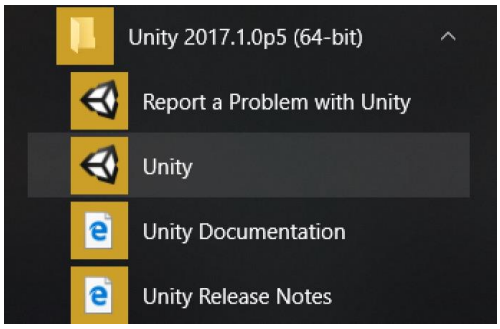
浸入式环境

目标：

- 团结--玩家运动

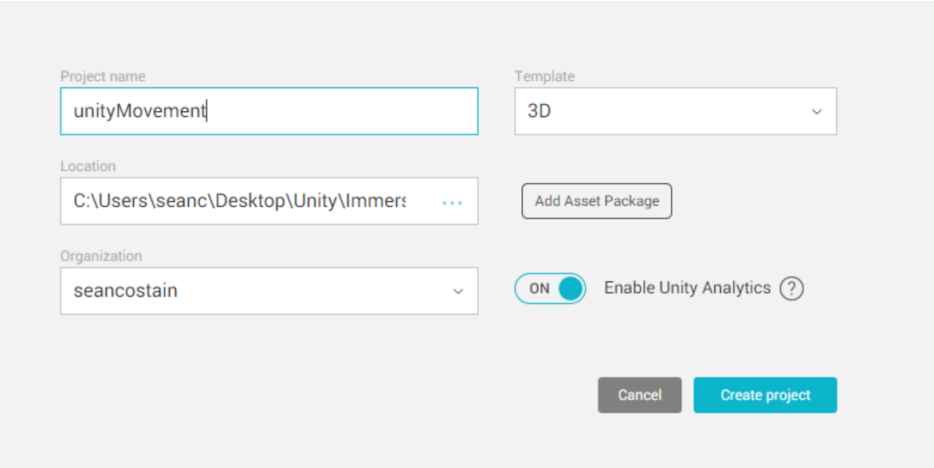
加载统一



I如果你没有一个帐户，你将不得不创建一个。否则，请使用您的注册帐户登录。

创建新项目，单击右上角的新图标。

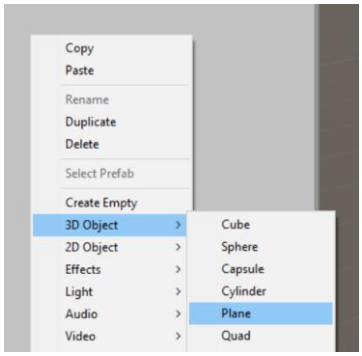
应用名称，然后单击"创建项目"。



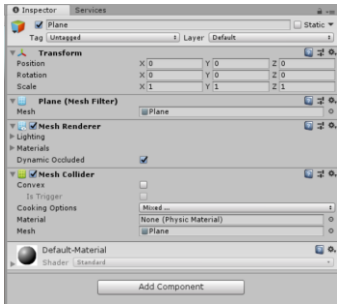
这将开门下面的场景

首先, 我们需要创造一个进入的世界。在这种情况下, 它不会是一个复杂的世界, 仅仅是理解正在发生的事情就足够了。

在层次结构窗格中右键单击, 然后选择 3d-> 平面

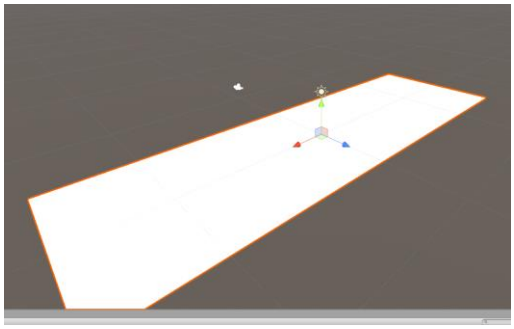


从这里, 点击飞机, 并在检查器面板中进行检查。

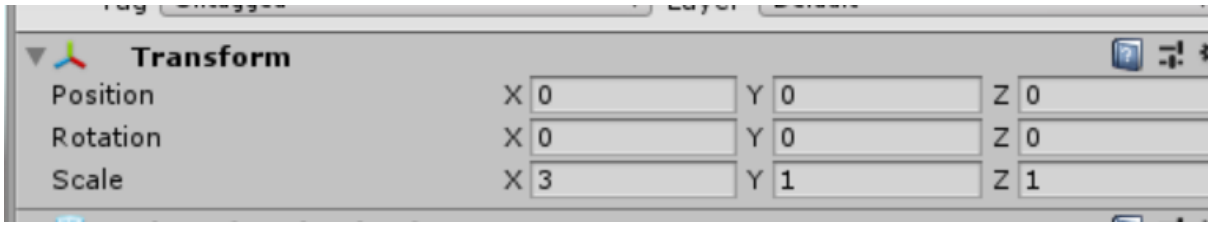


我们想要的 do 是加长平面, 这可以通过增加 x 平面上的刻度, 将值从1更改为3来实现。

这将为您提供以下内容:

