# Holograms 210 凝视

本节参考官网地址：（ <https://developer.microsoft.com/en-us/windows/mixed-reality/holograms_210> ）

## 本节前提（环境，工具，及基本编程能力）

1.[安装](https://developer.microsoft.com/en-us/windows/mixed-reality/install_the_tools)了正确[工具的](https://developer.microsoft.com/en-us/windows/mixed-reality/install_the_tools) Windows 10 PC 。

2.一些基本的C＃编程能力。

3.Hololens设备。

4.Unity版本已升级至2017.2.0以对应微软官网最新。

5. MixedRealityToolkit-Unity使用MixedRealityToolkit存储库中的代码，使得混合现实开发在Unity中更容易进行。本节将导入MixedRealityToolkit工具。（下载地址如下）

开发工具下载地址

Windows 10（专业版、企业版或教育版）： <https://www.microsoft.com/zh-cn/software-download/windows10>

Visual Studio 2017：<https://developer.microsoft.com/zh-cn/windows/downloads>

Unity2017.2.0或更高： <https://store.unity.com/download>

MixedRealityToolkit-Unity(基于Unity的混合显示开发工具包) :

<https://github.com/Microsoft/MixedRealityToolkit-Unity>

凝视是输入的第一种形式，它揭示了用户的意图和意识。Holograms 210(又名Project Explorer)深入研究了与gaze相关的Windows全息概念。我们将为光标和全息图添加上下文意识，充分利用你的应用了解用户的目光。

### 首先打开官网下载该节课程所需项目资源

<https://developer.microsoft.com/en-us/windows/mixed-reality/holograms_210>



**注意：**Unity不同版本对应的文件。

## 第一章 - Unity设置

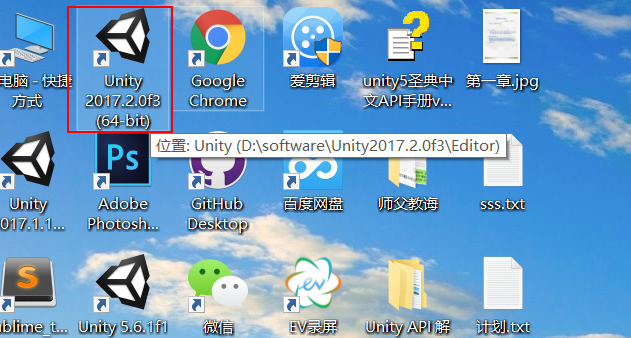
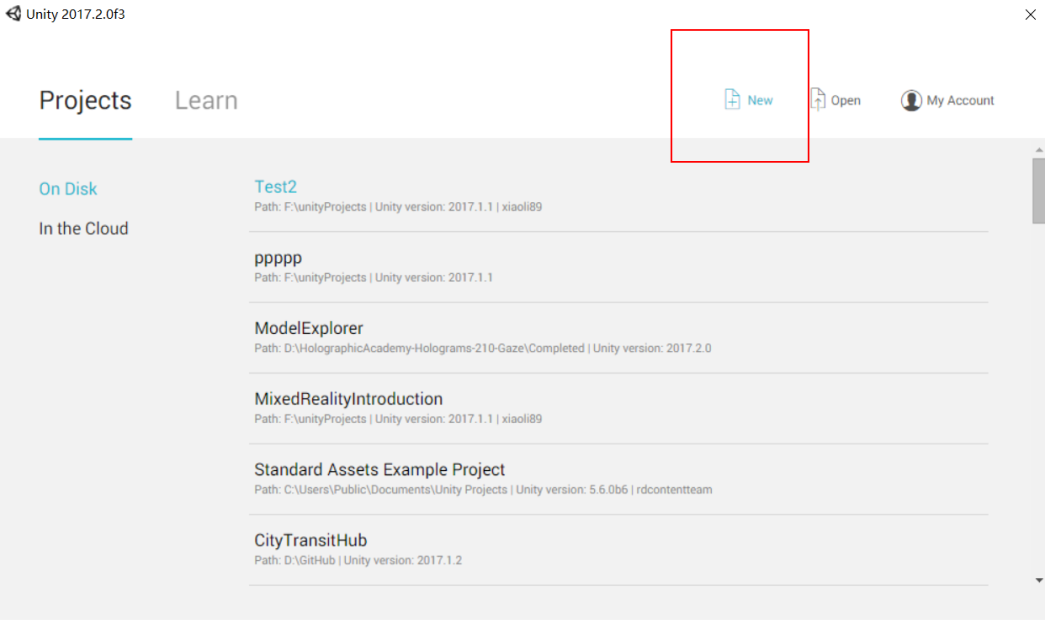
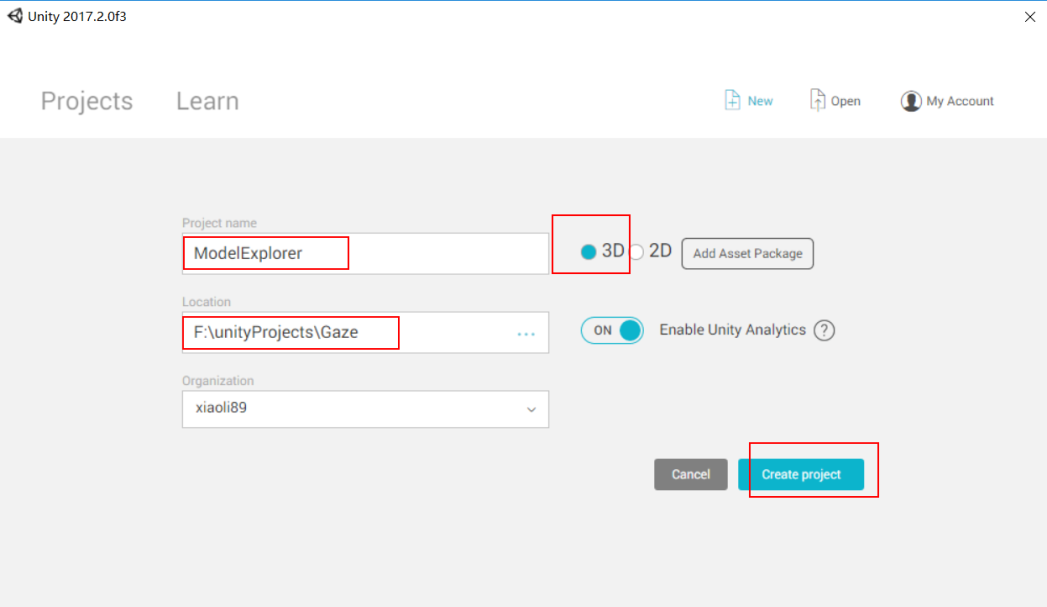
### 本章目标：

1.优化Unity对Hololens的开发。

2.导入资源并设置场景。

3.在Hololens中查看项目中的宇航员。

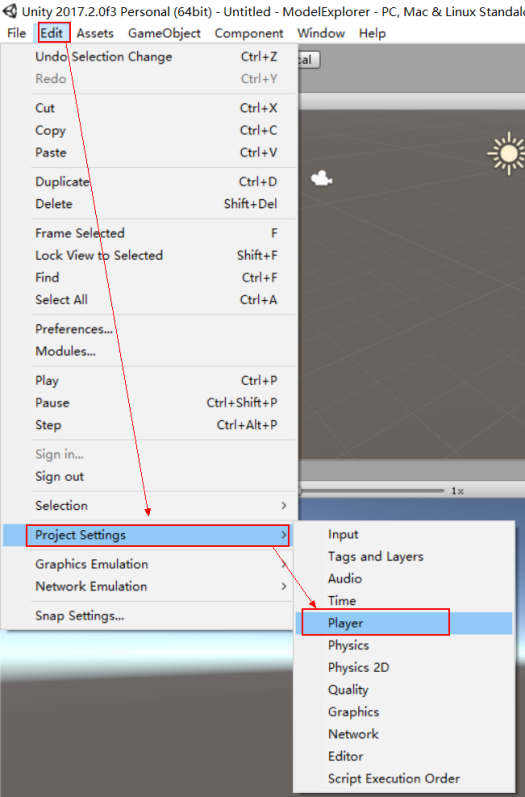
### 第一步：创建项目

1. 启动Unity
2. 选择New
3. 进入Project name将项目命名为ModelExplorer。
4. 进入Location选择项目的存储路径，新建一个文件夹命名为Gaze,保存至该文件夹下。
5. 确保项目设置为3D。
6. 单击Create project创建新项目。

