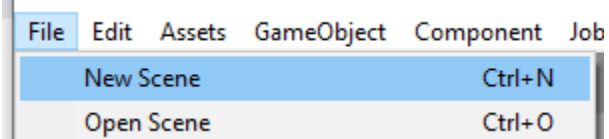


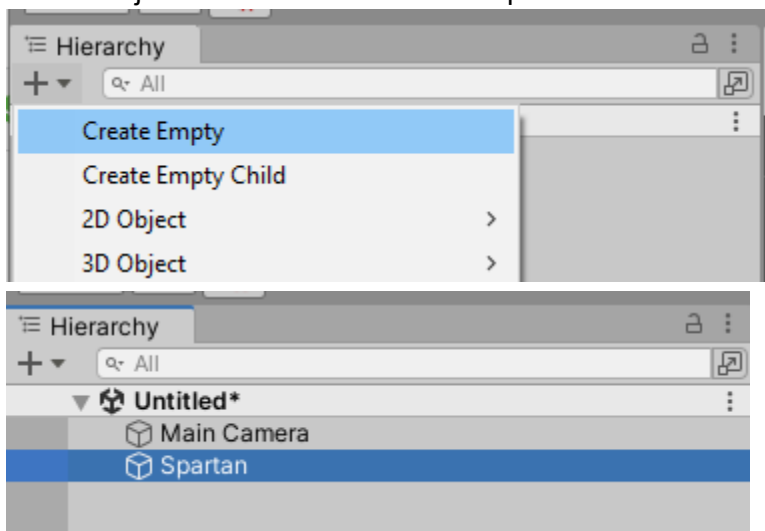
What is a Vector?

Unity Project Setup

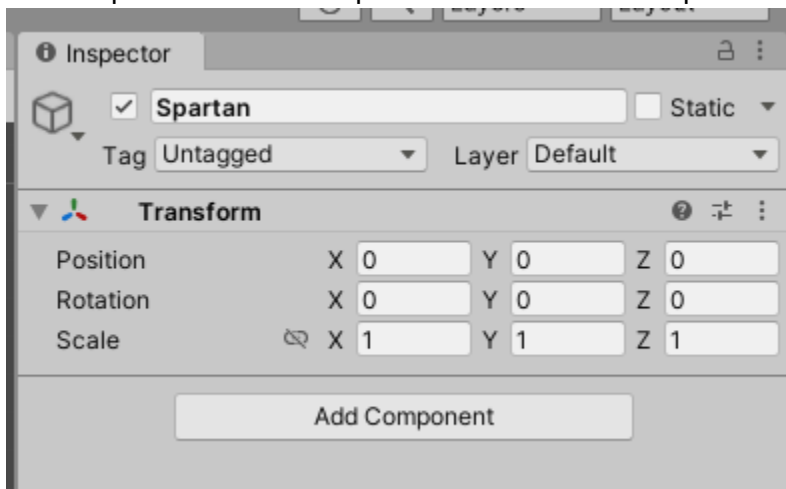
In your project create a new scene and call it Simple Vector Move



