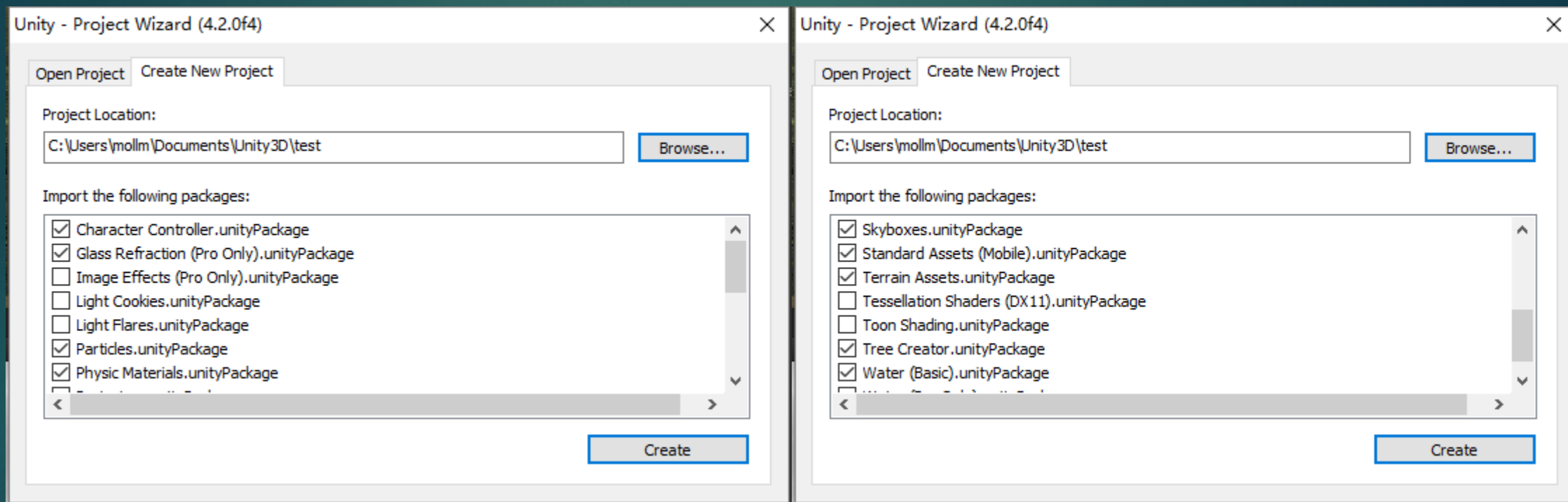


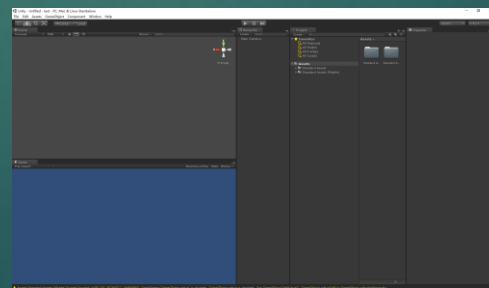
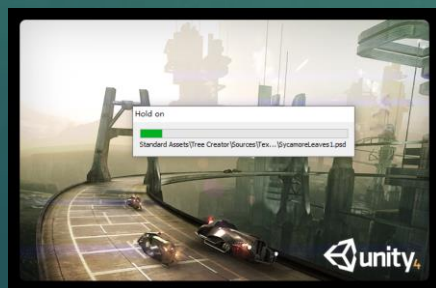
# Unity 3D 实现雨雪雾霾天气模型

MOLLMY 2016/06/28

# 新建工程，导入以下包：

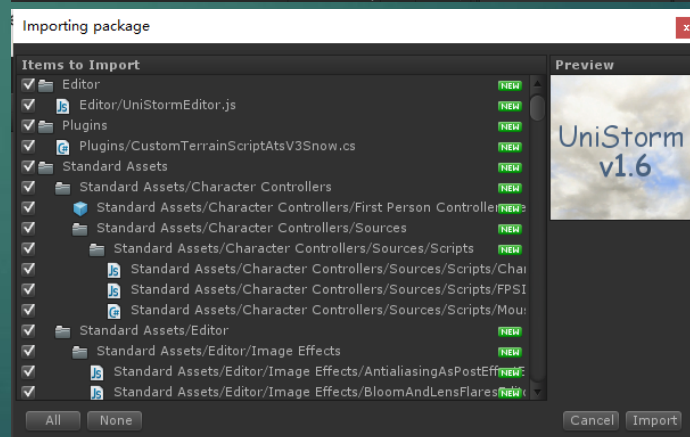
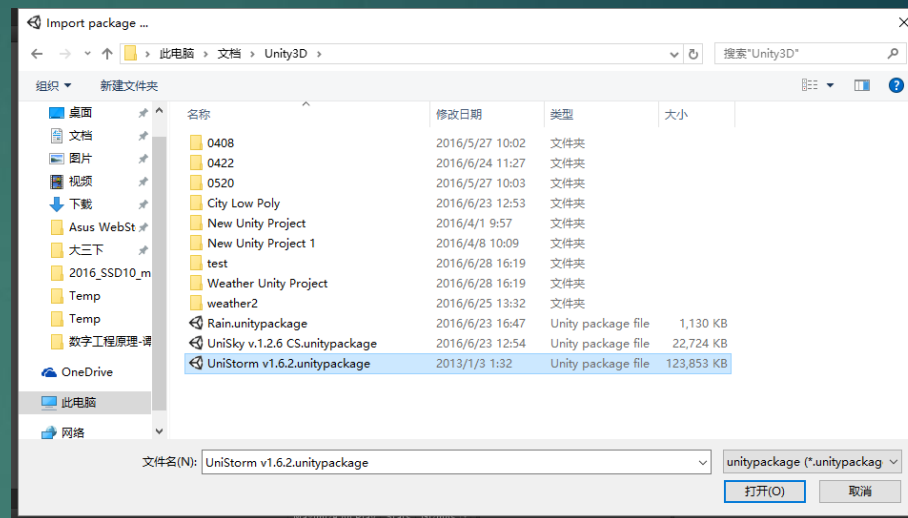
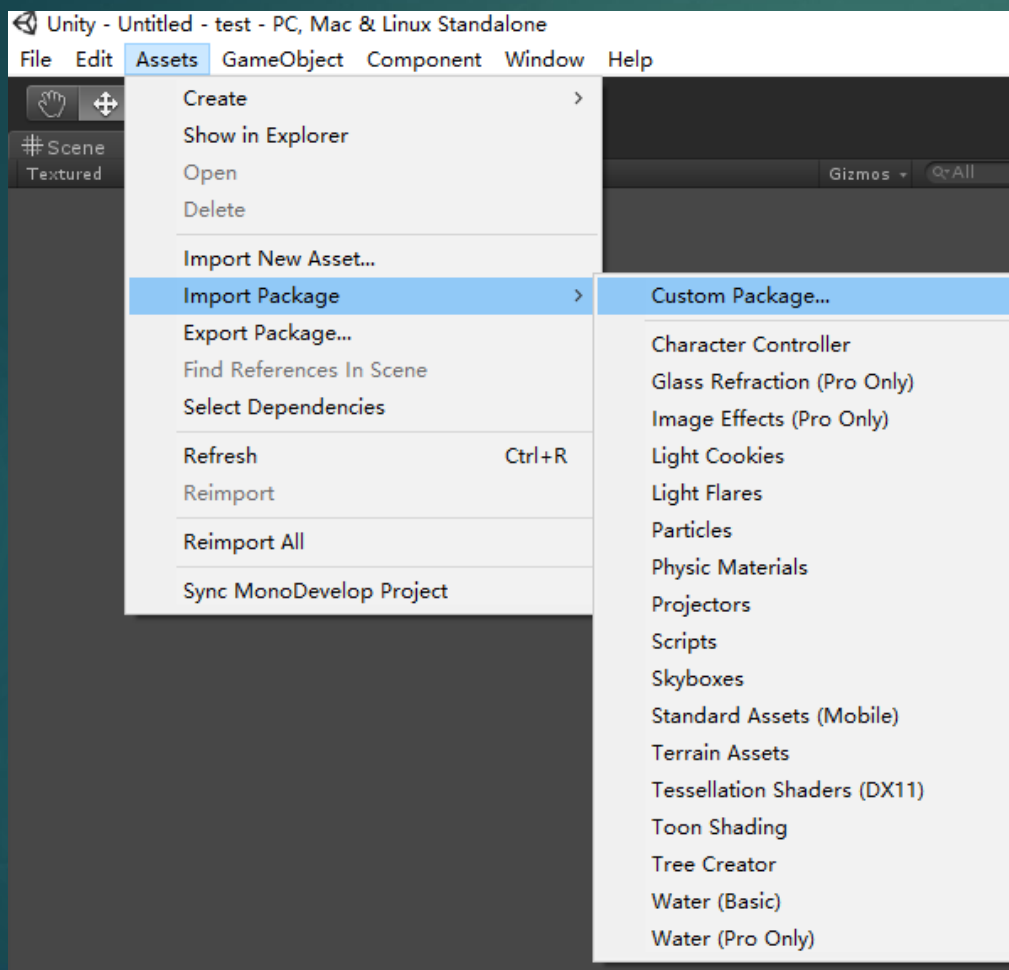


