

## Tilevania Common Errors

Check these error solutions prior to asking for help!

### **If you can't edit points on polygon collider:**

Click Points

Click Paths

Open Element

Change Size to 4 (for four corners)

Adjust X and Y values to line up with corners (this may be some trial and error)

### **Jump error- if your player jumps as soon as play is hit:**

Check project settings for gravity (should be negative)

Make sure there is no semicolon after checking for "Jump"

### **Jump error- player doesn't jump.**

Ensure capsule collider a little outside box

### **Change scene persist code if it isn't working properly (changes highlighted):**

```
//SCENE PERSIST CHANGES
```

```
using System.Collections;
```

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

```
using UnityEngine;
```

```
using UnityEngine.SceneManagement;
```

```
public class ScenePersist : MonoBehaviour
```

```
{
```

```
    private int sceneIndex; //create variable
```

```
    private void Awake()
```

```

{
    int currentSceneIndex = SceneManager.GetActiveScene().buildIndex; //get active scene number
    ScenePersist[] persists = FindObjectsOfType<ScenePersist>(); //Create an array to hold scene
    persist objects
    foreach (var persist in persists) //Check for objects that are a part of scene persist
    {
        if (persist != this) //Check for scene persist in the scene
        {
            if (persist.sceneIndex == currentSceneIndex)
            {
                Destroy(gameObject); //destroy persist object and return immediately (anything that is
                already in room upon awake)
                return;
            }
            else
            {
                Destroy(persist.gameObject); //destroy if doesn't have the current scene index
            }
        }
    }
    sceneIndex = currentSceneIndex; //set sceneIndex variable to current scene index
    DontDestroyOnLoad(gameObject); //create dont destroy on load object after scene index set
    correctly and all prior scene persists have been destroyed
}

```

**private void Update() //ADDED THIS TO DELETE VALUES AND RESET LEVEL ONCE SUCCESS SCENE REACHED - YOU CAN CHANGE THIS NUMBER IF YOU ADD MORE LEVELS**

```

{
    int currentSceneIndex = SceneManager.GetActiveScene().buildIndex;
    if (currentSceneIndex == 5)

```

```

    {
        Destroy(gameObject); //Delete scene persist object if still present in success screen (4 IS THE INDEX OF MY SUCCESS SCREEN - YOURS MAY BE DIFFERENT - CHECK BUILD SETTINGS)
    }

    if (currentSceneIndex == 0)
    {
        Destroy(gameObject); //Delete scene persist (to reset coins) after death and sent to menu (0 IS THE INDEX OF MY MENU SCREEN - YOURS MAY BE DIFFERENT - CHECK BUILD SETTINGS)
    }
}

```

#### **//GAME SESSION ADDITION**

```

private void Update() //ADDED THIS TO DELETE VALUES AND RESET LEVEL ONCE SUCCESS SCENE REACHED - YOU CAN CHANGE THIS NUMBER IF YOU ADD MORE LEVELS
{
    int currentSceneIndex = SceneManager.GetActiveScene().buildIndex;
    if (currentSceneIndex == 5)
    {
        Destroy(gameObject);
    }
}

```

#### **IF COIN DISSAPPEARS ON SECOND LEVEL**

##### **//LEVEL EXIT ADDITION**

```

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

```

```
using UnityEngine.SceneManagement;
```

```
public class LevelExit : MonoBehaviour
```

```
{
```

```
    [SerializeField] float LevelLoadDelay = 1f;
```

```
    [SerializeField] float LevelExitSlowMoFactor = 0.2f;
```

```
    private void OnTriggerEnter2D(Collider2D collision)
```

```
    {
```

```
        StartCoroutine(LoadNextLevel());
```

```
    }
```

```
    IEnumerator LoadNextLevel()
```

```
    {
```

```
        Time.timeScale = LevelExitSlowMoFactor;
```

```
        yield return new WaitForSecondsRealtime(LevelLoadDelay);
```

```
        Time.timeScale = 1f;
```

```
        var currentSceneIndex = SceneManager.GetActiveScene().buildIndex;
```

```
        Destroy(FindObjectOfType<ScenePersist>()); //ADDED THIS - Destroy scene persist - added to fix  
coins disappearing on second level
```

```
        SceneManager.LoadScene(currentSceneIndex + 1);
```

```
    }
```

```
}
```