# Stylize C# Script

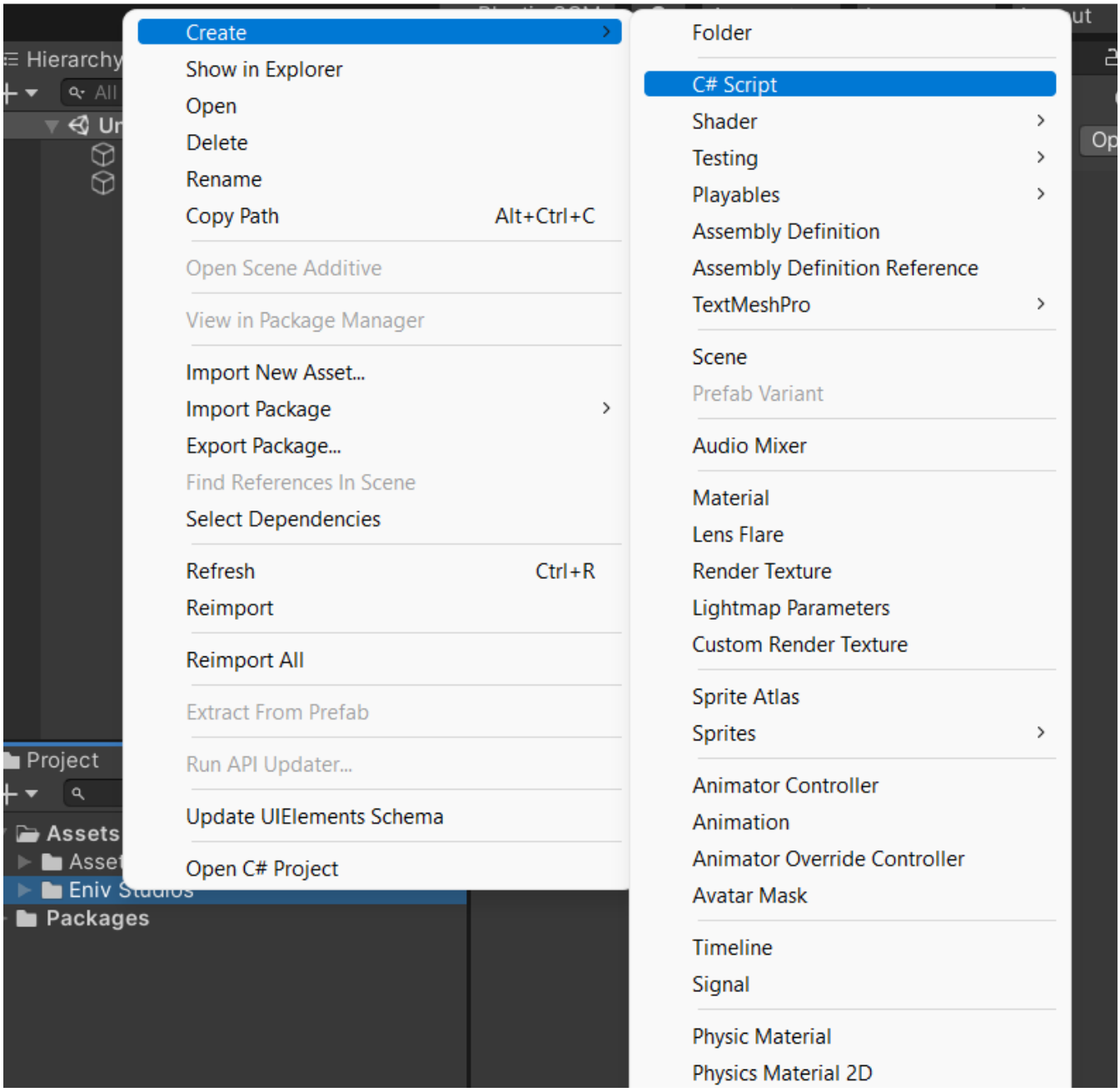
## Stylize C# Script

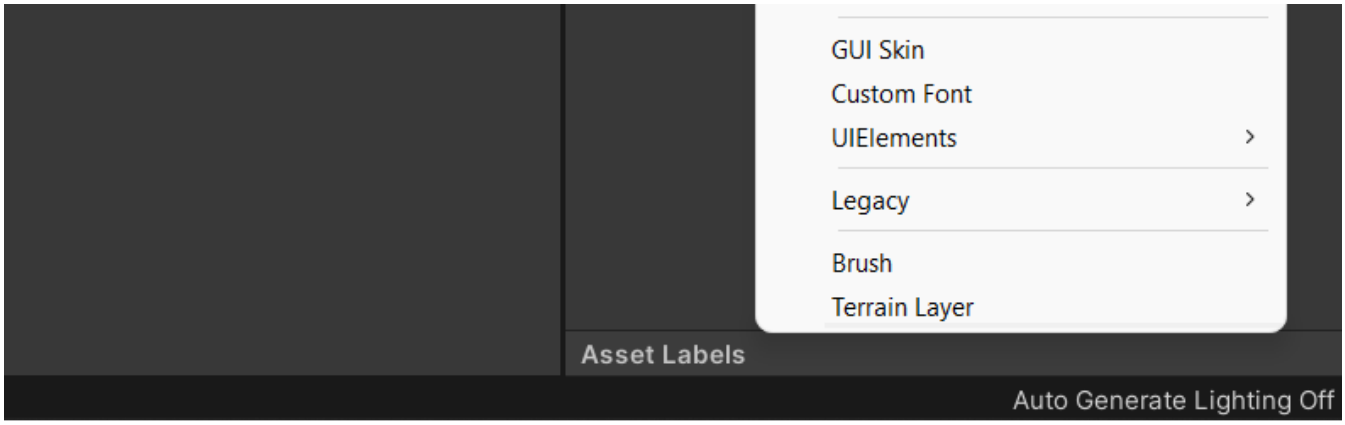
> ⓘ This is a free asset that can help you to write and stylize your C# script in unity Inspector.

---

