**Unity Scene Management and UI Project**

This Unity project demonstrates basic scene management and UI interactions. It includes two scenes: ‘HomeScene’ and ‘GameScene’. The project allows navigation between these scenes, implements a popup system using UI buttons, includes an arrow button in the ‘GameScene’ to return to the ‘HomeScene’, and provides a close button within the popup to hide it.

**Features**

1. **Scene Management**:
   * A button in the ‘HomeScene’ allows the player to transition to the ‘GameScene’.
   * An arrow button in the ‘GameScene’ returns the player to the ‘HomeScene’.
   * This is implemented through a script named ‘SceneChange’, which handles scene transitions.
2. **UI Buttons and Popups**:
   * In the ‘GameScene’, a button triggers the visibility of a popup message.
   * The popup can be dismissed by clicking a close button inside the popup.
   * Both the popup and close functionality are implemented using Unity's ‘OnClick()’ option, where buttons directly call ‘popupPanel.SetActive(true)’ and ‘popupPanel.SetActive(false)’.
3. **Back Navigation**:
   * An arrow button is placed in the upper left corner of the ‘GameScene’. Clicking it will return the player to the ‘HomeScene’.

**Project Structure**

**Scenes:**

* **HomeScene**: Contains a "Start Game" button. When clicked, it transitions the player to the ‘GameScene’.
* **GameScene**:
  + Contains a button that shows a popup message.
  + The popup includes a close button to hide it.
  + An arrow button is located in the top left corner to navigate back to the ‘HomeScene’.

**Script:**

* **SceneChange.cs**: This script is responsible for handling scene transitions.

**How to Use**

1. **Scene Transition**:
   * Attach the ‘SceneChange’ script to an empty GameObject in your scene (e.g., GameManager).
   * Create a UI Button in the ‘HomeScene’.
   * In the Button's ‘OnClick()’ event in the Inspector, drag the GameObject with the ‘SceneChange’ script, and select the ‘SceneChange.SceneChange’ method.
   * Pass the name of the scene you want to load as a parameter (e.g., GameScene).
2. **Back Button (Arrow Button) Implementation**:
   * In the GameScene, create a UI Button for the arrow in the top left corner.
   * In the Button's ‘OnClick()’ event in the Inspector, drag the GameObject with the ‘SceneChange’ script, and select the ‘SceneChange.SceneChange’ method.
   * Pass ‘HomeScene’ as the parameter to return to the ‘HomeScene’ when clicked.
3. **Popup Implementation**:
   * In the ‘GameScene’, create a Canvas with a UI Panel for the popup.
   * Initially, set the Panel's ‘GameObject.SetActive(false)’ to hide the popup.
   * Add a button to show the popup by calling ‘popupPanel.SetActive(true)’ using Unity's ‘OnClick()’ option in the Inspector.
   * Inside the popup, add a close button that hides the popup when clicked by calling ‘popupPanel.SetActive(false)’ using Unity's ‘OnClick()’ option.

**Setup Instructions**

1. **Import Scenes**:
   * Ensure both scenes (‘HomeScene’ and ‘GameScene’) are added to your Build Settings under **File > Build Settings**.
2. **Assign Buttons**:
   * Assign the "Start Game" button in the ‘HomeScene’ to trigger the scene transition to ‘GameScene’.
   * Assign the arrow button in the ‘GameScene’ to trigger the transition back to the ‘HomeScene’.
   * Assign buttons in the ‘GameScene’ to show/hide the popup using Unity's ‘OnClick()’ option to call ‘SetActive(true)’ and ‘SetActive(false)’.

**Conclusion**

This project demonstrates basic Unity scene management and UI functionality, including scene transitions, popups, and navigation between scenes using buttons. The popup functionality, including the close button, is handled directly through Unity's ‘OnClick()’ option for intuitive and simple UI management.

**License**

This project is for demonstration purposes and is free to use.