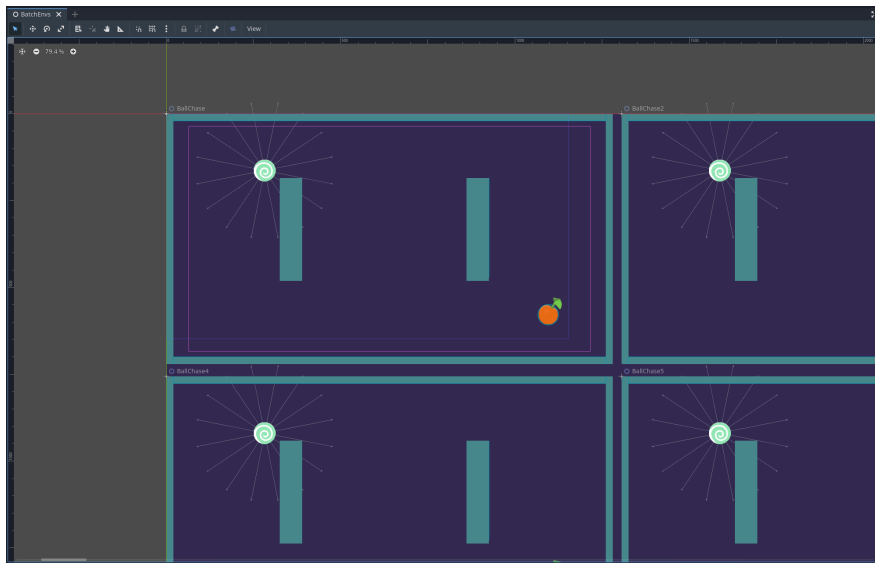
Interfaccia Principale

L'interfaccia principale di Godot è suddivisa in diverse modalità che possono essere selezionate tramite i tab situati nella parte superiore dell'interfaccia.



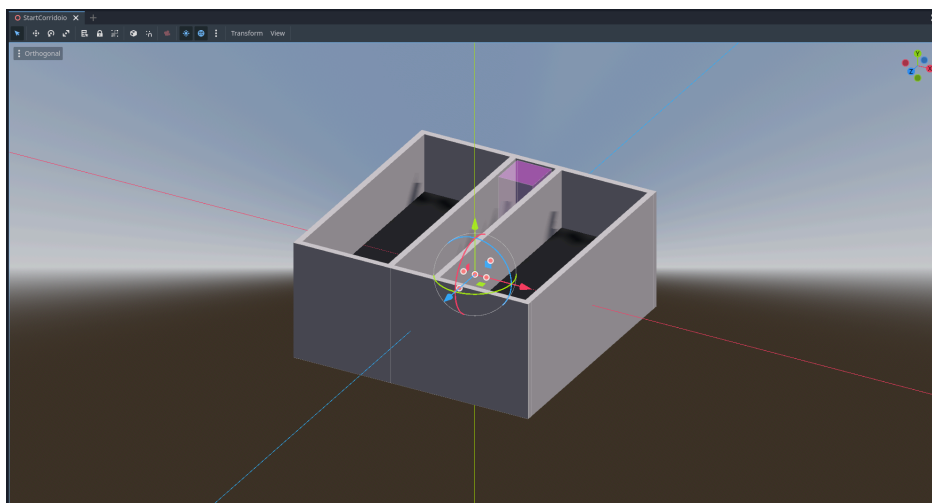
2D

Questa modalità è ottimizzata per lo sviluppo di giochi 2D. Mostra una vista ortografica del progetto su cui si può lavorare direttamente con sprite, animazioni e altri asset 2D.