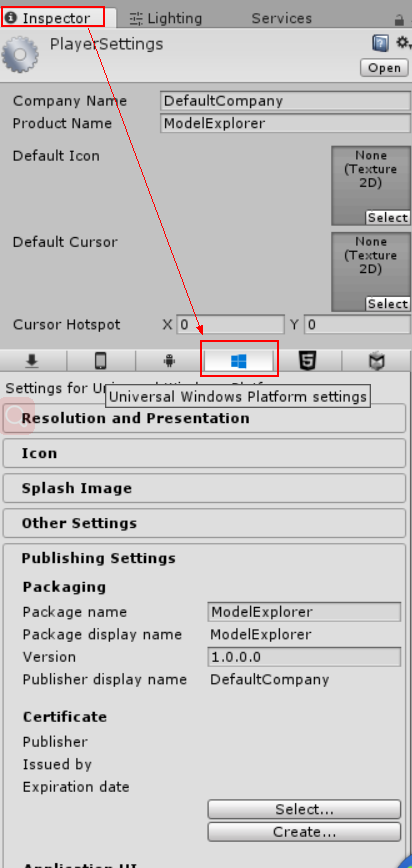
### 第二步：HoloLens的Unity设置

我们需要让Unity知道我们试图导出的应用程序应该创建一个全息视图而不是2D视图。我们通过添加HoloLens作为虚拟现实设备来做到这一点。

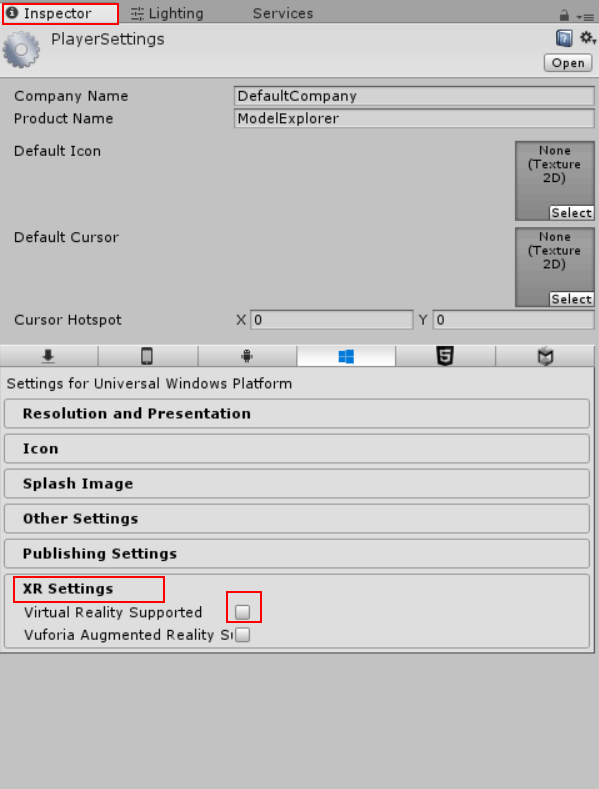
1. 到Unity中，Edit>Project Setting>Player

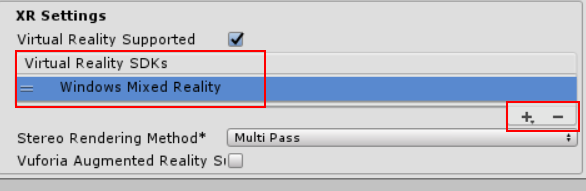


1. 到**Inspector**面板选择**Windows Store** Icon

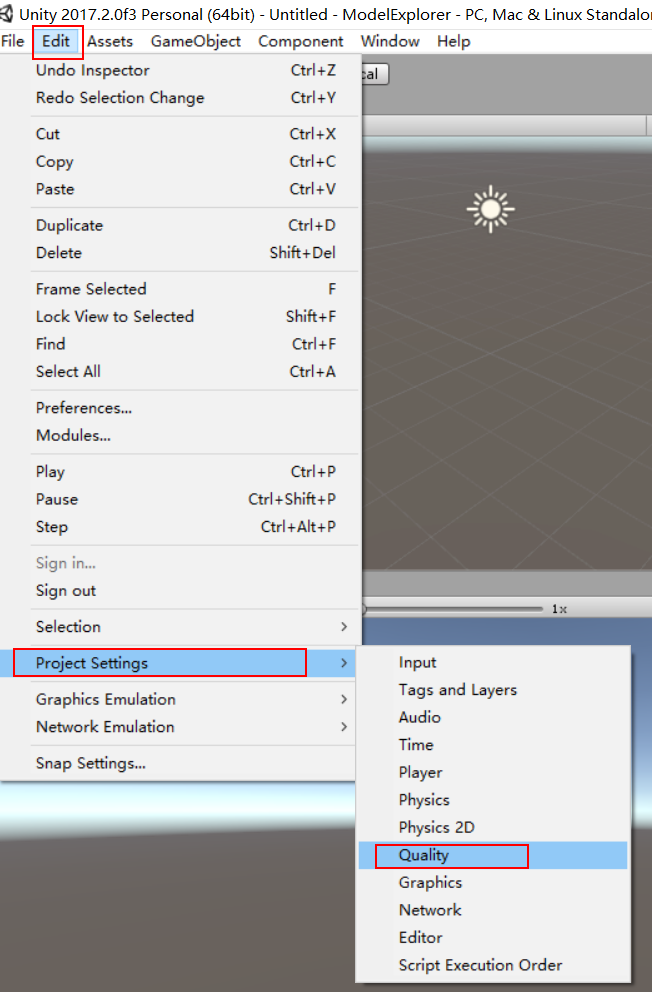


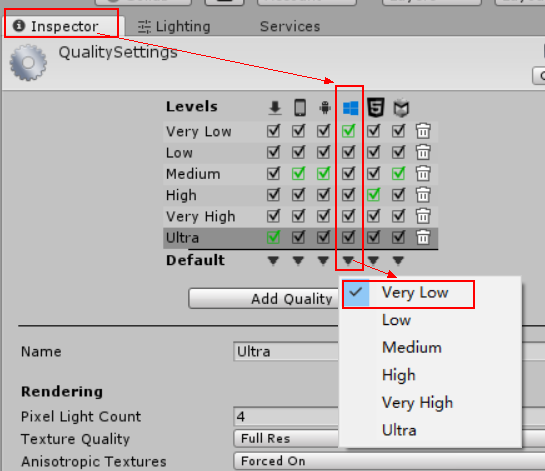
1. 展开 **XR Settings** 勾选**Virtual Reality Supported**后的选择框。会出现**Virtual Reality SDKs**为Windows Mixed Reality。如果没有，可以点击右下角的“+”号进行添加。（此处之前版本实展开Other Settings，在Rendering下设置的）





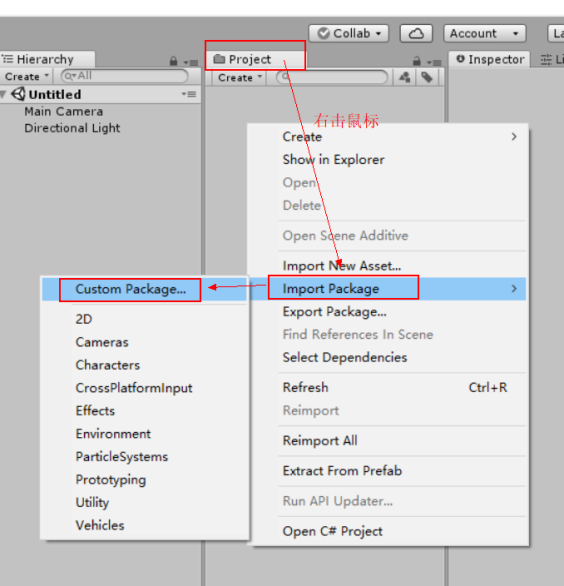
1. 到Unity中，Edit>Project Setting>Quality进行质量设置。到**Inspector**面板，点击**Windows Store** Icon列最下面的小三角，选择Very Low(最快)。

