We can create our own events (like onSceneChange). When an event gets triggered, it will call all the functions that are subscribed to it. This is a fantastic way of making each script more independent (since it doesn’t need a reference to the script that triggered the event). It’s also a great way to avoid checking the state of another script every Update(), like busy waiting.

To see an example in the code, check out GameController.cs and PlayerMovement.cs (in the *cleanup* branch if reading before merge).

# The script that triggers the event:

To create an event, declare it as follows:

public static event Action gameStarted;

Since it’s public and static, other scripts will be able to access it via the class, without any reference to an instance of the class. Much safer.

To trigger it, use:

gameStarted?.Invoke();

The question mark is a neat way of doing the equivalent:

if (gameStarted != null) gameStarted.Invoke();

Once triggered, all functions that are subscribed to it will get called.

If you want an event to pass parameters to the called functions, include the type(s) of the parameter(s) in <> and pass the value(s) when invoked.

public static event Action<GameController> gameActive;

…

gameActive?.Invoke(this);

# A script that subscribes to the event:

Create a function that matches the type specified for the event:

public void OnGameStarted()

{

…

}

public void OnGameActive(GameController game)

{

…

}

Now subscribe this function to the event in an OnEnable() method:

void OnEnable()

{

GameObject.gameStarted += OnGameStarted;

GameObject.gameActive += OnGameActive;

}

And unsubscribe it in an OnDisable() method (to avoid null reference errors when this script is destroyed):

void OnDisable()

{

GameObject.gameStarted -= OnGameStarted;

GameObject.gameActive -= OnGameActive;

}

Now, whenever the event is triggered, this function will get called.

# Notes:

You can subscribe as many different functions from as many different scripts to the same event as you like.

Beware that when the event is triggered, it will call ALL these functions on all objects in the scene.