## How To Use It?

1. Create a new **C# script** with any name :-






2. Write **using UnityEditor** and **namespace EnivStudios** (or your own custom namespace) :-

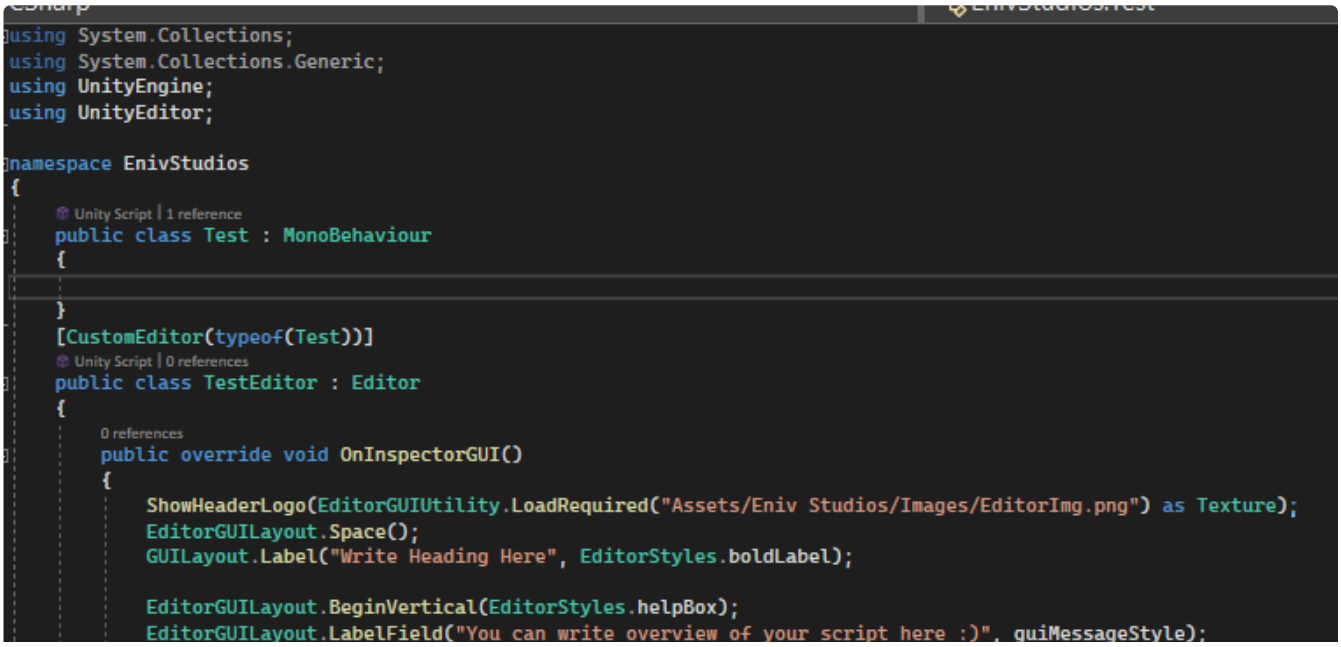
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEditor;

