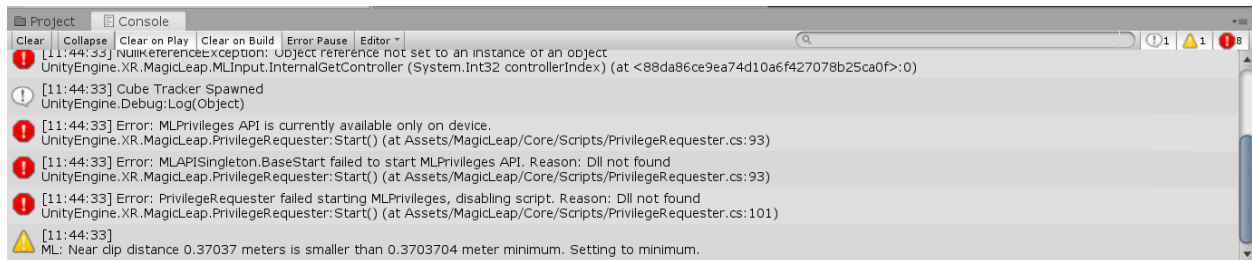


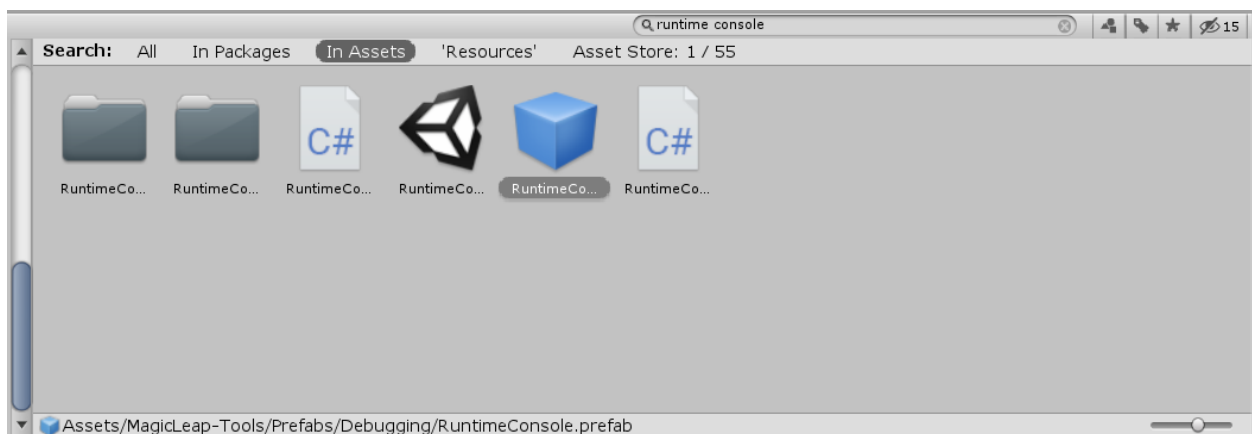
Adding a Debug Console to a Unity Magic Leap App

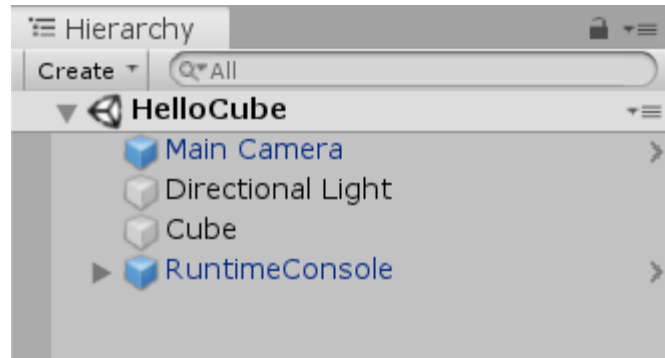
You can add a console to debug your Magic Leap application like the Unity console shown below by adding the **Runtime Console** prefab to your scene hierarchy.



STEP 1: Add the Runtime Console prefab

Look for the **Runtime Console** prefab under **MagicLeap-Tools > Prefabs > Debugging > RuntimeConsole** and drag it into your scene hierarchy.





