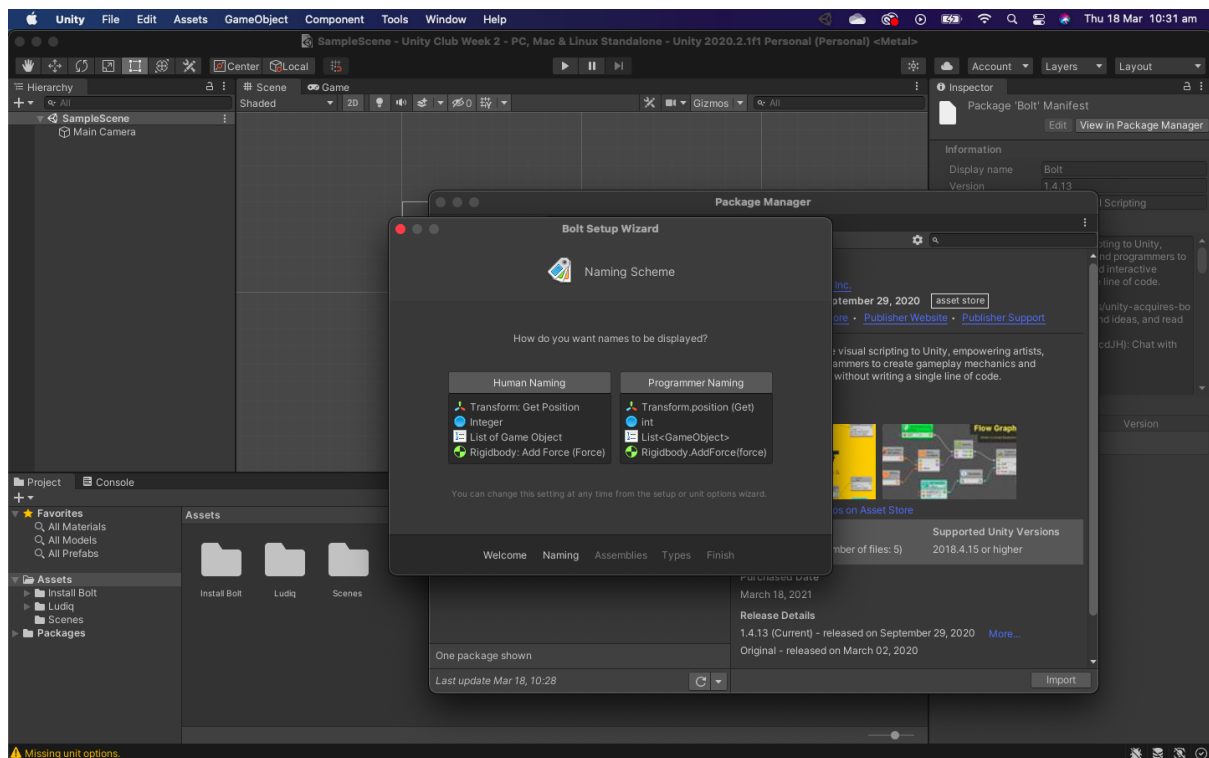Week 4

The lack of tutorials in Bolt has proven to be a struggle for me these past weeks. To be honest, there are moments when I wonder choosing to use Bolt is a good idea. This thing really scares me yo, what if I fail?!

But NO, the show must go on

It might sound stupid to hear that I chose the Programmer option because why the hell don't you just use C# amirite? BUT (I'm really trying to reason my decision here), after using Bolt all this time, I think that it is much easier for Beginner to troubleshoot if anything happens because you don't need to absolutely understand what is going on. While in C#, you are done for if you don't know how this comma work. On the other hand, I don't think that it would be easier for a beginner with 0 coding experience due to the lack of tutorials