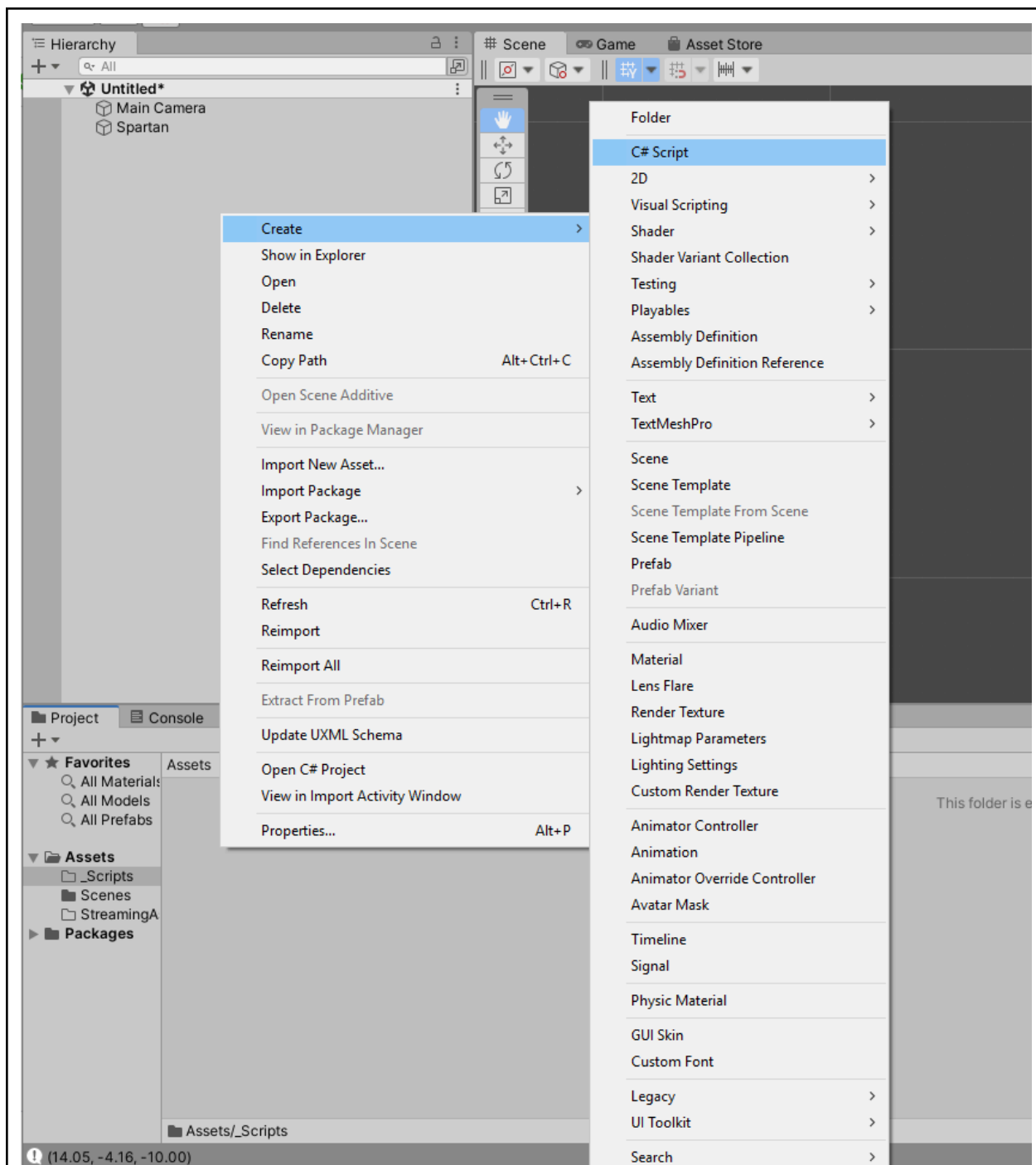
Add an object to be controlled. Call it Spartan.



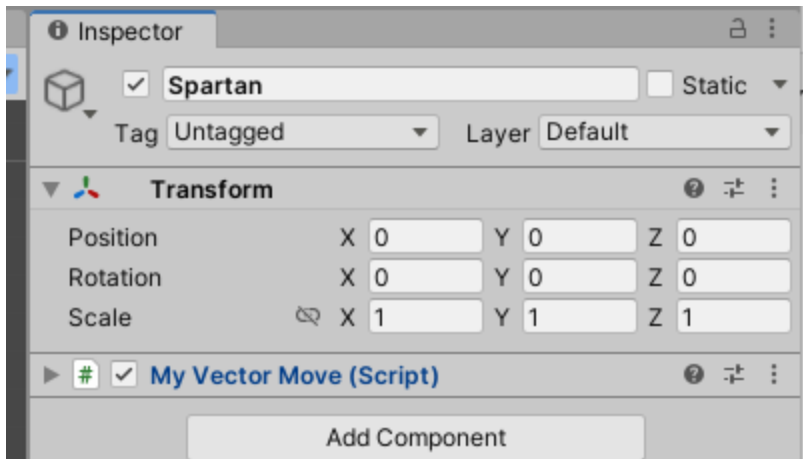
Select Spartan and set the position to Zero in the inspector.



