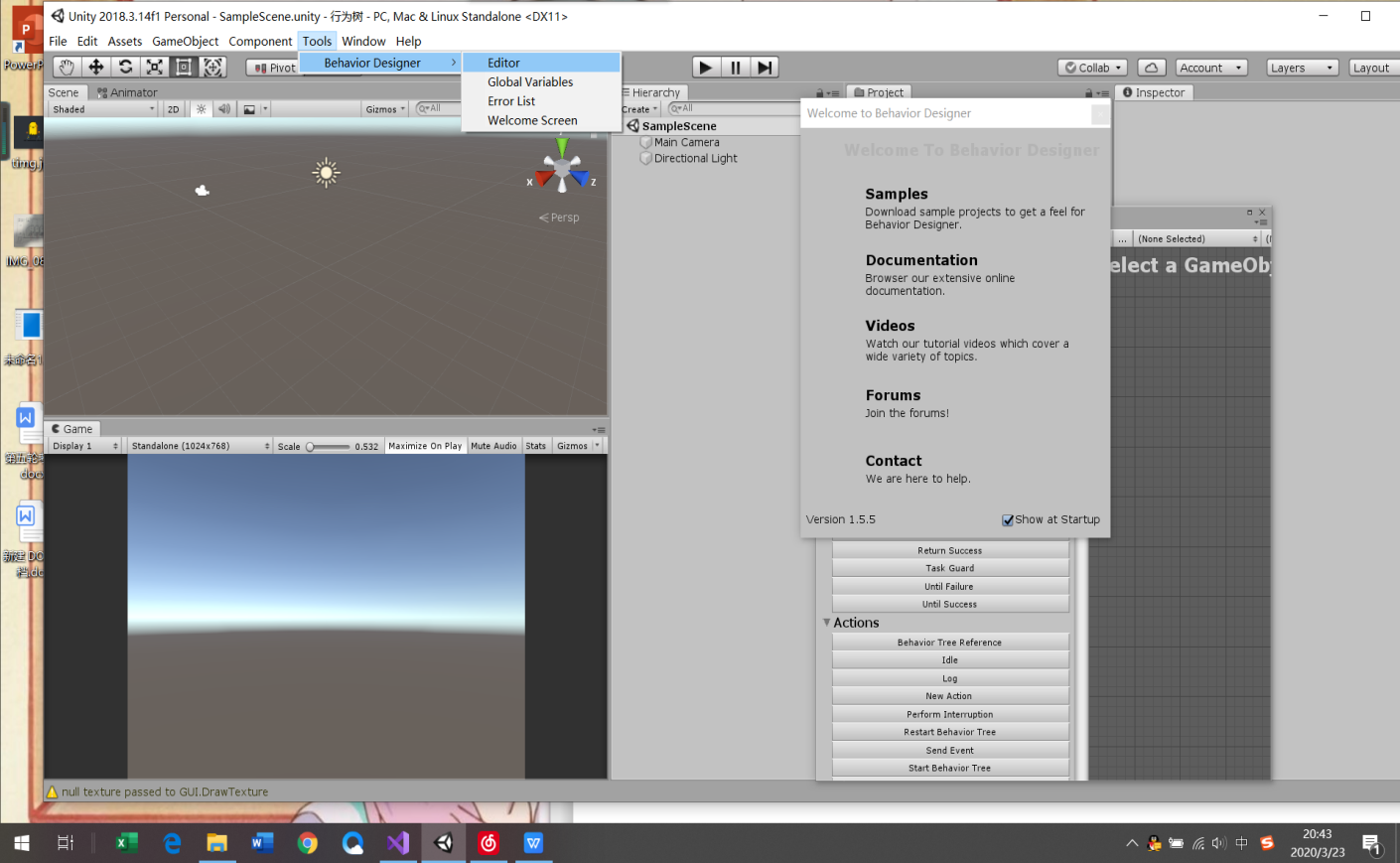
<https://www.cnblogs.com/Dearmyh/p/9316249.html> 博客

