Freeloader Documentation

v1.0.0

November 10, 2024

Stylish Esper

Table of Contents

Online Documentation

Freeloader's documentation may be updated occasionally. To find the latest version of the documentation, use this link. The online documentation is recommended in general for readability.

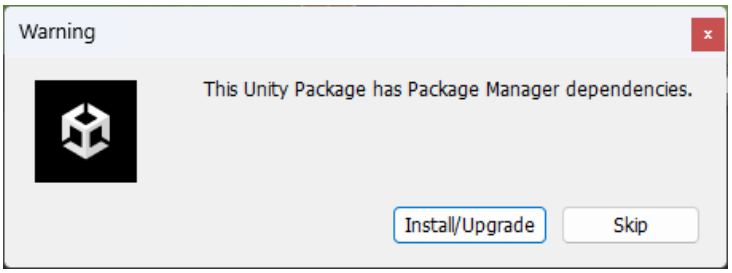
Installation

Asset Store

You can find the latest version of Freeloader in the asset store with this link.

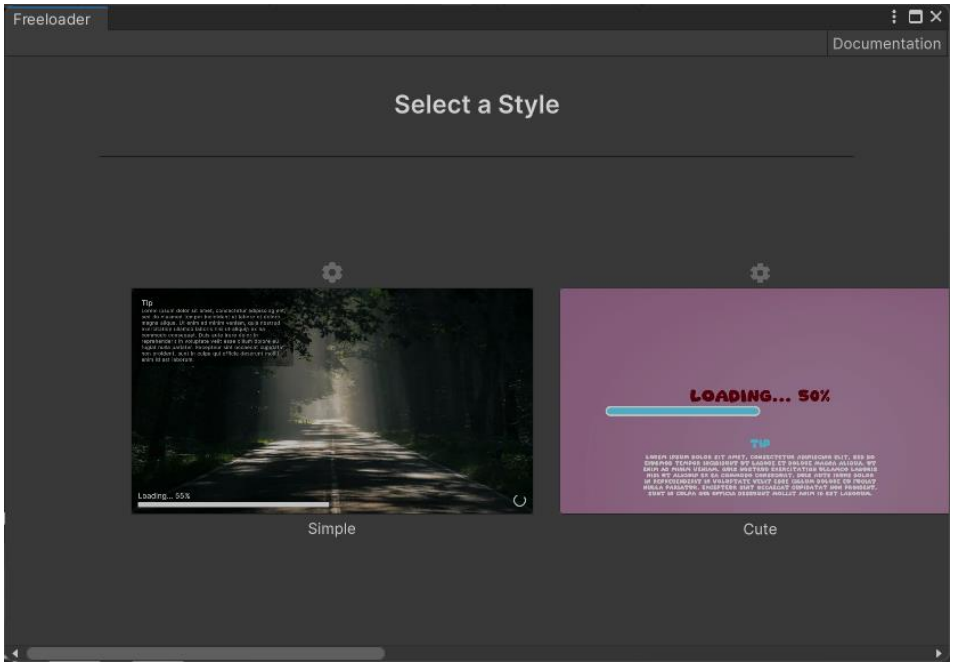
Installation Steps:

1. Get Freeloader from the asset store.
2. Open your Unity project, go into **Window > Package Manager**, switch to **My Assets** from the packages dropdown at the top-left, and then type **Freeloader** in the search bar.
3. Download and import the package. You may be prompted to install package manager dependencies. If so, click **Install/Upgrade**.