namespace EnivStudios
{
    Unity Script | 0 references
    public class Test : MonoBehaviour
    {

    }
}
```

3. Now just below where your **class** curly brackets ends paste this code :-

```
[CustomEditor(typeof(Test))]
  public class TestEditor : Editor
  {
      public override void OnInspectorGUI()
      {
          ShowHeaderLogo(EditorGUIUtility.LoadRequired("Assets/Eniv Studios/Images/Edito
          EditorGUILayout.Space();
          GUILayout.Label("Write Heading Here", EditorStyles.boldLabel);

          EditorGUILayout.BeginVertical(EditorStyles.helpBox);
          EditorGUILayout.LabelField("You can write overview of your script here :)", gu
```

```
            EditorGUILayout.EndVertical();
            EditorGUILayout.Space();

            EditorGUILayout.BeginVertical(EditorStyles.helpBox);

            DrawDefaultInspector();

            EditorGUILayout.EndVertical();
            EditorGUILayout.Space();

        }
        void ShowHeaderLogo(Texture tex)
        {
            var rect = GUILayoutUtility.GetRect(0f, 0f);
            rect.width = tex.width;
            rect.height = tex.height;
            GUILayout.Space(rect.height);
            GUI.DrawTexture(rect, tex);

            var e = Event.current;
            if (e.type != EventType.MouseUp) { return; }
            if (!rect.Contains(e.mousePosition)) { return; }
        }
        GUIStyle guiMessageStyle
        {
            get
            {
                var messageStyle = new GUIStyle(GUI.skin.label);
                messageStyle.wordWrap = true;
                return messageStyle;
            }
        }
    }
```

**Just Like This :-**



```
            EditorGUILayout.EndVertical();
            EditorGUILayout.Space();

            EditorGUILayout.BeginVertical(EditorStyles.helpBox);

            DrawDefaultInspector();

            EditorGUILayout.EndVertical();
            EditorGUILayout.Space();

        }
        1 reference
        void ShowHeaderLogo(Texture tex)
        {
            var rect = GUILayoutUtility.GetRect(0f, 0f);
            rect.width = tex.width;
            rect.height = tex.height;
            GUILayout.Space(rect.height);
            GUI.DrawTexture(rect, tex);