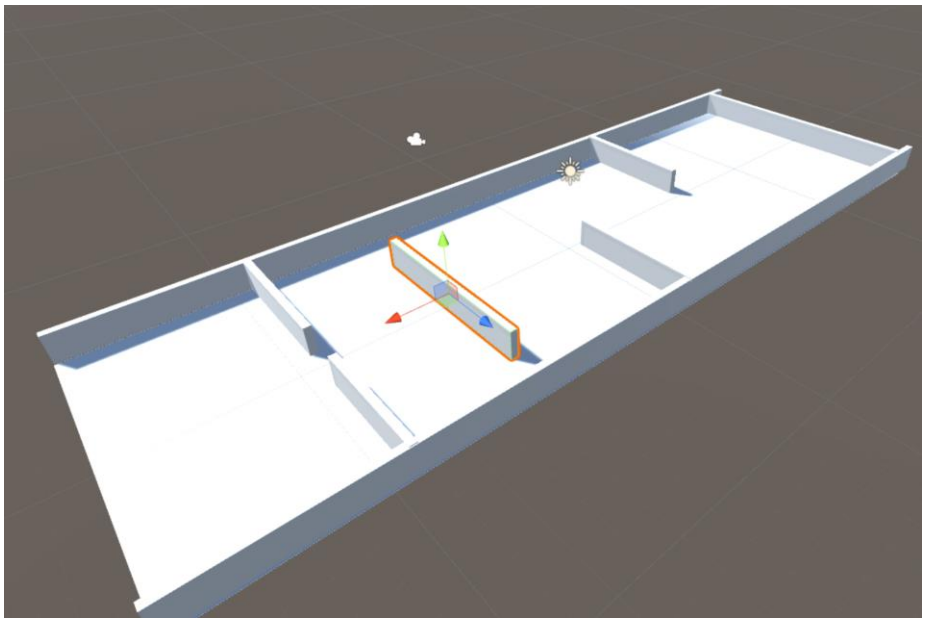


检查器的更改是:

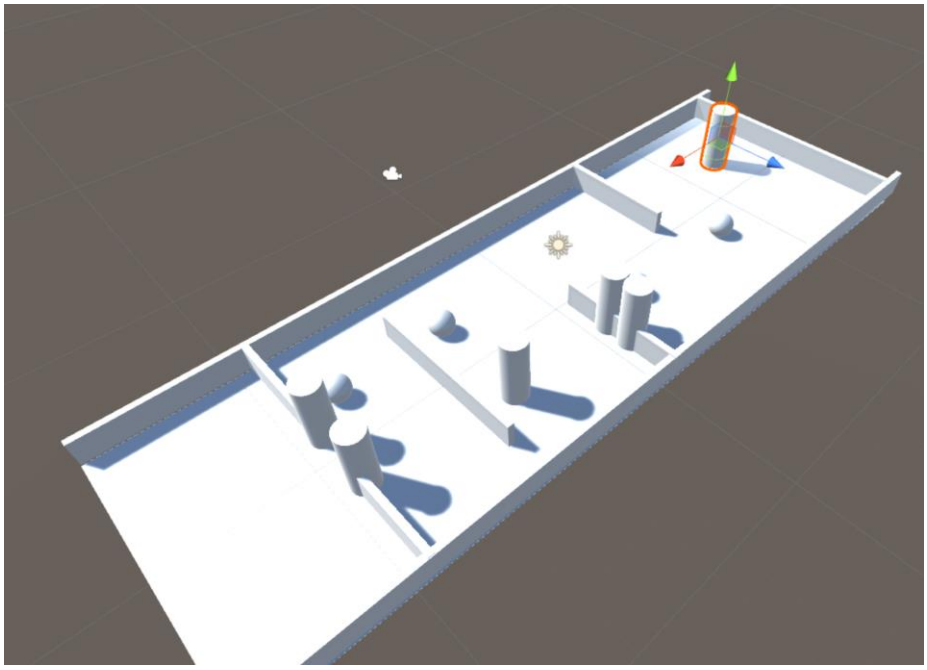


接下来, 在外部添加一些形状的3d 多维数据集, 并开始添加一些墙壁来运行。

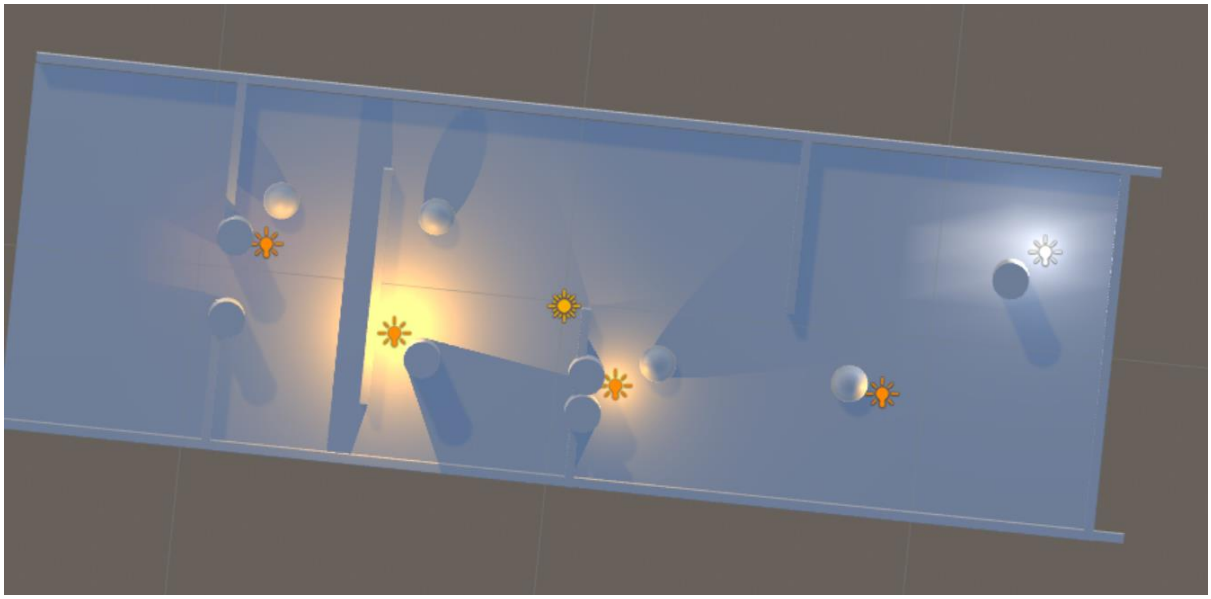
你想最终得到下面看起来像模型;n b 不一定是完美的搭配, 只是类似的东西。



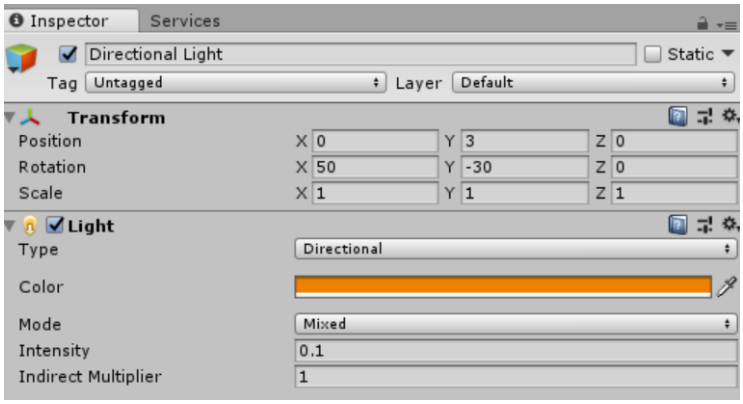
从这里, 填充它有一些圆柱体和球体

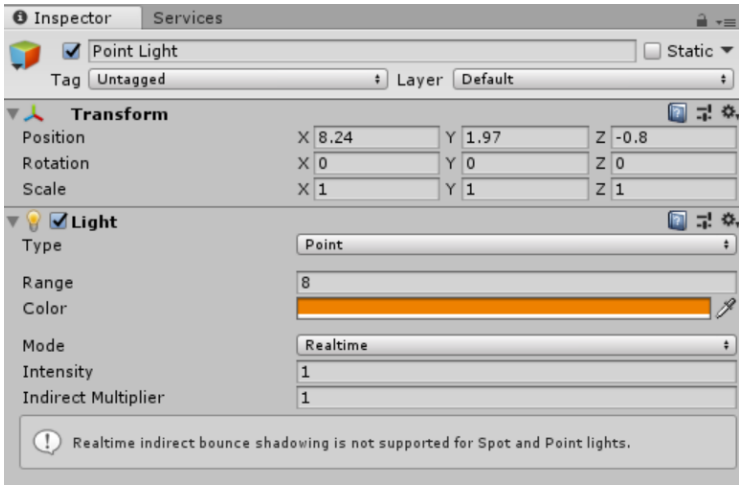


这些基元中的每一个都刚刚被旋转和缩放, 为我们提供了一个工作环境。 正如你所看到的, 我们开始做一个世界的位, 现在加入一些不同颜色的灯光提升世界。



这些灯只是点灯, 方向光的强度下降到0.1。

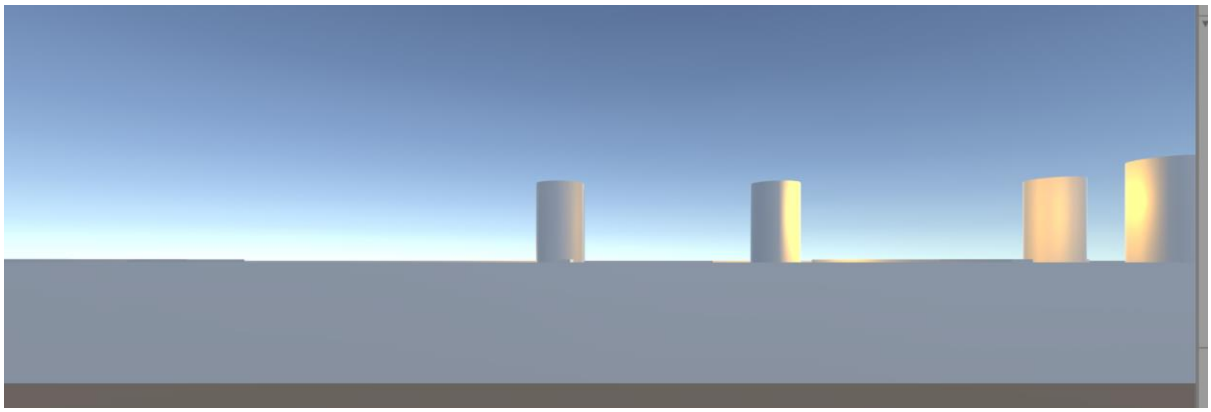




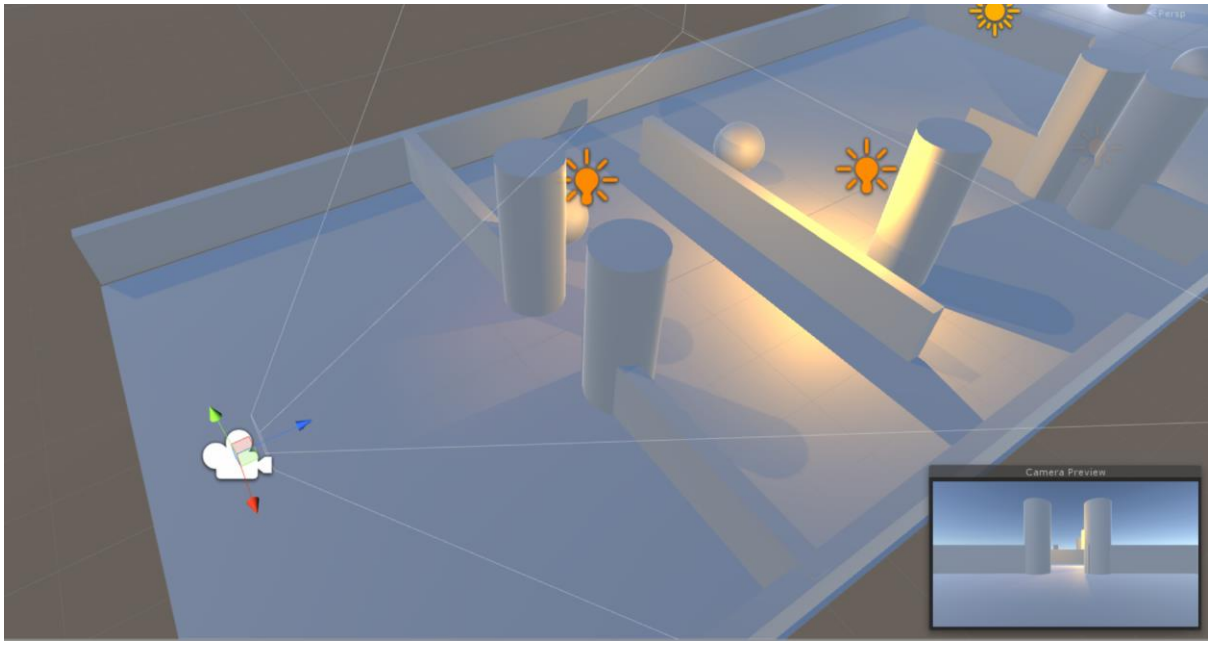