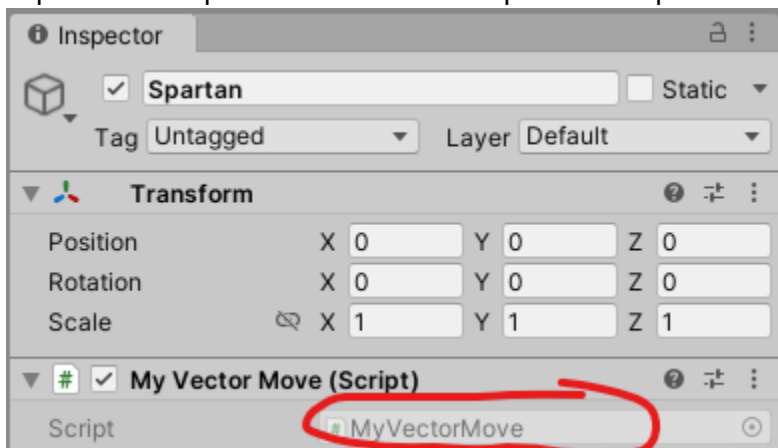
Now we will create a new script called MyVectorMove



Drag the script to the spartan game object.



Expand the script and double click to open the script file.



Make sure the code matches what is shown in the code segment below:

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

public class MyVectorMove : MonoBehaviour
{
    public Vector2 myDisplacementVector;
    public float mySpeedVariable;

    public GameObject ArrowPointingVector;
    Vector3 myScalar;

    public bool useMousePosition;
    public Vector3 location;
    float myMag;

    // Start is called before the first frame update
    void Start()
    {
```

```

}

// Update is called once per frame
void Update()
{
    /*
    This code will scale the arrow to make it point to were our guy will go
    */
    myScalar = new Vector3 (1f,1f,1f);
    if(useMousePosition)
    {
        Vector3 objectPos =
Camera.main.WorldToScreenPoint(ArrowPointingVector.transform.position);
        location = Camera.main.ScreenToWorldPoint(Input.mousePosition);
        Debug.Log(location);
        Vector3 dir = Input.mousePosition - objectPos;
        //dir.Normalize();
        myDisplacementVector = new Vector2(dir.x, dir.y);
        ArrowPointingVector.transform.rotation = Quaternion.Euler(0, 0,
Mathf.Atan2(myDisplacementVector.y, myDisplacementVector.x) * Mathf.Rad2Deg);

        Vector3 Loc_dir = location - gameObject.transform.position;

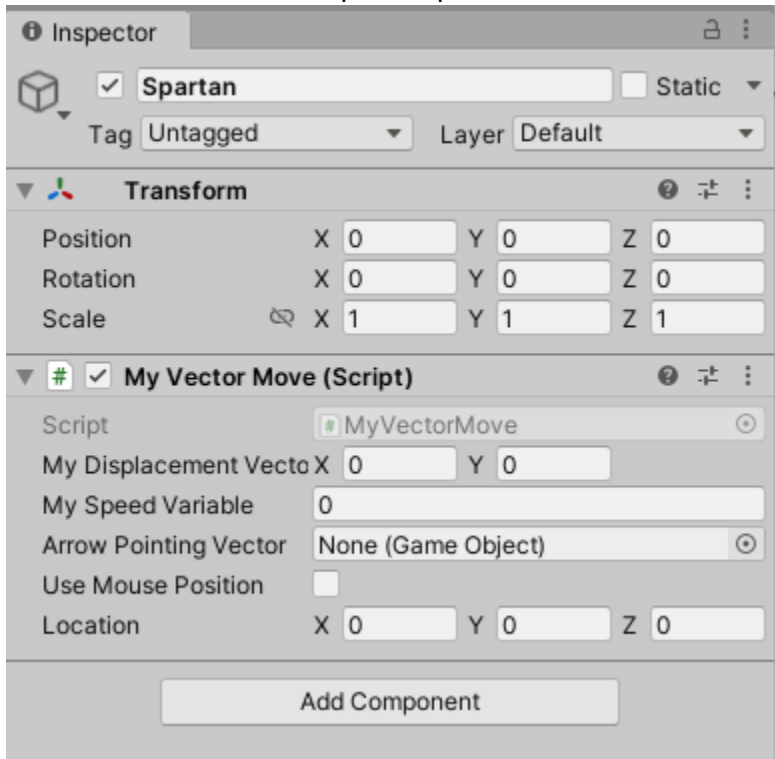
        location = new Vector3(Loc_dir.x, Loc_dir.y, 0f);
        //location = new Vector3(location.x, location.y, 0f);
        myMag = location.magnitude;
        ArrowPointingVector.transform.localScale = myScalar*myMag;
    }
    else
    {
        ArrowPointingVector.transform.rotation = Quaternion.Euler(0, 0,
Mathf.Atan2(myDisplacementVector.y, myDisplacementVector.x) * Mathf.Rad2Deg);
        myMag = myDisplacementVector.magnitude;
        ArrowPointingVector.transform.localScale = myScalar*myMag;
    }

    if(Input.GetKeyDown(KeyCode.Space ))
    {
        /*
        This Code uses the displacement vector and the speed scalar to move
        every time we press the spacebar
        */
        if(useMousePosition)
        {
            transform.Translate(location * mySpeedVariable , Space.World);
        }
        else
        {
            transform.Translate(myDisplacementVector * mySpeedVariable , Space.World);