<https://www.bilibili.com/video/BV1Kb411w7sf/?p=8> B站

Unity的行为树-->Behavior Designer

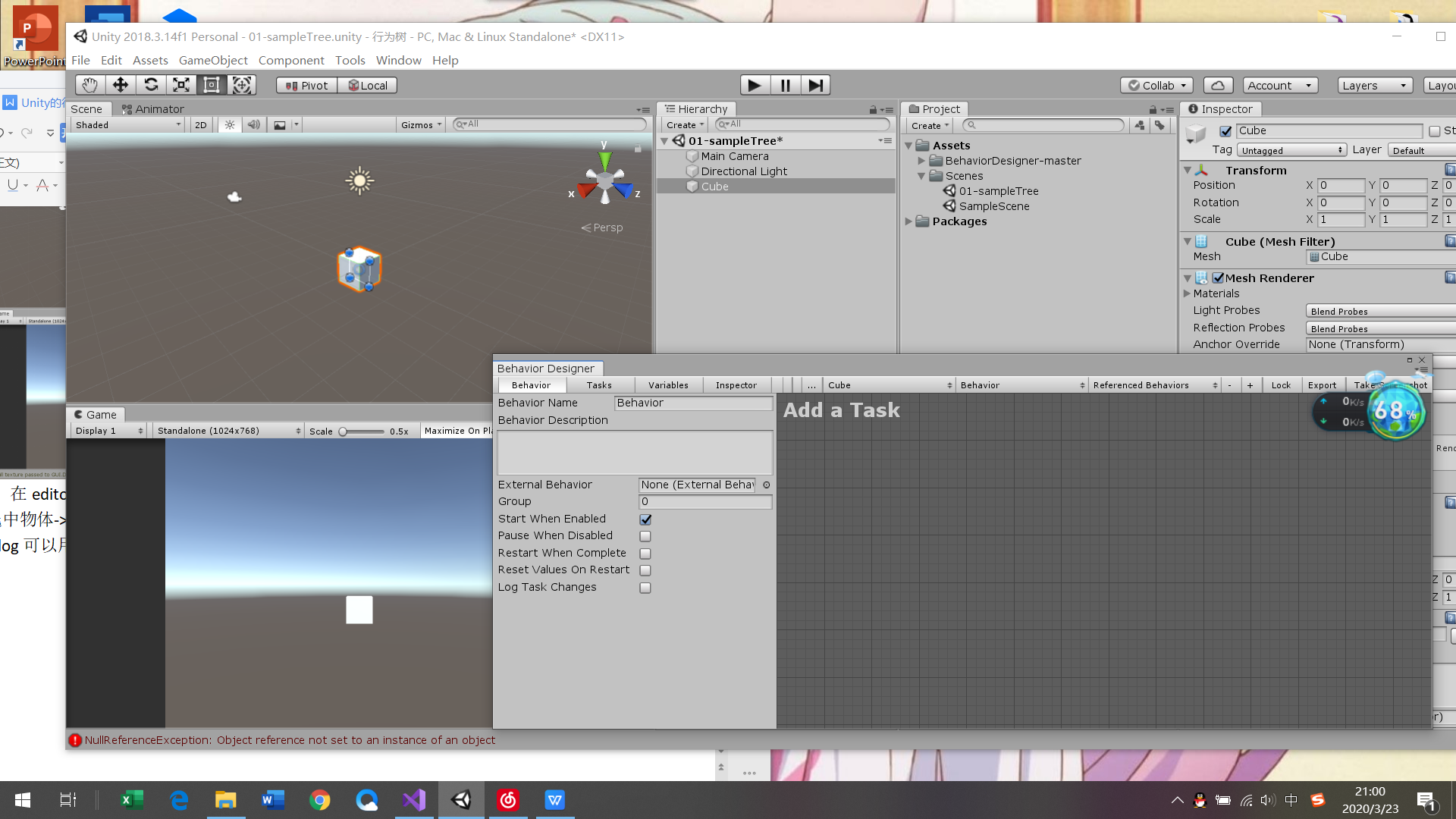
第一课学习;