点击create后进入下一页



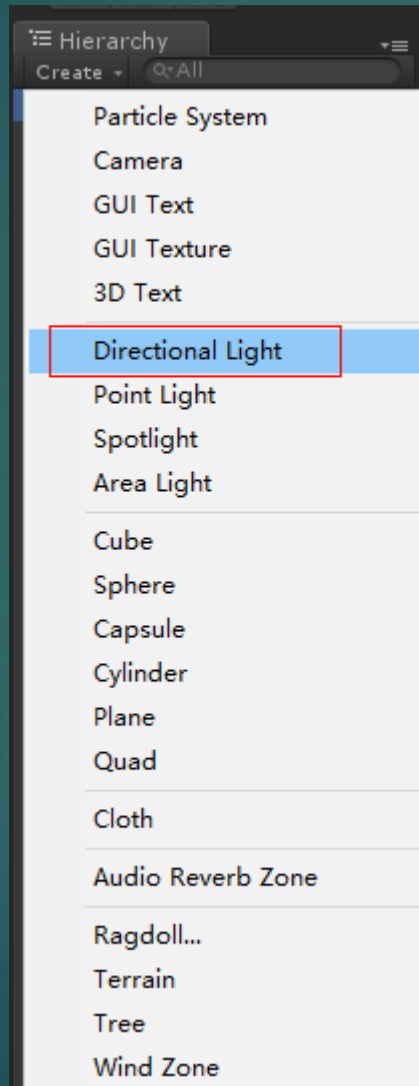
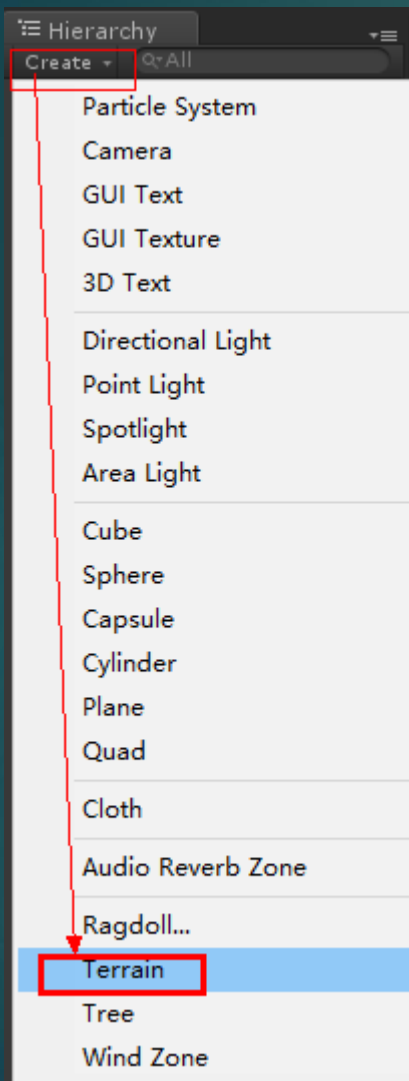
# 导入uniStorm包:

获取包: <http://pan.baidu.com/s/1sljAGix> 密码: yw5y

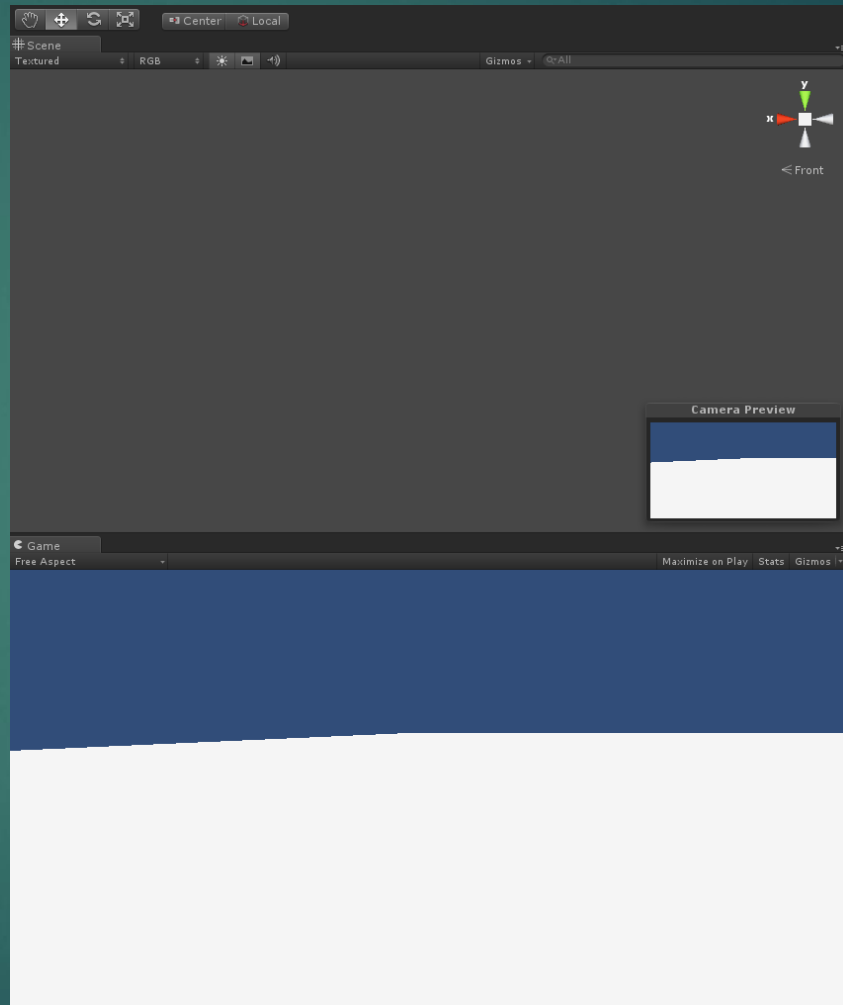


点击import全部导入

# 创建地形:

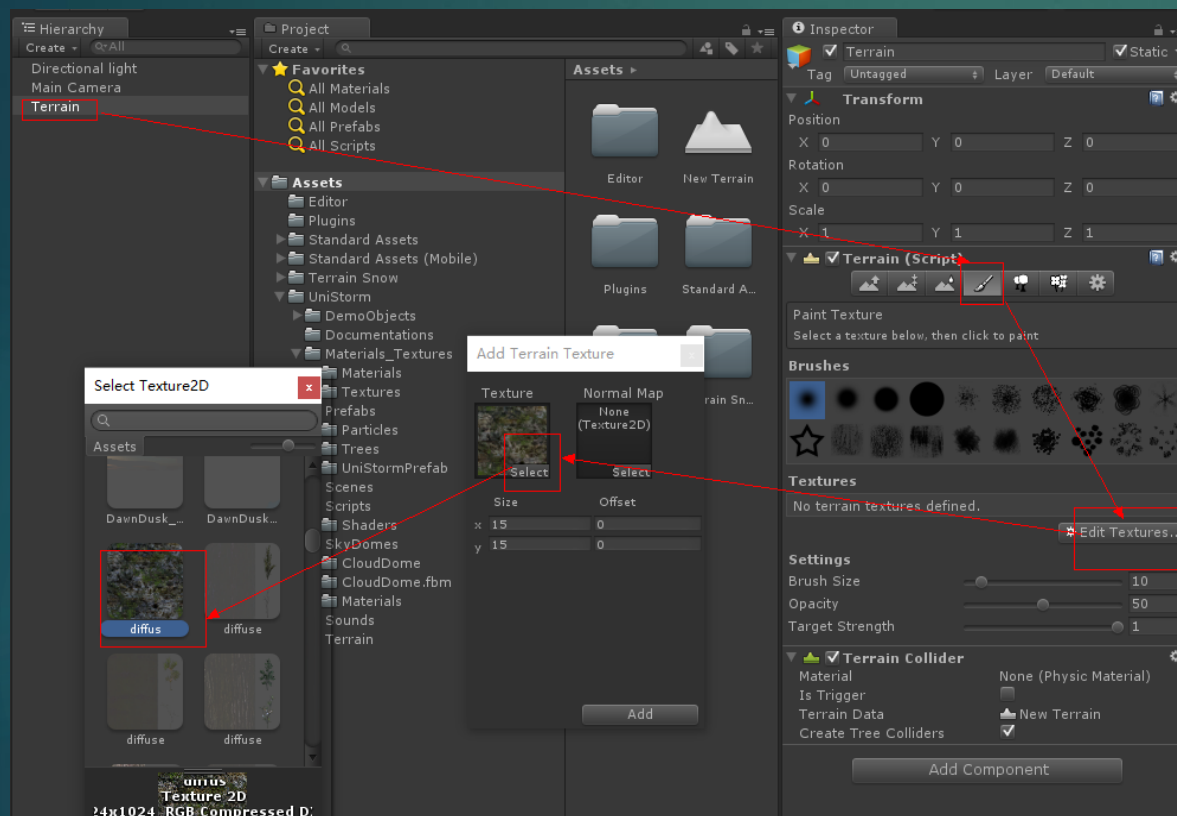


创建平行光