            var e = Event.current;
            if (e.type != EventType.MouseUp) { return; }
            if (!rect.Contains(e.mousePosition)) { return; }
        }
        1 reference
        GUIStyle guiMessageStyle
        {
            get
            {
                var messageStyle = new GUIStyle(GUI.skin.label);
                messageStyle.wordWrap = true;
                return messageStyle;
            }
        }
    }
}
```

**Make sure of following things :-**

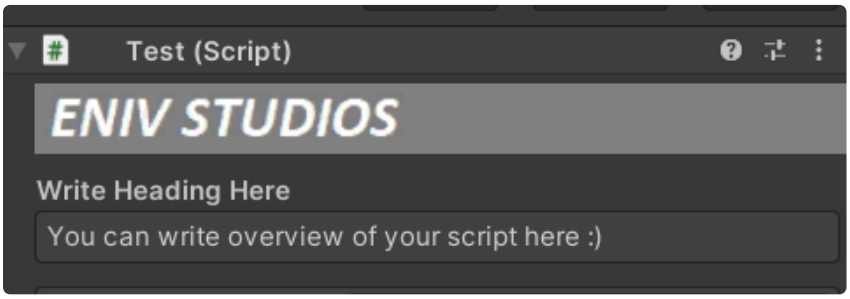
1st Line of code:- [CustomEditor(typeof(**your script name**))]

2nd Line of code :- public class **your script name**Editor : Editor

6th Line of code :- ShowHeaderLogo(EditorGUIUtility.LoadRequired("**Location of your header image**") as Texture);

8th Line of code : - GUILayout.Label("**Write Heading Here**", EditorStyles.boldLabel);

11th Line of code :- EditorGUILayout.LabelField("**You can write overview of your script here :)**", guiMessageStyle);




---

## How to use Eniv Inspector ?

First of all make sure that **namespace** in **Eniv Inspector script** and your all other **scripts** must be **same**.

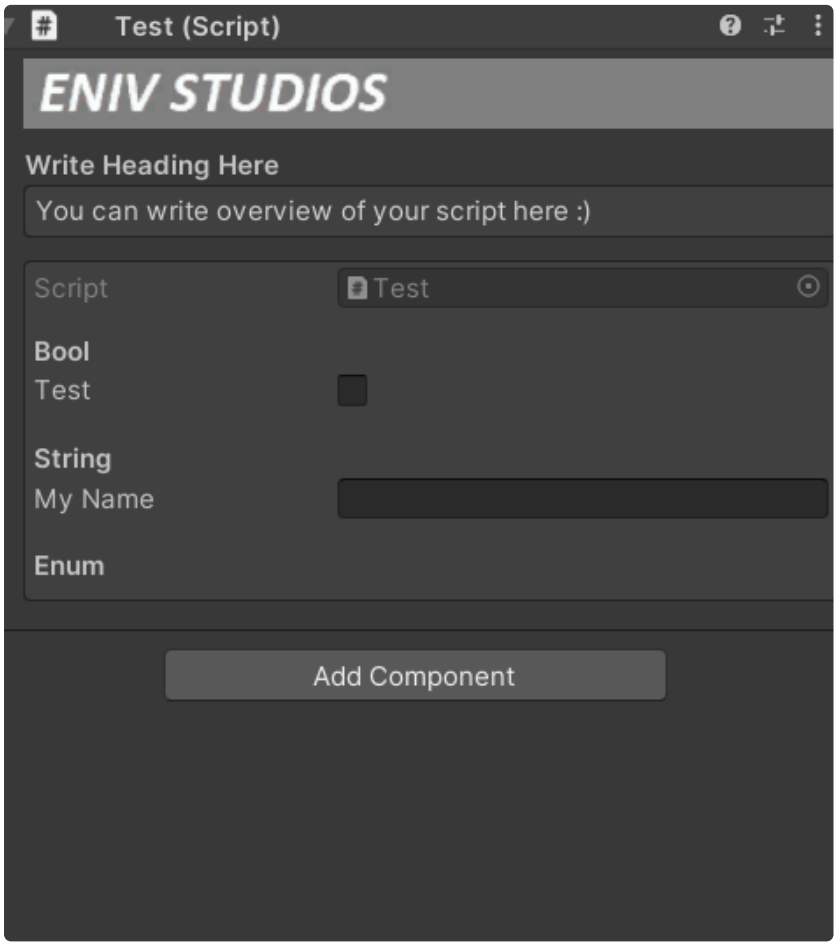
You can use your own namespace in eniv inspector script just change it in these two places.