1.导入Unity的asset文件夹以后上面会多出一个Tools->Behavior Designer->Editor



1. 在editor界面添加行为

选中物体->editor界面右键add BehaviorTree Behavior中可以修改树的描述



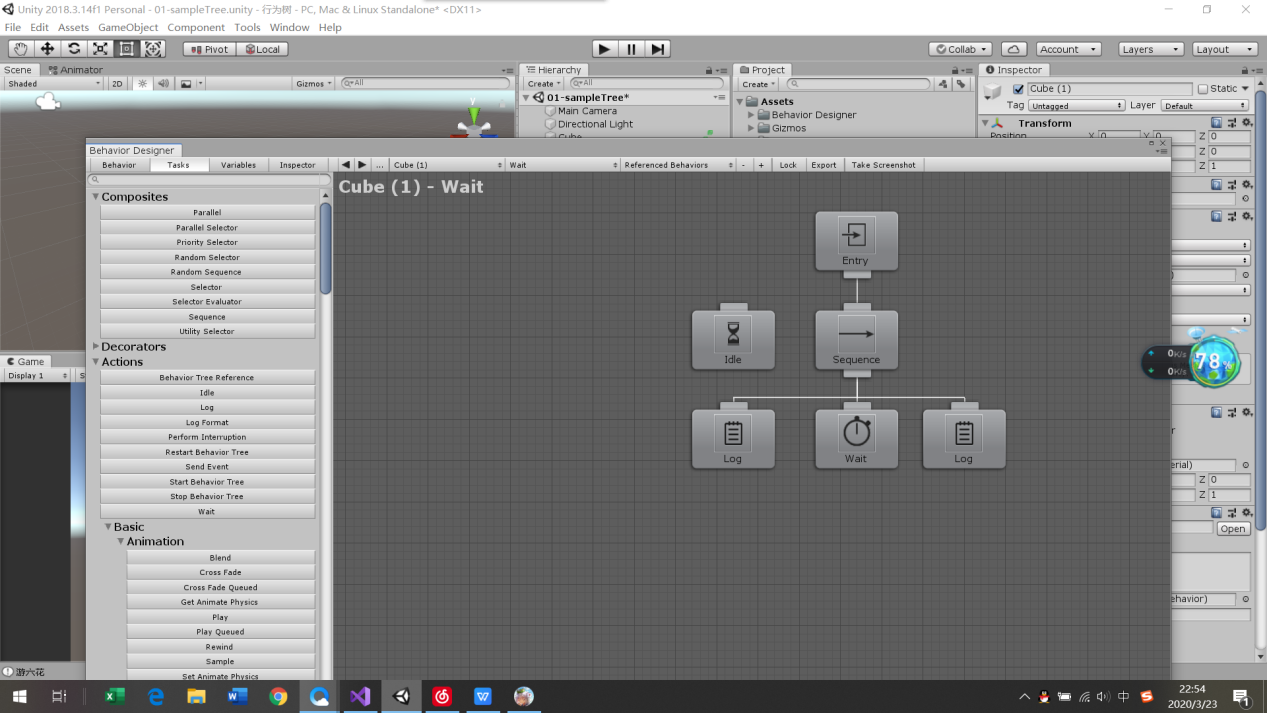
1. log可以用来输出某个东西 在inspector面板里面设定
2. 行为树的分支从左到右运行 越左边的结点优先级越高