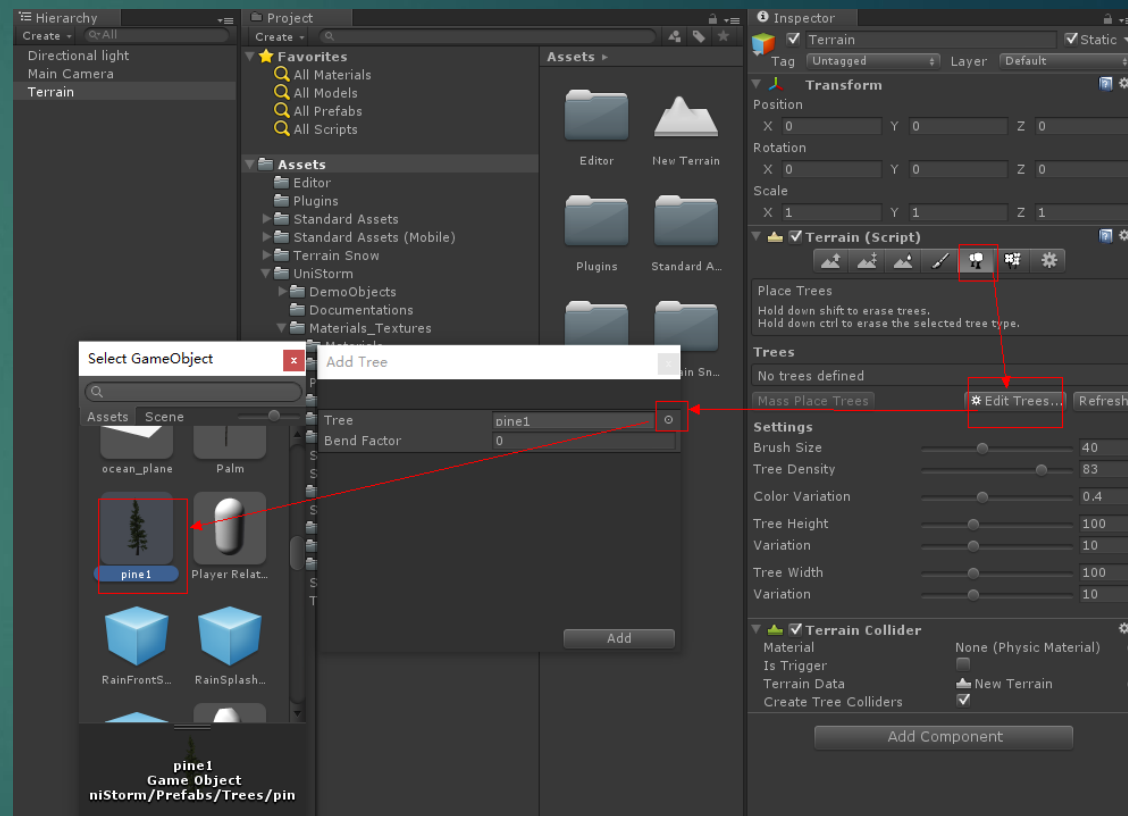


调整摄像机角度和位置

# 添加地形纹理、树木:

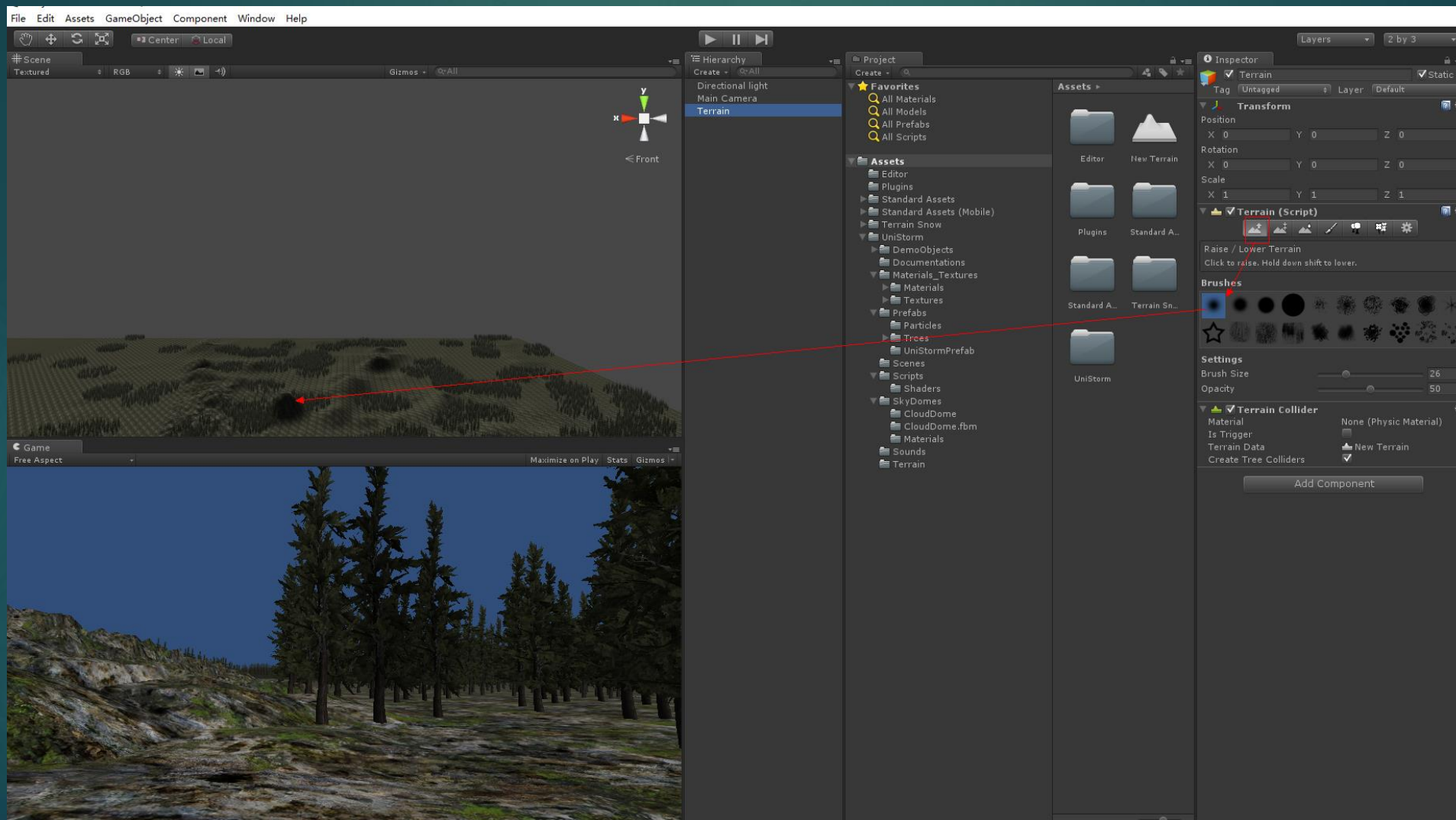


地表纹理



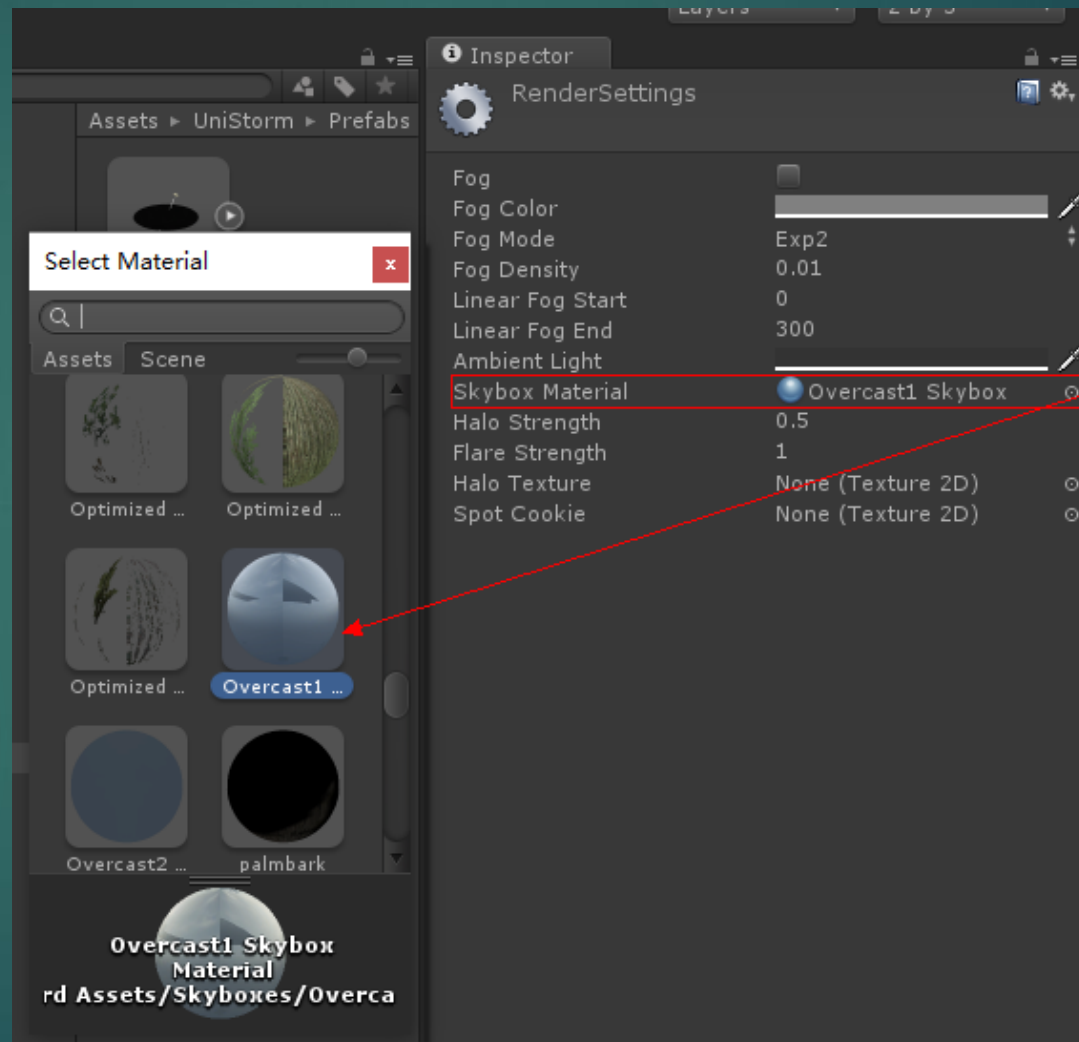
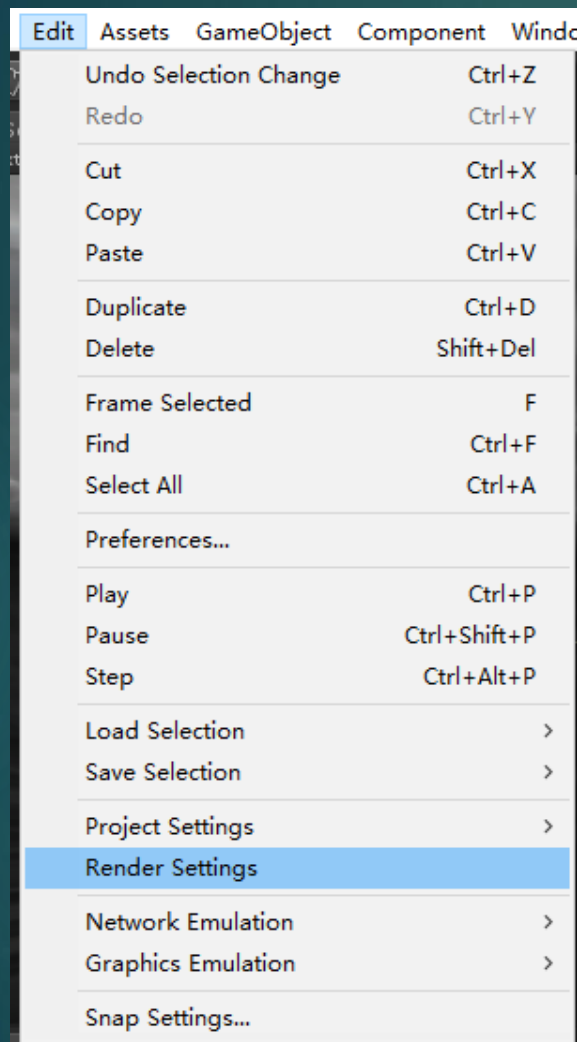
树木