从这里, 我们现在将定位我们将在整个环境中移动的相机。

如果您点击顶部的播放图标

您将看到以下内容



这是因为我们需要将相机定位到起点。要做到这一点。单击 "关闭播放", 这样您就可以在场景视图中, 然后将相机放置在我们环境的起始区域。



正如您在右下角看到的, 相机视图是显示。这就是我们想看到的。虽然, 如果我们做一个马泽伊-地牢, 我们会想增加的高度所有的墙壁。

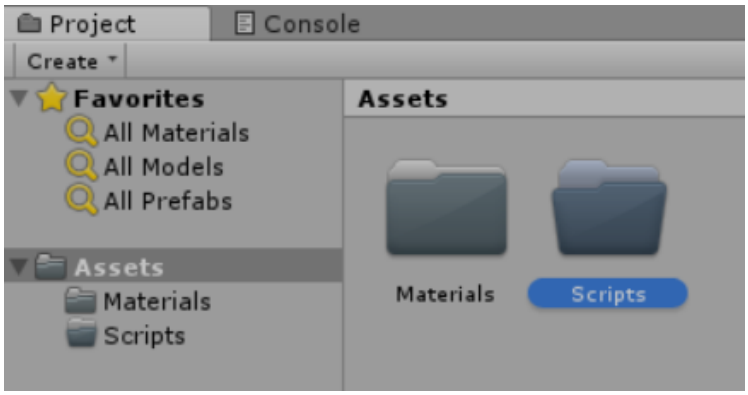
现在我们已经这样做了, 我们需要在相机中添加一个移动脚本。

在统一中, 有多种方法可以添加脚本, 我们可以创建脚本, 然后将其添加到对象, 在对象上创建脚本或获取已有的脚本并附加。

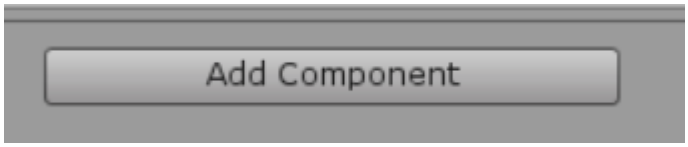
若要添加脚本, 我们将首先在 "资源" 部分创建一个名为 "脚本" 的文件夹, 然后使用 "添加组件" 创建脚本。完成此操作后, 我们将需要将新脚本拖到 "脚本" 文件夹中。