第二课学习：

3.1复杂任务（Composites混合） Sequence: 简单来说 就是可以连接多个任务 当且进当所有任务都执行成功且结束时返回正确进入下一步

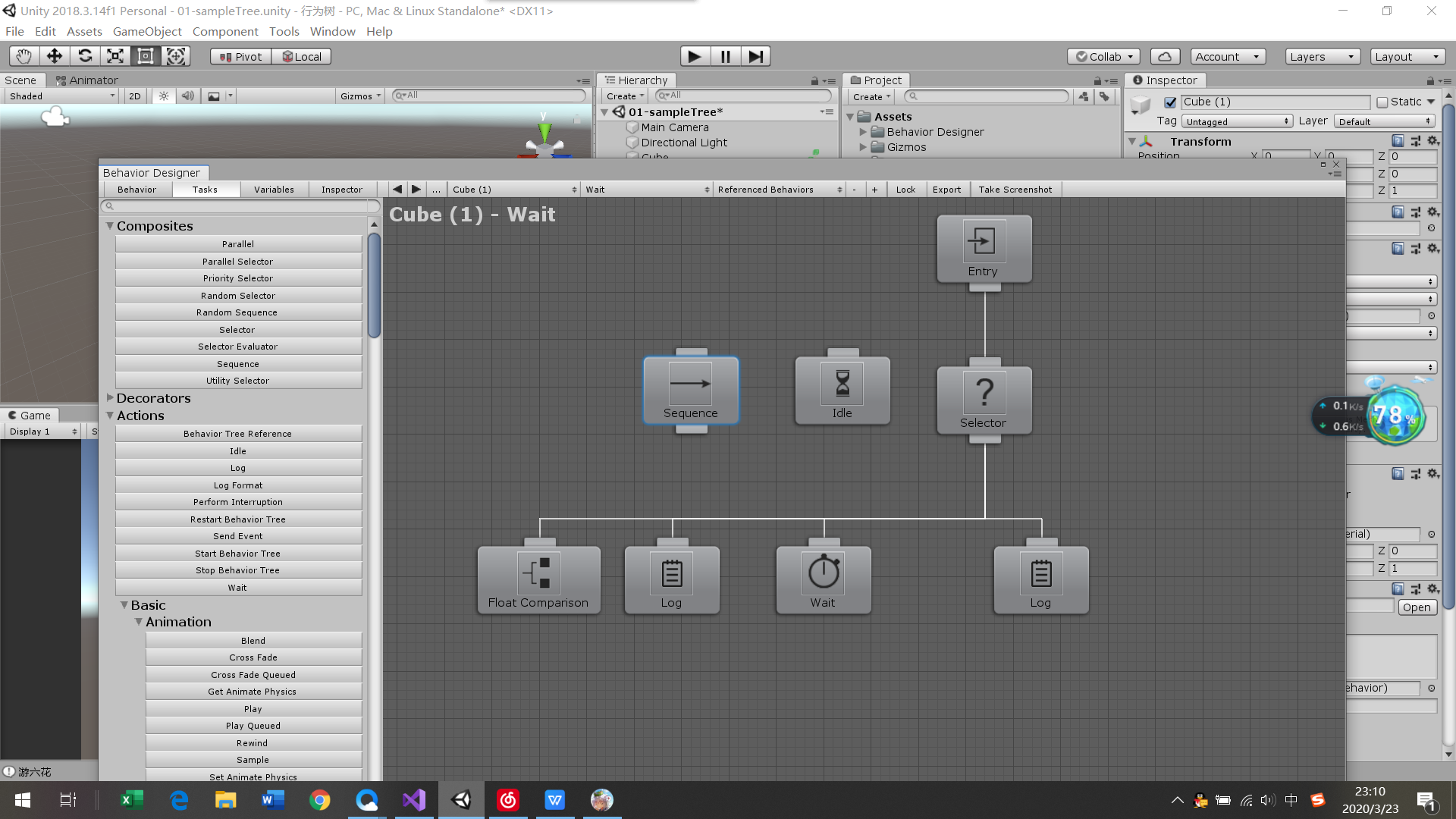
3.2 Idle :除非被打断后或者发生其他特殊情况一直保持running状态

Wait :等待一段时间后返回成功 等待时间内持续running



第三课学习：

3.1 复杂任务（Composites混合）Selector 子任务中有一个成功就会返回成功其他子任务不再执行 相比Sequence就是或



3.2 Float Comparison 对比数值大小 第一个数小于第二个数返回成功

第四课学习：

无

第五课学习：

1.在Variable面板创建变量 然后在inspector面板中点击小圆圈可以将这个变量赋予这个动作里面的对应参数

2.每个行为都有很多不同的赋值的地方 比如物体 坐标 浮点数 等等

3.通过selector和sequence就足够实现很多复杂的逻辑

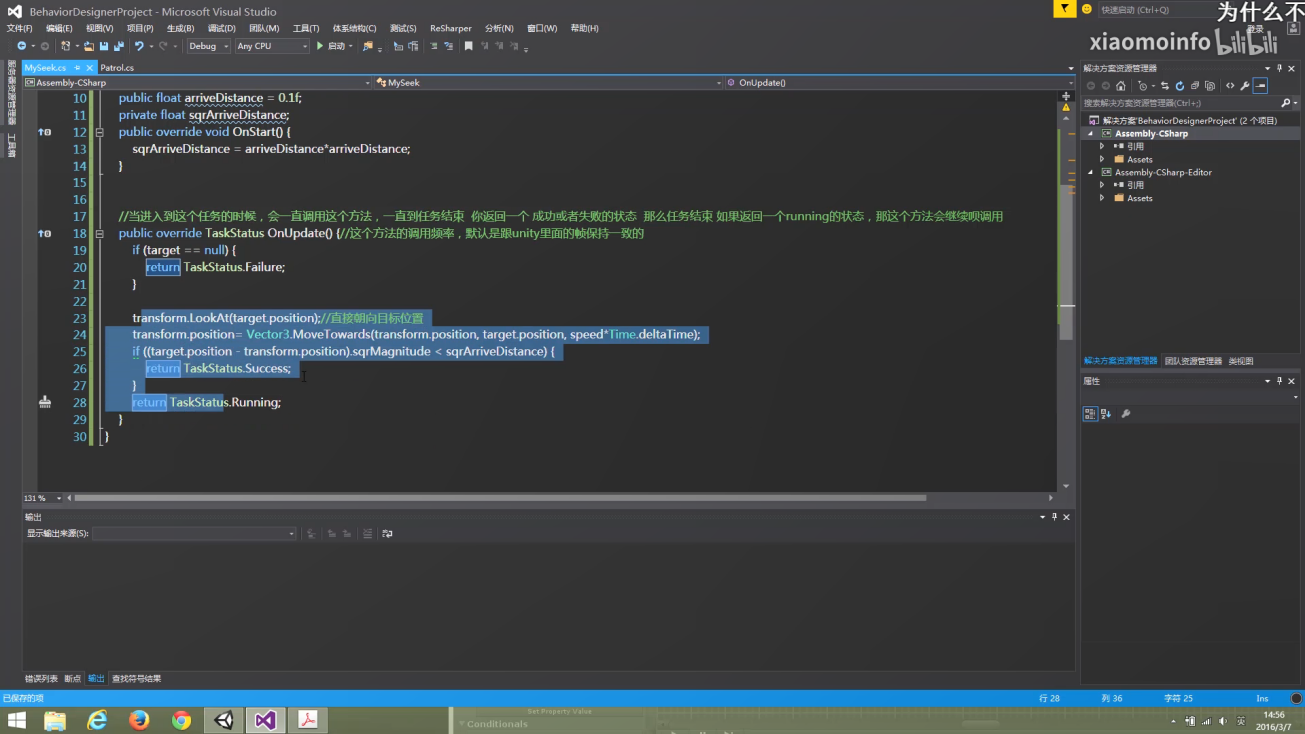
Ps：关于中断不同类型的解释

<https://blog.csdn.net/u012632851/article/details/89647097?locationNum=11&fps=1原理>

<https://www.cnblogs.com/01zxs/p/9863715.html> 详解举例

其实起因就是因为一些结点一旦运行起来如果没有外部的干预就不会自行终止 必须加上中断的不同类型来干预他状态的改变！！

第六课学习：



1.自己写行为树的脚本要选择一个基础的类来继承比如Action

写完以后这个行为就会出现在对应类的下方

1. 用TaskStatus.Failure/Success/Running等来传递行为的状态 目前只知道这些
2. 在当中gameobject身上添加了Nav Mesh Agent组件

这是个什么东西呢

<https://blog.csdn.net/ti1an123/article/details/42430421>

1. 行为树的脚本应该引用BhaviorDesinger.xxxx命名空间，并且行为树中有自己的OnUpdate/OnAwake等方法，要使用override TaskStatus重载，其拥有自己的规则.