# 添加山丘:



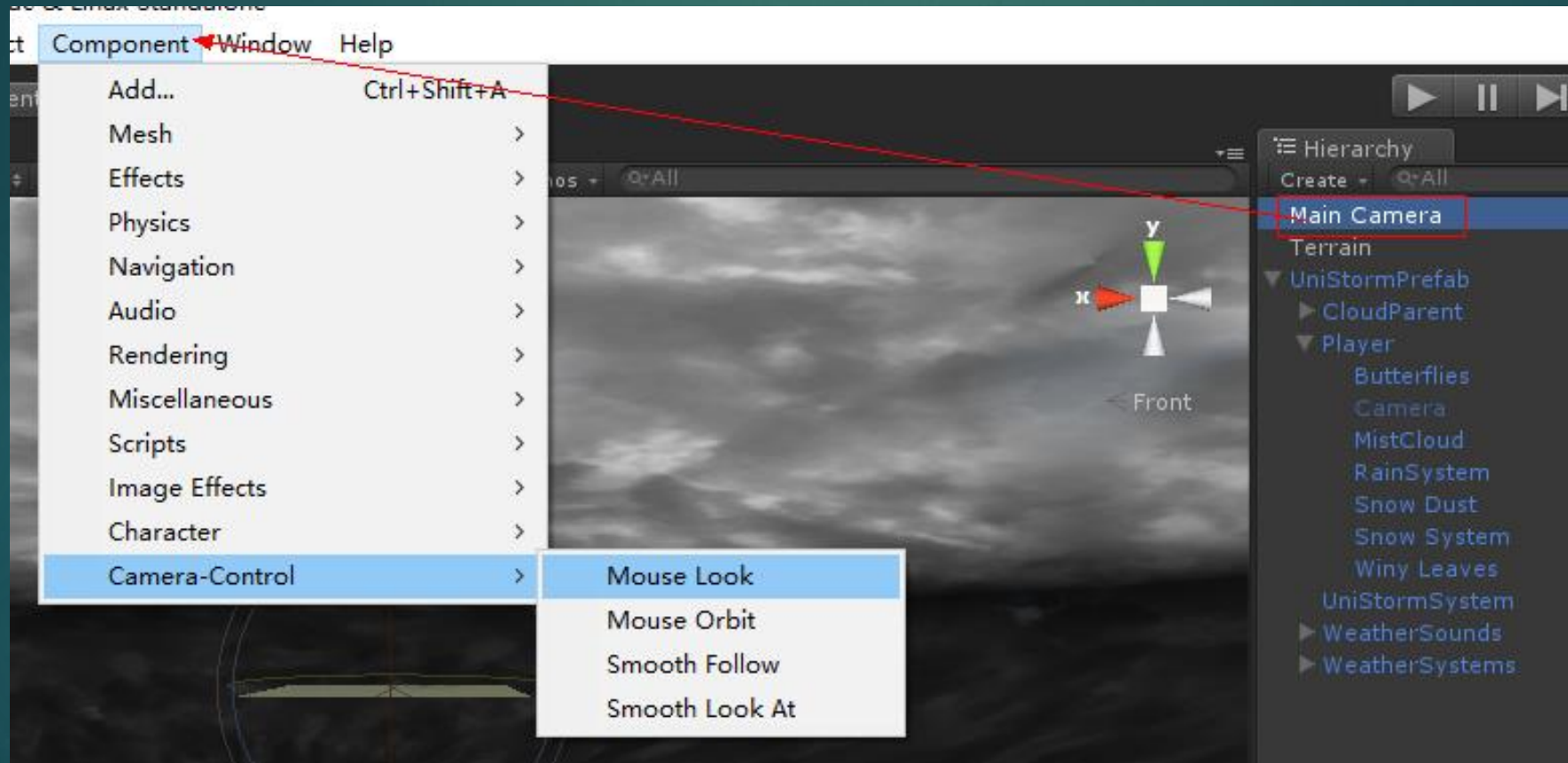


# 添加天空盒子:



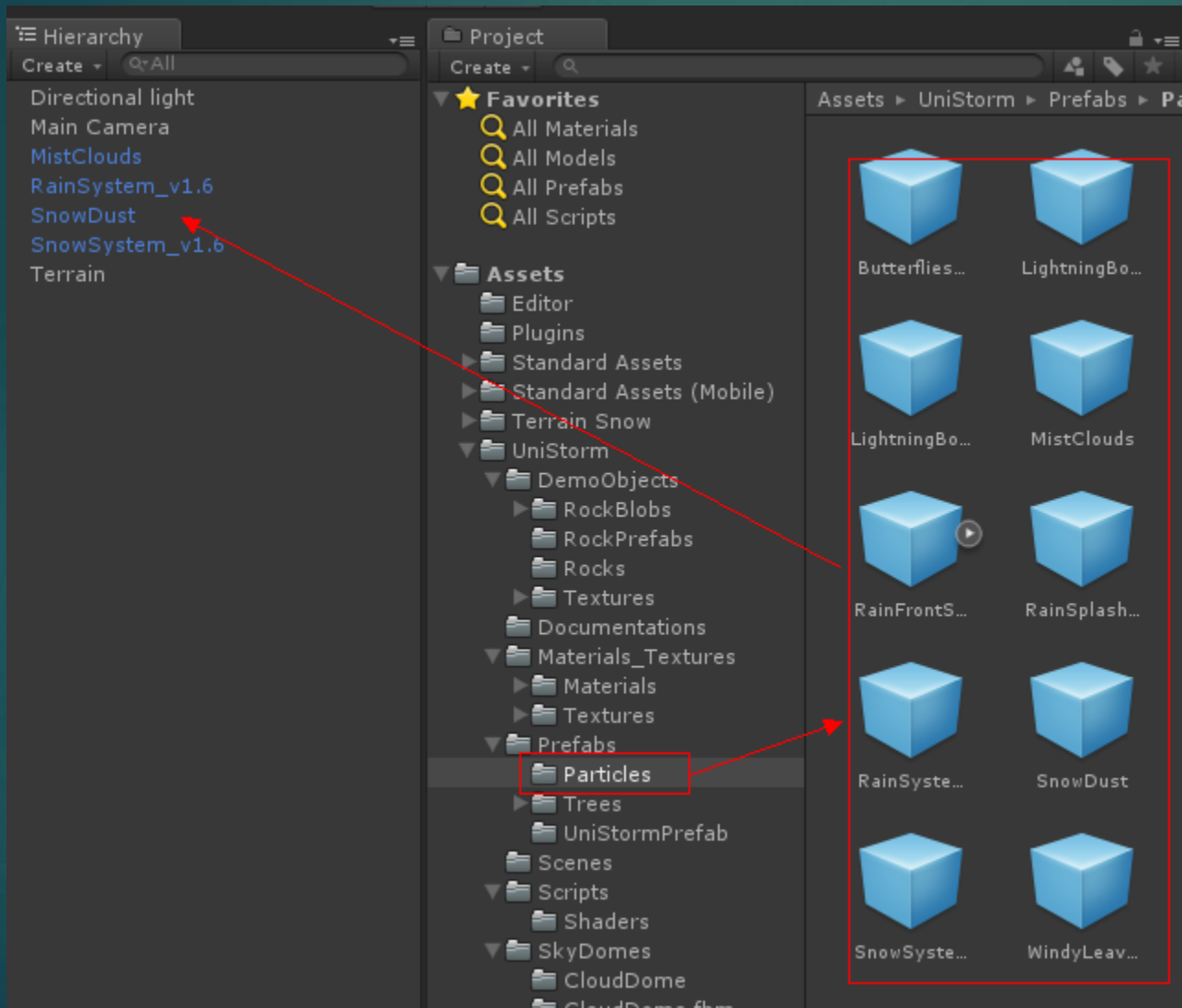
# 给摄像机加MouseLook脚本

使视角可以随鼠标的移动而转换



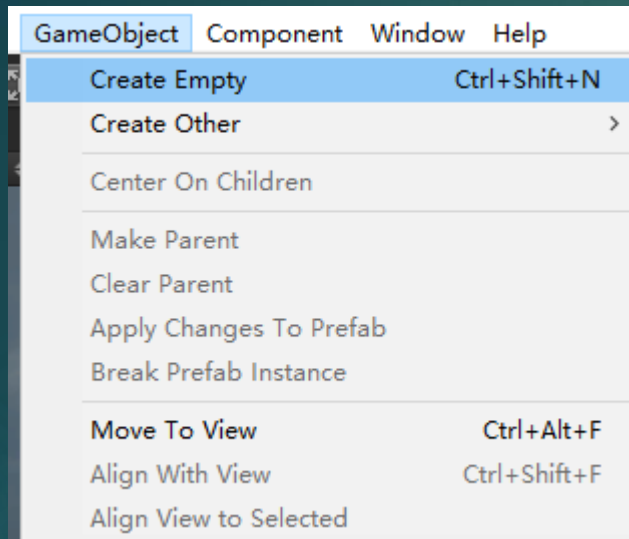


# 加入雨、雪、雾霾预设：

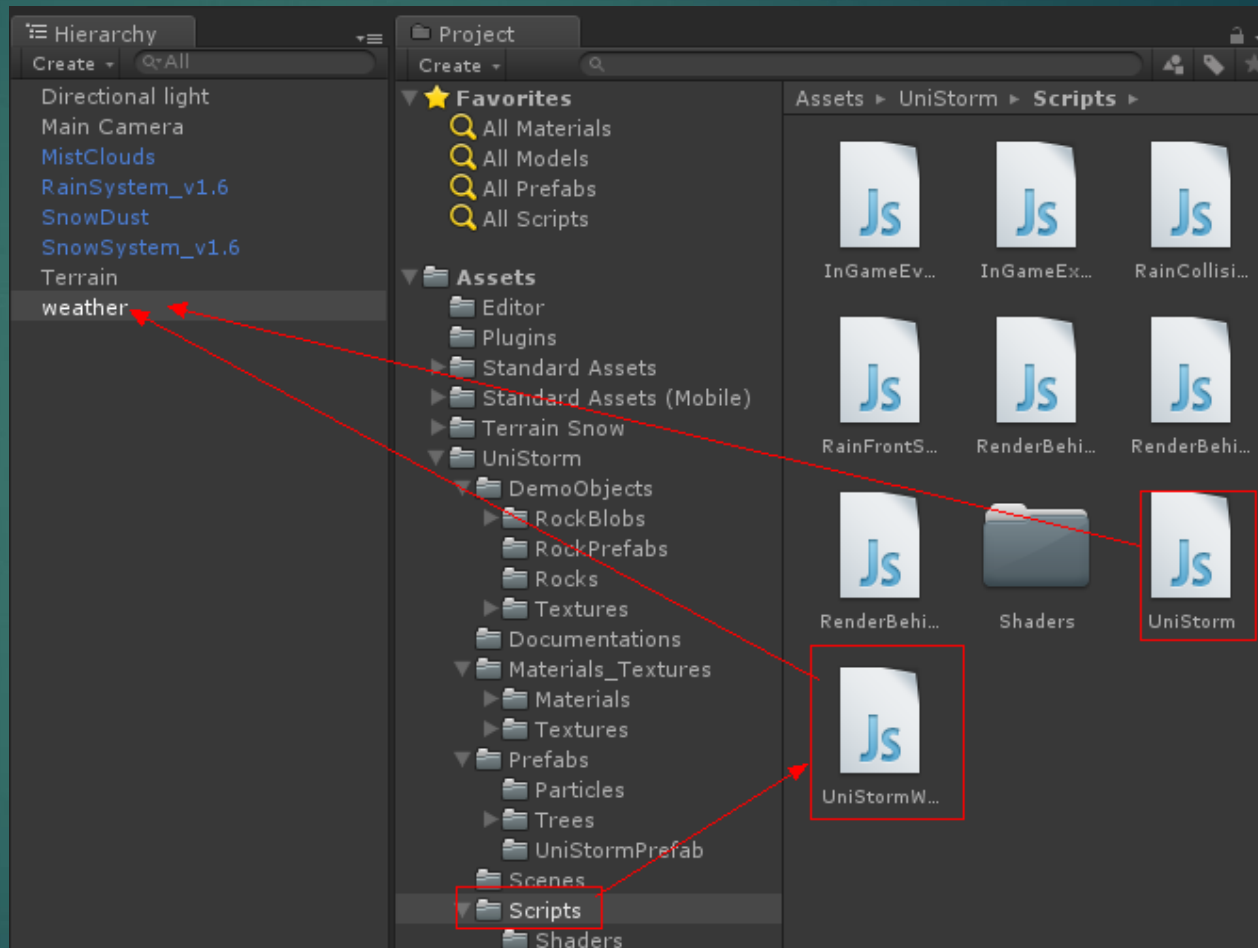


将需要的particles拖入Hierarchy中，左侧Hierarchy中蓝色项目为雨雪雾霾对象。

# 创建天气对象:

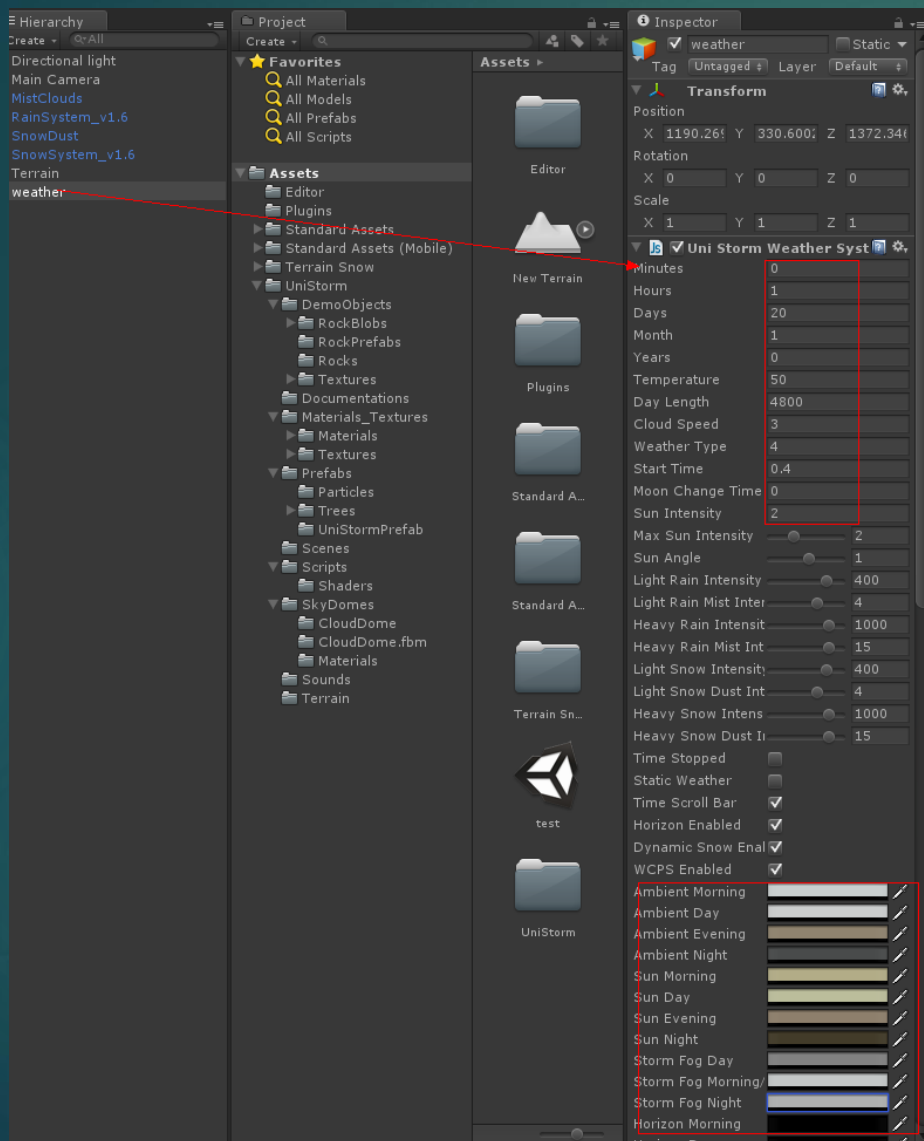


点击Create Empty创建一个空对象，重命名为Weather。



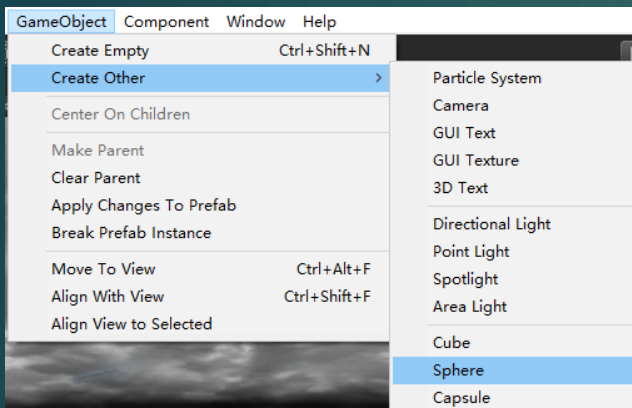
将UniStorm和UniStormSystem两个脚本赋给weather对象。

