

unity对话

步骤：

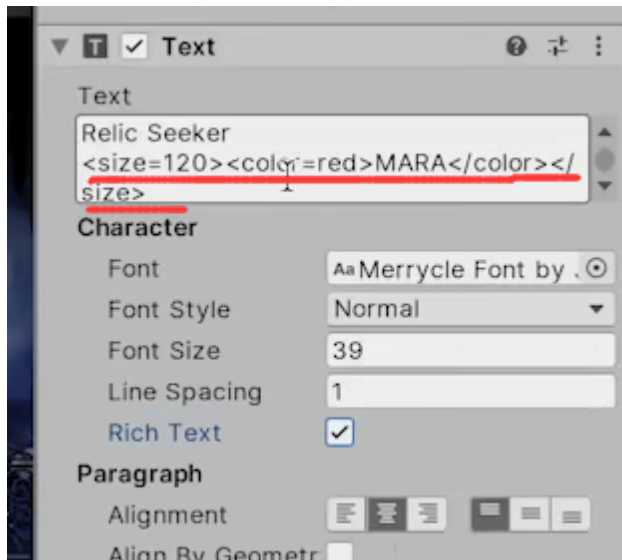
- 1, 制作对话UI, 获得文本文件,
 - 2, 创建方法, 把文本文件分割成一行一行, (分割关键字sprite), 然后把每一行的文字存放到一个列表中, (foreach),
 - 3, 显示文本, 把列表中的每一行文本在UI上显示出来。
- 细节, 逐字逐字的显示文本, (把每一行的文本当成一行字符, 每一行的字符单个单个显示),

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.UI;
5
6 public class Dialogle : MonoBehaviour
7 {
8     [Header("UI")]
9     public Text textLable;
10    public Image faceImage;
11    [Header("文本文件")]
12    public TextAsset textFile;
13    public int index;
14    public float textSpeed;//文本速度
15
16    public Sprite face1, face2;
17    private bool textFine;
18    List<string> textList = new List<string>();
19
20    private void Awake()
21    {
22        GetTextFromFile(textFile);
23    }
24    private void OnEnable()
25    {
26        textFine = true;
27        StartCoroutine(SetTextUI());
28        //textLable.text = textList[index];
29        //index++;
30    }
31    private void Update()
```

```
32 {
33     if (Input.GetKeyDown(KeyCode.R) && index == textList.Count)
34     {
35         gameObject.SetActive(false);
36         index = 0;
37         return;
38     }
39
40     if(Input.GetKeyDown(KeyCode.R)&&textFine)
41     {
42
43         //textLable.text = textList[index];
44         //index++;
45         StartCoroutine(SetTextUI());
46     }
47 }
48 void GetTextFromFile(TextAsset file)//读取文件
49 {
50     textList.Clear();
51     index = 0;
52     var dateList = file.text.Split('\n');//把文本切割成一行一行的
53     foreach (var line in dateList)
54     {
55         textList.Add(line);//把dateList的每一行加载到textList列表当中
56     }
57 }
58 IEnumerator SetTextUI()
59 {
60     textFine = false;
61     textLable.text = "";
62     switch (textList[index].Trim().ToString())
63     {
64         case "A":
65             facelImage.sprite = face1;
66             index++;
67             break;
68         case "B":
69             facelImage.sprite = face2;
70             index++;
71             break;
72     }
73     for (int i = 0; i < textList[index].Length; i++)
74     {
75         textLable.text += textList[index][i];
76
77         yield return new WaitForSeconds(textSpeed);
78     }
79     textFine = true;
```

```
80     index++;
81 }
```

```
}
```



文本小技巧

对话管理脚本

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.UI;
5
6 public class Dailong : MonoBehaviour
7 {
8     public static Dailong instance;
9     public GameObject dailong; // 获取对话框UI
10    public Text dialogueText, nameText; // 获取对话文本UI, 名字文本UI
11
12    [TextArea(1, 3)]
13    public string[] dialogueLine; // 获取对话内容
14    [SerializeField] private bool isScrolling;
15
16    [SerializeField] private int currentLine; // 对话行数
17    [SerializeField] private float textSpeed; // 文本滚动速度
18
19    private void Awake()
20    {
21        if(instance == null)
22        {
```

```
23     instance = this;
24 }
25 else
26 {
27     Destroy(gameObject);
28 }
29 }
30 private void Start()
31 {
32     FindObjectOfType<PlayerControl>().isMove = true;
33     dialogueText.text = dialogueLine[currentLine];
34 }
35 private void Update()
36 {
37     if (dailong.activeInHierarchy)
38     {
39         if (Input.GetMouseButtonUp(0))
40         {
41             if (isScrolling==false)//当文本不在滚动的时候，才可以显示下行文本
42             {
43                 currentLine++;
44                 if (currentLine < dialogueLine.Length)
45                 {
46                     //当你点击下一行的时候，有可能不是你对话的文本,所以检查文本是否是名字
47                     CheckName();
48                     //dialogueText.text = dialogueLine[currentLine];
49                     StartCoroutine(ScrollingText());
50                 }
51             }
52             else
53             {
54                 dailong.SetActive(false);
55                 FindObjectOfType<PlayerControl>().isMove = true;
56             }
57         }
58     }
59 }
60 public void ShowDlogue(string[] _lines ,bool isName)
61 {
62     dialogueLine= _lines;//把传送进来的文本框赋值显示出来的
63     currentLine = 0;
64     CheckName();//检查是否为名字
65
66     // dialogueText.text = dialogueLine[currentLine];
67     StartCoroutine(ScrollingText());
68
69     dailong.SetActive(true);
70 }
```

```

71     nameText.gameObject.SetActive(isName);
72
73     FindObjectOfType<PlayerControl>().isMove = false;
74 }
75 private void CheckName()//检查这一行是否为名字，若是名字，则把名字传输到显示名字的ui，然后跳过
    这一行
76 {
77     if (dialogueLine[currentLine].StartsWith("n-"))
78     {
79         nameText.text = dialogueLine[currentLine].Replace("n-", "");//将n-替换成null;
80         currentLine++;
81     }
82 }
83 //滚动文字效果
84 IEnumerator ScrollingText()
85 {
86     isScrolling = true;
87     dialogueText.text = "";//保证每一次开始都是从null开始
88     foreach (char letter in dialogueLine[currentLine].ToCharArray())
89     {
90         dialogueText.text += letter;//把每个字母单独拆开放进一个临时数组，然后在单独显示出来
91         yield return new WaitForSeconds(textSpeed);
92     }
93     isScrolling = false;
94 }
95 }
96

```

对话者脚本，把脚本放到需要对话的角色身上

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Talkable : MonoBehaviour
6 {
7     [SerializeField] private bool isEnter;
8     [TextArea(1,3)]
9     public string[] lines;
10    [SerializeField] private bool isName;
11    private void OnTriggerEnter2D(Collider2D collision)
12    {
13        if (collision.CompareTag("Player"))
14        {
15            isEnter= true;
16        }
17    }
18 }

```

```
17  }
18  private void OnTriggerExit2D(Collider2D collision)
19  {
20      if (collision.CompareTag("Player"))
21      {
22          isEnter= false;
23      }
24  }
25  private void Update()
26  {
27      if(Input.GetKeyDown(KeyCode.E)&&isEnter&&Dailong.instance.dailong.activeInHierarchy==false)
28      {
29          Dailong.instance.ShowDlogue(lines, isName);
30      }
31  }
32 }
33
```