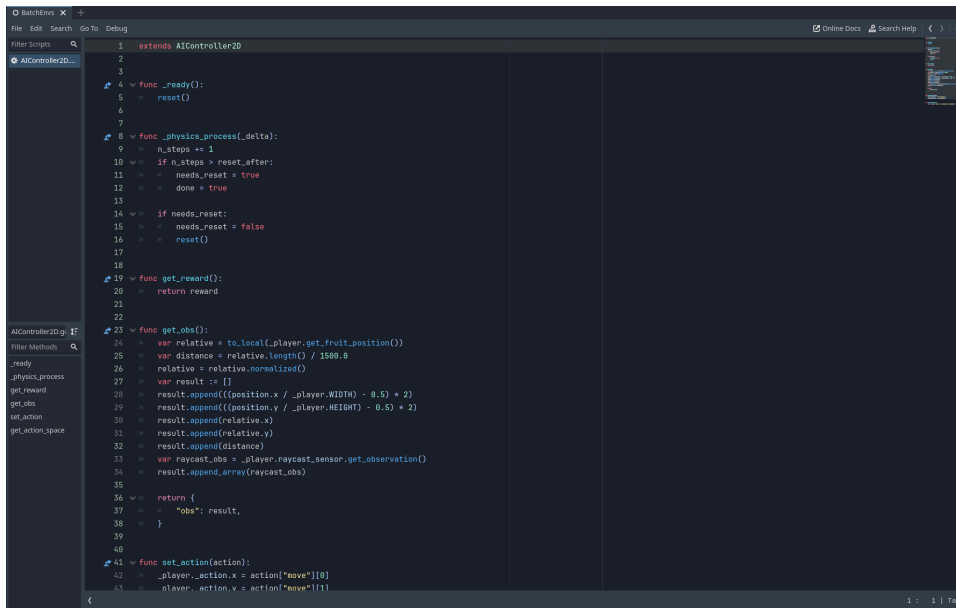
3D

Fornisce gli strumenti e le viste necessarie per lavorare con scene 3D. Qui puoi posizionare oggetti tridimensionali, gestire luci, camere e altre proprietà specifiche dei giochi 3D.



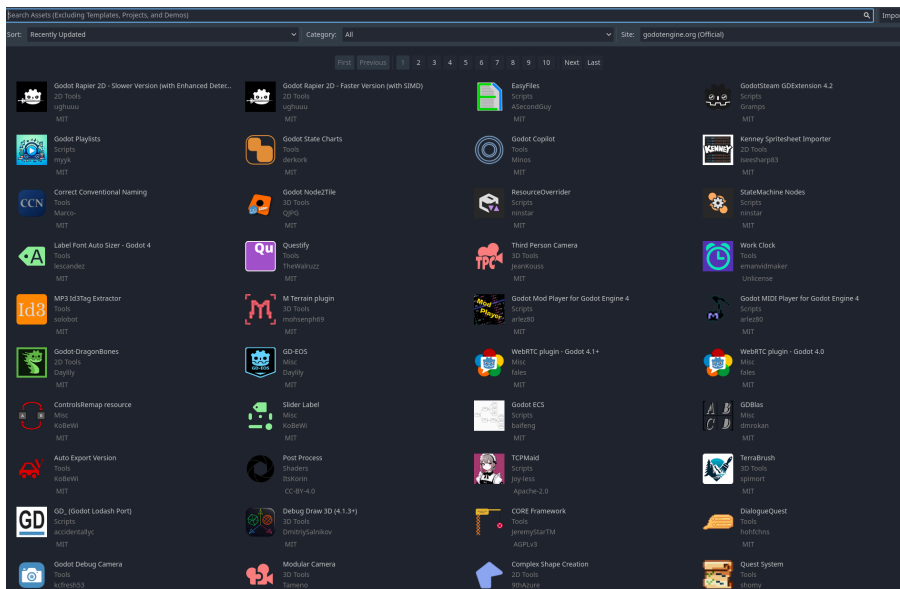
Script

Questo ambiente è dedicato alla scrittura e alla modifica di codice. Supporta vari linguaggi come GDScript, VisualScript, e C#. Offre funzionalità come evidenziazione della sintassi, autocompletamento e un debugger integrato.

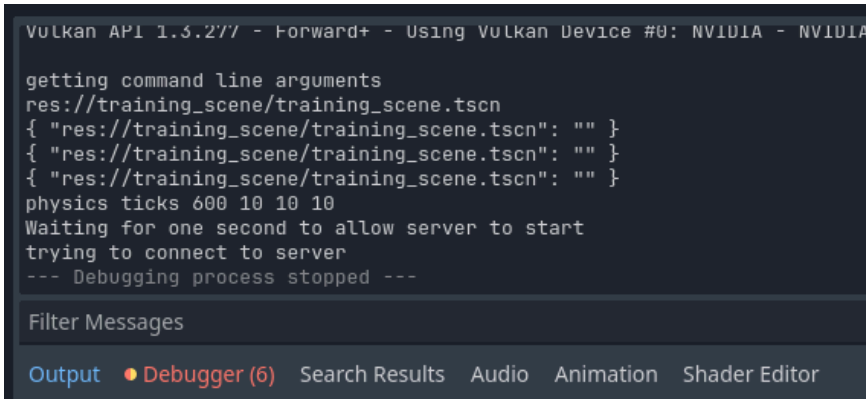


AssetLib

Questo tab permette agli utenti di accedere a una libreria di asset che possono essere scaricati e integrati nei loro progetti. Questi includono script, modelli, texture e altro.