# 给天气对象赋值：

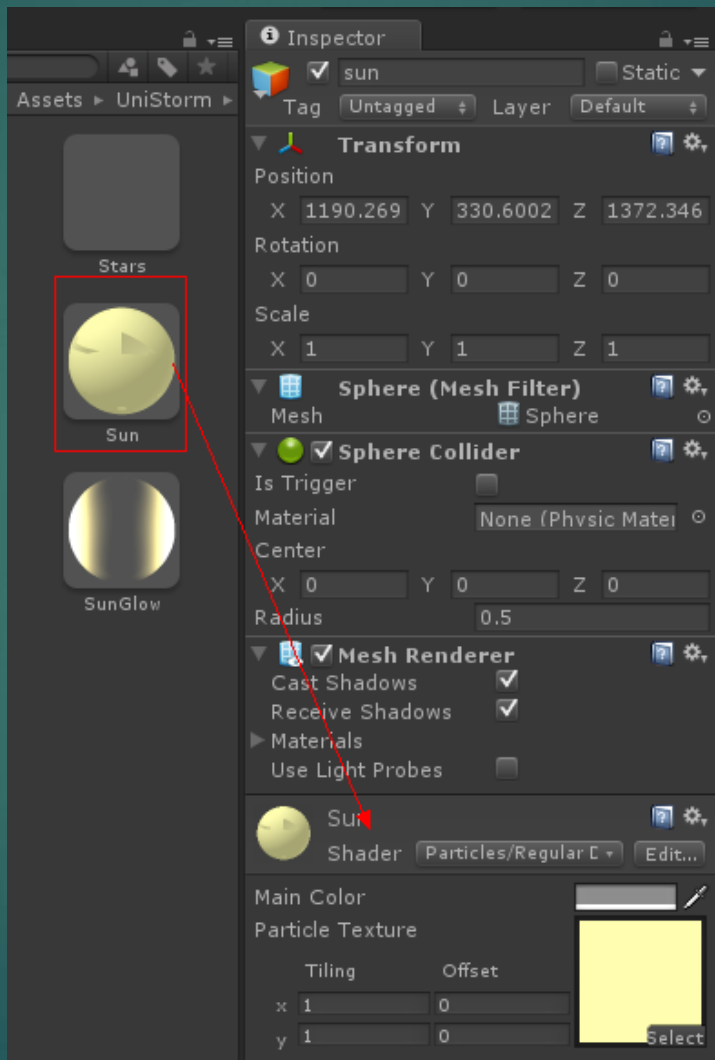


脚本文件中有针对不同时间平行光位于地形的不同位置的设置，所以可以给时间变量赋初始值，还可以选择在场景中显示时间滑动条，滑块的位置代表一天中不同的时间。根据不同天气的实际情况选择不同的颜色，这里大体选择相似颜色，如需要更好的效果需要仔细调色。暂时先给这些变量赋值，接下来添加日、月、云和声音对象后再给后面的变量赋值。

# 添加太阳对象：



创建球体对象，重命名为sun



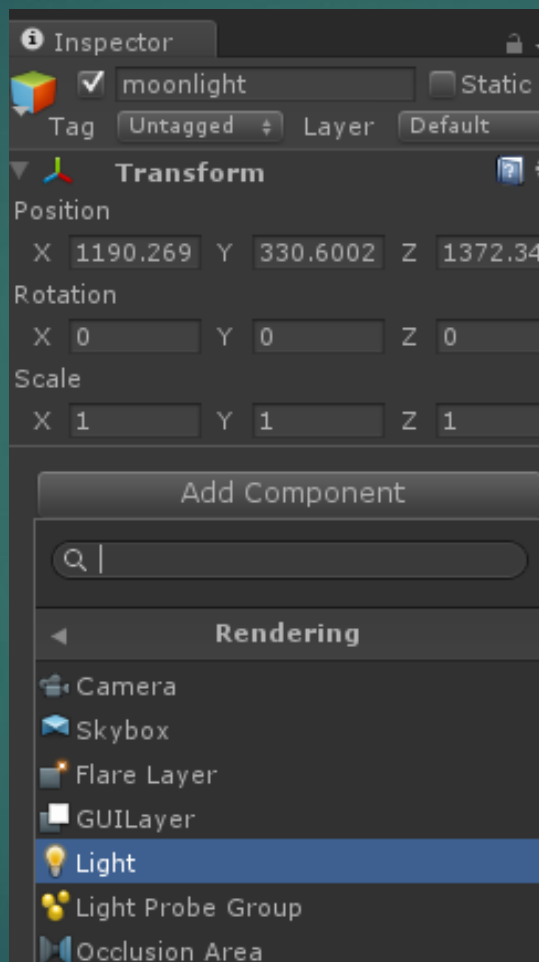
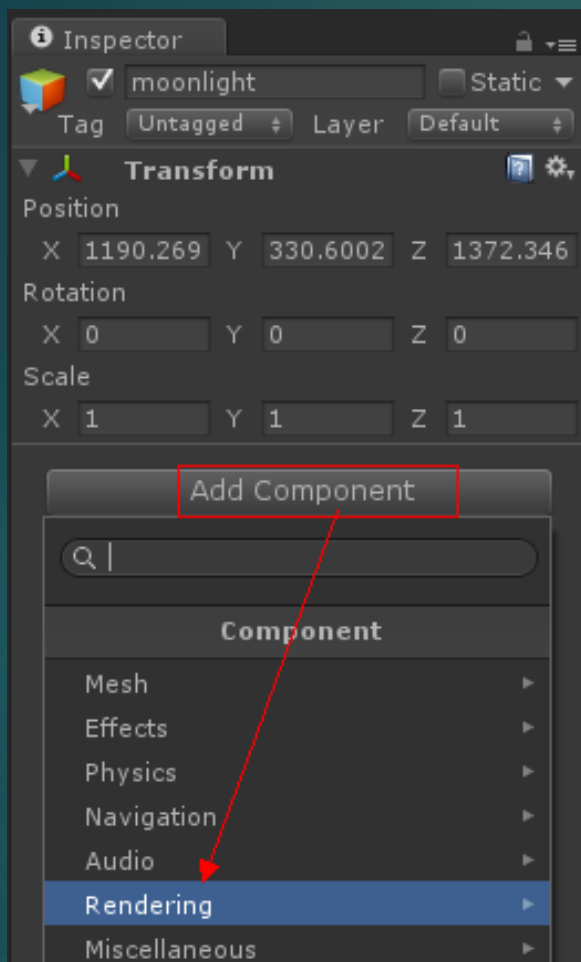
将SkyMaterials文件夹下的Sun素材赋给sun对象。

类似地——

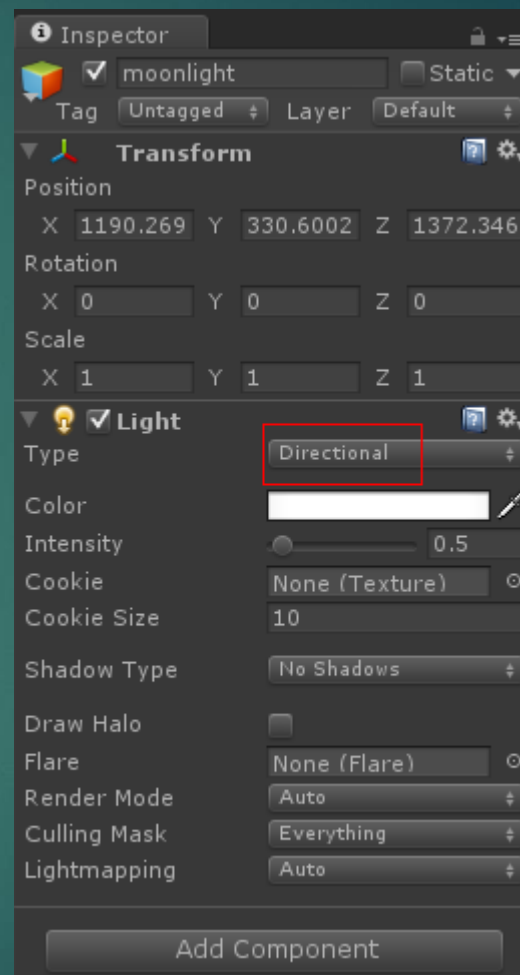
日光对象：新建名为sunglow的球体对象，将SunGlow素材赋给sunglow对象。

