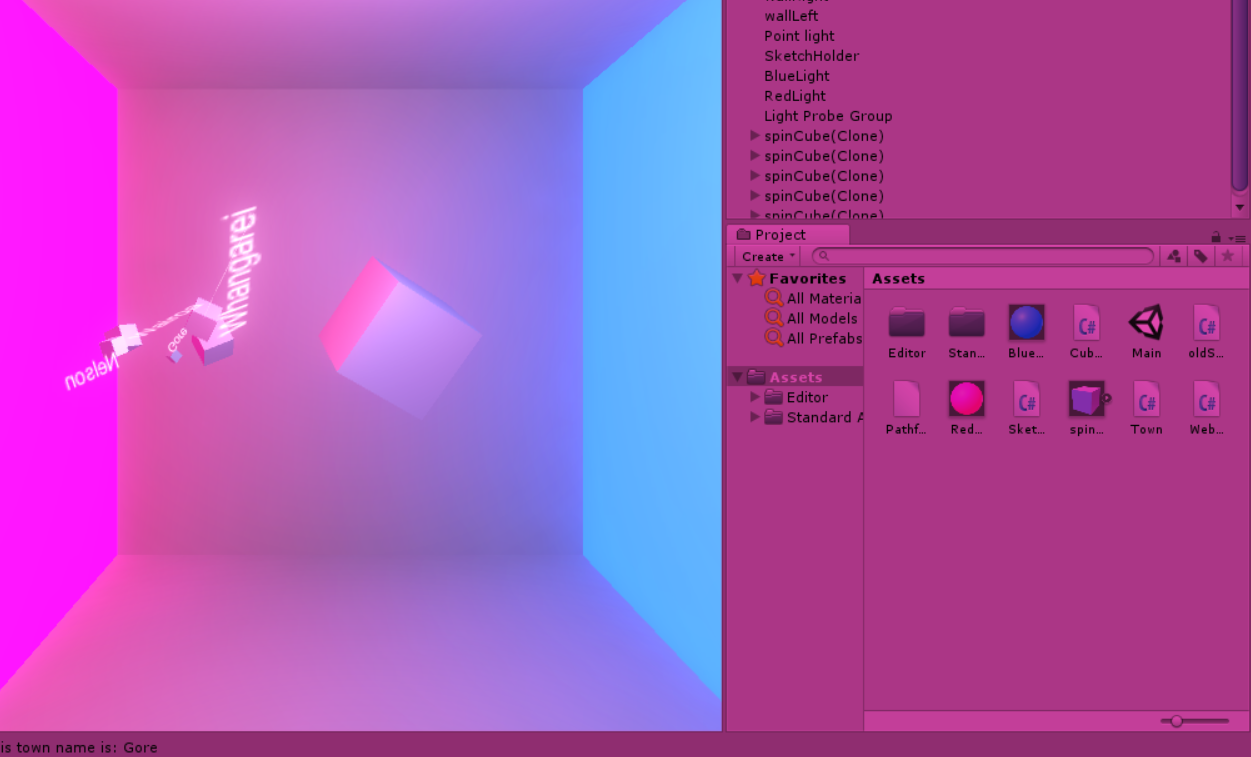
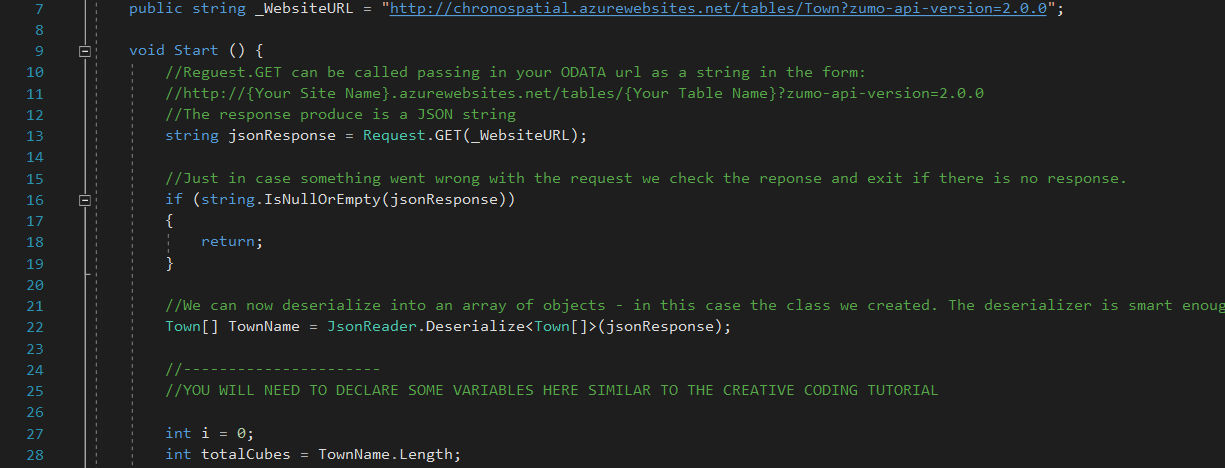