Okay, enough with the chitty chat, here are my hours and hours in Bolt

The characters need to move around in top-down style

Pretty much the easiest one because of the availability of a very straightforward tutorial. I think that the graph down here are quite telling themselves, most people will be able to understand what is happening
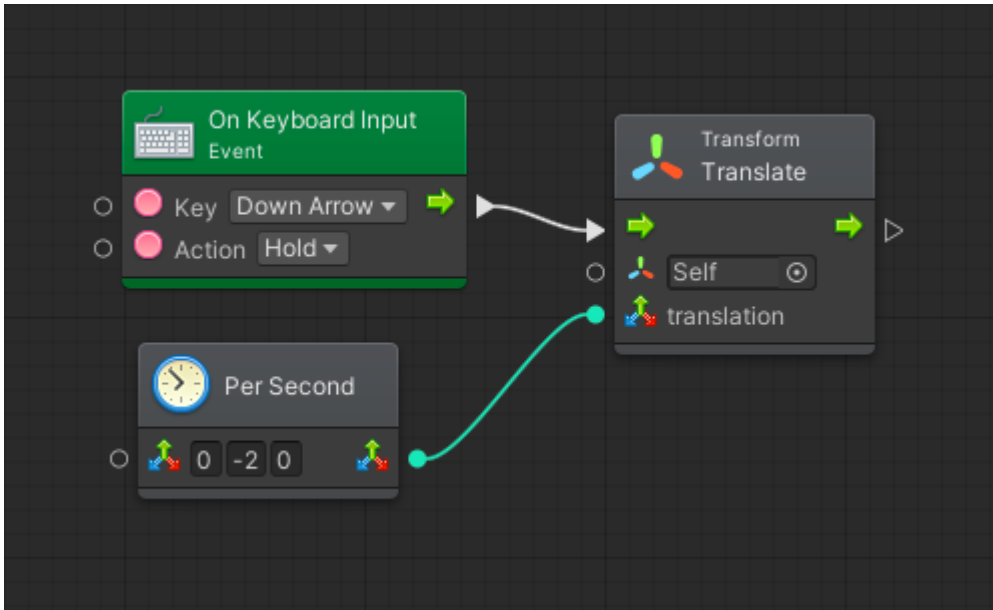
Instead of just putting a bunch of nodes inside a Flow Chart, I created a bunch of SuperUnit (basically Flow Chart that can be dragged in other Flow Chart). It is a very efficient way to clean things up
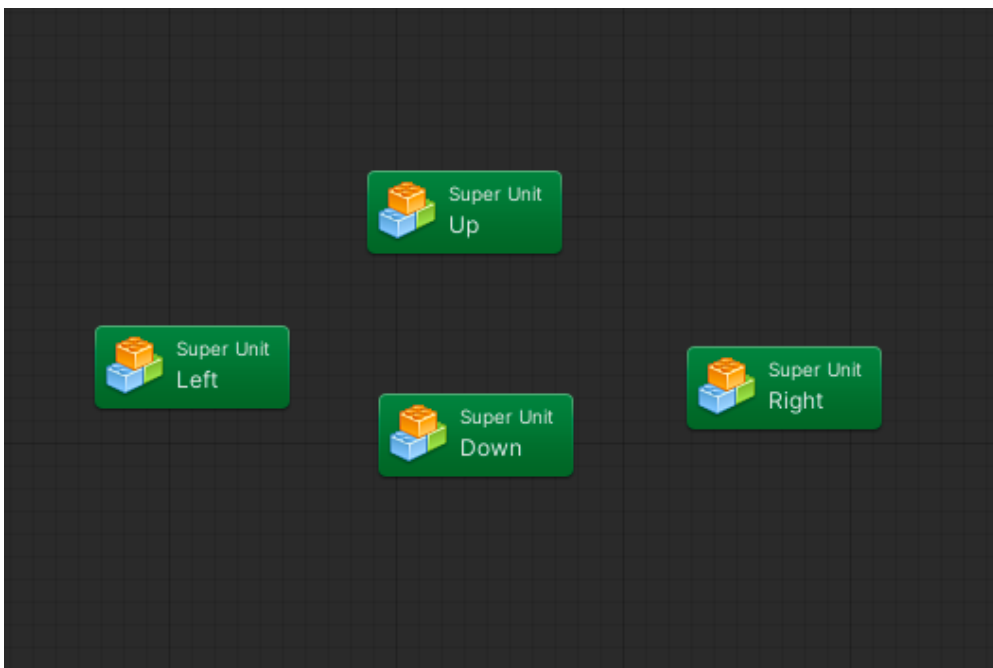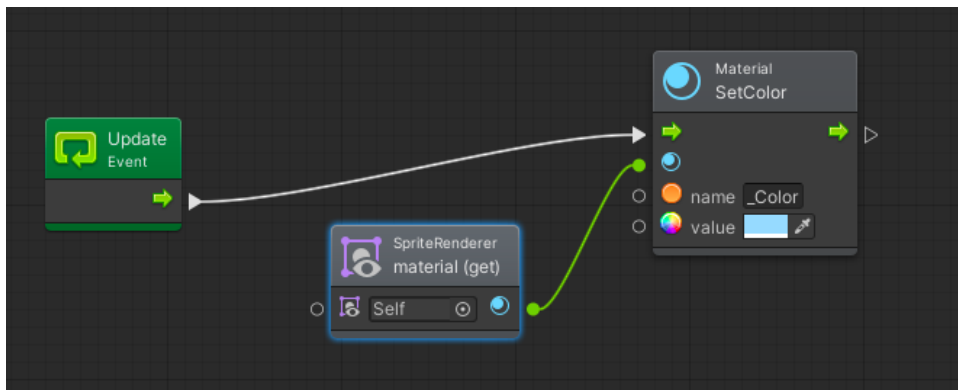