STEP 2: Use the Main Camera prefab from the Magic Leap Toolkit

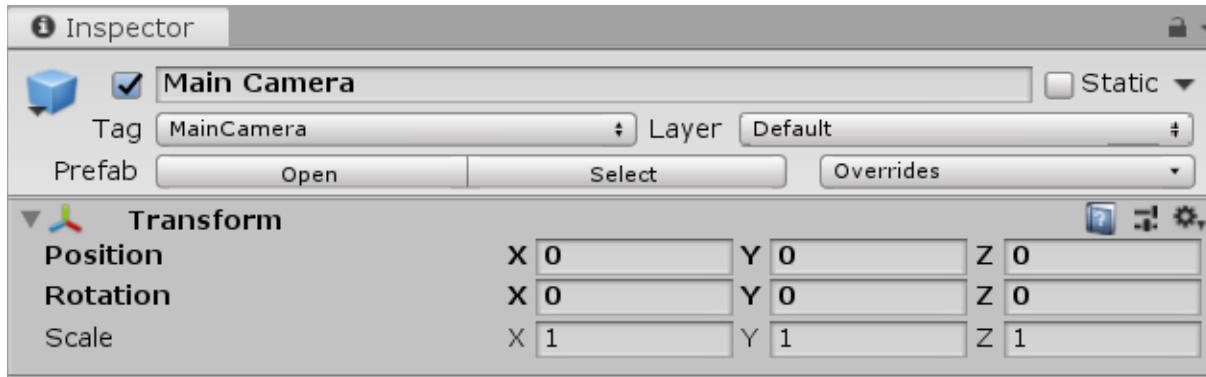
If you have not done so, make sure that your enabled camera in the scene is the **Main Camera** prefab that comes with the Magic Leap Toolkit. It can be found under **MagicLeap > Core > Prefabs > Main Camera**



Add the Main Camera anywhere to your Scene Hierarchy.

STEP 3: Position the Main Camera at the origin

This is very important as the debug console will not attach properly to the main camera if its position is not set to [0, 0, 0].



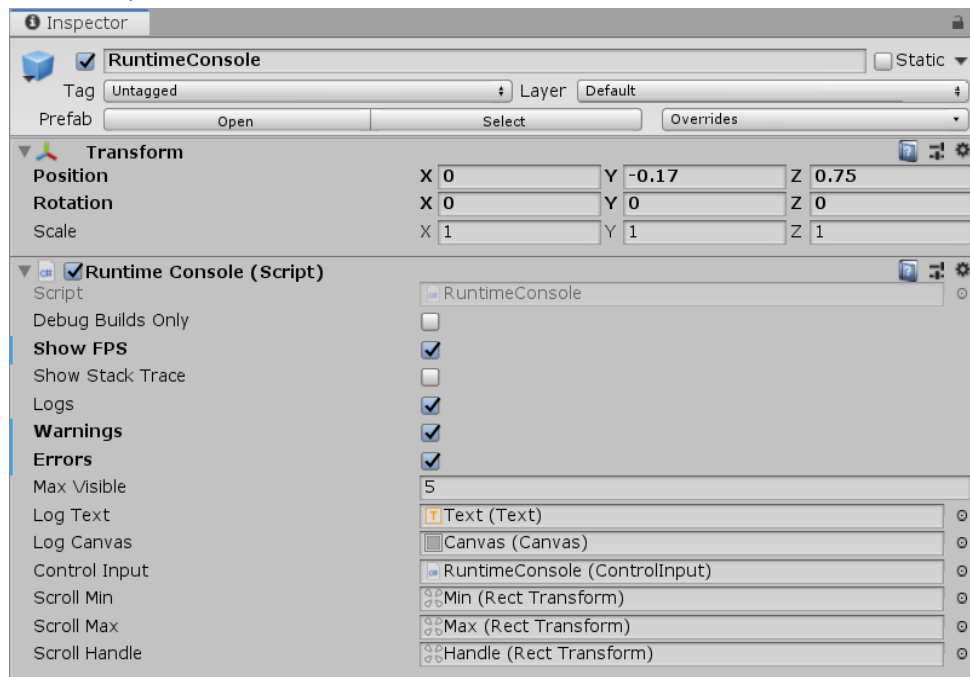
STEP 4: Use the Debug.Log() command to print to the console!

Now you are ready to use the Debug.Log() command anywhere in your project scripts and see the log appear in your runtime console.