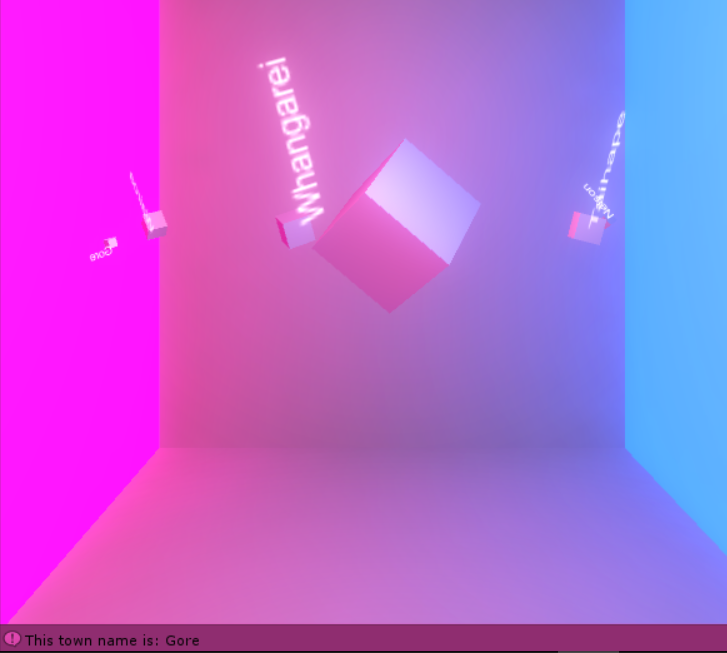
**Github URL:** <https://github.com/slia356/Lab-Test-2>