月球对象：创建名为moon的球体对象，将MoonPhasesMaterials文件夹下的MoonPase素材赋给moon对象。

# 添加月光对象:



创建空对象，命名为moonlight，在moonlight的Inspector视图中点击Add Component → Rendering → light

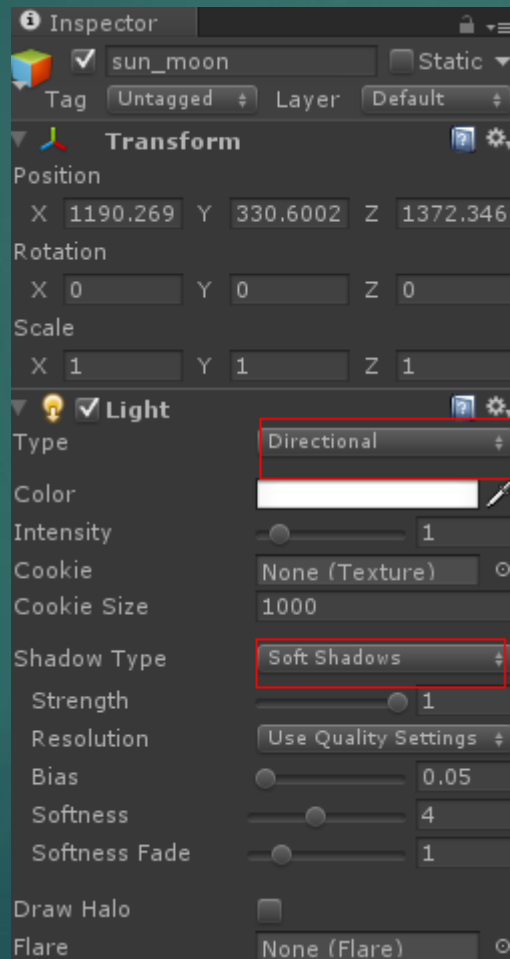


将light的type选择 Directional。

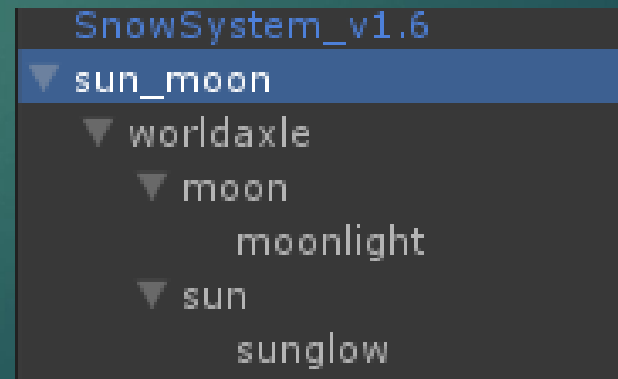
# 组合日月对象：

组合完成后应仔细调整位置，使滑动时间滑块时，太阳或月亮刚好从摄像机视野范围内经过。

新建名为sun\_moon的空对象，添加light组件并按上图设置

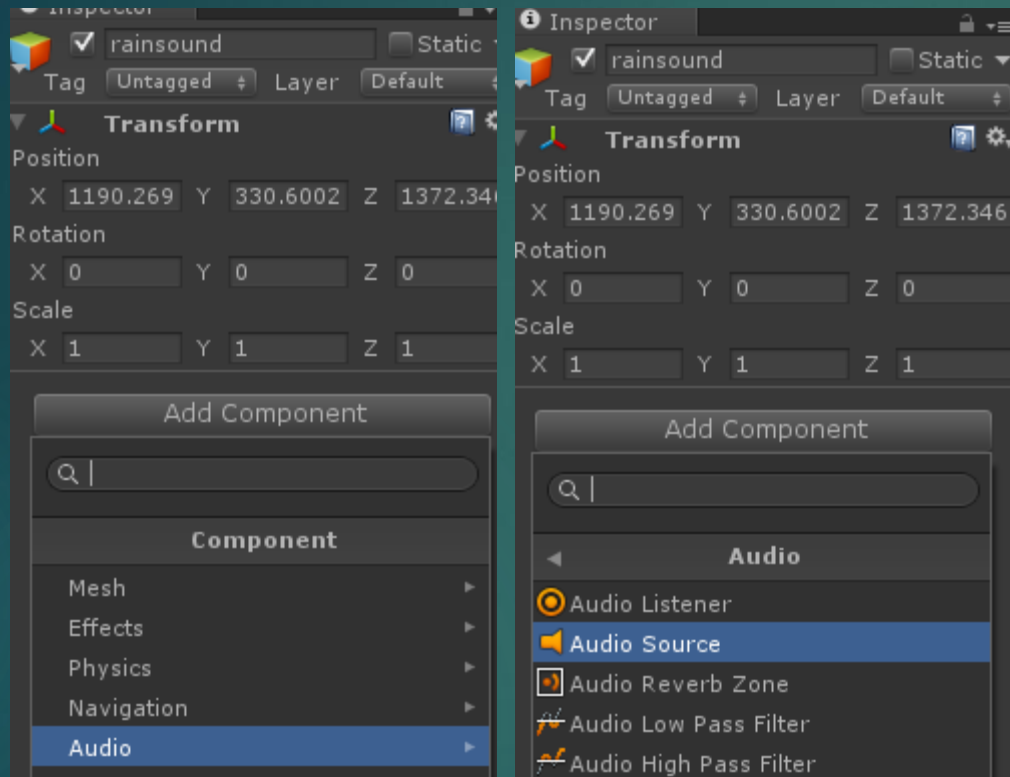


- 新建名为worldaxle的球体对象；
- 将moonlight对象放至moon对象下，将sunglow对象放至sun对象下；
- 将sun和moon放至worldaxle对象下；
- 将worldaxle放至sun\_moon对象下，是它们形成下图所示父子关系：

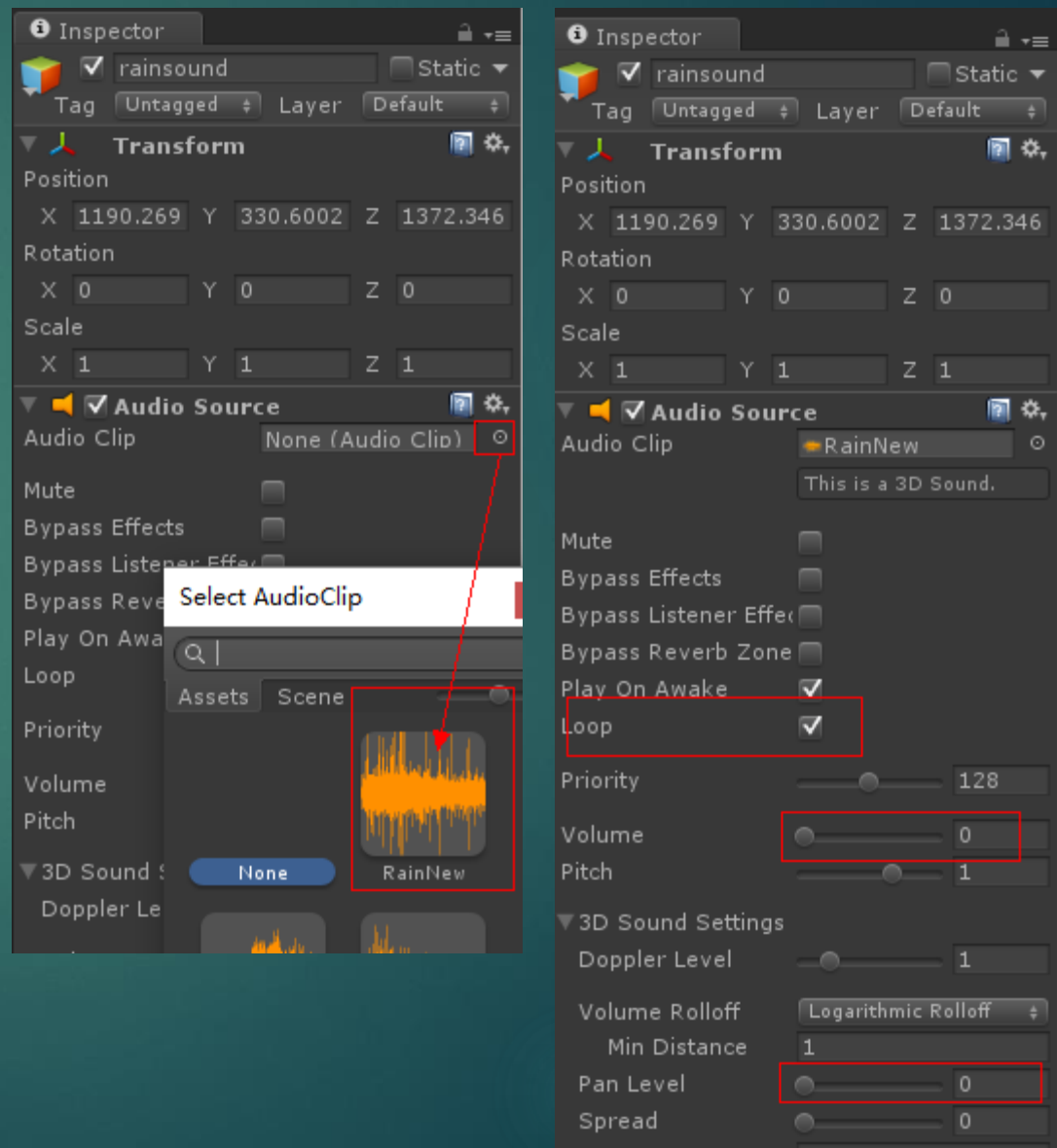




# 添加声音对象:

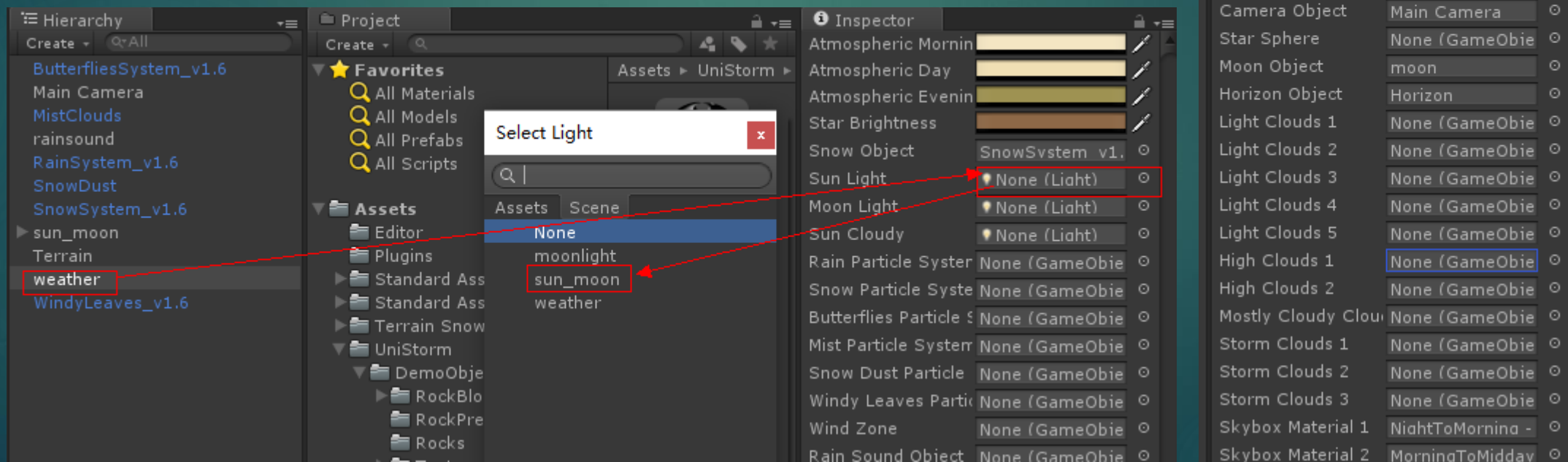


新建空对象，重命名为rainsound，Add Component → Audio → Audio Source 添加声音组件，按右图所示添加声音源并设置。类似地，可添加其他声音对象。