Please refer to the video "OnCollisionChangeColor" to view the full experiment

When the main character goes to another room, the colors of the rooms need to switch

You might wonder why I didn't just jump in and search for something like "Switch color in Unity" and be done with it or something. As a beginner, I find it easier to start with smaller things first instead of just staring at a big code chart or something. Also, there is no such tutorial in C#, let alone in Bolt. There is not even 1 simple setColor tutorial in Bolt. So most of these things is me clicking around, or drafting nodes I found from tutorials
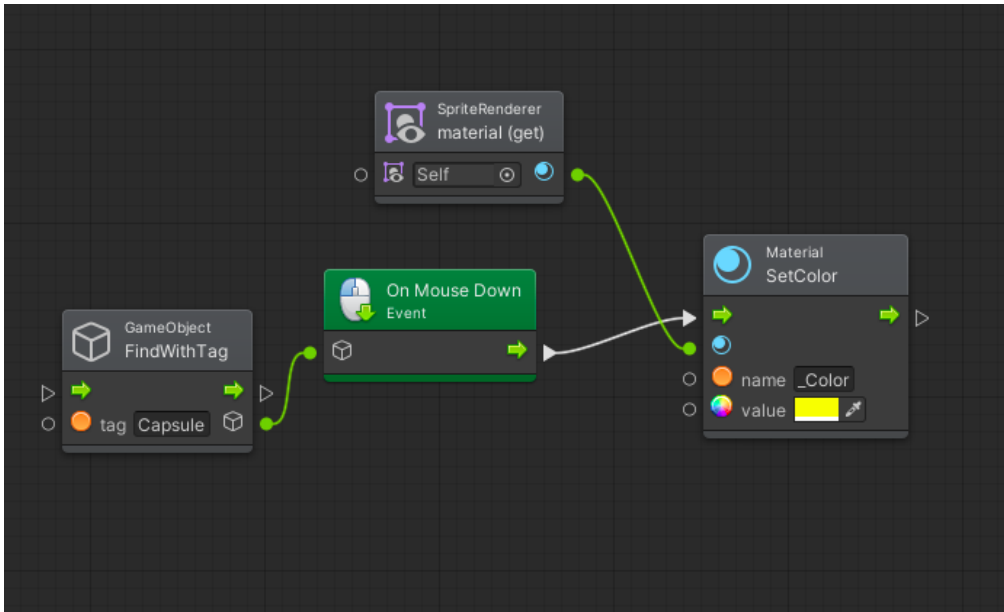
Change the color of objects

Well looking at this now, it is quite telling itself. You guys won't understand how much time I spent on this simple thing