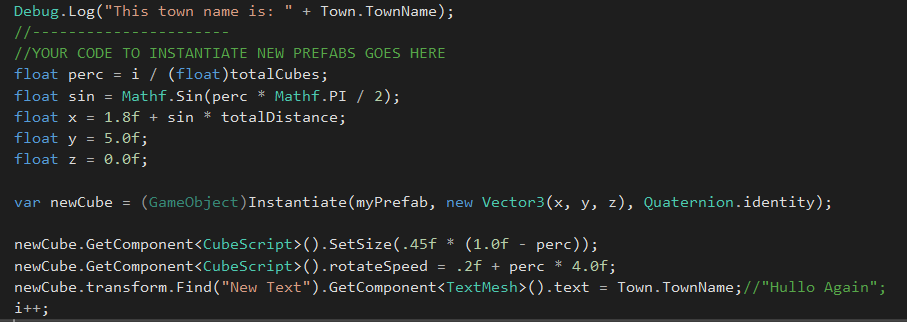
1. **Create a spinning cube for each Town**





1. **Attach the TownName to it**

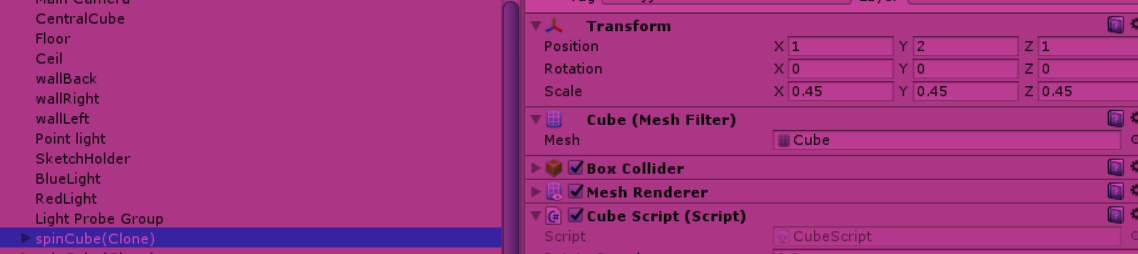


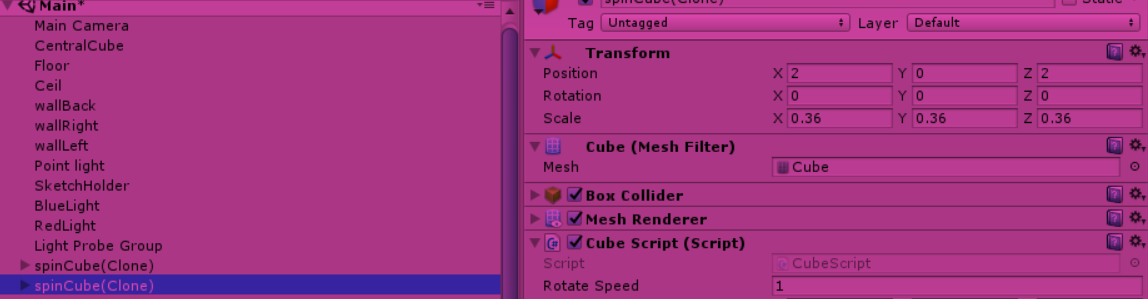


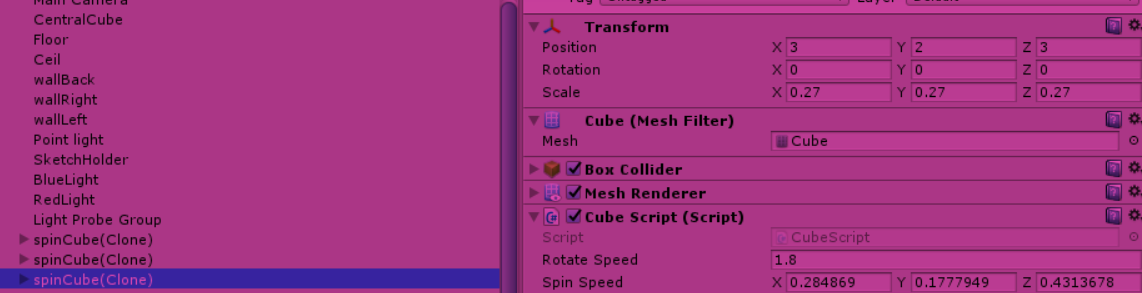
1. **Place Cubes at their Latitude, Altitude, Longitude coordinates**

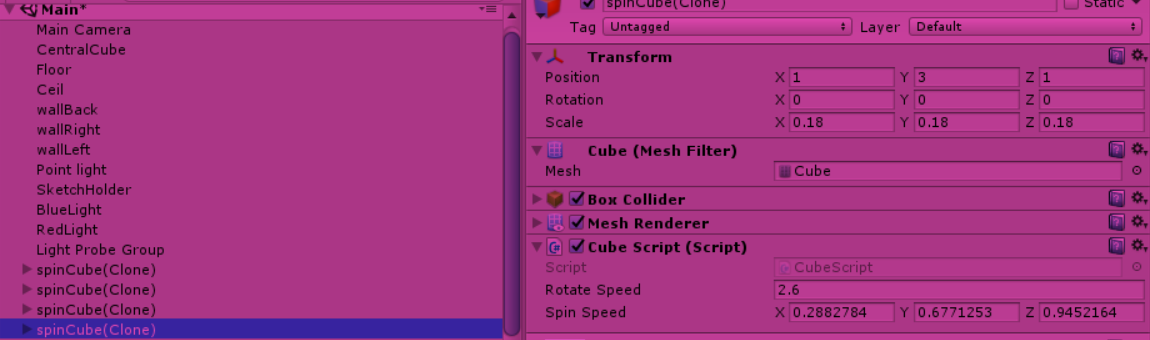


This screenshot is bit vague, will put the details (x,y,z value) of each town (cube) from inspector below:





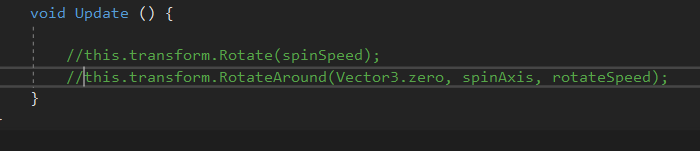






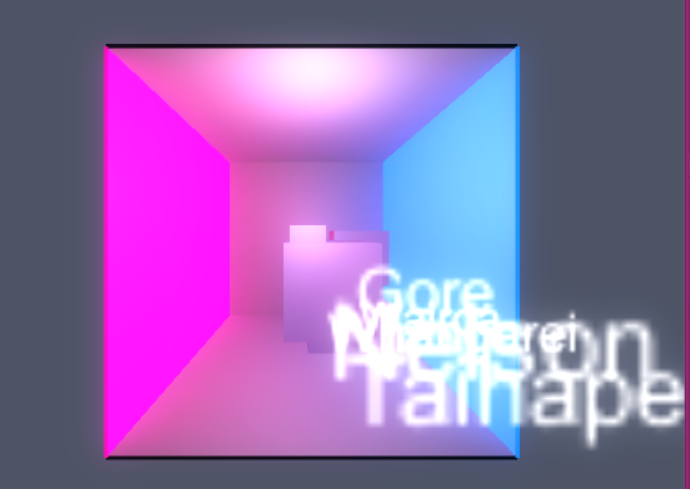
1. **Stop them spinning**

I did this by commenting out the spinning function from Cube Script:

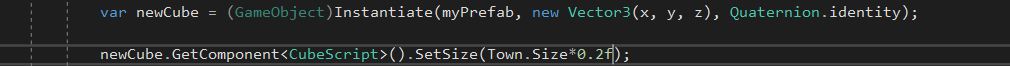




1. **Set the size of each cube**



However, the actual size for each town is quite big, so to show a clearer layout, I timed 0.2f to the actual size to make it smaller:



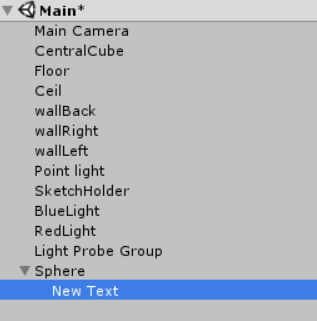
And now it looks like:

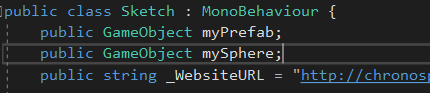


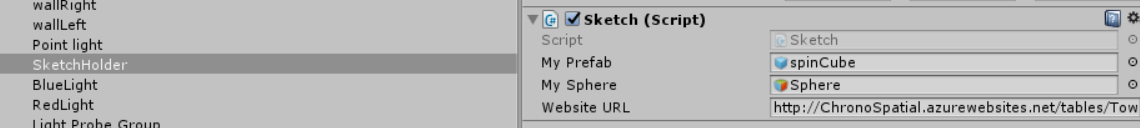
Looks better 😊

1. **Set Prefab based on “Symbol” value**

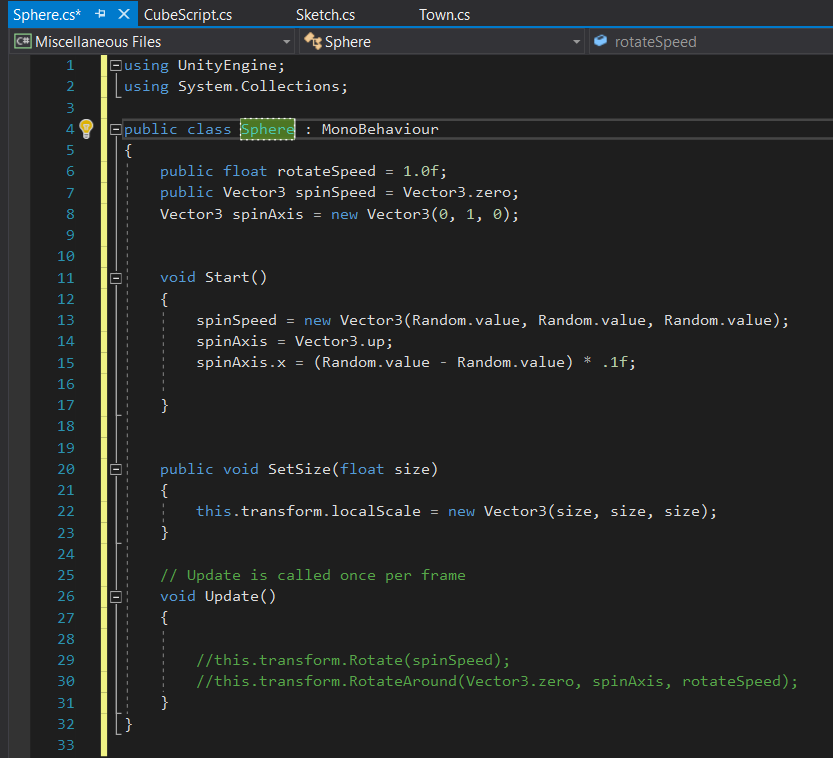
Create a new prefab ‘Sphere’ in hierarchy and add 3D text onto that.