向 "资产" 添加新文件夹

右键单击 > 创建者 > 文件夹将其称为脚本

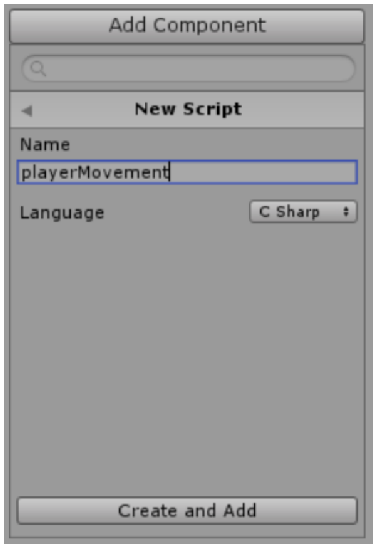


从这里单击层次结构中的 player 对象, 然后转到 "检查器" 面板, 然后单击 "添加组件"。

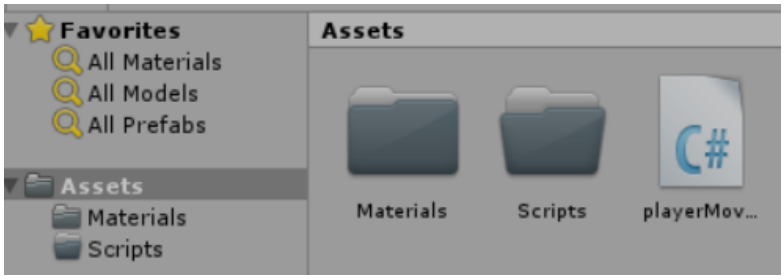


选择新脚本

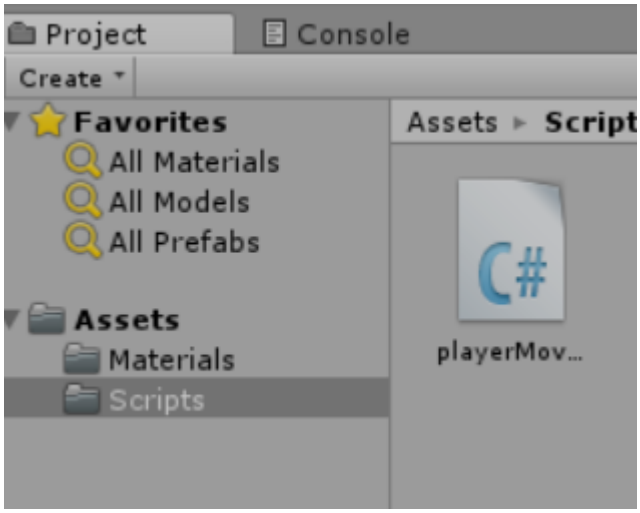
为脚本命名, 在这种情况下, 它被称为玩家移动.我们使用的语言 c 夏普, 你会看到它记录为 c# 有时。然后点击创建和添加



将脚本从 "资产根" 文件夹拖到脚本文件夹中

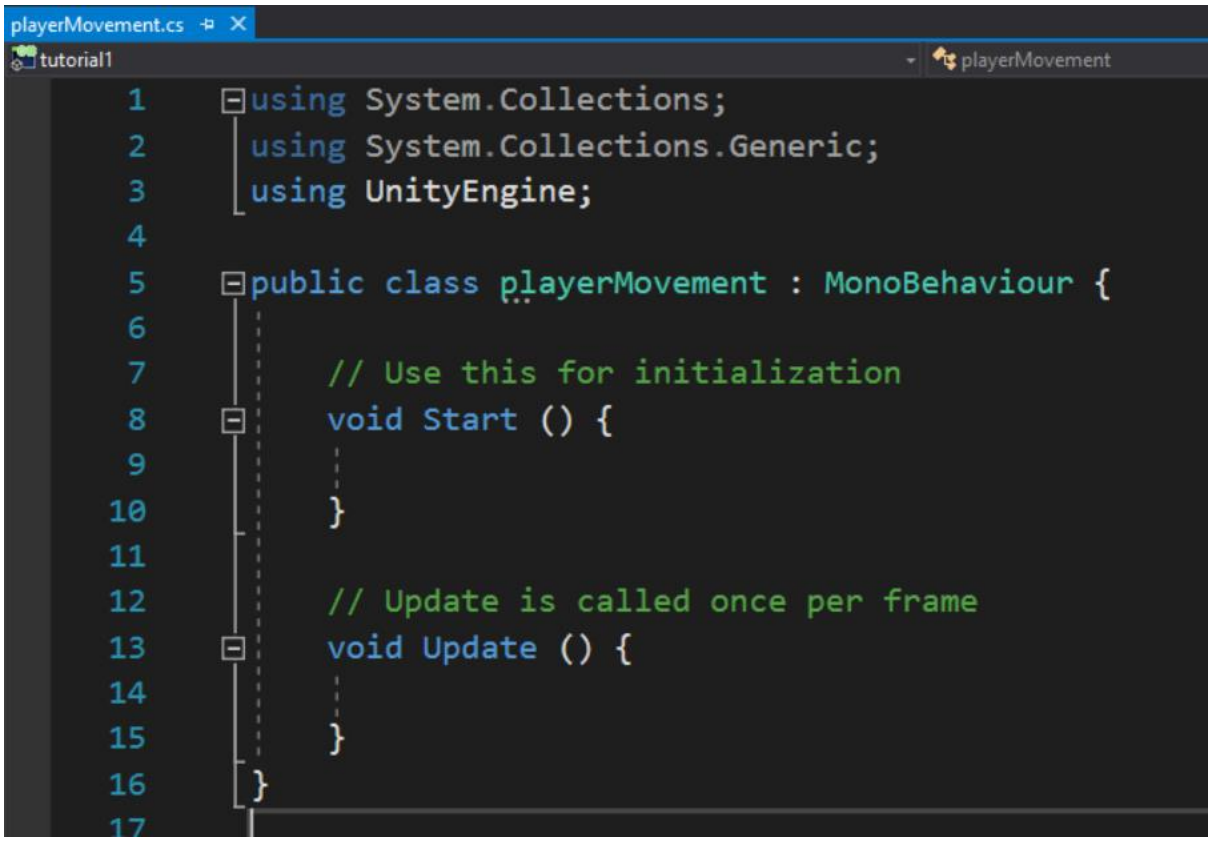


然后进入脚本文件夹



双击新创建的脚本。

随后将在编辑中开机, 这种情况下, 编辑使用的是2017年视觉工作室。您应该看到以下内容:



添加以下代码.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class playerMovement : MonoBehaviour {

