```
using UnityEngine;
using TMPro;

public class ArtInteraction : MonoBehaviour
{
    public TextMeshProUGUI titleText;
    public TextMeshProUGUI descriptionText;
    public GameObject artPopupPanel;
    public CanvasGroup artCanvasGroup; // For fade-in and fade-out
    private Camera mainCamera;
    private bool isPopupVisible = false;
    private AudioSource currentAudioSource = null;
    public float fadeDuration = 0.5f;

    void Start()
    {
        mainCamera = Camera.main;

        // Ensure the cursor is visible and unlocked
        Cursor.lockState = CursorLockMode.None;
        Cursor.visible = true;

        // Debugging checks for missing assignments
        if (mainCamera == null) Debug.LogError("Main Camera is not assigned. Make
sure your camera is tagged as 'MainCamera'.");
        if (titleText == null) Debug.LogError("Title Text is not assigned in the
Inspector.");
        if (descriptionText == null) Debug.LogError("Description Text is not
assigned in the Inspector.");
        if (artPopupPanel == null) Debug.LogError("Art Popup Panel is not assigned
in the Inspector.");
        if (artCanvasGroup == null) Debug.LogError("CanvasGroup for Art Popup Panel
is not assigned.");

        artPopupPanel.SetActive(false); // Hide the pop-up initially
        artCanvasGroup.alpha = 0; // Start fully invisible
    }

    void Update()
    {
        if (Input.GetMouseButtonDown(0)) // Left-click
        {
            Ray ray = mainCamera.ScreenPointToRay(Input.mousePosition);
            RaycastHit hit;

            if (Physics.Raycast(ray, out hit))
            {
                Debug.Log("Raycast hit: " + hit.transform.name); // Debug: Log the
name of the object hit

                ArtPiece artPiece = hit.transform.GetComponent<ArtPiece>();
                AudioSource audioSource =
hit.transform.GetComponent<AudioSource>();

                if (artPiece != null)
                {
                    Debug.Log("ArtPiece found: " + artPiece.title); // Debug: Log
the title of the ArtPiece
                    ToggleArtDetails(artPiece, audioSource);
                }
                else
                {
                    Debug.LogWarning("Raycast hit something without an ArtPiece
component.");
                }
            }
            else
            {
                Debug.LogWarning("Raycast did not hit any object.");
            }
        }
    }

   void ToggleArtDetails(ArtPiece artPiece, AudioSource audioSource)
{
    if (isPopupVisible && currentAudioSource == audioSource)
    {
        StartCoroutine(FadeCanvasGroup(artCanvasGroup, 1, 0, fadeDuration));
        isPopupVisible = false;

        // Stop the audio if it's playing
        if (currentAudioSource != null && currentAudioSource.isPlaying)
        {
            currentAudioSource.Stop();
        }
        currentAudioSource = null;
    }
    else
    {
        // Debug log to verify text updates
        Debug.Log("Updating title and description: " + artPiece.title + " | " +
artPiece.description);

        // Update UI text
        titleText.text = artPiece.title;
        descriptionText.text = artPiece.description;

        artPopupPanel.SetActive(true);
        StartCoroutine(FadeCanvasGroup(artCanvasGroup, 0, 1, fadeDuration));
        isPopupVisible = true;

        // Stop any previously playing audio
        if (currentAudioSource != null && currentAudioSource.isPlaying)
        {
            currentAudioSource.Stop();
        }

        // Start playing the new audio
        currentAudioSource = audioSource;
        if (currentAudioSource != null)
        {
            currentAudioSource.Play();
        }
    }
}

    private System.Collections.IEnumerator FadeCanvasGroup(CanvasGroup canvasGroup,
float startAlpha, float endAlpha, float duration)
```

```
    {
        float time = 0;
        canvasGroup.alpha = startAlpha;

        while (time < duration)
        {
            time += Time.deltaTime;
            canvasGroup.alpha = Mathf.Lerp(startAlpha, endAlpha, time / duration);
            yield return null;
        }

        canvasGroup.alpha = endAlpha;

        if (endAlpha == 0)
        {
            artPopupPanel.SetActive(false);
        }
    }
}
```