# 给weather对象的其他变量赋值:

按下图方式初始化各个对象，完成后如右图所示



# 编辑脚本使符合要求:

```
3978
3979 function OnGUI () {
3980
3981     if (timeScrollBar == true)
3982     {
3983         //Allows a scrolling GUI bar that controls the time of day by the user
3984         startTime= GUI.HorizontalSlider( Rect(10,5,200,20), startTime, 0,1.0);
3985     }
3986
3987     if (commandPromptActive)
3988     {
3989         stringToEdit = GUI.TextField (Rect (10, 430, 40, 20), stringToEdit, 10);
3990     }
3991     if(GUI.Button(Rect(10,30,50,30),"雨"))
3992     {
3993         monthCounter = 8;
3994         temperature = 35;
3995         weatherForecaster = 12;
3996     }
3997     if(GUI.Button(Rect(10,65,50,30),"雪"))
3998     {
3999         monthCounter = 1;
4000         temperature = 5;
4001         weatherForecaster = 3;
4002     }
4003     if(GUI.Button(Rect(10,100,50,30),"晴"))
4004     {
4005         weatherForecaster = 7;
4006     }
4007     if(GUI.Button(Rect(10,135,50,30),"雾霾"))
4008     {
4009         weatherForecaster = 1;
4010     }
4011 }
4012
```

在UniStormWeatherSystem.js脚本的OnGUI函数内加红框所示代码段，以方便我们快速切换到要求的雨、雪、雾霾天气模式下。

# 完成

如要更加精致的效果，还需要对各个对象以及场景中的草木进行微调，UniStorm包中有一些地形纹理，可用于在地形上绘制小路，制作湖泊等。可继续增加windsound对象以实现风声音的效果，增加cloud对象以展示在不同天气下天空的云彩变化。时间有限不一一截图介绍。

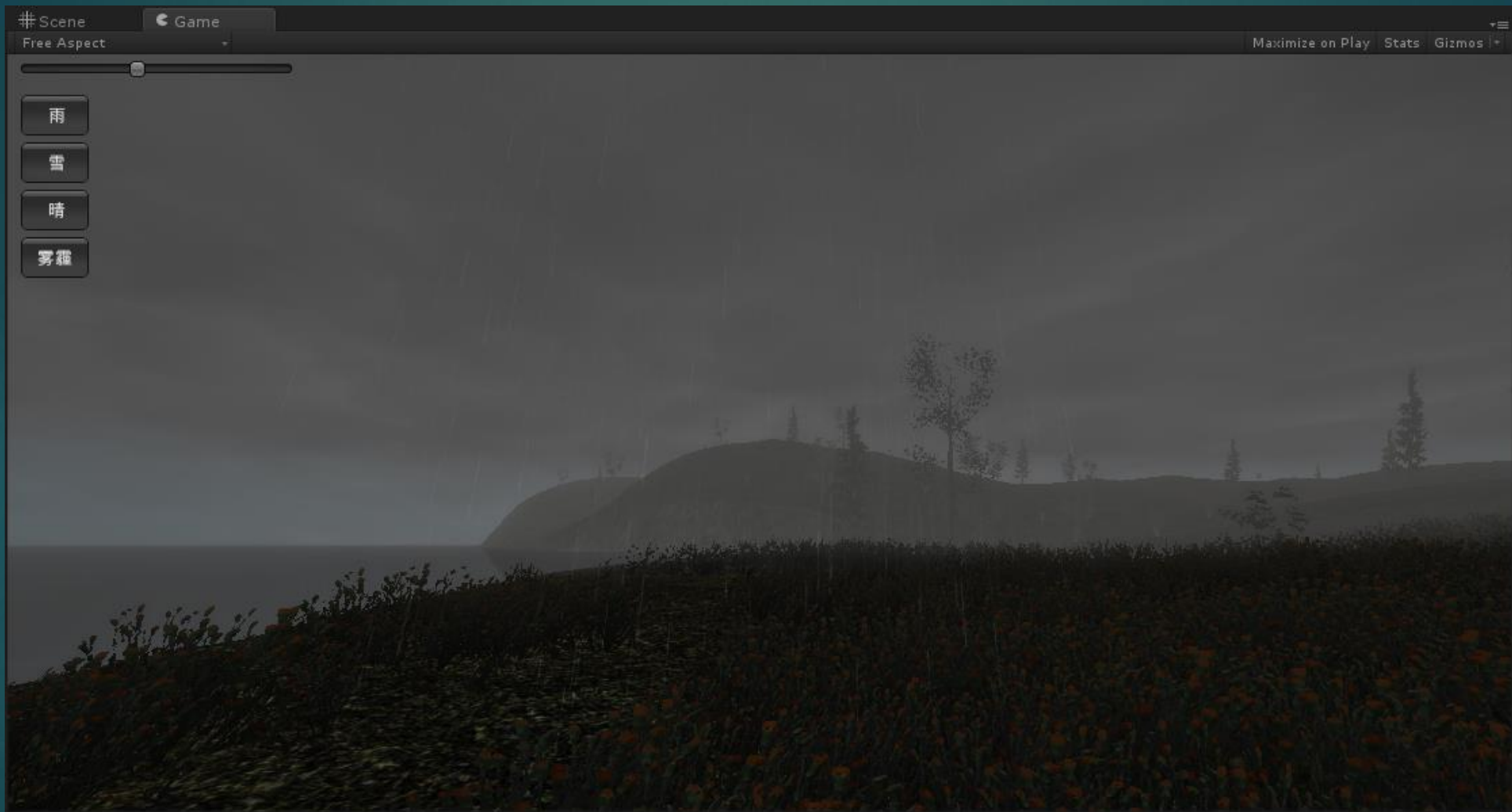




# 效果展示 —— 雪：



# 效果展示 —— 雨：

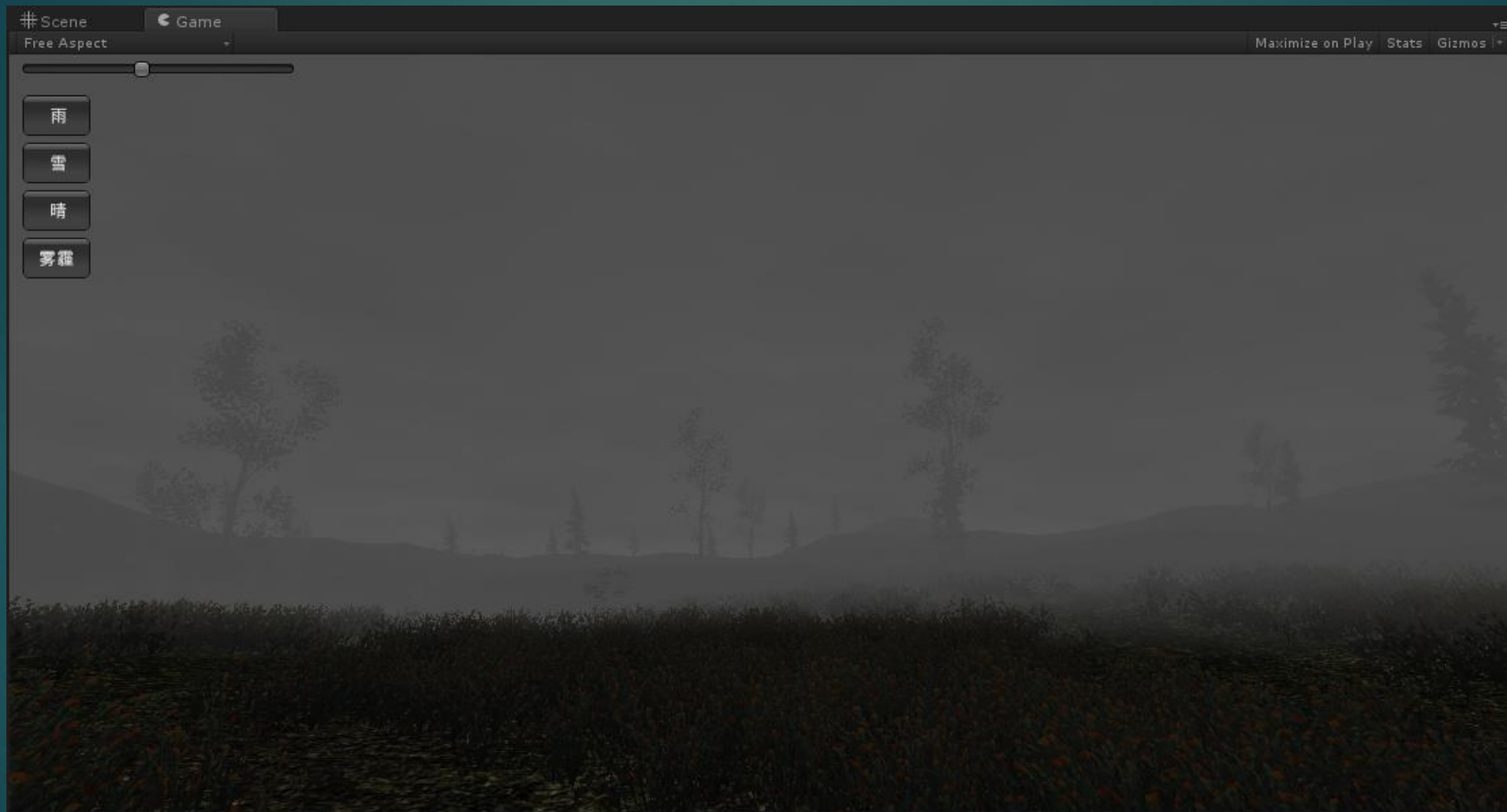




# 效果展示 —— 晴：



# 效果展示 —— 雾霾：



THE END