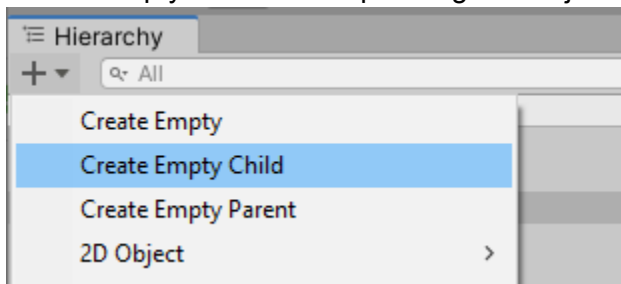
```

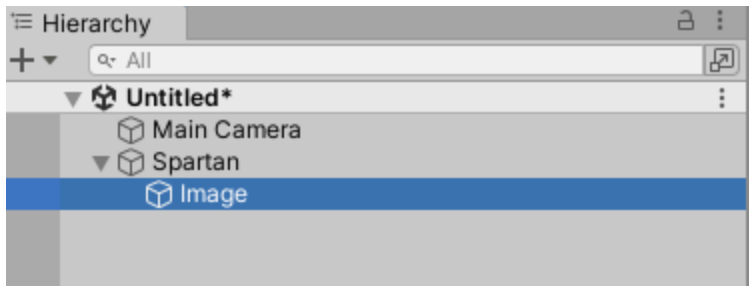
```
}  
}  
}  
}
```

Save the code and return to Unity.
Notice how the Vector scrip has updated.

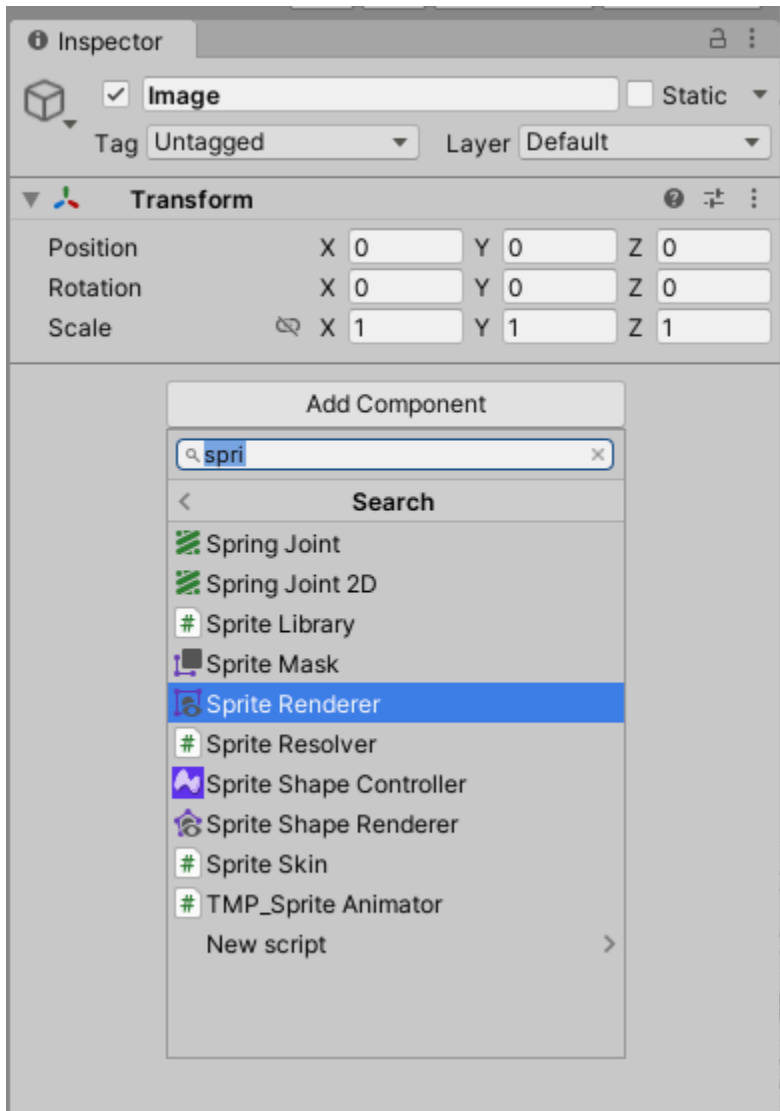


Next we will add a game object to manage our images.