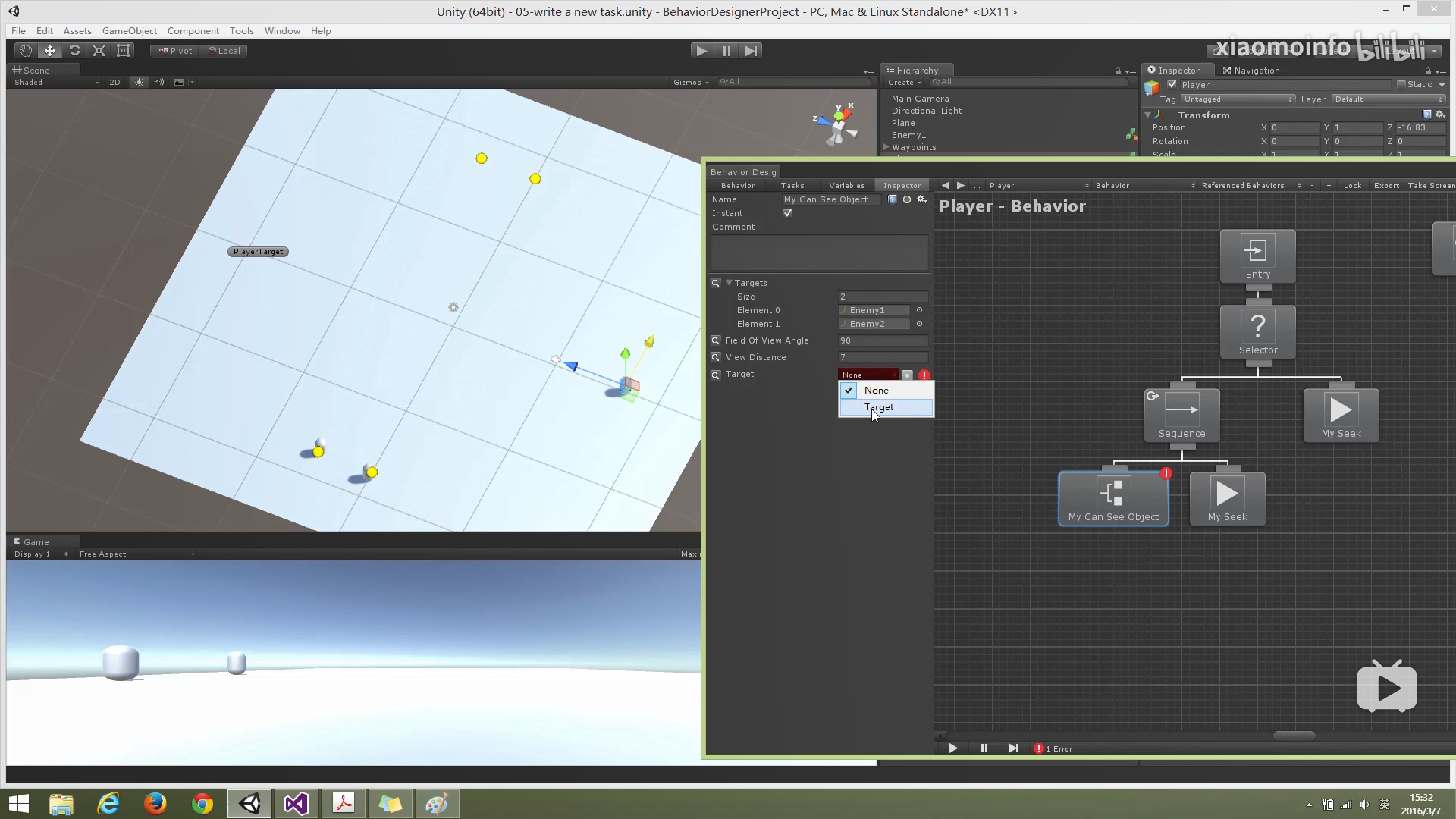
第七课

Sharedxxx类型的变量 ,当一个变量是shared类型的时候，可以让variable面板里面的变量和其联系在一起，这样就可以使其能够让其他shared类型的变量得到它的值

