

Sistema de Diálogos

Índice.

Este proyecto tiene contenido para hacer un sistema de diálogo sencillo, bonito y en poco tiempo.

Contiene scripts para movimiento básico de personaje, script de diálogo para un NPC, script para activar y desactivar un texto o imagen cuando tu “player” este cerca, sprites de personajes muy lindos (como Miku) entre otras cosas.

Instalación.

Para la instalación solo hay que seguir unos sencillos pasos.

1. Para más comodidad, puedes entrar a este link <https://github.com/Sireth6/Sistema-de-dialogos> para entrar de una forma más sencilla y rápida.
2. En el caso de no ser posible entrar al link, puedes encontrar el repositorio buscando GitHub en Google, después, en el buscador de GitHub, introducirás el nombre “Sireth6/Sistema-de-dialogos”
3. Una vez hayas entrado a la página, dale clic al botón verde con la leyenda “code” y le darás clic en “Download Zip”

4. Se descargará un archivo .zip o .rar.
5. Darás clic derecho al archivo y después en “Extraer aquí”.
6. Te irás a la carpeta “Sistema de diálogos” y te aparecerán 4 carpetas más.
7. Finalmente arrastra esas 4 carpetas a tu proyecto de Unity.

Contenido.

- Script de movimiento

Este script funciona para que tu personaje 2D se mueve de izquierda a derecha (en eje X) con las teclas A y D las cuales también puedes modificar a tu gusto.

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

public class PlayerMovement : MonoBehaviour
{
    public float speed;
    public float dirX;
    public SpriteRenderer spr;
    public Animator anim;

    // Update is called once per frame
    void Update()
    {
        dirX = Input.GetAxis("Horizontal");

        transform.Translate(Vector3.right * dirX * speed);

        if (dirX > 0)
        {
            spr.flipX = false;
        }
        if (dirX < 0)
        {
            spr.flipX = true;
        }

        if (Input.GetKeyDown(KeyCode.A))
        {
            anim.SetBool("Run", true);
        }

        if (Input.GetKeyUp(KeyCode.A))
        {
            anim.SetBool("Run", false);
        }

        if (Input.GetKeyDown(KeyCode.D))
        {
            anim.SetBool("Run", true);
        }

        if (Input.GetKeyUp(KeyCode.D))
        {
            anim.SetBool("Run", false);
        }
    }
}
```

- Script de diálogo para NPC

Este Script es para hacer el panel de diálogo en un NPC, primero hay que agregar un panel de texto con un Text Mesh Pro, después, agregamos en el script las variables necesarias. Con este código tu panel de dialogo tendrá variables publicas para poder modificar el texto y los diálogos a tu gusto.

```
public class Dialogue : MonoBehaviour
{
    [SerializeField] private GameObject dialogueMark;
    [SerializeField] private GameObject Press;
    [SerializeField] private GameObject dialoguePanel;
    [SerializeField] private TMP_Text dialogueText;
    [SerializeField, TextArea(4, 6)] private string[] dialogueLines;
    [SerializeField] private TMP_Text nter;

    private float typingTime = 0.05f;
    private bool isPlayerInRange;
    private bool didDialogueStart;
    private int lineIndex;

    // Update is called once per frame
    void Update()
    {
        if(isPlayerInRange && Input.GetButtonDown("Submit"))
        {
            if(!didDialogueStart)
            {
                StarDialogue();
            }
            else if (dialogueText.text == dialogueLines[lineIndex])
            {
                NextDialogueLine();
            }
            else
            {
                StopAllCoroutines();
                dialogueText.text = dialogueLines[lineIndex];
            }
        }
    }
}
```

```

private void StartDialogue()
{
    didDialogueStart = true;
    dialoguePanel.SetActive(true);
    dialogueMark.SetActive(false);
    Press.SetActive(false);
    lineIndex = 0;
    Time.timeScale = 0f;
    StartCoroutine>ShowLine());
}

private IEnumerator ShowLine()
{
    dialogueText.text = string.Empty;

    foreach(char ch in dialogueLines[lineIndex])
    {
        dialogueText.text += ch;
        yield return new WaitForSecondsRealtime(typingTime);
    }

private void NextDialogueLine()
{
    lineIndex++;
    if(lineIndex < dialogueLines.Length)
    {
        StartCoroutine>ShowLine());
    }
    else
    {
        didDialogueStart = false;
        dialoguePanel.SetActive(false);
        dialogueMark.SetActive(true);
        Press.SetActive(true);
        Time.timeScale = 1f;
    }

private IEnumerator ShowLine()
{
    dialogueText.text = string.Empty;

    foreach(char ch in dialogueLines[lineIndex])
    {
        dialogueText.text += ch;
        yield return new WaitForSecondsRealtime(typingTime);
    }
}

```

```

private void OnTriggerEnter2D(Collider2D collision)
{
    if (collision.gameObject.CompareTag("Player"))
    {
        isPlayerInRange = true;
        dialogueMark.SetActive(true);
        Press.SetActive(true);
    }
}

private void OnTriggerExit2D(Collider2D collision)
{
    if (collision.gameObject.CompareTag("Player"))
    {
        isPlayerInRange = false;
        dialogueMark.SetActive(false);
        Press.SetActive(false);
    }
}

private IEnumerator ShowLine()
{
    dialogueText.text = string.Empty;

    foreach(char ch in dialogueLines[lineIndex])
    {
        dialogueText.text += ch;
        yield return new WaitForSecondsRealtime(typingTime);
    }
}

```

- Script para interactuar,
Este script servirá para mostrar un texto flotante o símbolo para después poder interactuar con ellos (los textos que aparecen en el proyecto son imágenes, recuerda que para cambiar éstos, tienes que cambiar la imagen del texto por otra, ya que no son textos reales)

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

public class Interact : MonoBehaviour
{
    [SerializeField] private GameObject moveText;

    private bool isPlayerInRange;

    // Update is called once per frame
    void Update()
    {

    }

    private void OnTriggerEnter2D(Collider2D collision)
    {
        if (collision.gameObject.CompareTag("Player"))
        {
            isPlayerInRange = true;
            moveText.SetActive(true);
        }
    }

    private void OnTriggerExit2D(Collider2D collision)
    {
        if (collision.gameObject.CompareTag("Player"))
        {
            isPlayerInRange = false;
            moveText.SetActive(false);
        }
    }
}
```

Recomendaciones.

El objetivo de este proyecto fue crear un sistema de diálogo, como recomendación, usa tu NPC como prefab para que lo puedas usar en otros NPC que tengas diálogo sin tener que repetir todos los pasos, al igual para los objetos que tengan algún tipo de interacción.