Add an empty child to the Spartan game object. Call it image.



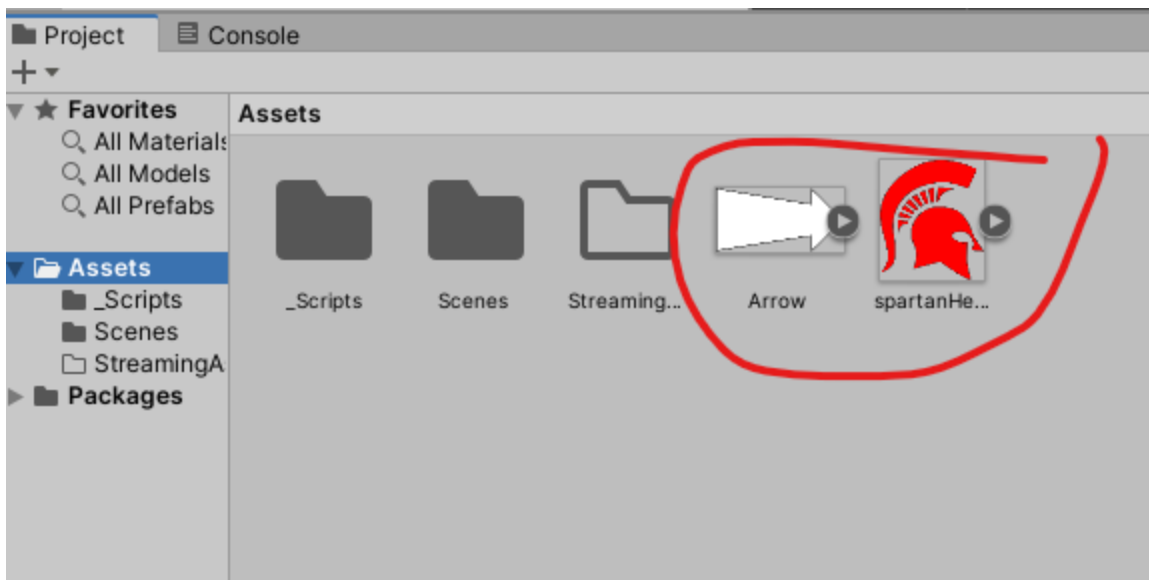


We will add a sprite render game object to the image game object. Select the Image game object then in the inspector select the add component button. Select Sprite Render from the list.