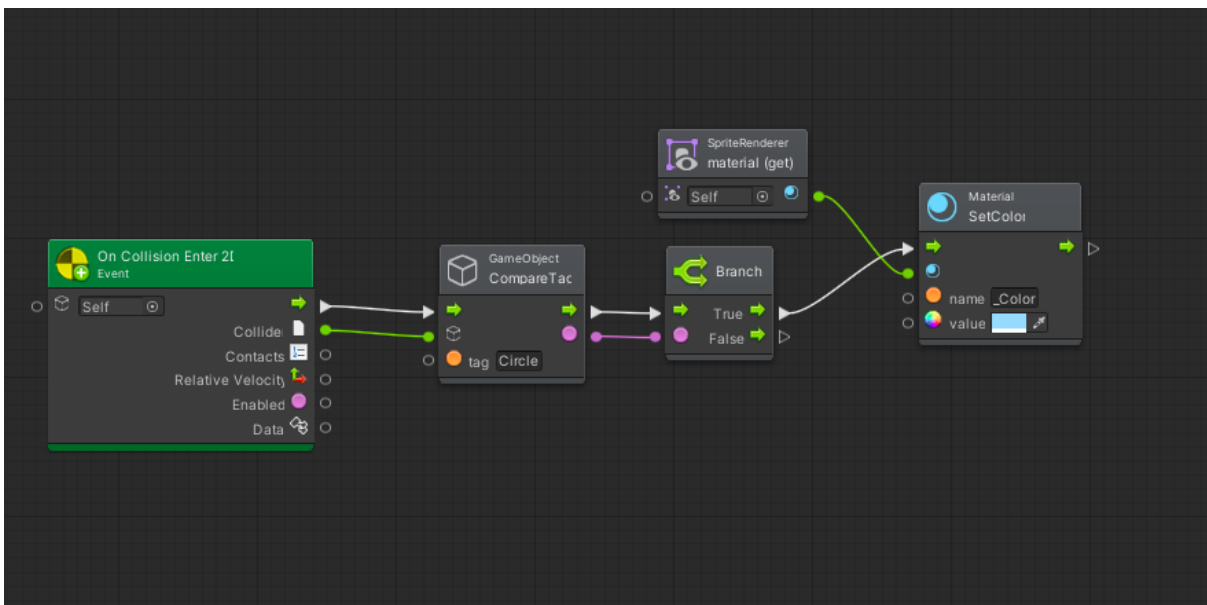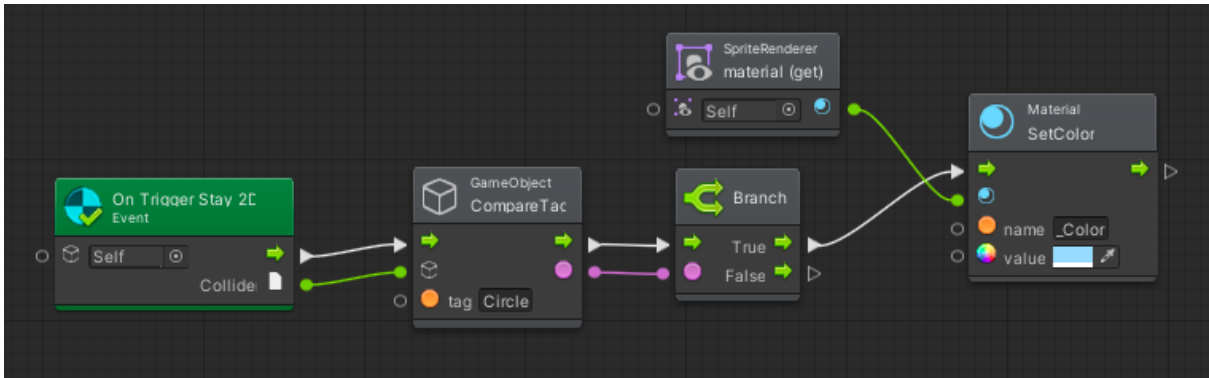If you guys want to turn the above into Super Unit, here is how you do it
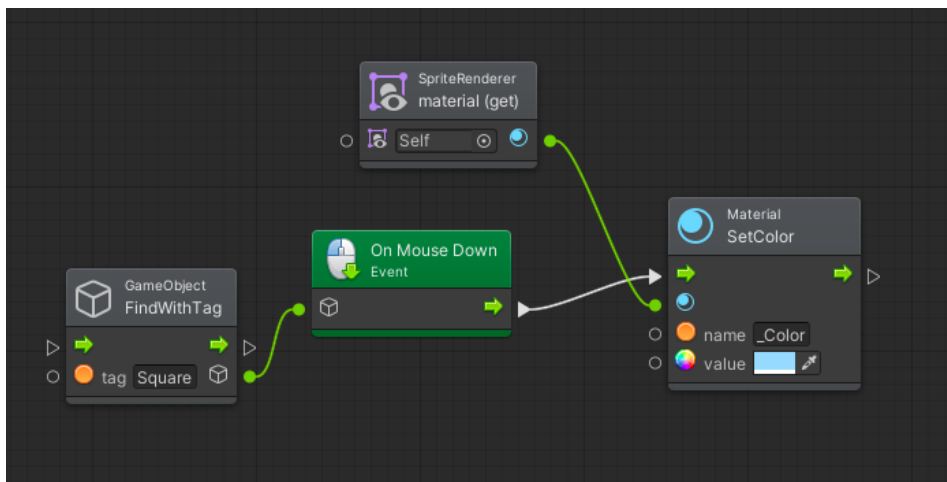
Each of the graphs above is calling a different object

Please refer to the video "SetColorWhenPlay" to view the full experiment

Use a trigger to change color

Note to myself, when a Player meet An Object: Object [Collider [is Trigger]] + no
Rigidbody; Player [Collider [no "is Trigger"]] + Rigidbody + Gravity = 0