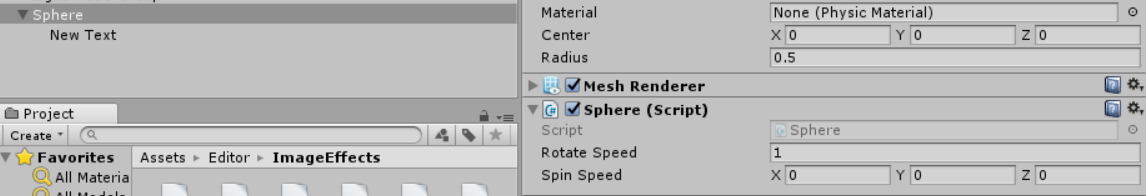


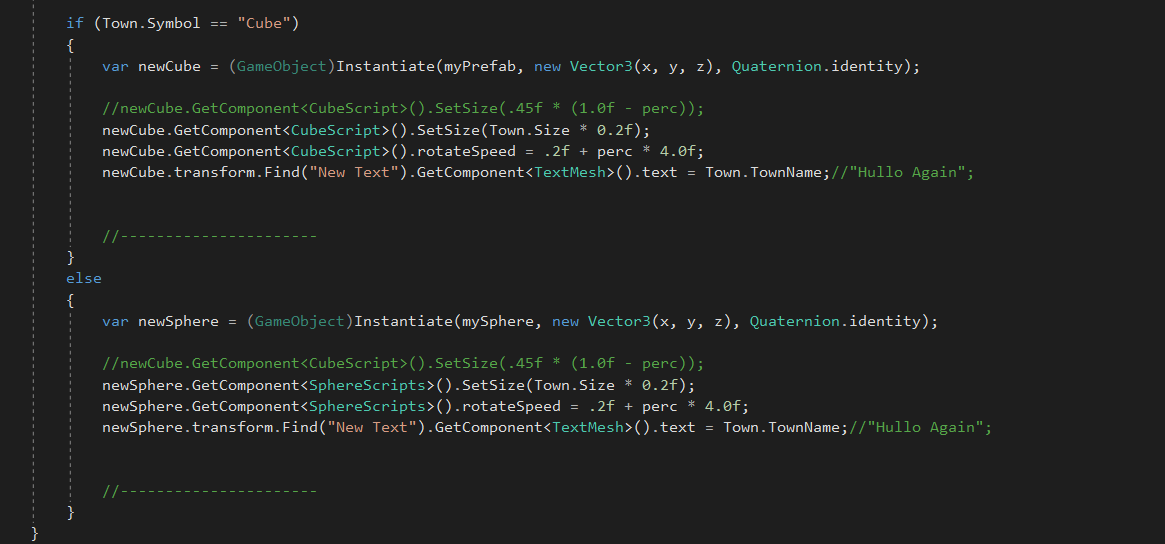


Then create the Sphere c# script file (when copy n paste from cube script, make sure to change the class):



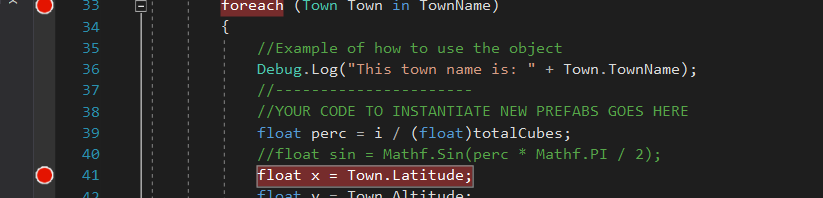
Attach the Sphere Script onto the prefab:







1. **Set a breakpoint and examine values in array of Towns**



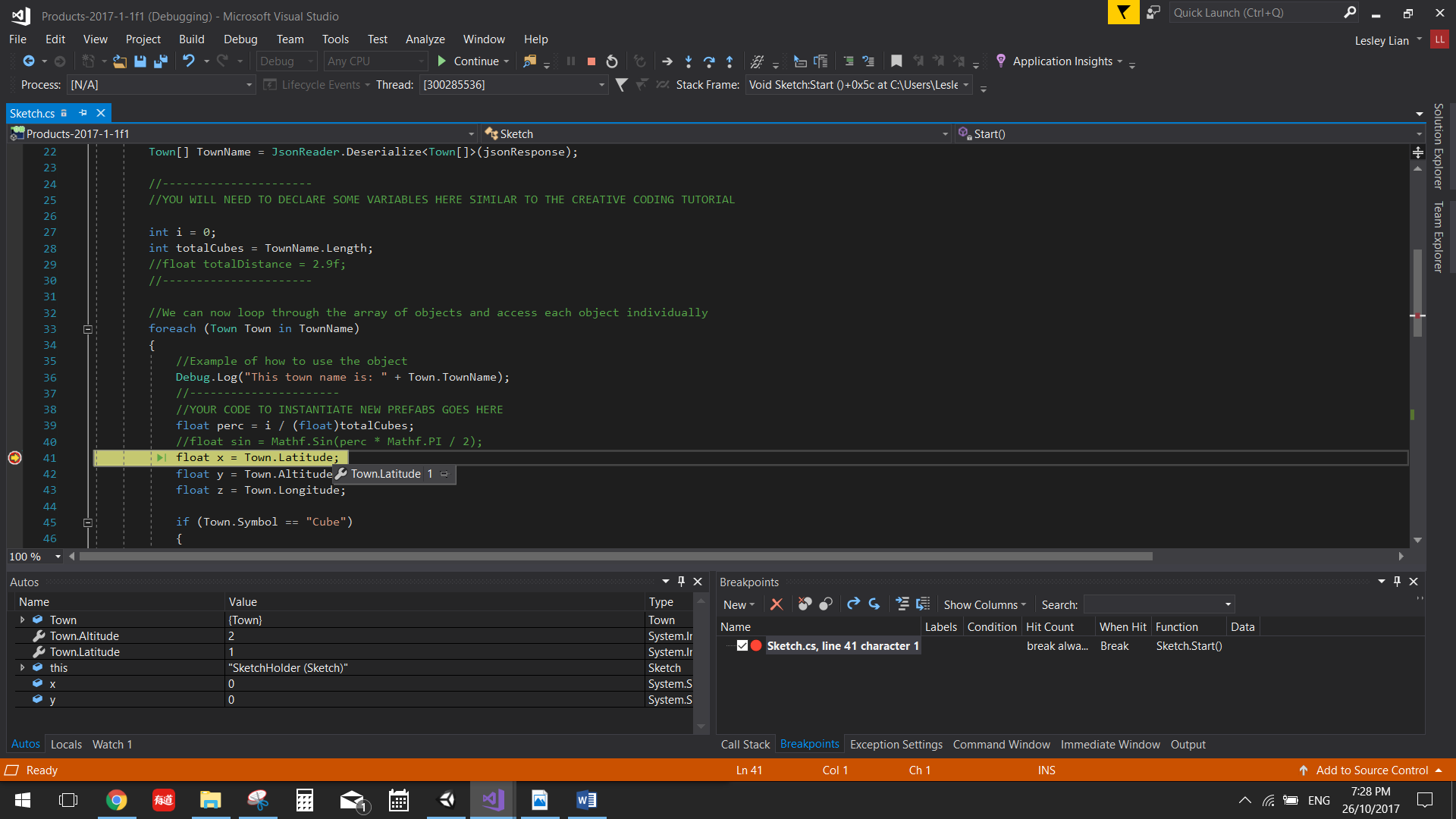
Then play the debug



Then run the unity



Then it will back to visual studio and display the info/details on the code, to tell us how codes function:



1. **Make spheres a different colour to cubes**

Let’s make spheres in red

