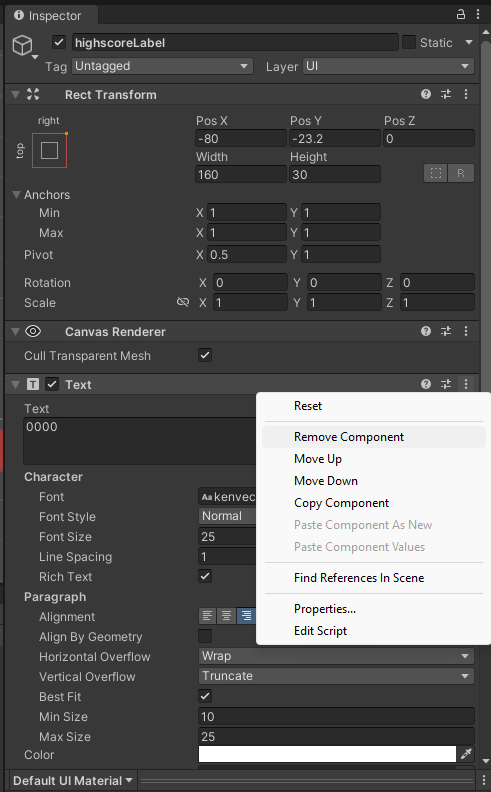
Updating to TMPro

The Text method while still in Unity, I assume for backwards compatibility, does not work as of 2019. The “Text” method is depreciated which is a fancy way of saying it doesn’t work with the newer versions and it is not a bug. So if you are using a newer version on Unity (Like me) and not 5.x like the author you cannot use his code here. I believe the Text method depreciated in 2019 and this book was published in 2016.

That being said, here's how you can get it to work anyways in a slightly different way than the book.

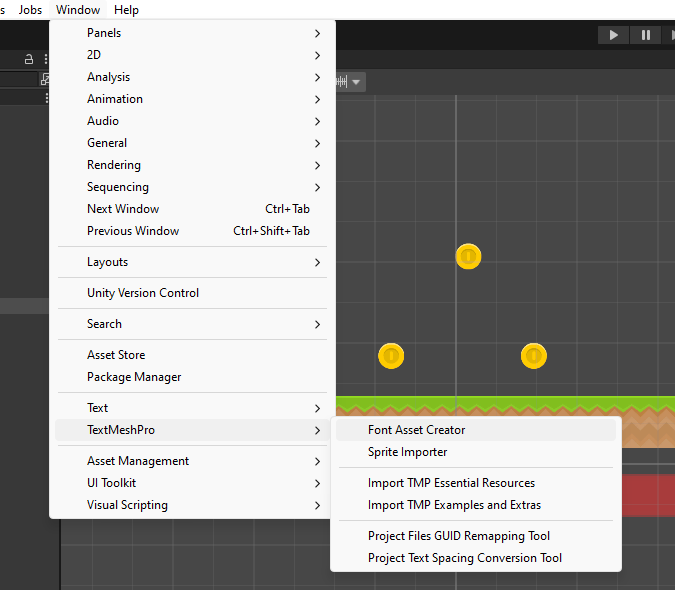
Remove the Text component from your Label, while the components work in the Unity engine the code you write to reference them will NOT compile and not work, unless you know something I do not.



Now, Add Component, TextMeshPro - Text (UI)

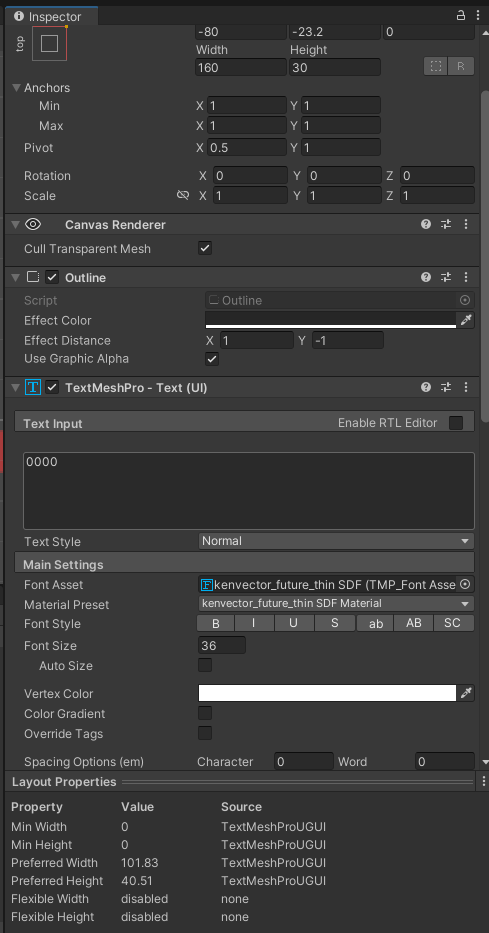
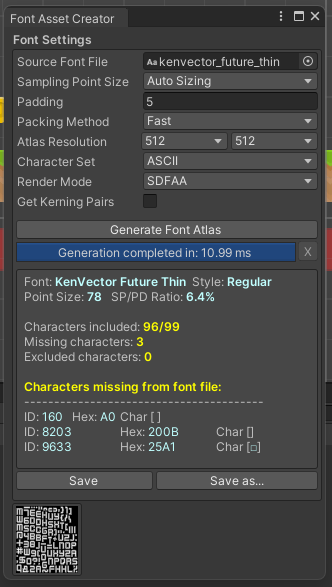
If you are like me you will want all your fonts to match so go to

Window->TextMeshPro -> Font Asset Creator



Drop the kenvector\_future\_thin into the Source font file. Click “Generate Font Atlas”, then save.

You can now drag this into the “Font Asset” box to have matching fonts with the rest of the GUI. Odds are you will also need to reposition the location of both <scoreLabel> and <highscoreLabel> to get them to line up on the GUI.



You will need to do this for the score, highscore, and coin counter.

Now for the coding part.

Firstly you will need the additional libraries in your file so the compiler can find the associated methods. (The complete example text is at the end).

Include:

using TMPro;

using System;

Then we will have to declare our variable differently.

We declare our variables like this…

public TextMeshProUGUI coinLabel;

Under void Start()

TextMeshProUGUI[] allLabels = GetComponents<TextMeshProUGUI>();

Under void Update()

if (GameManager.instance.currentGameState == GameState.inGame) {

coinLabel.SetText(GameManager.instance.collectedCoins.ToString());

scoreLabel.SetText((PlayerController.instance.GetDistance().ToString("F1"));

highscoreLabel.SetText(PlayerPrefs.GetFloat("highscore", 0).ToString("F0")); }

If you want the bigger breakdown you will need to look into the TextMeshProUGUI methods and can explore other options as well, this is merely the first working option I was able to find. Good luck!