    // Use this for initialization
    void Start () {

    }

    // Update is called once per frame
    void Update () {
        float vval = Input.GetAxis("Vertical");
        float hval = Input.GetAxis("Horizontal");
        float currentHeight = 0.0f;

        transform.Translate(hval, currentHeight, vval);
    }
}
```

那么, 这一切意味着什么呢？

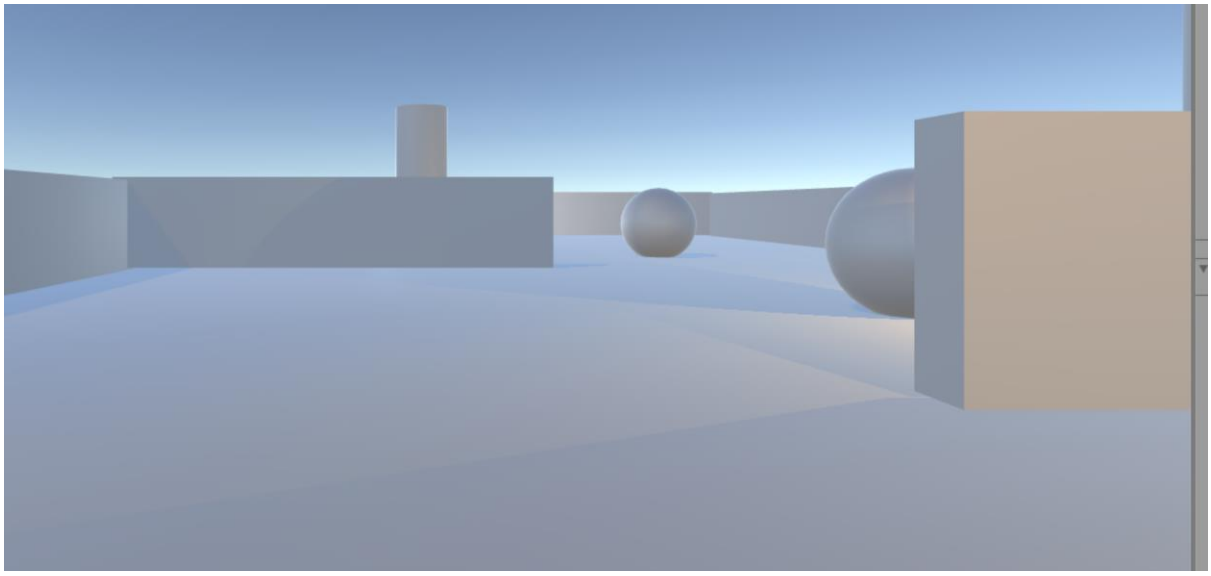
浮动值是一个数字,有能力包含小数。 瓦尔,赫瓦尔和电流高度都是变量, 名称可能已经什么但最好使用有意义的名字, 以方便理解。 变量是一个内存对象, 它可以包含不同的值, 但始终由一个名称引用。

当我们处理3d 世界时, 所有对象都有3轴,X, Y和 z。因此, 我们需要在每个轴中都有值。

从这里, 保存脚本。然后回到团结。

现在, 您点击顶部的播放图标

然后使用W, A, s、d 或光标键, 以在环境中移动相机。



在我的系统上, 相机移动得非常快。因此, 我对代码做了一个小的修改。此更改旨在操纵相机的速度。

首先, 我添加了一个名为 "速度" 的公共变量。

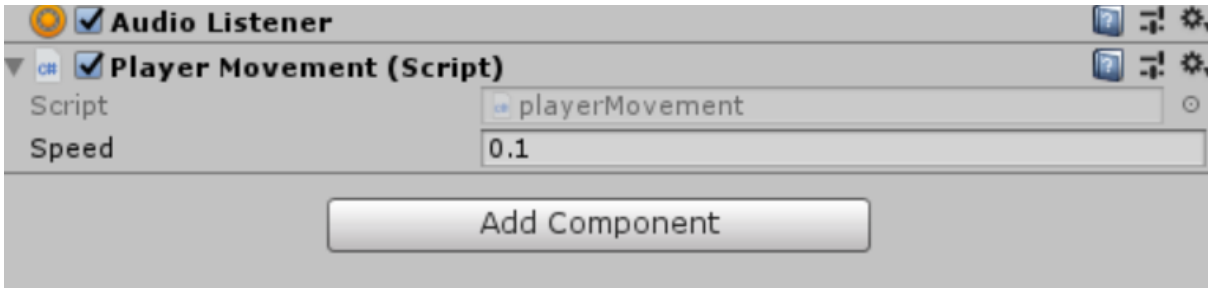
这样做的原因是, 它允许我手动分配一个数字的速度, 直到我找到一个数字, 我很高兴, 然后我可以代码的速度修饰符到程序, 在这种情况下, 它是0.1。

其次, 我成倍增加赫瓦尔和瓦尔在翻译之前的速度, 以这种方式, 一个新的数字被指定, 并降低了我的相机速度。

新代码应如下所示:

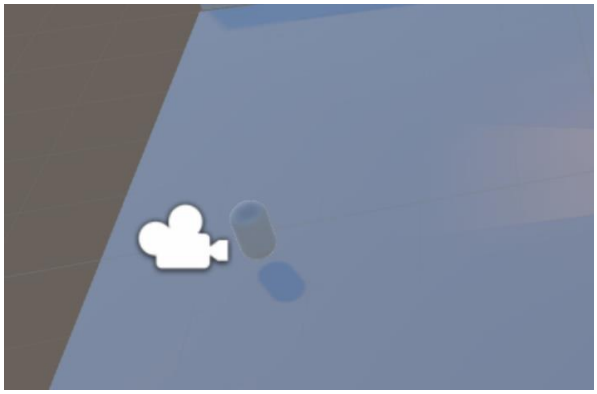
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class playerMovement : MonoBehaviour {

    public float speed = 0.1f;

    // Use this for initialization
    void Start () {

    }

    // Update is called once per frame
    void Update () {
        float vval = Input.GetAxis("Vertical");
        float hval = Input.GetAxis("Horizontal");
        float currentHeight = 0.0f;
        hval = hval * speed;
        vval = vval * speed;

        transform.Translate(hval, currentHeight, vval);
    }
}
```