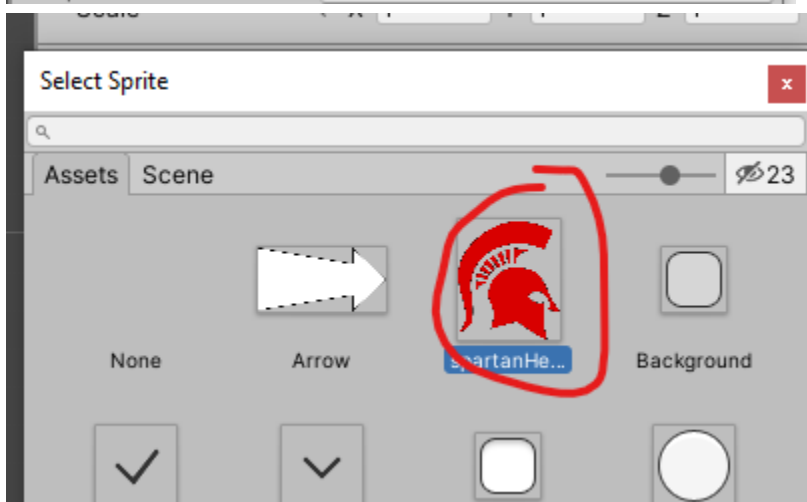
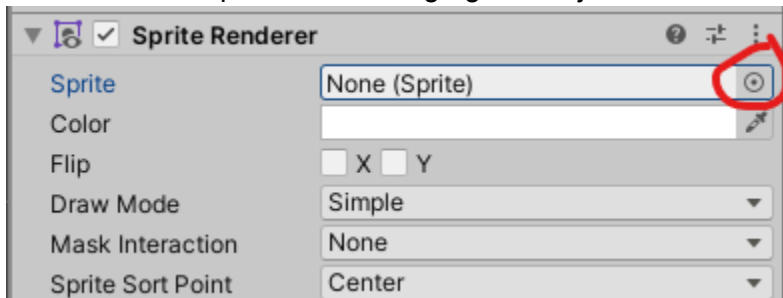


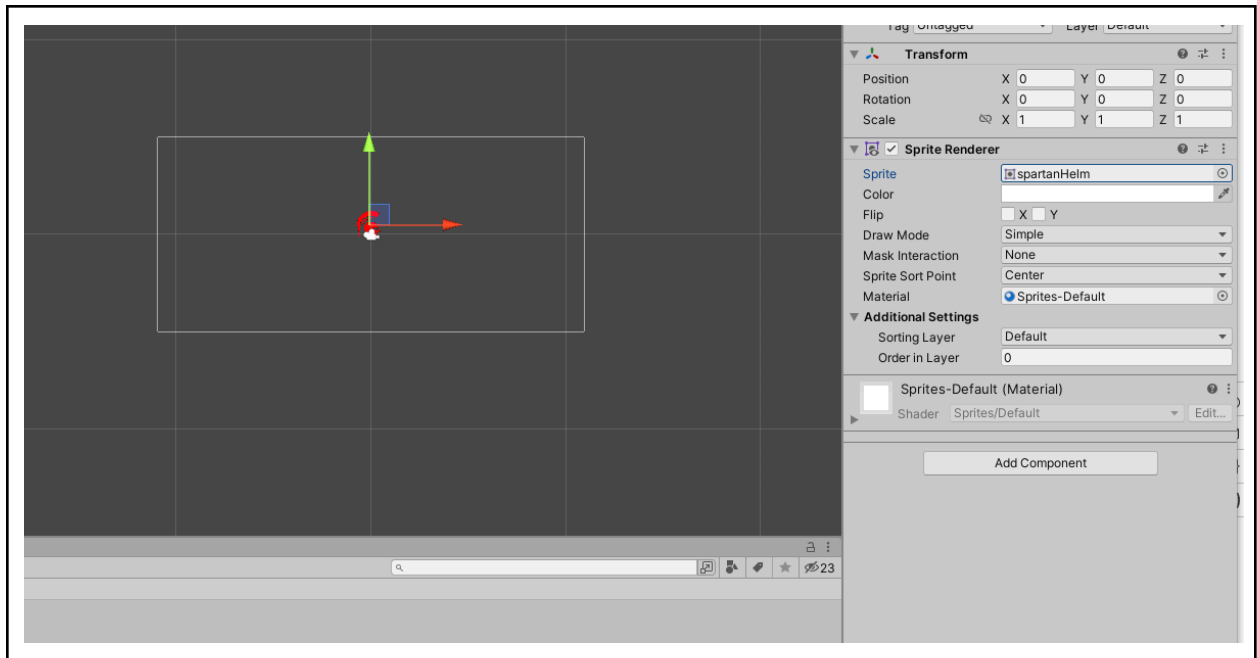
Also add a sprite render to the main spartan game object.

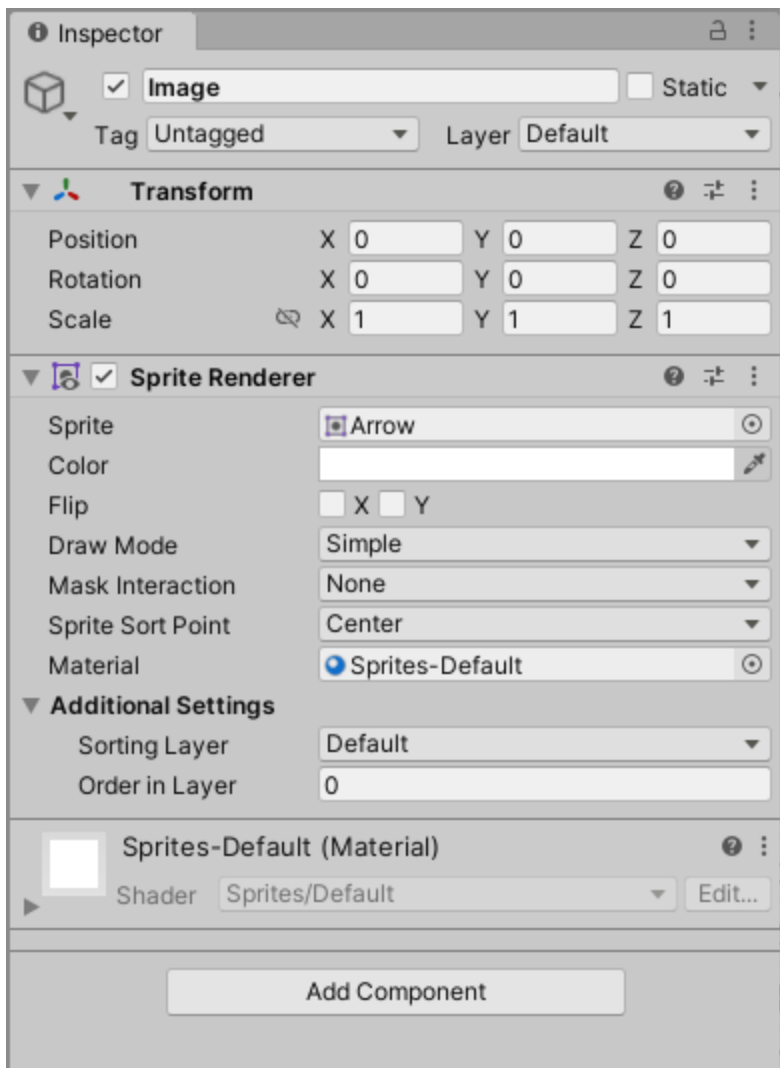
Drag the spartan helmet image, and the arrow to the project assets window.