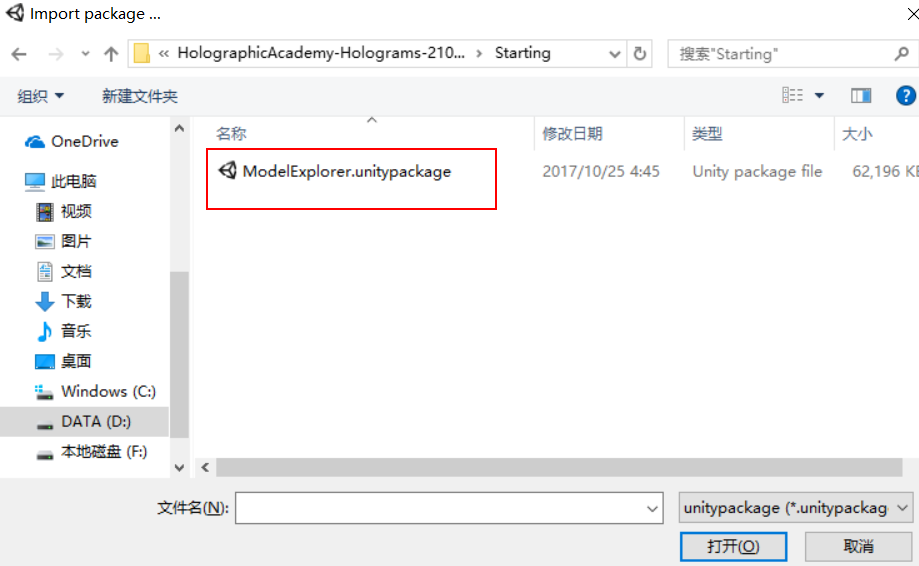


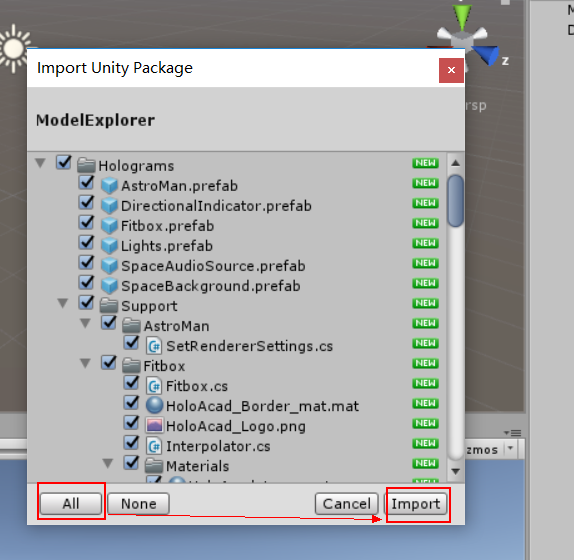


### 第三步：导入资源

1.在Project面板右击鼠标，选择Import Package > Custom Package 导航至已下载的项目资源包所在路径，选择**ModelExplorer.unitypackage** 资源包,然后选择All,点击**Import**导入Unity 。

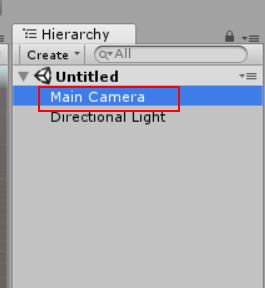




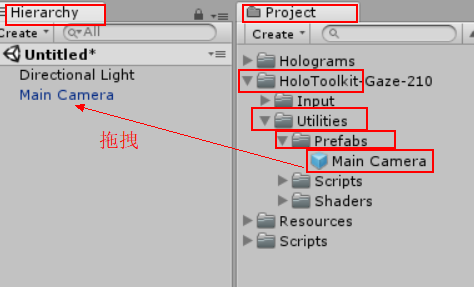


### 第四步：设置场景

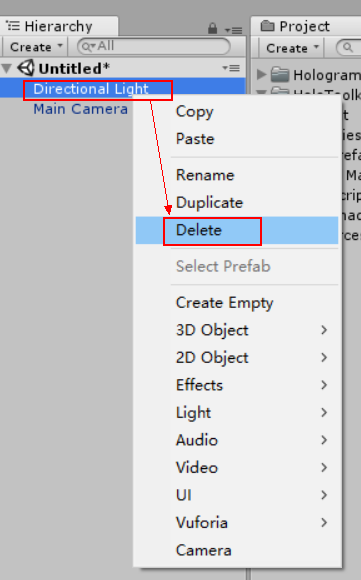
1.在**Hierarchy**面板选中Main Camera,按Delete键删除。



2.  在**HoloToolkit**文件夹下展开**Utilities**文件夹，接着展开**Prefabs**文件夹， 将**Main Camera**从**Prefabs**文件夹拖拽至**Hierarchy**面板中。

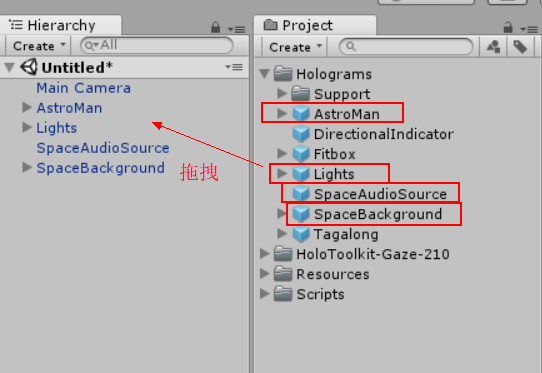


3. 右键单击层次结构中的方向指示灯，然后选择删除。

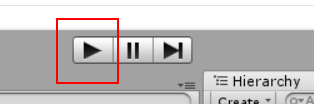


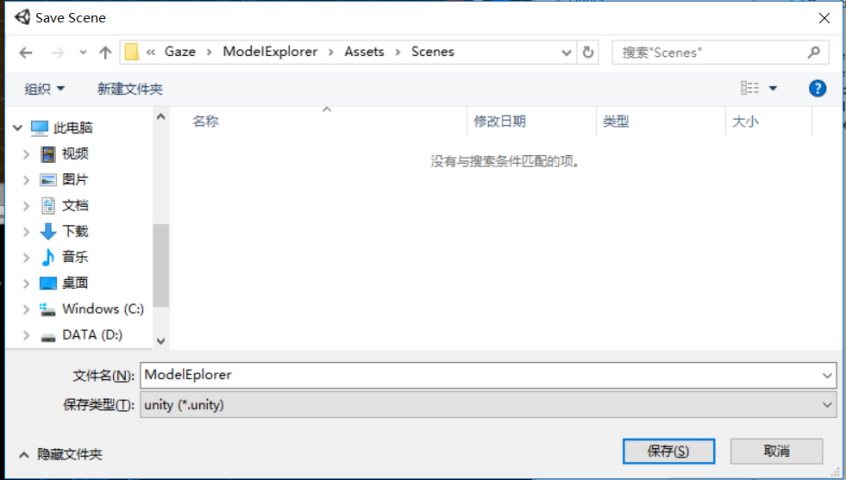
4.在Project面板，将**Holograms**文件夹下的下列资源分别拖入**Hierarchy**面板。