```
using System;
using System.Linq;
using System.Collections.Generic;
using UnityEngine;
using System.Reflection;

namespace EnivStudios
{
    [AttributeUsage(AttributeTargets.Field)]
    24 references
    public class EnivInspector : PropertyAttribute
    {
        public readonly string FieldToCheck;
        public readonly string[] CompareValues;
        public readonly bool Inverse;
        20 references
        public EnivInspector(string fieldToCheck, bool inve
        {
            FieldToCheck = fieldToCheck;
            Inverse = inverse;
            CompareValues = compareValues.Select(c => c.ToS
        }
    }
}
#if UNITY_EDITOR
namespace EnivStudios
{
    using UnityEditor;

    [CustomPropertyDrawer(typeof(EnivInspector))]
    0 references
```

**You can use bools, strings and enums using Eniv Inspector like this :-**

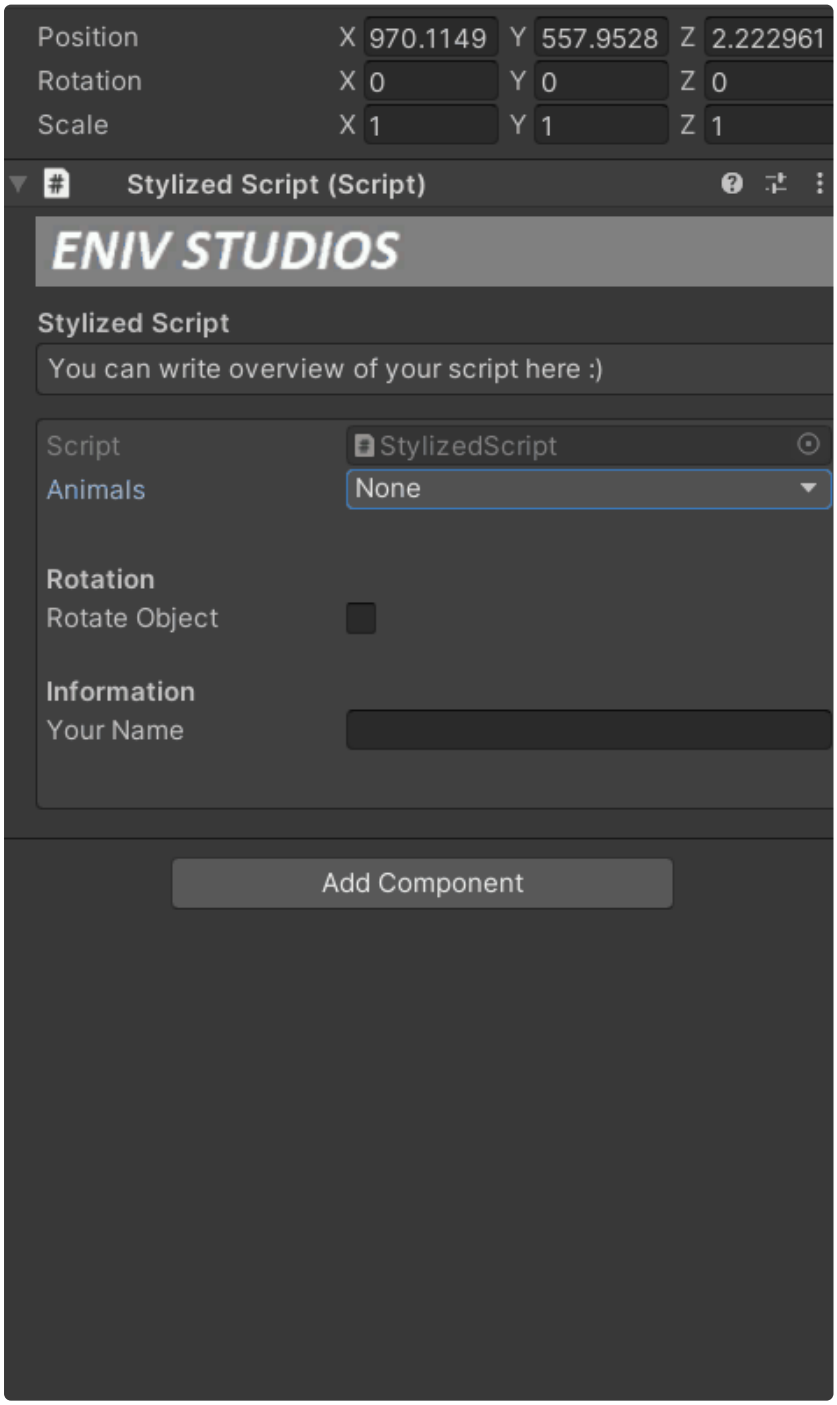
```
Unity Script | 1 reference
blic class Test : MonoBehaviour