在使用的时候注意

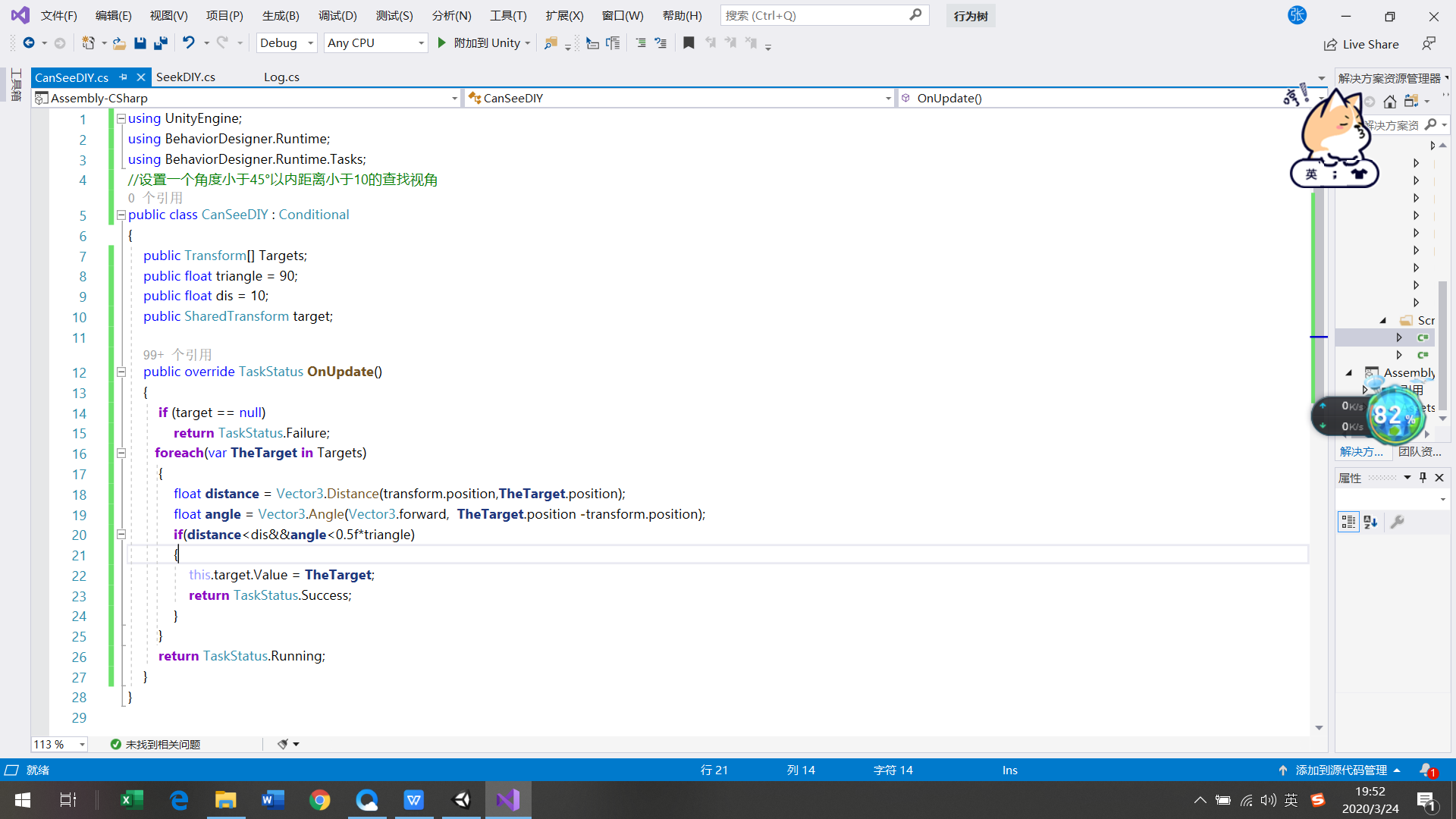
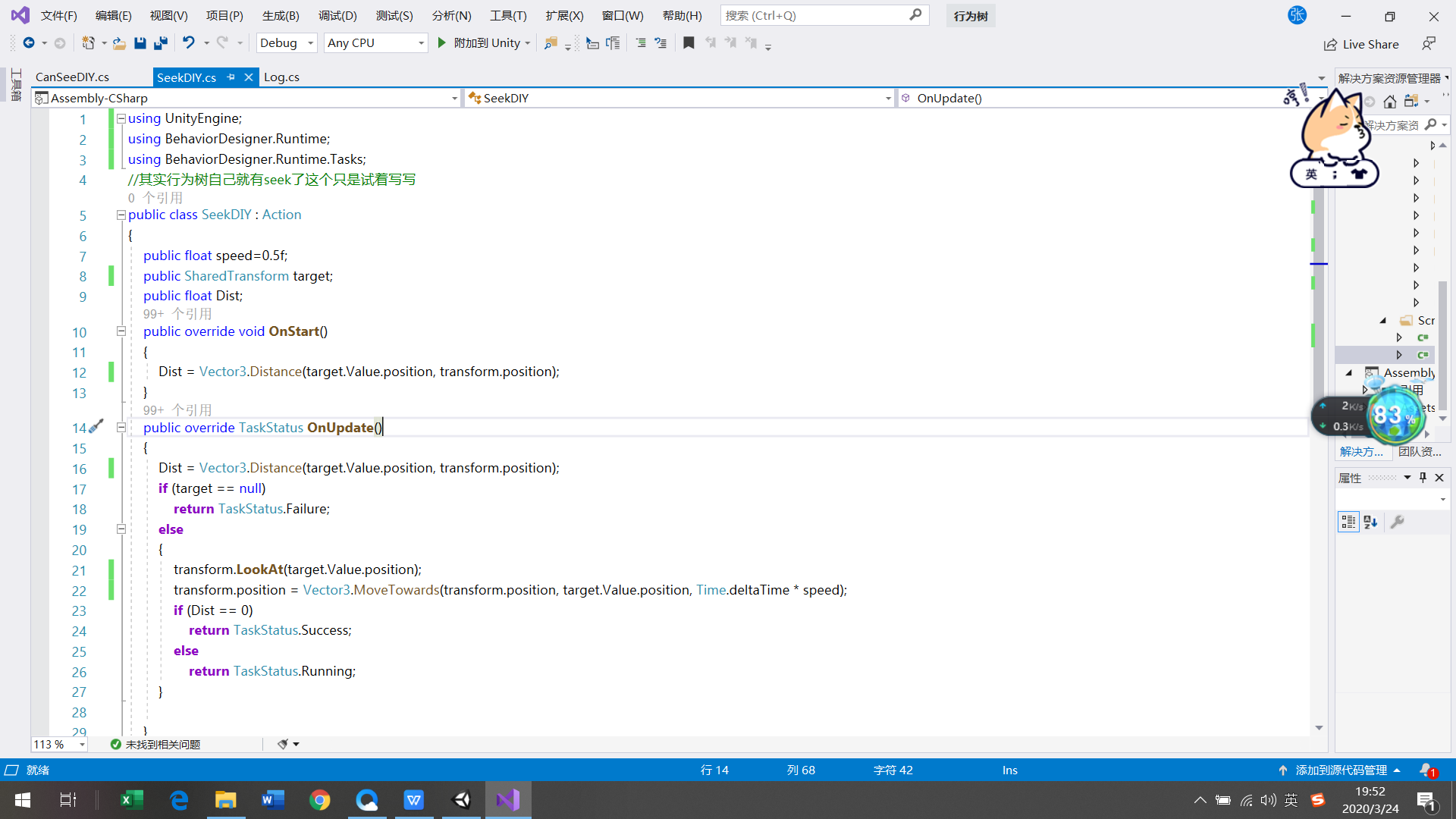
**1.需要在脚本头部引入：using BehaviorDesigner.Runtime;**

**2.使用的时候，额外加上.Value就可以啦。**



Ps：变量后面的小圆圈是shared变量才有的

此处的Target就是variable面板里面的

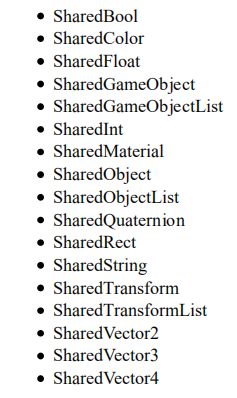
 

1. ：

局部变量行为树中所有任务共享

全局变量所有行为树共享

可以直接使用的Shared类型：



# Unity AI 行为树

<https://blog.csdn.net/dingxiaowei2013/article/details/89249609>

<https://blog.csdn.net/q414620221/article/details/78596732>