This CompareTag is a little bit tricky. So in my test, I name my player "Circle" and my room "Square" (due to the way I lazily draw them). And I'm going to explain how I do this in details (because I might forget how I did it in the future so I need to write it down
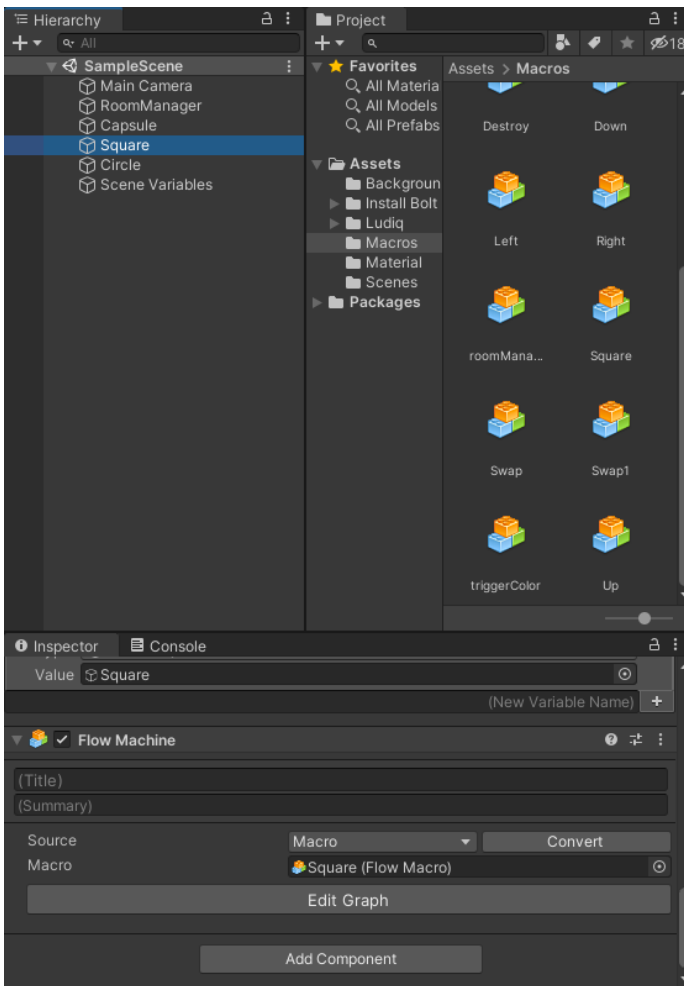
In this picture, you can see that I'm selecting my Square in the Hierarchy. And in the Square, there is a Flow Machine component. Moving down please
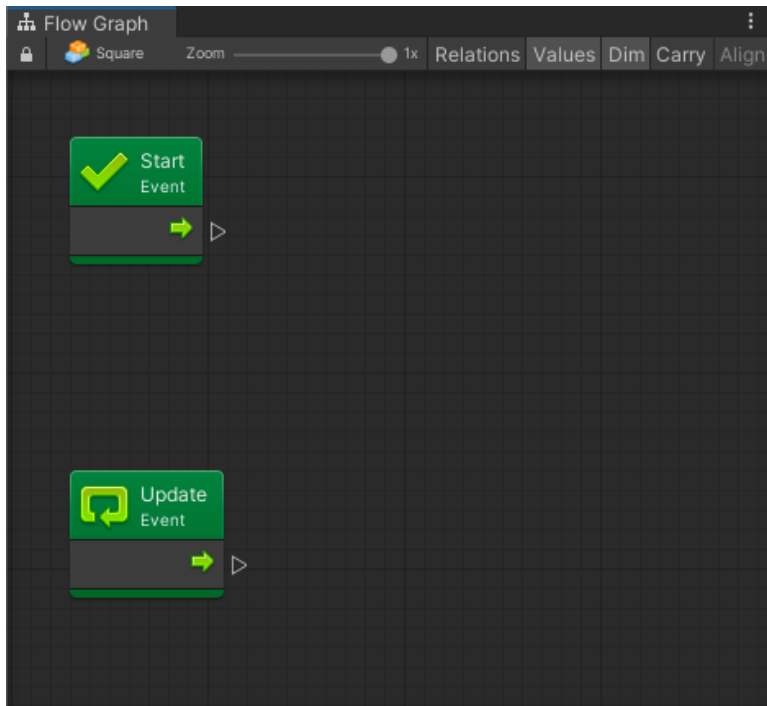
Here is how that component looks like. Look at the top-left, you can see the name of the Flow Chart / Super Unit
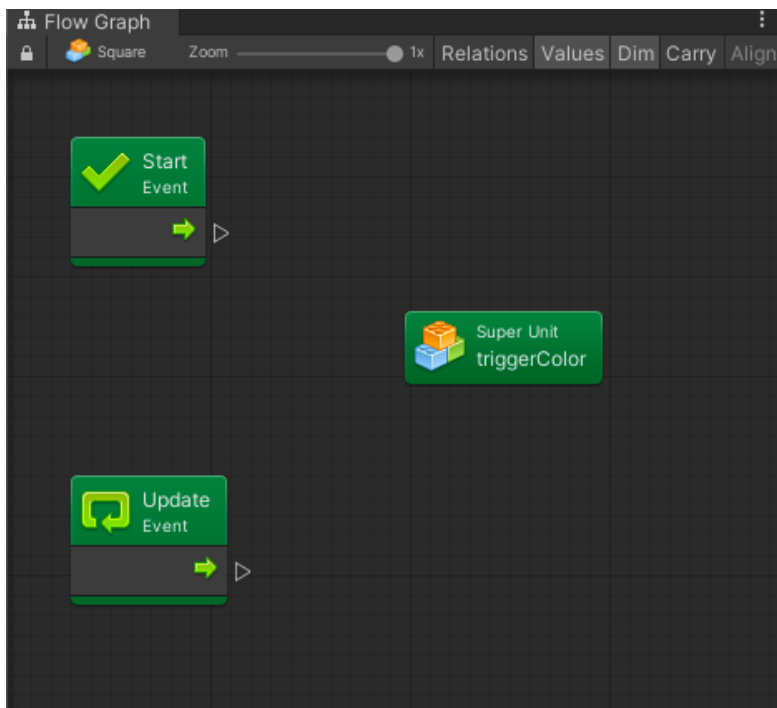
In order for the script to work, you will need to drag the Super Unit responsible for the function into the Flow Chart of the object.