如果你在检查器中注意到统一, 脚本区域内就会有一个新的盒子, 如果你愿意, 你可以在里面更改号码。

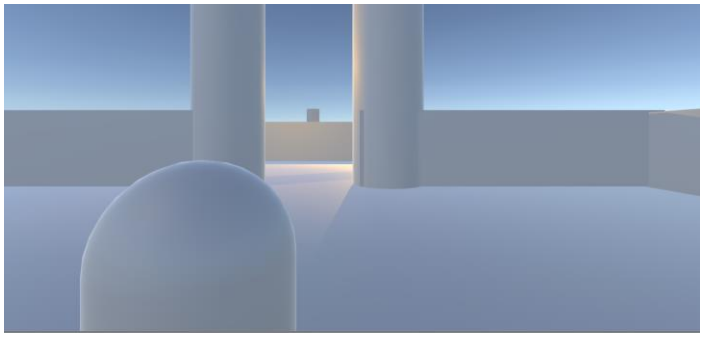


现在我们有了周围的动作, 让我们在相机中添加一个玩家, 所以这个动作并不总是第一人称, 而是更多的第三人称。

我们将使用一个圆柱体作为球员。创建一个胶囊, 然后将其放置在相机前面。

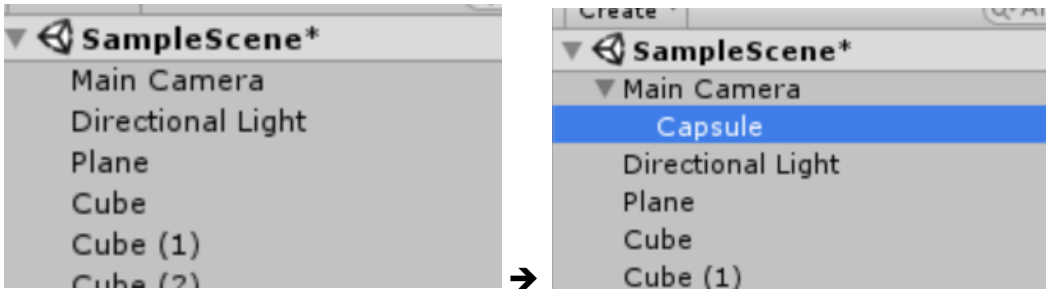


尽管胶囊如果漂浮, 从相机的角度来看, 它看起来非常好。

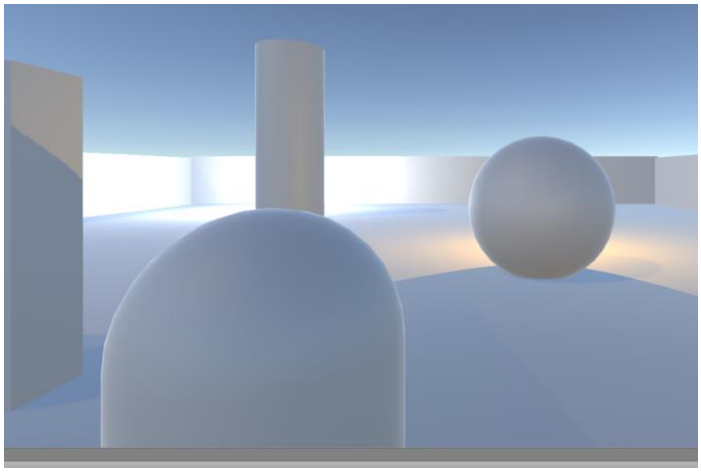


而这才是最重要的, 要创造角色处于正确高度的假象。

现在, 测试游戏。如果你运行它起来你会看到相机移动, 我们的太空舱停留在我们开始它的位置。要解决这个问题, 我们需要把胶囊变成相机的孩子。这可以通过将胶囊拖放到主相机上来实现。



