

Easy Expandable Text Box


This text asset is great! \(^o^)/

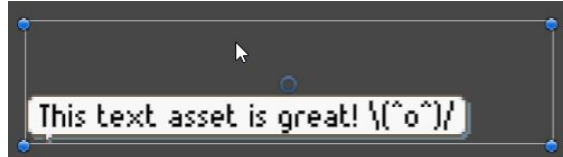
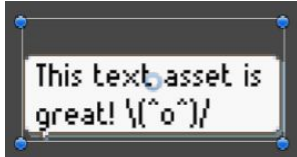
Table of Contents

Setup	2
Script Variables	2
Change the text via script	4
Create My Own Font	4

Setup

Go inside the asset folder (Plugins/Easy Expandable Text Box), then to the Prefabs folder and drag and drop the prefab to the hierarchy, inside a Canvas (if the prefab is invisible, then you haven't dropped the prefab inside a Canvas).

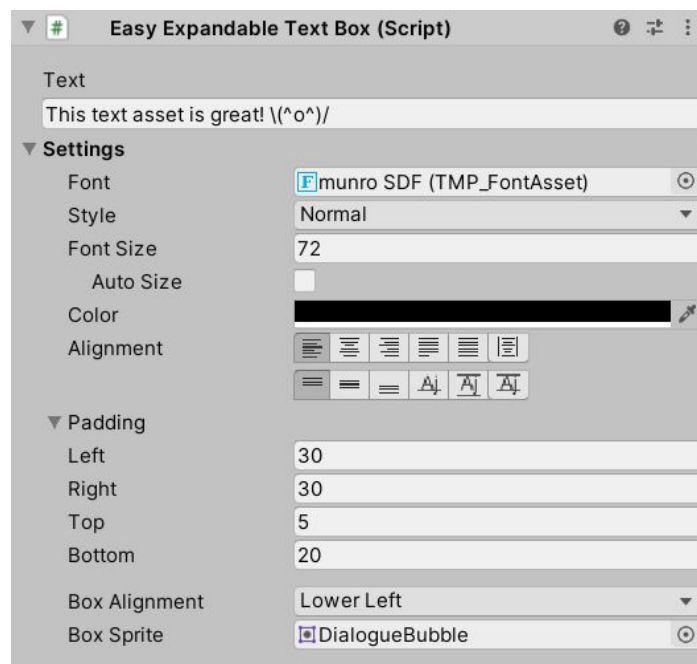
Once done, modify the max size of the box with the Rect Tool  (above the hierarchy).



Be aware that this asset **needs** TextMeshPro to function, so if your project doesn't have it imported, it will appear a window asking to do so. Just click "Import TMP" and the package will be imported.

And you're done! Now you can modify all the variables to adjust the asset to your project!

Script Variables



The variables are the same as the TextMeshPro asset provides, plus some added features:

Text: String variable, modifies what the Text Box displays.

Font: TMP_FontAsset variable, changes the font of the rendered text. To create a font for TextMeshPro go to the “Create my own font” section.

Style: FontStyle variable, modifies the way the text is rendered (Bold, Italics, etc).

Font Size: Int variable, changes the size of the text.

Auto Size: Bool variable, when activated the font scales based on the size of the container, disables the “Font Size” variable, as it is useless.


Min: Int variable, it is displayed when “Auto Size” is true, it is the minimum size the text will be rendered, independently of the container size.

Max: Int variable, it is displayed when “Auto Size” is true, it is the maximum size the text will be rendered, independently of the container size.

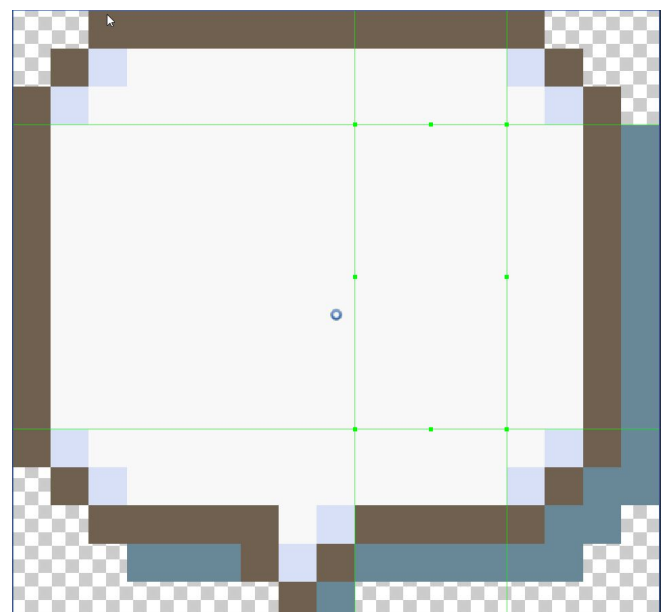
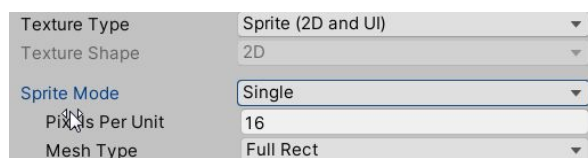
Color: Color variable, changes the color of the text.

Alignment: TextAlignmentOptions variable, changes the alignment of the text.

Padding: Int variables, each one adds padding to the text inside the box, adjust it if your text shows out of it.

Box Alignment: TextAnchor variable, changes the alignment of the box relative to the max box size (Set with the Rect Tool ).

Box Sprite: Sprite variable, defines the sprite that will act as the box container. In most cases, it needs to be set to Full Rect mode, and changed accordingly in the Sprite Editor (adjust the expandable section with the green bars as shown in the image, everything outside this section will not be modified when increasing the box size).

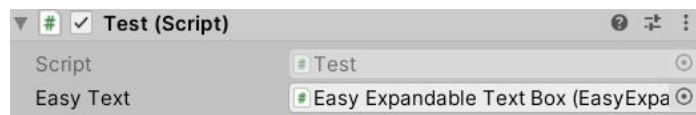


Change the text via script

To modify the text, first you have to add it in your script as a `EasyExpandableText` variable:

```
public EasyExpandableText easyText;
```

Then, assign the variable in the inspector:



And finally call the `EasyMessage(string message, float timeBetweenCharacters)` function:

```
easyText.EasyMessage("Example Text", 0.25f);  
easyText.EasyMessage("Example Text"); (Example without time)
```

By default, the text is rendered one character at a time, being `timeBetweenCharacters` (the second argument) the time in seconds that a character takes to show up, if the second argument is not written, then it will use the default value of 0.125.

In case you just want the text to be instant, set this value to 0.

This function can also be called with a coroutine, allowing another coroutine to wait until the text has displayed:

```
yield return new WaitForSeconds(easyText.EasyMessage("I'm waiting until my text is  
rendered!"));
```

Create your own font

TextMeshPro only accepts its own format for the fonts, but they're very easy to create.

First, go to the "Window" tab (At the top of the screen) and select "Text Mesh Pro", then, "Font Asset Creator".

Put your font inside the "Source Font File", and click the "Generate Font Atlas" button.

Then, the "Save" button, and save it wherever you want.

The saved file is the `TMP_FontAsset` that you need!