  [Header("Bool")]
  [SerializeField] public bool test;
  [EnivInspector("test", false)][SerializeField] private int number = 2;

  [Header("String")]
  [SerializeField] public string myName = "Shubham";
  [EnivInspector(nameof(myName), false, "Shubham")][SerializeField][TextArea] string myDescription;

  3 references
  [SerializeField] enum states {None,Bat,Rat}
```

```
[Header("Enum")]
[EnivInspector(nameof(myName), false, "Shubham")] [SerializeField] states animals;
[EnivInspector(nameof(animals), false, states.Bat)] [SerializeField] [Range(1, 10)] private int batSpeed = 2;
[EnivInspector(nameof(animals), false, states.Rat)][SerializeField] string batDescription = "Gets eaten by cat";
```




## More Complex Form of using bools , enums and string using Eniv Inspector :-

```
references
SerializeField] private enum state { None,Cat, Dog, Rabbit}
SerializeField] private state animals;
EnivInspector(nameof(animals), false, state.Cat)][SerializeField] string catName = "Smokey";
EnivInspector(nameof(animals), false, state.Cat)][SerializeField] int catSpeed = 3;
EnivInspector(nameof(animals), false, state.Dog)][SerializeField] string dogName = "Rambo";
EnivInspector(nameof(animals), false, state.Dog)][SerializeField] int dogSpeed = 4;
EnivInspector(nameof(animals), false, state.Rabbit)][SerializeField] string rabbitName = "Maggi";
EnivInspector(nameof(animals), false, state.Rabbit)][SerializeField] int rabitSpeed = 5;

Header("Rotation")]
SerializeField] private bool rotateObject;
EnivInspector("rotateObject",false)] [SerializeField][Range(1, 10)] private int rotationSpeed;
EnivInspector("rotateObject", false)][SerializeField] private Vector2 rotationDirection;

Header("Information")]
SerializeField] private string yourName;
EnivInspector(nameof(yourName), false, "Shubham")][SerializeField] int shubhamAge = 19;
EnivInspector(nameof(yourName), false, "Shubham")][SerializeField] bool likesUnity;
EnivInspector("likesUnity", false)][SerializeField][TextArea] string description;


EnivInspector(nameof(yourName), false, "Akash")][SerializeField] int akashAge = 19;
EnivInspector(nameof(yourName), false, "Akash")][SerializeField] bool dontLikeUnity;
references
SerializeField] private enum anotherState { noHobby, Badminton, Hacking }
EnivInspector("dontLikeUnity", false)][SerializeField] private anotherState hobbies;
EnivInspector(nameof(hobbies), false, anotherState.Badminton)][SerializeField] int playingSince = 2020;
EnivInspector(nameof(hobbies), false, anotherState.Badminton)][SerializeField] string favouritePlayer = "Name Doesn't Matter";
EnivInspector(nameof(hobbies), false, anotherState.Badminton)][SerializeField][TextArea] string gameDescription;

EnivInspector(nameof(hobbies), false, anotherState.Hacking)][SerializeField] int startedAt = 2017;
EnivInspector(nameof(hobbies), false, anotherState.Hacking)][SerializeField] string favHackerType = "White Hat";
EnivInspector(nameof(hobbies), false, anotherState.Hacking)][SerializeField][Range(1, 10)] int totalExperience = 5;
```




## If you face any problem feel free to contact me :)

My Email Id :- 4d4467@gmail.com