Does that make sense? I'm sorry I'm never good at explaining. I can't even explain in my own native language

Please refer to the videos "OnCollisionChangeColor" and "OnTriggerChangeColor" to view the full experiment

Also, please view to the video "OnClickSetColor" and "OnClickChangeColoSpecial"

Switch

Now let's look at what we already have. I know how to Set Color to objects in Bolt, I know how to trigger an event in Bolt, now I just need to know how to switch.

Sound easy right? I can do this

Looking down here, this is how an enum was created in Bolt. So you will need to create a new empty object name RoomManager, and you add a variable like this

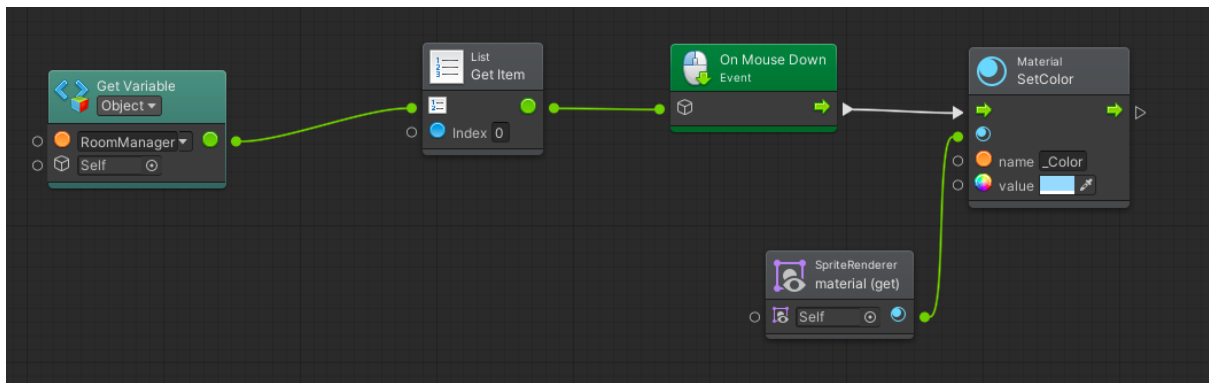And of course, you need to have a new Macros attached to it
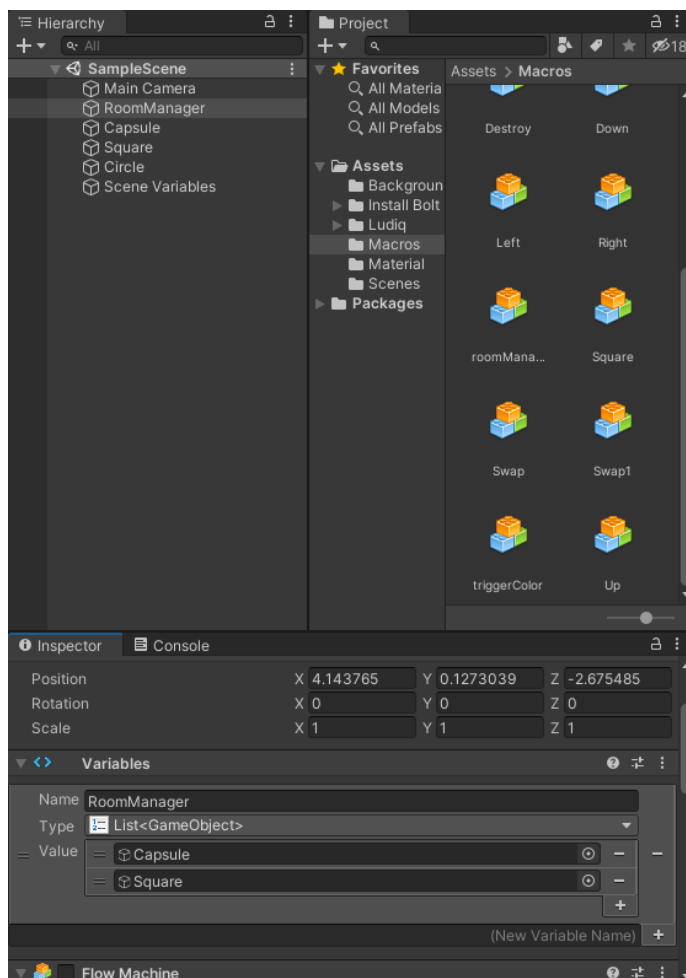
Moving down