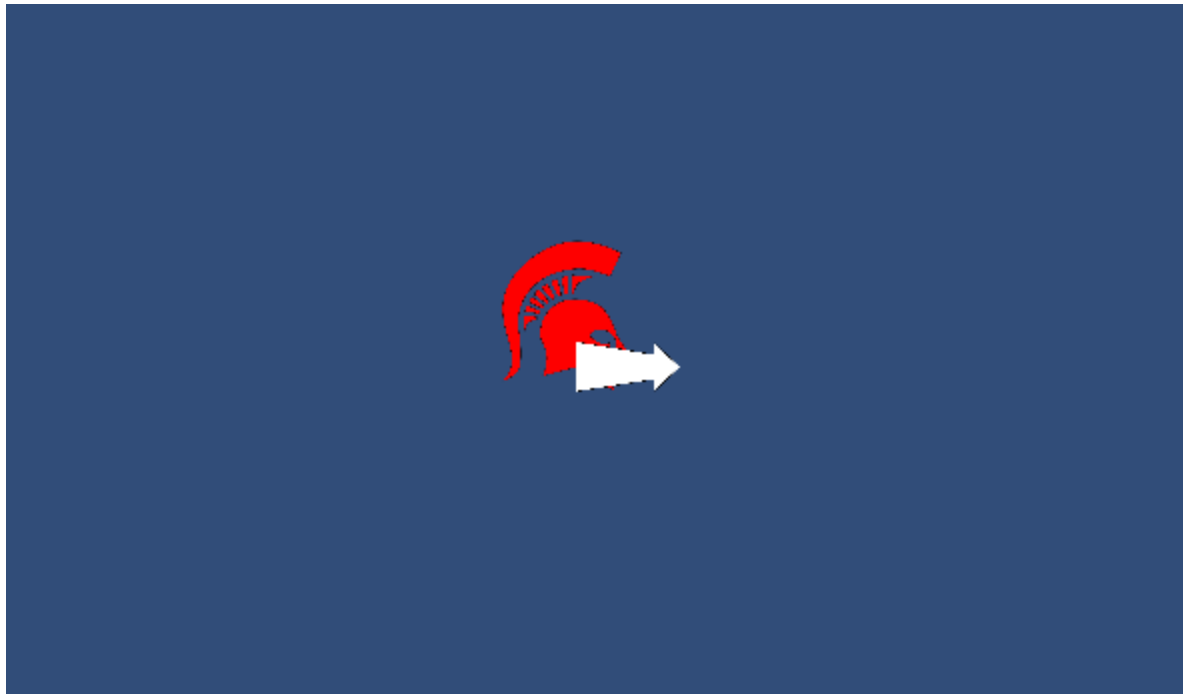


Now we can set the sprite to be the spartan helmet for the parent game object, and set the arrow to be the sprite for the image game object.

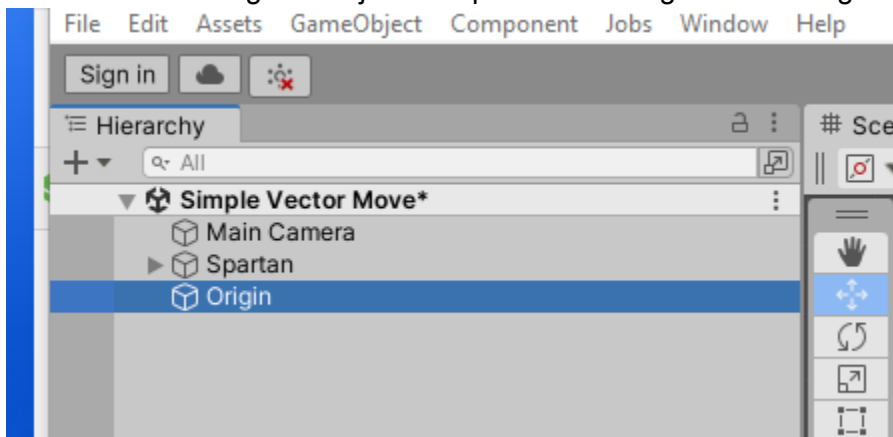




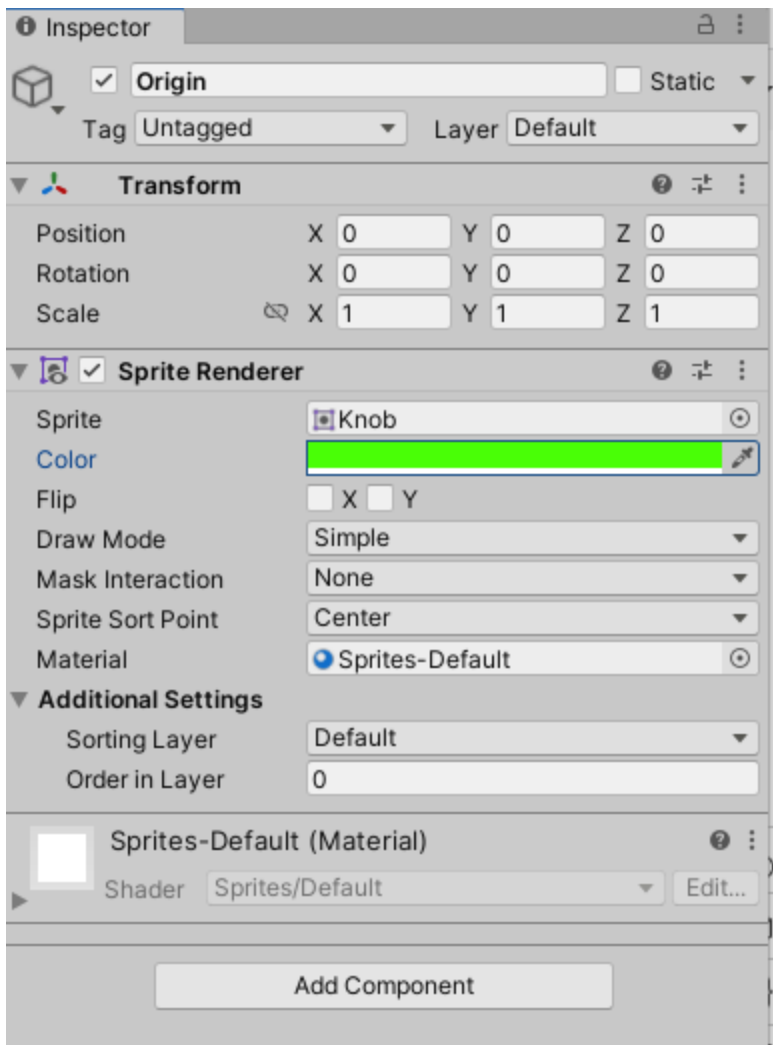




We will now add a game object to represent the origin. Add new game object and call it origin.



We will use it as a marker. Zero it's position and add a sprite render and select the knob image. Set the color to green.



Play test the game and ask the instructor to provide additional explanations at your workstation.