

Step 1: Implement the Interactable Interface

Create the `IInteractable` Interface: This interface should be in the project's `Scripts` folder.

```
using UnityEngine;

public interface IInteractable
{
    void Interact();
}
```

Step 2: Create a Script for Your Interactable Object

For each new interactable object (e.g., lights, buttons, doors), create a separate script that implements the `IInteractable` interface.

Example: Interactable Light

For a light object that turns on/off when interacted with:

```
using UnityEngine;

public class InteractableLight : MonoBehaviour, IInteractable
{
    [SerializeField] private Light lightSource;

    public void Interact()
    {
        if (lightSource != null)
        {
            lightSource.enabled = !lightSource.enabled; // Toggle light
            Debug.Log("Light toggled!");
        }
    }
}
```

Example: Interactable Button

For a button that changes some text when interacted with:

```
using UnityEngine;
using TMPro; // Assuming TextMeshPro is used for text
```

```

public class InteractableButton : MonoBehaviour, IInteractable
{
    [SerializeField] private TextMeshProUGUI textDisplay;

    public void Interact()
    {
        if (textDisplay != null)
        {
            textDisplay.text = "Button Pressed!";
            Debug.Log("Button text updated!");
        }
    }
}

```

Example: Interactable Cube (Basic) For a cube that changes color when interacted with:

```

using UnityEngine;

public class InteractableCube : MonoBehaviour, IInteractable
{
    public void Interact()
    {
        GetComponent<Renderer>().material.color = Color.red;
        Debug.Log("Cube color changed!");
    }
}

```

Step 3: Add the Script to Your Object in the Scene

1. Attach your custom script (e.g., `InteractableLight`, `InteractableButton`) to the `GameObject` in the scene.
2. Ensure the `GameObject` has a `Collider` component (like `Box Collider` or `Sphere Collider`) so it can detect raycast interactions.
3. Optional: Tag or layer the object as "Interactable" to keep it organized.