**AstroMan，Lights，SpaceAudioSource，SpaceBackground**

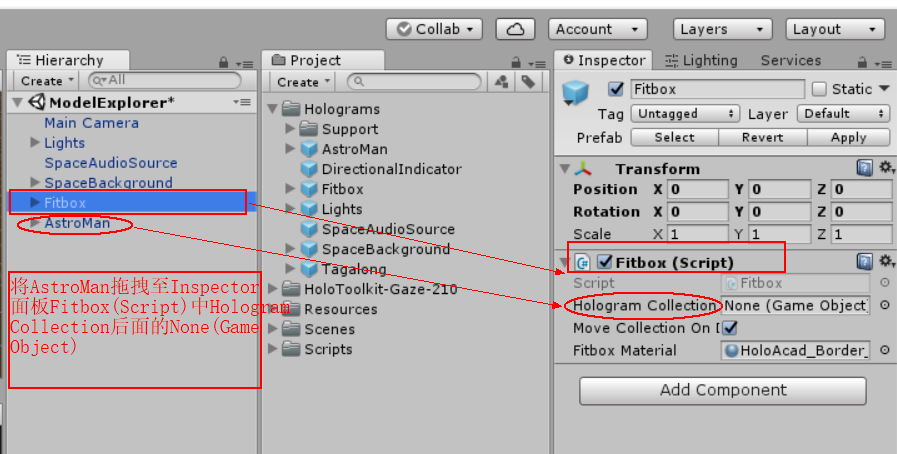


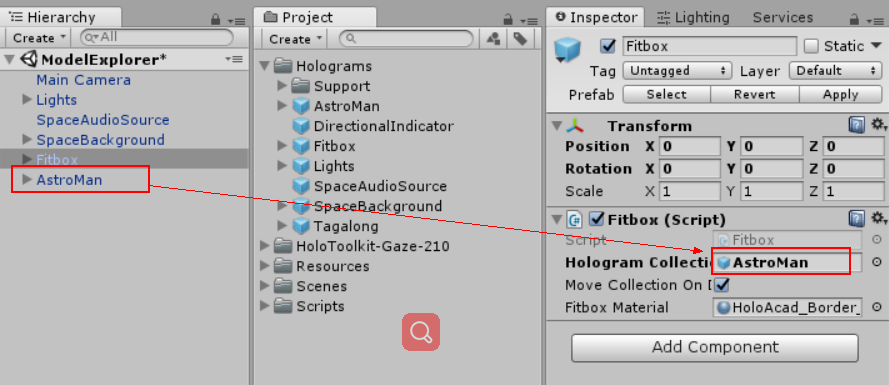
1. 现在可以点击播放查看场景中的宇航员，再次点击播放按钮即可停止播放。**Ctrl+S**保存场景到新建的**Scenes**文件夹下，命名为**ModeIExplorer**。





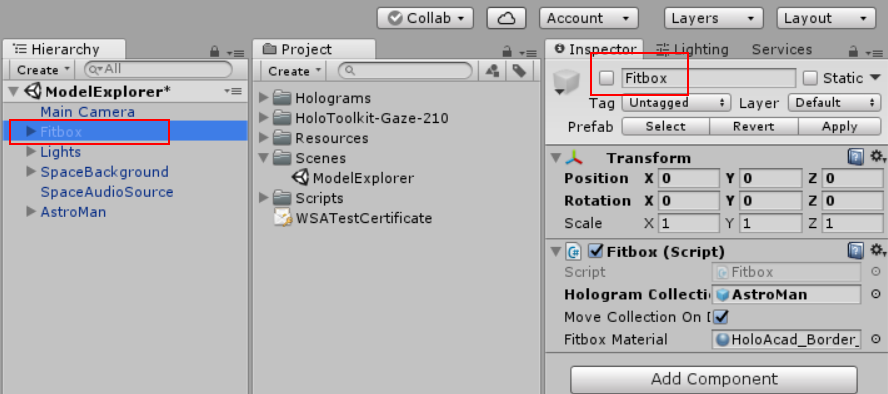
1. 将**Project**面板中**Holograms**文件夹下的**Fitbox**拖入**Hierarchy**面板， 然后在**Hierarchy**面板中选中**Fitbox，**将**AstroMan**拖拽至**Inspector**面板下的Hologram Collection 。



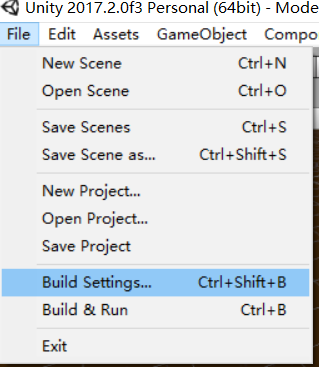
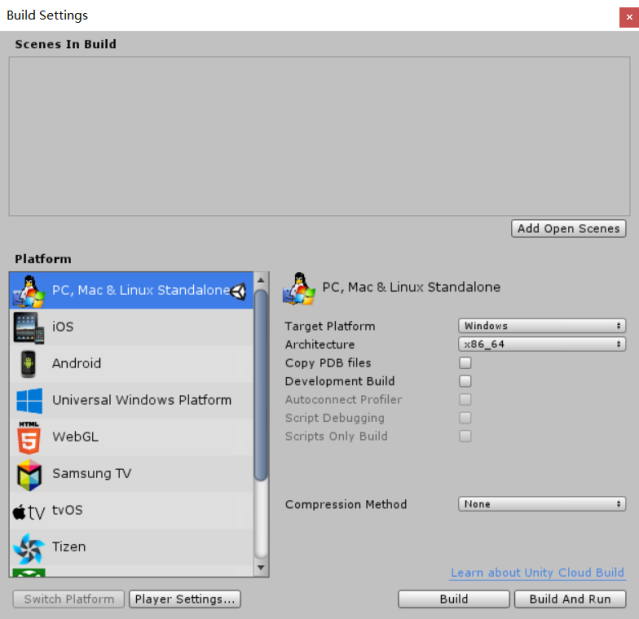
****

