

○○○

```
/// <summary>
/// Start conversation when player enters NPC trigger
/// </summary>
private void OnTriggerEnter(Collider other)
{
    if (other.CompareTag("Player"))
    {
        StartConversation();
        GetComponent<BoxCollider>().enabled = false; // prevent re-triggering
    }
}

/// <summary>
/// Activates conversation UI and starts typing text
/// </summary>
private void StartConversation()
{
    playerScript.you_are_in_conversation = true;
    playerScript.conversation_manager_script.the_container.SetActive(true);
    ContinueConversation();
}

/// <summary>
/// Handles advancing conversation or finishing it
/// </summary>
public void ContinueConversation()
{
    if (conversationIndex >= conversationTexts.Length)
    {
        FinishConversation();
        return;
    }

    playerScript.conversation_manager_script.the_text.text = string.Empty;
    playerScript.conversation_manager_script.select_options.SetActive(false);
    playerScript.conversation_manager_script.button_for_continue.SetActive(false);
    charIndex = 0;
    playerScript.conversation_manager_script.the_npc = this;

    StartCoroutine(TypeText());
}

/// <summary>
/// Types each letter of the current conversation text gradually
/// </summary>
private IEnumerator TypeText()
{
    while (charIndex < conversationTexts[conversationIndex].Length)
    {
        playerScript.conversation_manager_script.the_text.text +=
conversationTexts[conversationIndex][charIndex];
        charIndex++;
        yield return new WaitForSeconds(0.05f);
    }

    // Show options or continue button after text is done
    bool hasOptions = (multipleOptions.Length > conversationIndex) &&
multipleOptions[conversationIndex];
    playerScript.conversation_manager_script.select_options.SetActive(hasOptions);
    playerScript.conversation_manager_script.button_for_continue.SetActive(!hasOptions);

    conversationIndex++;
}

/// <summary>
/// Ends conversation and optionally teleports player
/// </summary>
private void FinishConversation()
{
    playerScript.conversation_manager_script.the_container.SetActive(false);

    if (teleportPosition != null)
    {
        transitionEffect.SetActive(true);
        StartCoroutine(TeleportPlayer());
    }
    else
    {
        playerScript.you_are_in_conversation = false;
    }
}

/// <summary>
/// Teleports player to target position with optional scene effects
/// </summary>
private IEnumerator TeleportPlayer()
{
    if (bossFightScene != null)
        bossFightScene.SetActive(true);

    playerScript.cc.enabled = false;
    playerScript.enabled = false;

    yield return new WaitForSeconds(1f);

    player.transform.position = teleportPosition.transform.position;

    if (waterObject != null)
        waterObject.SetActive(false);

    playerScript.enabled = true;
    playerScript.cc.enabled = true;

    yield return new WaitForSeconds(2f);

    playerScript.you_are_in_conversation = false;

    yield return new WaitForSeconds(2f);

    transitionEffect.SetActive(false);
}
```