Altre Funzioni e Strumenti



```
Vulkan API 1.3.277 - Forward+ - Using Vulkan Device #0: NVIDIA - NVIDIA  
getting command line arguments  
res://training_scene/training_scene.tscn  
{ "res://training_scene/training_scene.tscn": "" }  
{ "res://training_scene/training_scene.tscn": "" }  
{ "res://training_scene/training_scene.tscn": "" }  
physics ticks 600 10 10 10  
Waiting for one second to allow server to start  
trying to connect to server  
--- Debugging process stopped ---
```

Filter Messages

Output ● Debugger (6) Search Results Audio Animation Shader Editor

Output

Mostra output di stampa dal codice, informazioni di logging durante l'esecuzione del gioco e altri messaggi di sistema.

Debugger

Ti aiuta a trovare e correggere gli errori nel tuo gioco. Include funzionalità come breakpoints, step execution e visualizzazione dello stack di chiamate.

Search Results

Dopo aver eseguito una ricerca nell'ambito del progetto, i risultati vengono visualizzati qui.

Audio

Permette di gestire le impostazioni audio del progetto come i mix e i bus.

Animation

Permette di creare e modificare animazioni per vari oggetti e proprietà nel tuo gioco.

Shader Editor

Permette di scrivere e modificare shader direttamente all'interno dell'editor.

Pannelli di Lavoro

Scene

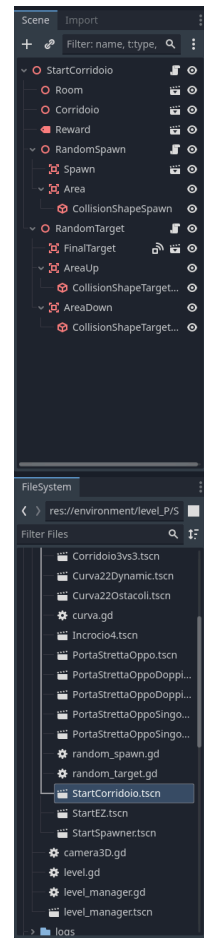
È il cuore della creazione di livelli e gameplay in Godot. Qui puoi organizzare la gerarchia dei nodi che compongono la tua scena. Ogni nodo può avere diversi componenti e child nodes.

Import

Dopo aver selezionato un asset nel FileSystem, questo pannello ti permette di modificare le impostazioni di importazione per quell'asset specifico, come la qualità delle texture o altre opzioni di importazione per i modelli 3D.

FileSystem

È l'esploratore di file del progetto dove tutti i file e le cartelle del progetto sono organizzati e accessibili. Puoi trascinare e rilasciare risorse direttamente nelle tue scene o script.



Inspector

Qui puoi vedere e modificare le proprietà dei nodi selezionati nella scena. Per ogni nodo, questo pannello mostra tutte le proprietà che possono essere modificate, come trasformazioni, materiali, script e variabili esposte.

Node

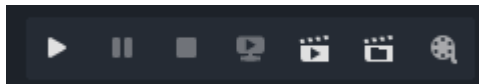
Fornisce un elenco di segnali e gruppi associati al nodo selezionato. Questo è utile per gestire la comunicazione tra nodi e l'appartenenza a gruppi specifici.

History

Registra le modifiche recenti e permette di navigare tra di esse, simile alla funzione "undo/redo" di molti altri programmi.



Controlli di Esecuzione



Situati nella parte superiore destra dell'interfaccia, questi controlli includono:

Run Project

Avvia l'esecuzione dell'intero progetto. Questo è il modo standard per testare il gioco come se fosse in esecuzione in un ambiente di produzione.

Pause Project

Mette temporaneamente in pausa l'esecuzione del progetto. È utile per debuggare specifici momenti o comportamenti all'interno del gioco senza fermare completamente l'esecuzione.

Stop Running Project

Ferma l'esecuzione del progetto che è attualmente in corso. Questo comando riporta l'utente all'editor.

Remote Debug

Consente di connettersi a un'istanza remota del gioco per il debugging. Questo è particolarmente utile per testare il gioco su piattaforme o dispositivi differenti da quelli su cui è in esecuzione l'editor.

Run Current Scene

Avvia l'esecuzione della scena attualmente aperta nell'editor. È un modo rapido per testare una singola scena senza dover eseguire l'intero progetto.

Run Specific Scene

Permette di selezionare e avviare l'esecuzione di una scena specifica senza dover navigare fino a essa nell'editor.

Enable Movie Maker Mode

Questo è un comando speciale che facilita la registrazione di sequenze di gameplay o cinematiche del gioco, offrendo controlli ottimizzati per la cattura di video.