### 第五步：构建应用程序

**注意：**开始构建应用程序前，在Hierarchy面板选中Fitbox,然后在Inspector面板把Fitbox前面选择框的勾勾掉，使其处于未激活状态。



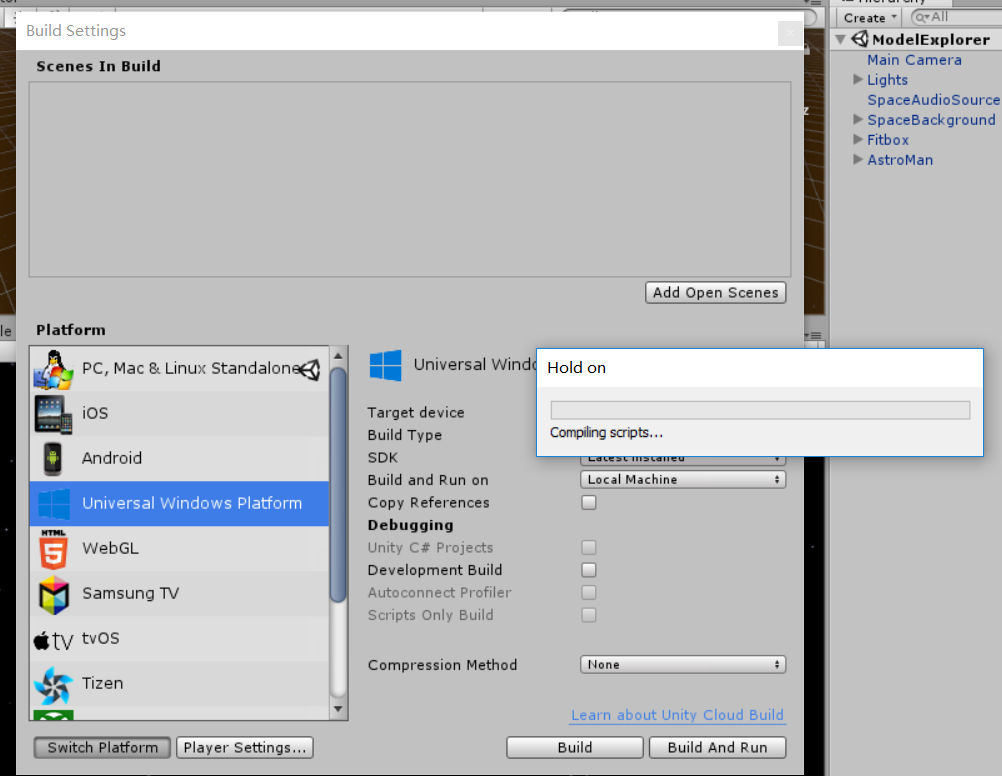
#### 1.选择File > Build Settings,打开构建设置。

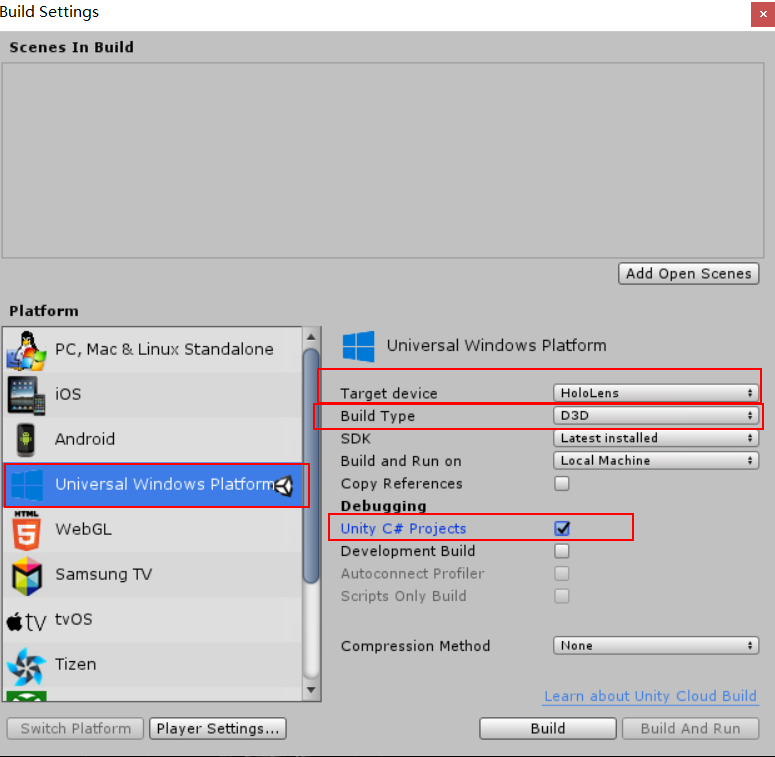
 

#### 2.平台设置

选择**Universal Windows Platform** **，**点击**Switch Platform**将构建目标转化为**Universal Windows** 平台**。Target device >Hololens ， Build Type >D3D ，**

**Unity C # Projects**勾选**。**

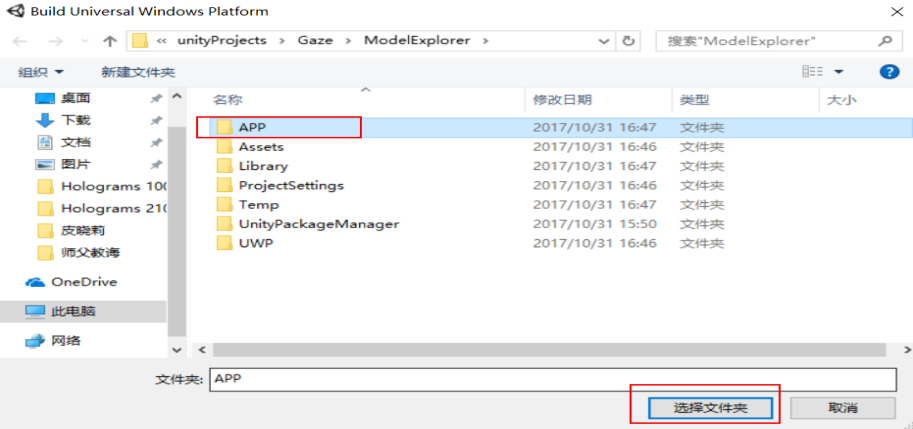
****



#### 3.选择项目保存路径

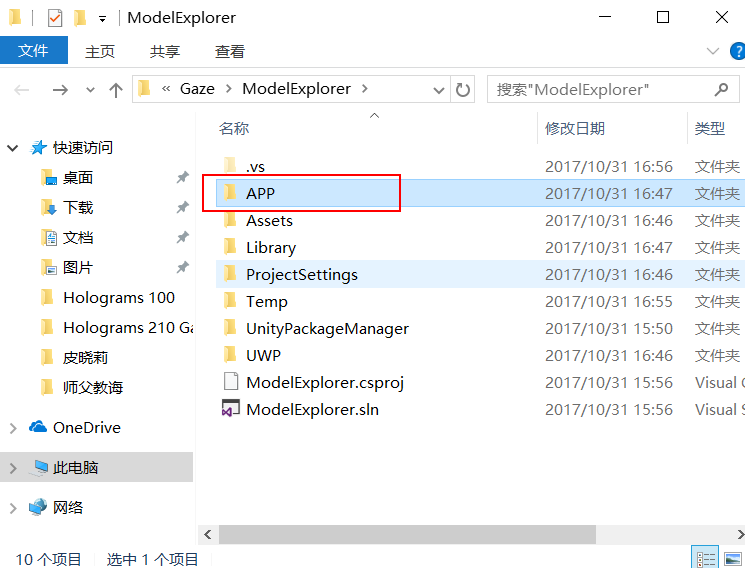
单击**Add Open Scenes**,将要打包的场景加入**Scenes In Build** 。单击**Build,**创建一个名为“App”文件夹，单击“App”文件夹，按该文件夹。