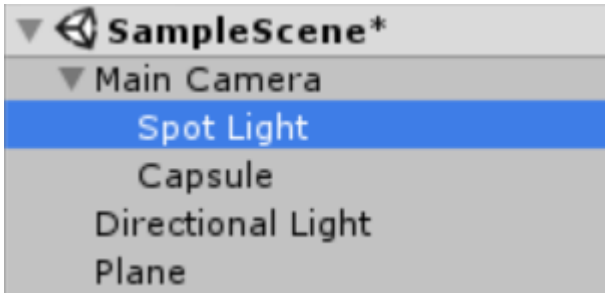
从这里开始, 再次运行游戏并进行测试。

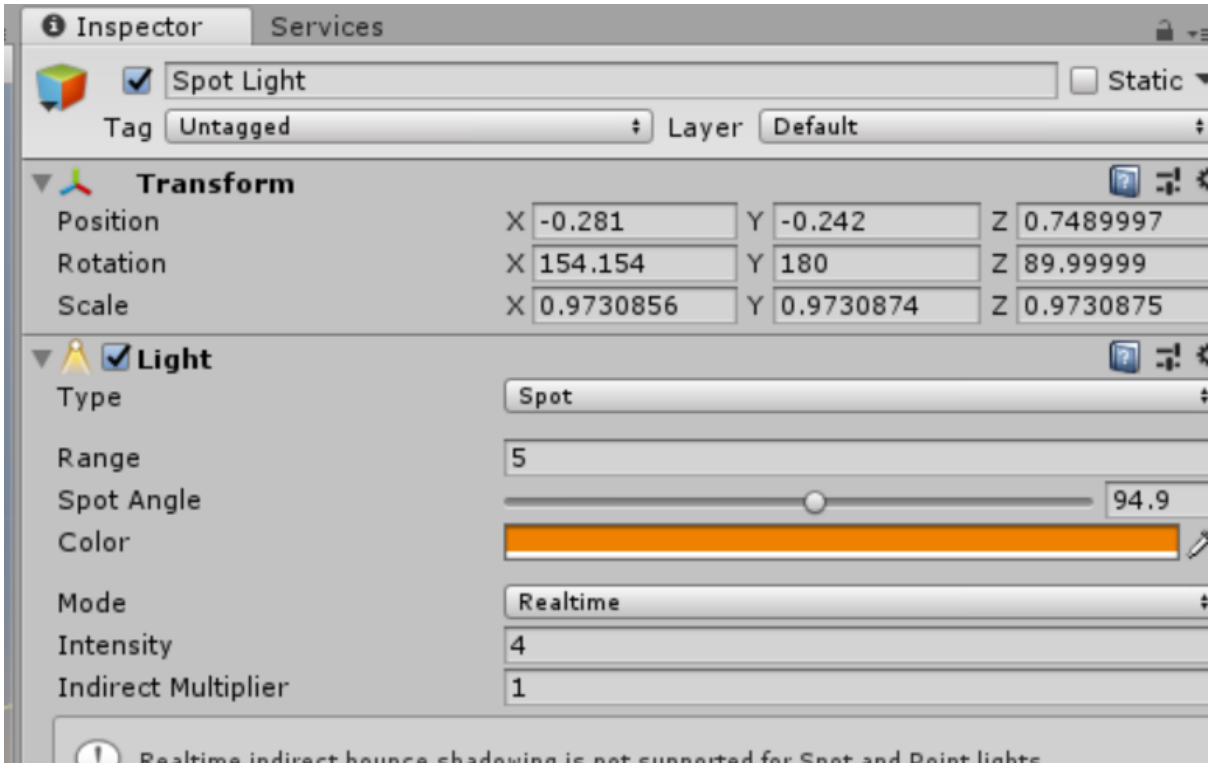


现在, 当我们跑来跑去的时候, 太空派玩家就会和镜头呆在一起.接下来, 我们将为角色添加一个聚光灯, 这样他们就可以照亮方向。

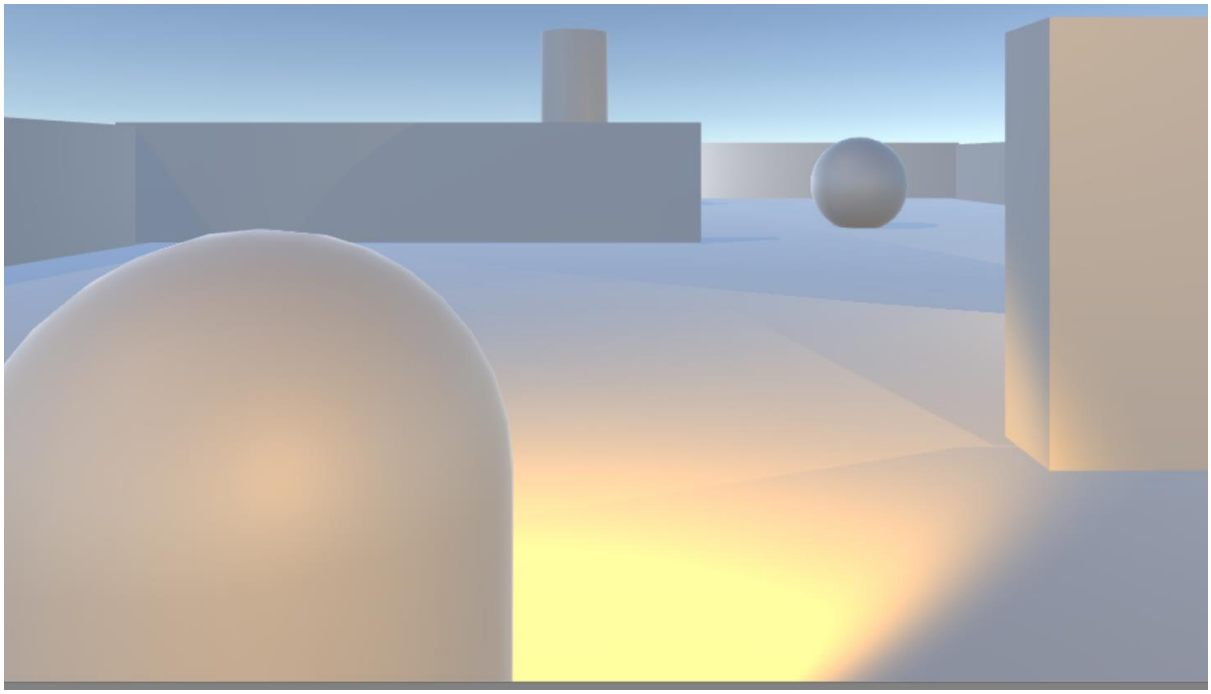
和以前一样, 创建聚光灯, 将其定位为相机的孩子, 然后进行测试。

改变颜色和范围的聚光灯, 给它额外的氛围。





在环境中四处走动时，这应该会给你以下视图。



只是为了好玩，增加墙壁的大小，所以玩家看不到马泽西地牢的样子。