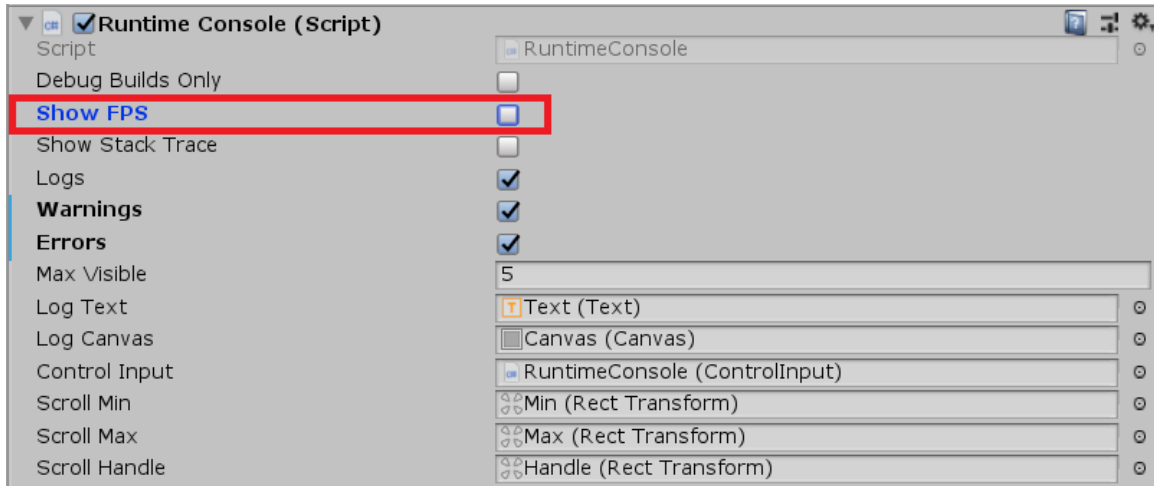
```
// Start is called before the first frame update
void Start()
{
    Debug.Log("Runtime Console is working now!");
}
```

STEP 5: Adjust Console Settings

You can adjust console settings by clicking on the Runtime Console prefab and editing the settings under *Runtime Console (Script)*

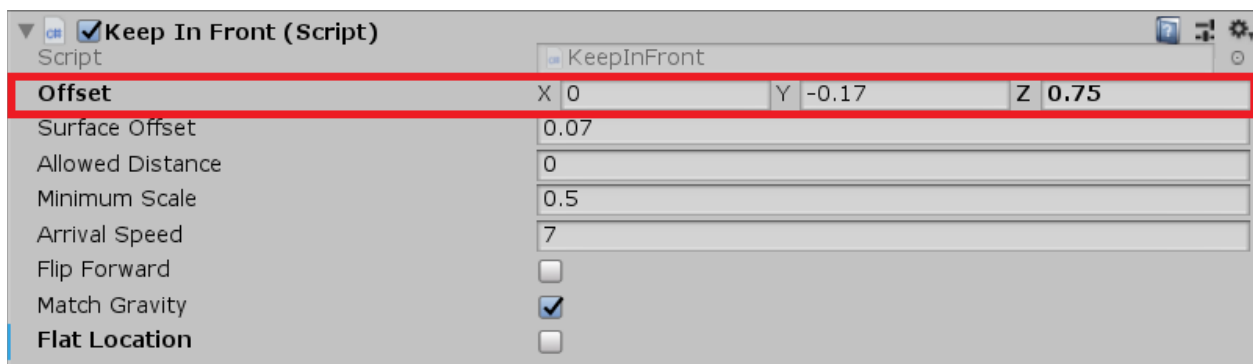


At runtime, an Average FPS log will appear every second and this may cause inconvenience if you are trying to debug your code and the Average FPS log is continuously moving your logs out of the way. If you don't need to know the average FPS just turn off the **Show FPS** setting as shown below.



You can also turn off **Warnings** and **Errors** in the Warnings and Errors section of the Runtime Console Component.

To adjust the position and rotation of the console, go to the *Keep In Front* component of the Runtime Console Prefab and edit the **Offset** section.



Adjusting the Y component of the position moves the runtime console either up or down. The X component moves it horizontally and the Z component will move the console further away or closer to you.