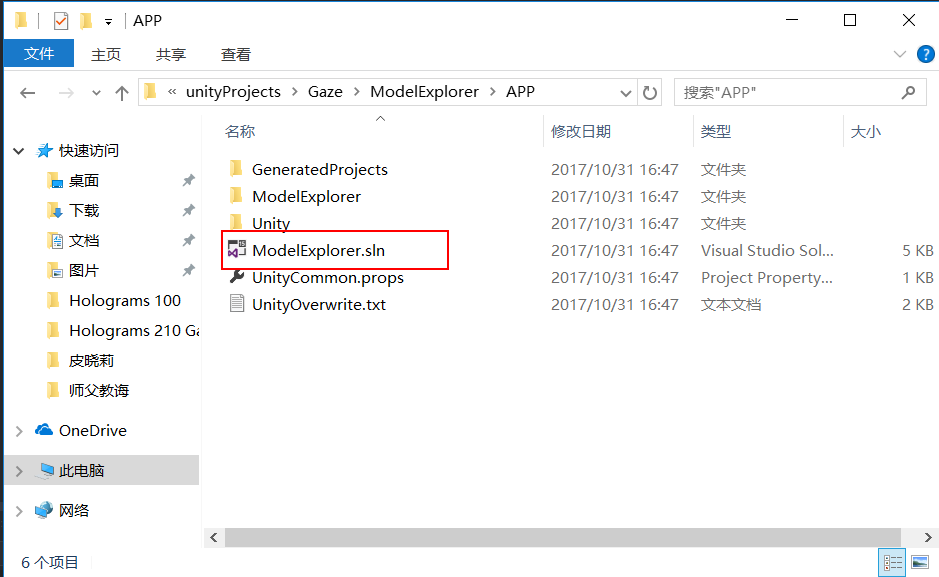




#### 4.打开项目解决方案

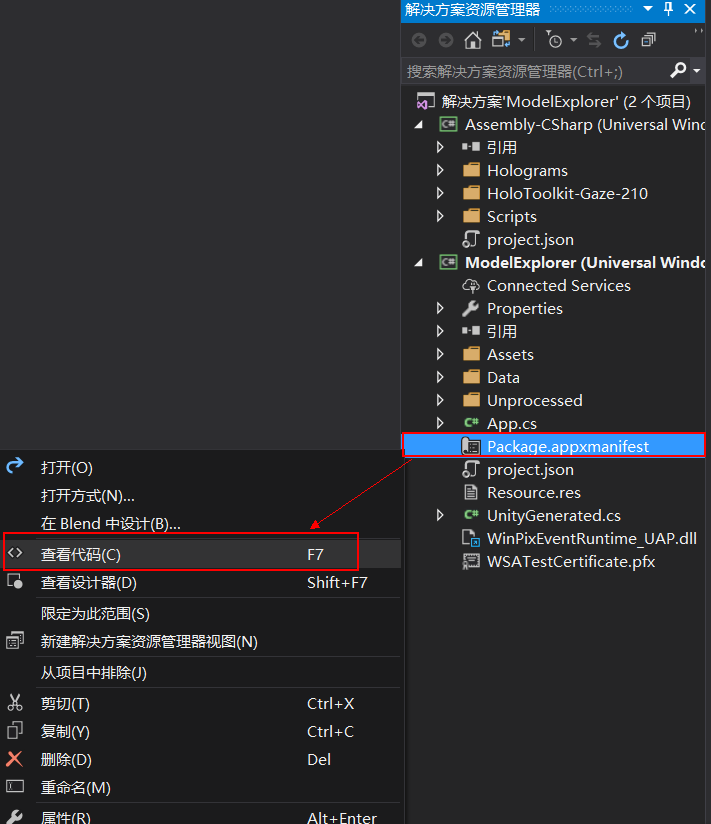
Build完成后出现下面文件夹，点开文件夹APP,用Visual Studio将文件夹APP中的ModelExplorer.sln打开。

****



#### 5. Package.appxmanifest文件的设置

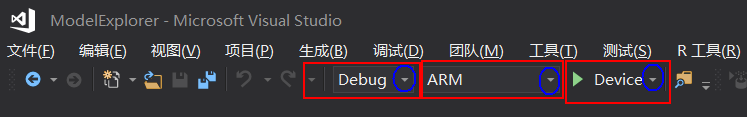
在Visual Studio中，右键单击解决方案资源管理器中的**Package.appxmanifest**，然后选择查看代码，找到指定**TargetDeviceFamily**行，并将Name = **“Windows. Universal”**更改为Name =**“Windows.Holographic”**。然后Ctrl+S保存 Package.appxmanifest。





#### 6. Visual Studio顶部工具栏的设置

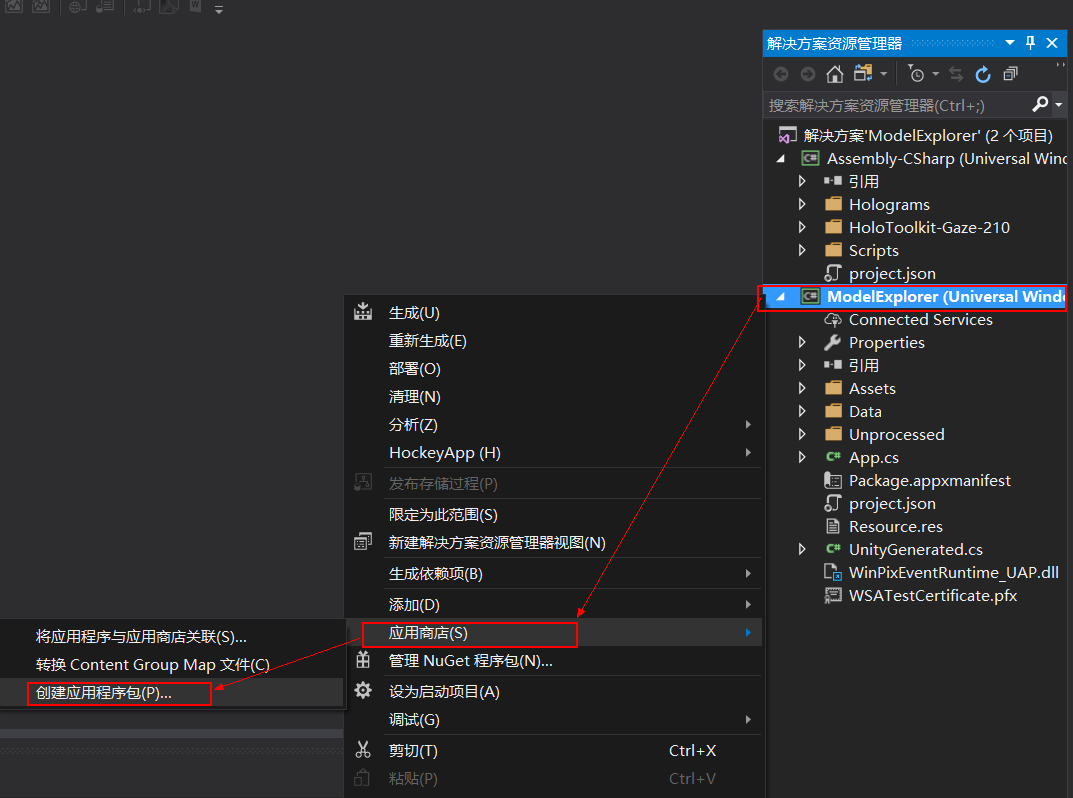
使用Visual Studio中的顶部工具栏，将目标从Debug更改为**Release**，ARM更改为**x86**。Device处变为本地计算机。