Add a Loading Screen

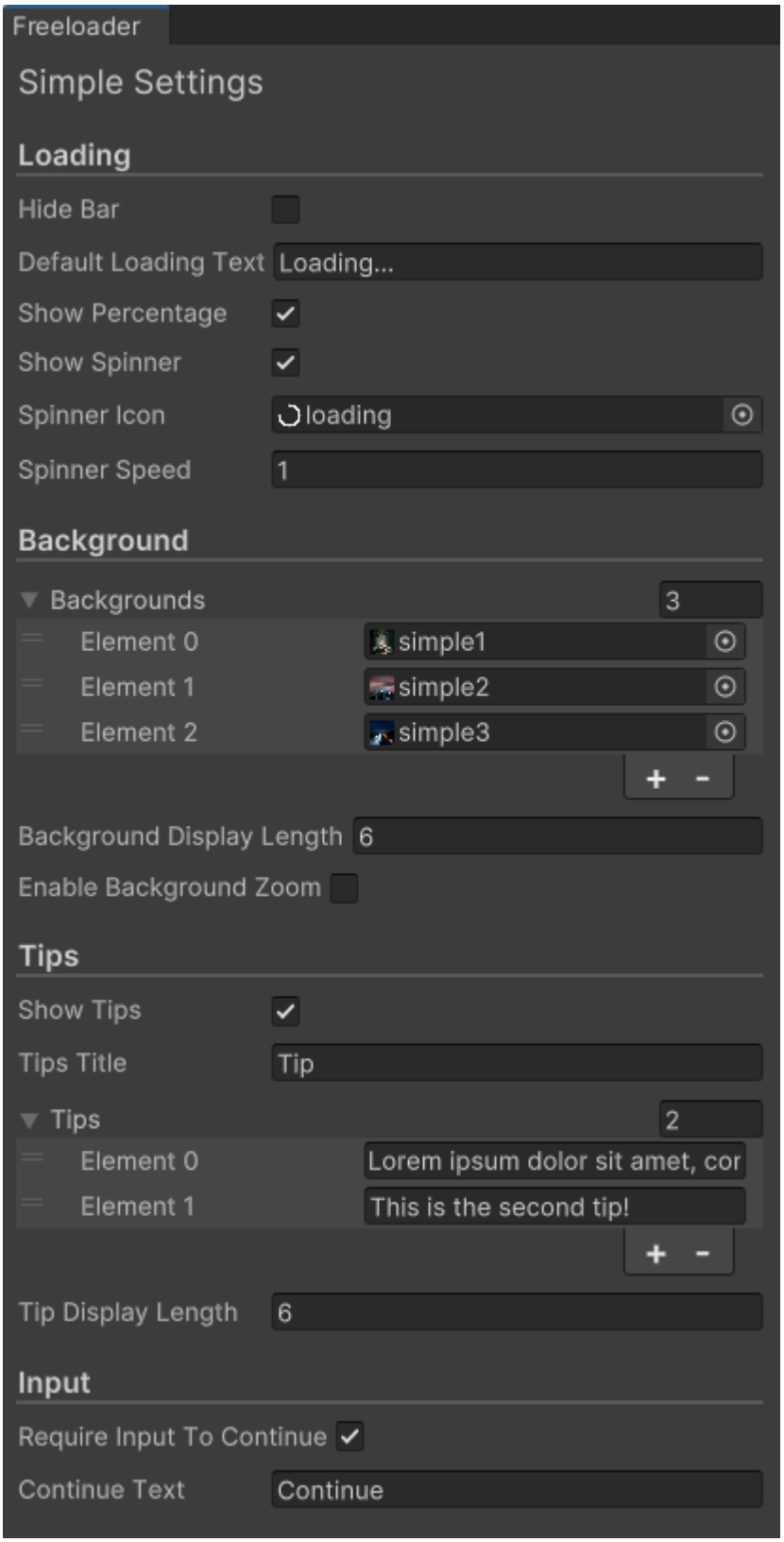
To add a loading screen, open the Freeloader window from Window > Freeloader.



Simply click on one of the styles and it will be added to your scene. Ensure there is only one instance of a loading screen.

Customization Settings

Above each style option is a settings icon. Clicking it will open the customization settings. You can adjust these settings as needed.


Freeloader
Simple Settings
Loading
Hide Bar
Default Loading Text
Loading...
Show Percentage
Show Spinner
Spinner Icon
loading
Spinner Speed
1
Background
Backgrounds
3
Element 0
simple1
Element 1
simple2
Element 2
simple3
Background Display Length
6
Enable Background Zoom
Tips
Show Tips
Tips Title
Tip
Tips
2
Element 0
Lorem ipsum dolor sit amet, cor
Element 1
This is the second tip!
Tip Display Length
6
Input
Require Input To Continue
Continue Text
Continue

More Customizations

Any more customizations outside of the settings would have to be done on your own either through code or through the UI Builder window. All loading screens are made with the UI Builder. You can open the loading screens in the UI Builder by double clicking their .uxml file. You can find these in Assets/StylishEsper/Freeloader/LoadingScreens in their respective folders.

Loading

Freeloader is a simple asset meant for loading scenes. You can use Freeloader to load scenes as long as you have an instance of a loading screen in your initial scene.

Loading a Scene

Use the Load method to load a scene.

Load By Scene Name

```
string sceneName = "scene name";
LoadingScreen.Instance.Load(sceneName)
```

Load By Build Index

```
int buildIndex = 1;
LoadingScreen.Instance.Load(buildIndex);
```

Tracking Progress

The progress of the loading the scene itself is tracked automatically. To track anything else that your game will need to load during the loading screen will have to be added manually. You can use the LoadingProgressTracker struct for this purpose.

Adding a Progress Tracker

Before you begin to load a scene, you can add multiple progress trackers.

The LoadingScreen.Instance.Load methods accepts an array of LoadingProgressTracker's as it's second parameter.

Step 1: Create a Float

Create a float field that will be updated as whatever it is you need to load progresses. Ensure that the reference to this field isn't lost due to a scene being unloaded.

```
private static float progress;
```

Step 2: Creating the Tracker

Create a LoadingProgressTracker that uses a getter so that it can get the value of the progress field as necessary.

The first parameter is the text displayed in the loading screen. The second parameter is the progress getter.

```
var process = new LoadingProgressTracker("Please wait...", () =>
progress);
```

Step 3: Adding the Tracker

This tracker can be added with the loading screen's Load method. You can't add a tracker while loading is ongoing, only before.

```
// Where sceneName is a string that contains the name of the scene you
want to load
LoadingScreen.Instance.Load(sceneName, process);
```