Look at the graph, down here, this is how you call an item in a list. Strangely enough, you need to have 2 separate Variables nodes, calling the items from the same nodes will result in nothing
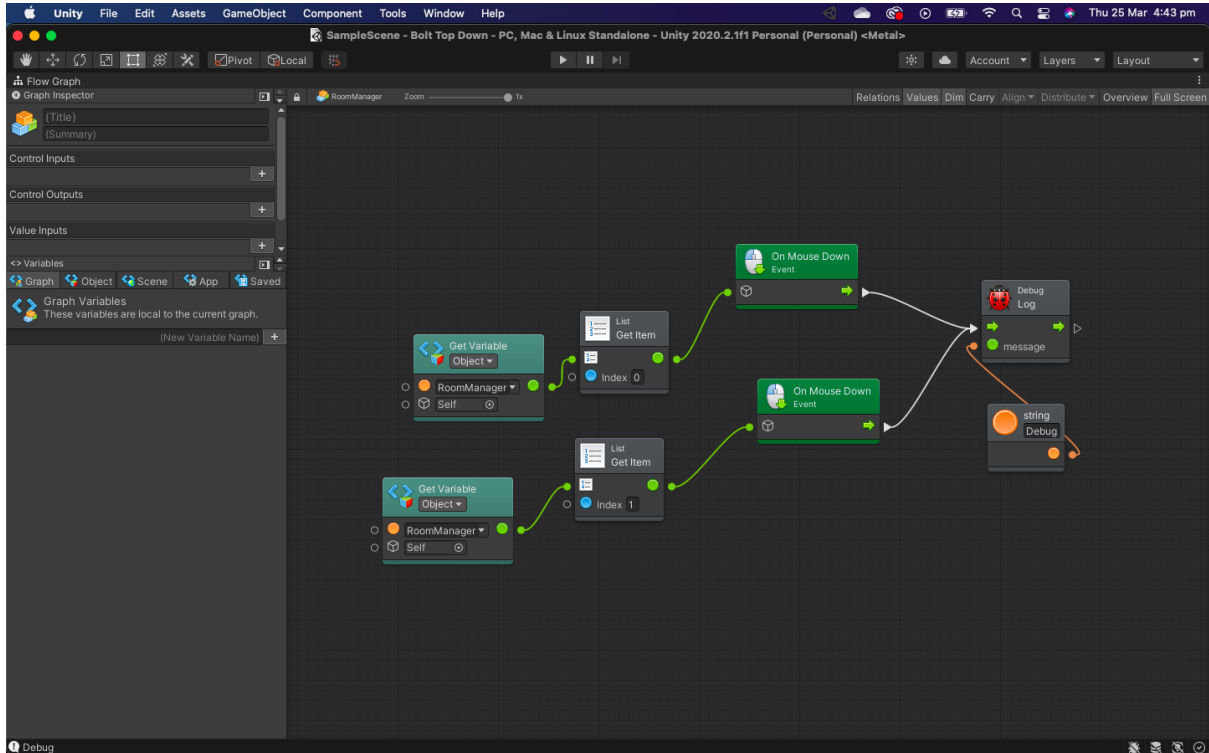
Now, this Flow Chart above works. Please refer to the full video in order to see the full experience

But this Flow Chart, for some reasons, doesn't work. Even though logically, it must works. When you look at how I set the color above, you will see that it is supposed to work.