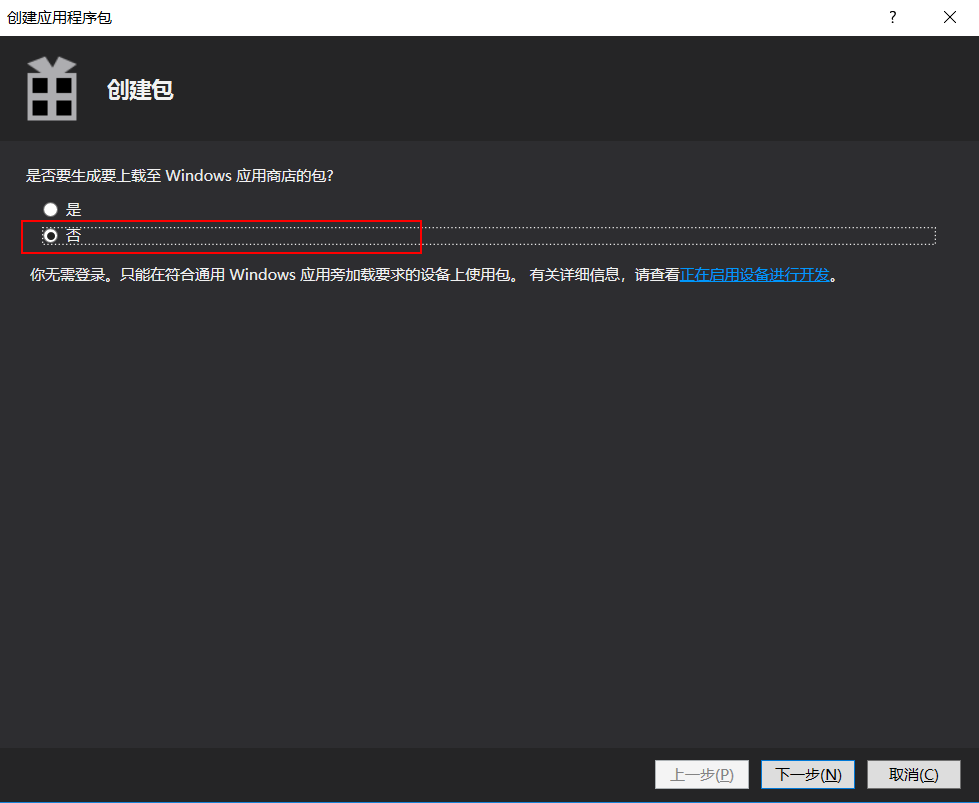
#### 7.开始创建应用商店程序

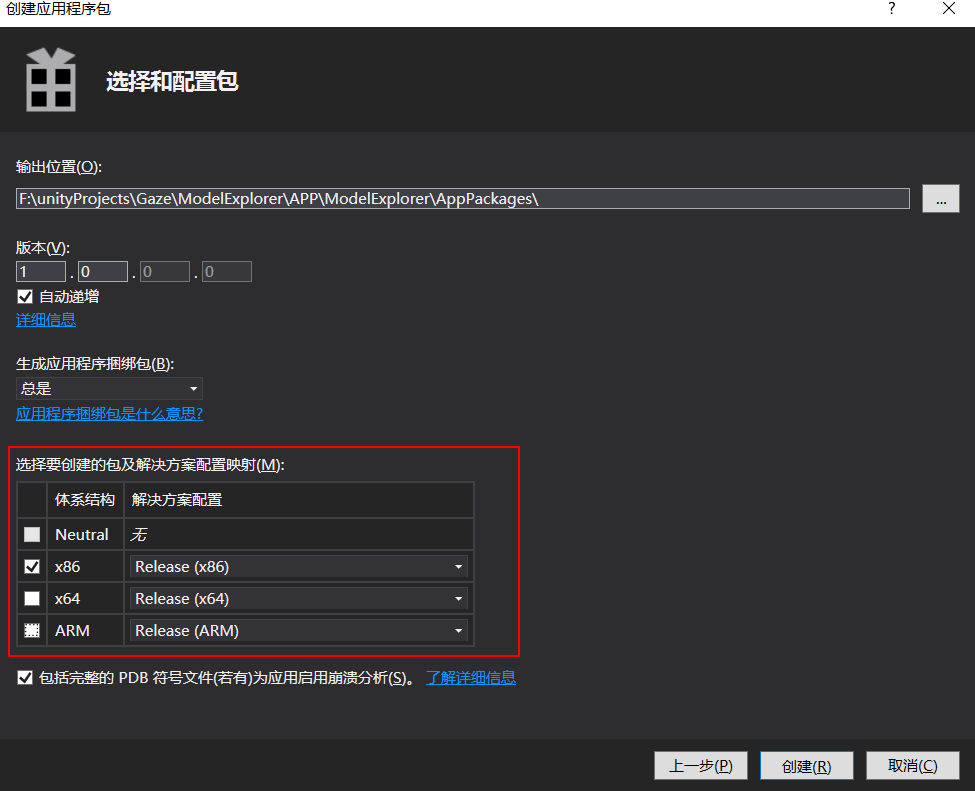
右键单击项目解决方案，选择应用商店，开始创建应用程序包。

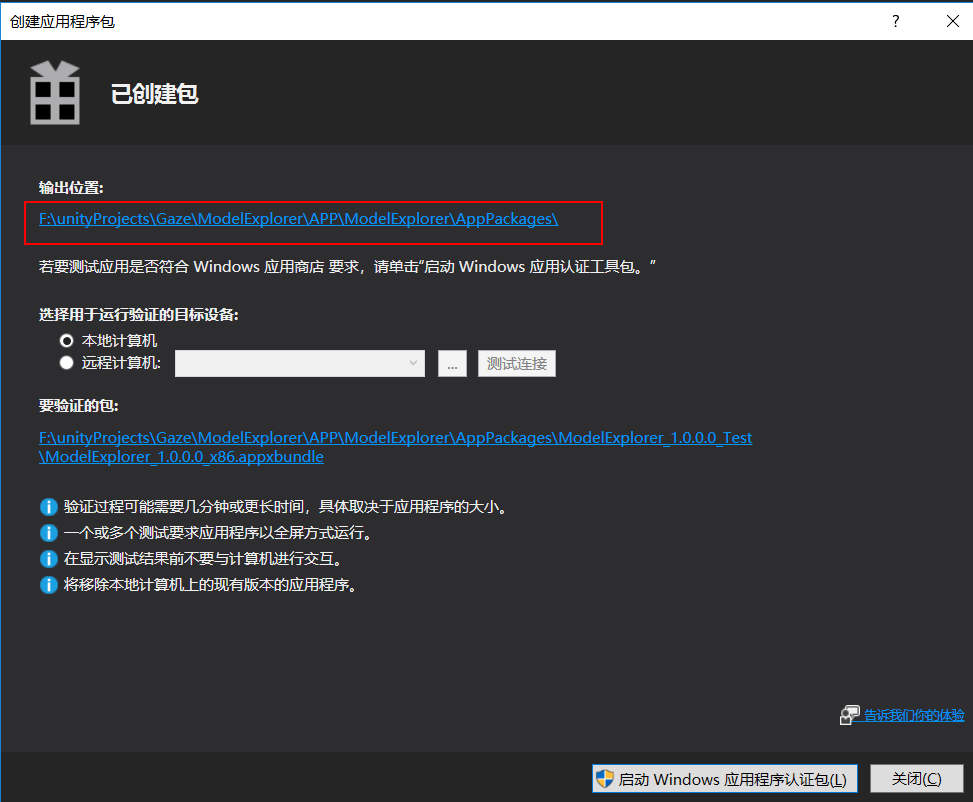


#### 8.选择和配置包

出现创建应用程序窗口，点否，点击下一步。然后选择x86,Release(x86),点击创建。创建完毕，窗口显示应用程序包输出位置。







#### 9.在Hololens上查看

将应用程序部署到Hololens,便可在Hololens上查看。

## 第二章 – 凝视光标和反馈

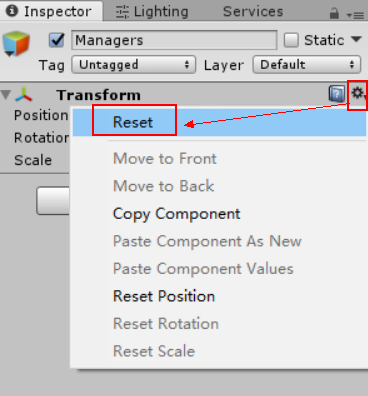
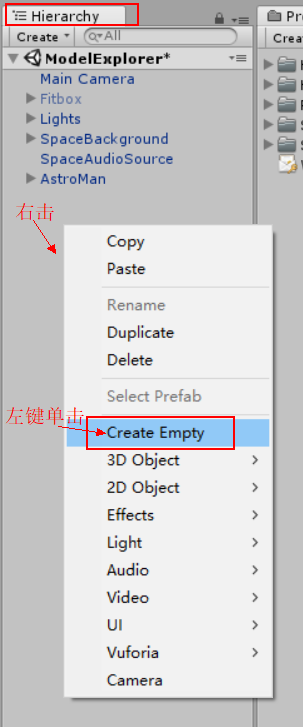
### 本章目标

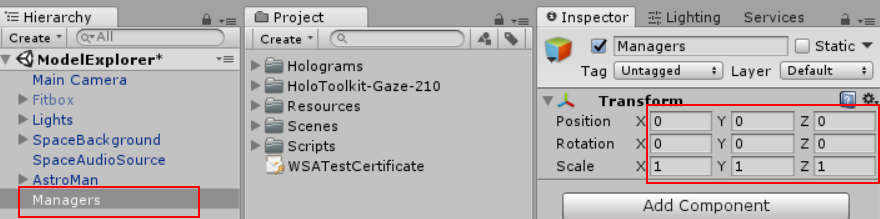
1.光标视觉设计和行为。2.基于注视的光标反馈。3.基于凝视的全息图反馈。

我们的光标需要满足始终存在,不能太小或太大，避免阻碍内容等一些光标基本设计原则。

### 第一步：Hierarchy面板创建Manager，Cursor

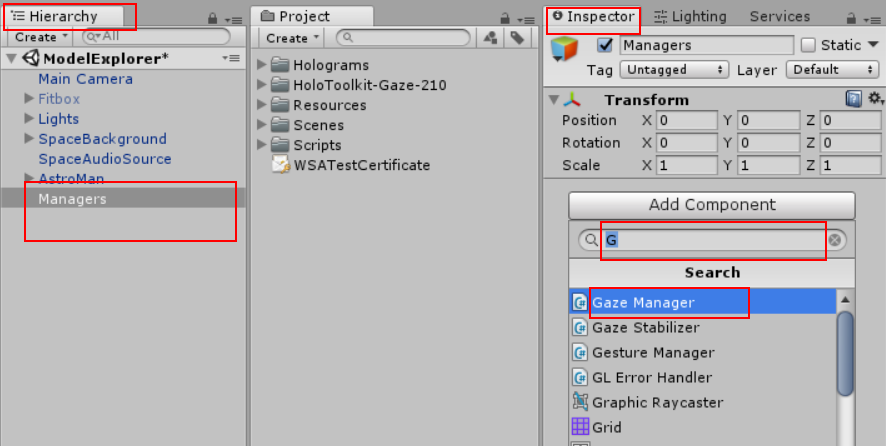
在Hierarchy层次面板右键单击，然后左键单击选择Create Empty创建空物体，然后改名为Managers。在Inspector面板左键单击Transform组件的设置图标，点击Reset将位置，旋转参数全部置为零，缩放置为1。

