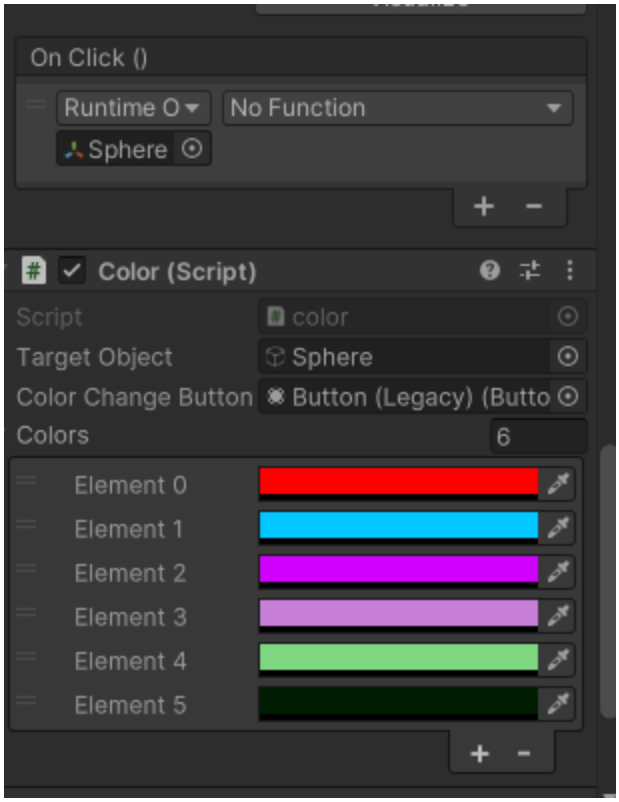
Color Change button

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
public class color : MonoBehaviour
{
public GameObject targetObject;
public Button colorChangeButton;
public Color[] colors; // Array of colors to cycle through
private int currentColorIndex = 0;
private void Start()
{
// Add a listener to the button click event
colorChangeButton.onClick.AddListener(ChangeObjectColor);
// Initialize the target object's color
targetObject.GetComponent<Renderer>().material.color =
colors[currentColorIndex];
}
private void ChangeObjectColor()
{
// Increment the color index and wrap around if needed
currentColorIndex = (currentColorIndex + 1) % colors.Length;
// Change the target object's color
targetObject.GetComponent<Renderer>().material.color =
colors[currentColorIndex];
}
}
```



Player score

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

public class score : MonoBehaviour
{
    // Start is called before the first frame update
    public Transform player;
    public Text scoreText;



    // Update is called once per frame
    void Update()
    {
        scoreText.text=player.position.z.ToString();
    }
```

```
}
```

Switch scene

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
public class SceneLoader : MonoBehaviour
{
    public string scene2;

    public void LoadScene()
    {
        SceneManager.LoadScene(scene2);
    }
}
```

Move

```
using UnityEngine;

public class move : MonoBehaviour

{
public float moveSpeed = 10f;
public float turnSpeed = 50f;
private Rigidbody rb;
private void Start()
{
rb = GetComponent<Rigidbody>();
}
private void FixedUpdate()
{
```

```
float horizontalInput = Input.GetAxis("Horizontal");
float verticalInput = Input.GetAxis("Vertical");
Vector3 movement = transform.forward * verticalInput * moveSpeed *
Time.fixedDeltaTime;
Quaternion rotation = Quaternion.Euler(0f, horizontalInput * turnSpeed *
Time.fixedDeltaTime, 0f);
rb.MovePosition(rb.position + movement);
rb.MoveRotation(rb.rotation * rotation);
}}
```

Move

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

public class NewBehaviour : MonoBehaviour



{
  public Rigidbody rr;
    // Start is called before the first frame update
 //public float forwardForce=2000f;

 void FixedUpdate()
 {
float moveSpeed = 10;
float moveSpeed1 = 30;
//Define the speed at which the object moves.

float horizontalInput = Input.GetAxis("Horizontal");
//Get the value of the Horizontal input axis.

float verticalInput = Input.GetAxis("Vertical");
//Get the value of the Vertical input axis.
```

```
transform.Translate(new Vector3(horizontalInput, 0, 0) * moveSpeed *
Time.deltaTime);
transform.Translate(new Vector3(0, 0, verticalInput) * moveSpeed1 *
Time.deltaTime);
}
}
```

```
using UnityEngine;

public class KeyInputMover : MonoBehaviour
{
    public float moveSpeed = 5f; // Speed at which the object moves

    void Update()
    {
        // Initialize movement vector
        Vector3 movement = Vector3.zero;

        // Check for specific key inputs
        if (Input.GetKey(KeyCode.W)) // Move forward
        {
            movement += Vector3.forward;
        }
        if (Input.GetKey(KeyCode.S)) // Move backward
        {
            movement += Vector3.back;
        }
        if (Input.GetKey(KeyCode.A)) // Move left
        {
            movement += Vector3.left;
        }
        if (Input.GetKey(KeyCode.D)) // Move right
        {
            movement += Vector3.right;
        }
```

```
        // Apply the movement
        transform.Translate(movement * moveSpeed * Time.deltaTime);
    }
}
```

Throw

```
using UnityEngine;

public class ThrowObject : MonoBehaviour
{
    public float throwForce = 10f; // Adjust the force applied to throw the object

    private Rigidbody rb;

    void Start()
    {
        rb = GetComponent<Rigidbody>();
    }

    void Update()
    {
        if (Input.GetMouseButtonDown(0)) // Left mouse button
        {
            Throw();
        }
    }

    void Throw()
    {
        // Reset the velocity to ensure a consistent throw
        rb.velocity = Vector3.zero;

        // Apply a force to throw the object
        rb.AddForce(transform.forward * throwForce, ForceMode.VelocityChange);
    }
}
```

Follow

```
using UnityEngine;

public class Follow : MonoBehaviour
{
    public Transform objectToFollow;
    public float followSpeed = 1;

    void Update()
    {
        // calculate the distance between this object and the target object
        // and move a small portion of that distance each frame:

        var delta = objectToFollow.position - transform.position;
        transform.position += delta * Time.deltaTime * followSpeed;
    }
}
```

BUTTON PRESSED

```
using UnityEngine;
using UnityEngine.UI;

public class ButtonHandler : MonoBehaviour
{
    // This method will be called when the button is pressed
    public void PrintMessage()
    {
        Debug.Log("Button was pressed!");
    }
}
```

Link the Button to the Script:

- Select the button in the `Hierarchy` window.

- In the `Inspector` window, scroll down to the `Button` component and find the `On Click ()` section.
- Click the `+` button to add a new event listener.
- Drag the GameObject that has the `ButtonHandler` script attached to the field where it says `None (Object)`.
- From the dropdown menu, select the `ButtonHandler > PrintMessage` method.