If I can figure out what's wrong with this Flow Chart, I will be able to switch the color between 2 objects by using an if statement (Branch in Bolt)

I'm almost there, I just need time

Please refer to the video "GetItem" to view the full experiment

Switch color AND switch color on top of that

This sentence doesn't make any sense, I know. How can I explain it again?
So what I want to do is to swap 2 things. I kind of know the logic in C#, again, there is no tutorial in Bolt, so I will need to translate the logic to Bolt.

- CurrentRoom = NewRoom
- NewRoom = CurrentRoom

So we already swap the color of the room right? Now I will need to renew it so that where the character is standing will be the CurrentRoom, and the next room the character move to will be the NewRoom

This is something that I will need to figure out after I solved my current problem with the SetItemSetColor graph

The enemy kills the main character when she enters the room

Here is how I make it so that 1 object can destroy the other
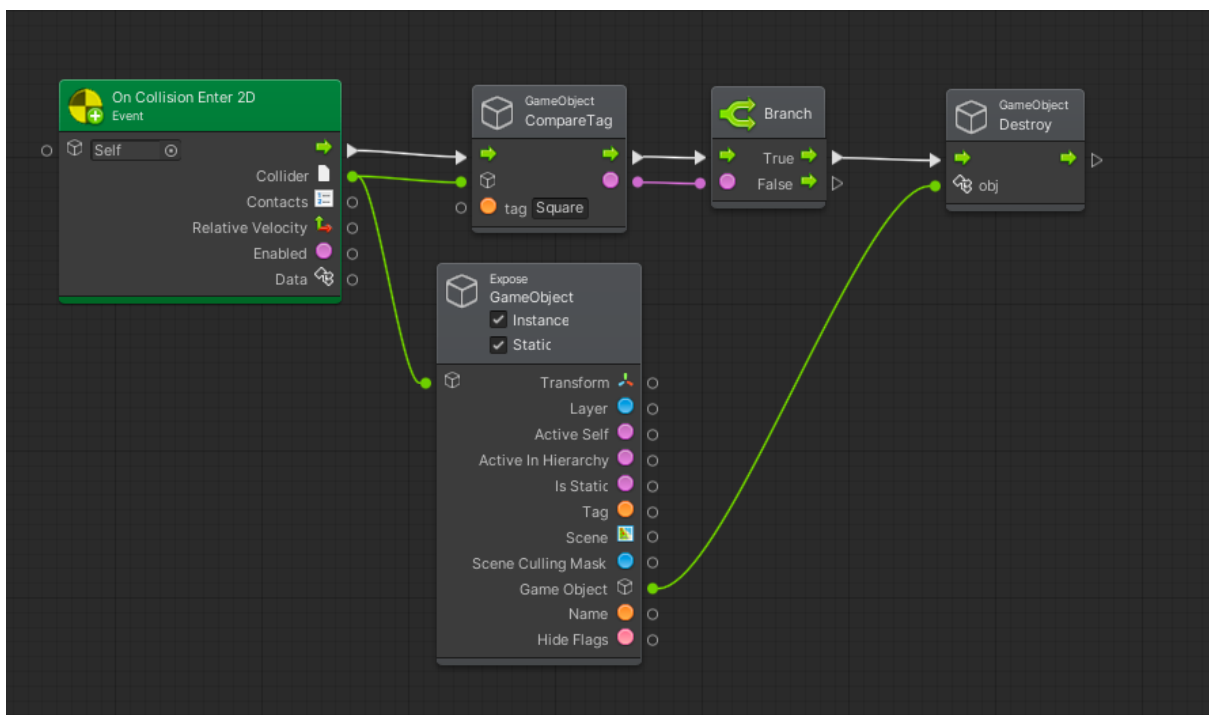
Please refer to the video "OnCollisionDestroy" to view the full experiment

The main character wins when she reaches the portal

Conclusion

One of the reasons why I create this document is that I need to sort out all of my logic. After hours and hours of staring at Unity, things are starting to go all over the place in my head. Hopefully, this will help me figure out why I'm stuck

I also want to talk a little bit about my timeline. The hardest part of my project is to create the color switching mechanism. There are several reasons why: I just begin to use Unity, there is no tutorial, and so on. So I think the moment I figure this out, I will be able to rush the rest (like enemy controller, there are a lot of tutorials on this). My aim is to finish this low fidelity prototype before the start of Week 6, or at the very least, I give myself until the start of Week 7. Then after that, I will use the rest of my time to finish the assets