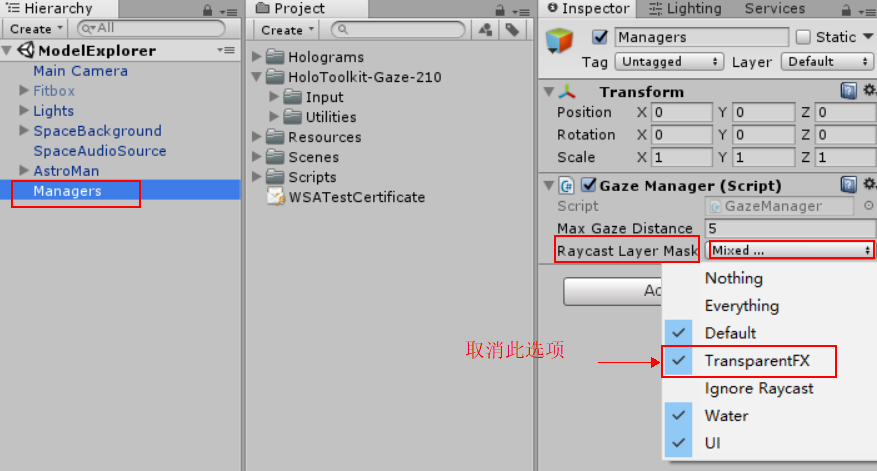




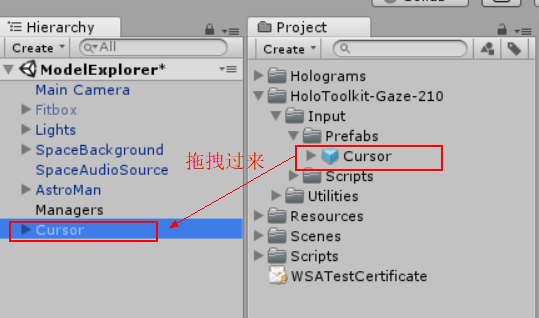
### 第二步：给Manager，Cursor 添加所需组件

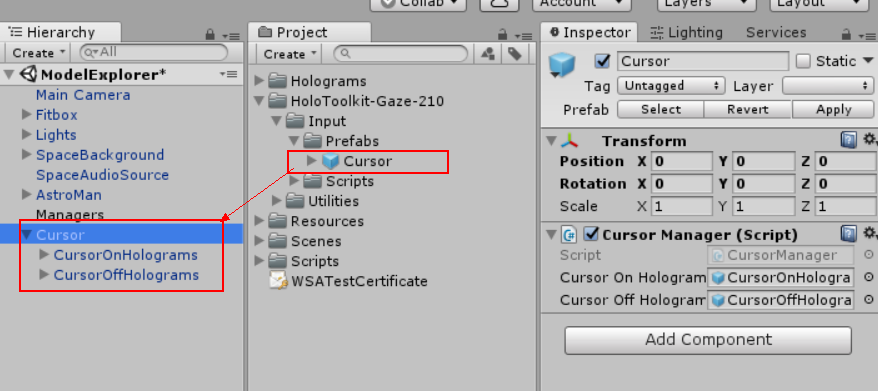
在Hierarchy面板选中Manager,到Inspector面板点击Add Component，在搜索框输入Gaze Manager,其实输入G时，下面就出现Gaze Manager,选中Gaze Manager为Managers添加Gaze Manager脚本组件。然后选择该组件内RaycastLayerMask下拉菜单并取消选“TransparentFX”。





在Inspector面板HoloToolkit \ Input \ Prefabs文件夹中，找到Cursor预制体，将其拖入Hierarchy面板。





HoloToolkit提供的Gaze Manager和Cursor Manager的脚本可以帮助我快速开发，当然我们也可以自己写。（首先编辑GazeManager.cs文件以执行以下任务：进行物理射线，存储Raycast投射到物体点的位置和法线，如果Raycast没有击中任何东西，请将位置设置为正常。可以找到注释来编写代码，标签为“编码练习” - 每个注释对应一行代码。 其次，您需要编辑CusorManager.cs以实现以下功能：决定哪个光标应该是活动的，根据是关闭

还是在全息图上更新光标，将光标放在用户注视的位置。 接下来，需要编辑InteractibleManager.cs和Interactible.cs才能实现以下功能：

在InteractibleManager.cs中，获取注视射线投射点并保存相关的GameObject。

发送GazeEntered消息，如果注视可以与之交互的对象。

如果远离可以与之交互的对象，发送GazeExited消息。

处理Interactible.cs中的GazeEntered和GazeExited回调。）

## 第三章 – 目标技术

### 目标

使目标全息图更容易。稳定自然头部运动。

### 步骤：Manager添加Gaze Stabilizer组件

在Hierarchy面板选中Manager,到Inspector面板点击Add Component，搜索Gaze Stabilizer,点击该组件添加。



## 第四章 - 定向指示器

### 目标

在光标上添加directional indicator方向指示器，以帮助找到全息图。

我们将使用DirectionalIndicator.cs文件，它将完成以下功能：

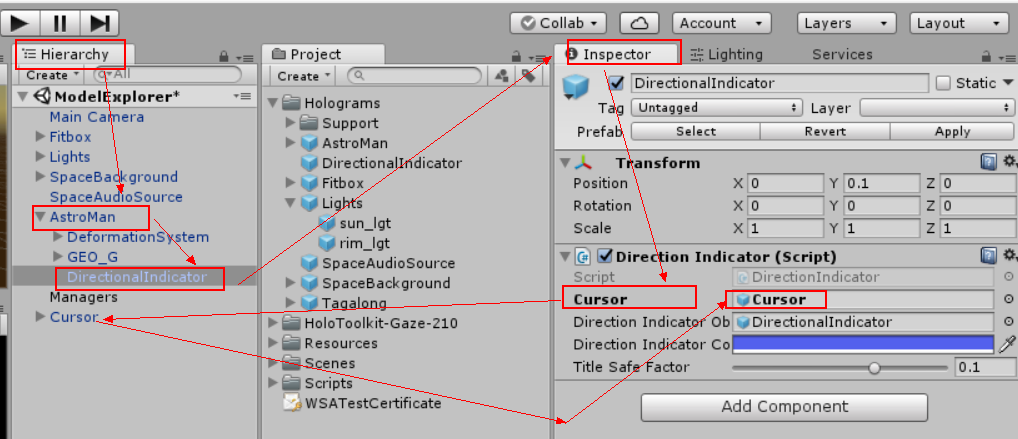
如果用户没有注视全息图，则显示方向指示器。

如果用户正在注视全息图，则隐藏方向指示器。

更新方向指示器以指向全息图。

### 第一步：为DirectionalIndicator指定Cusor

在Hierarchy面板选中AstroMan > DirectionalIndicator,再到Inspector面板，将Hierarchy面板中的Cusor拖拽至DirectionalIndicator属性中的Cusor。



### 第二步：部署至Hololens观察

如之前一般，构建并部署到Hololens, 观察方向指示器对象如何帮助您找到全息宇航员。

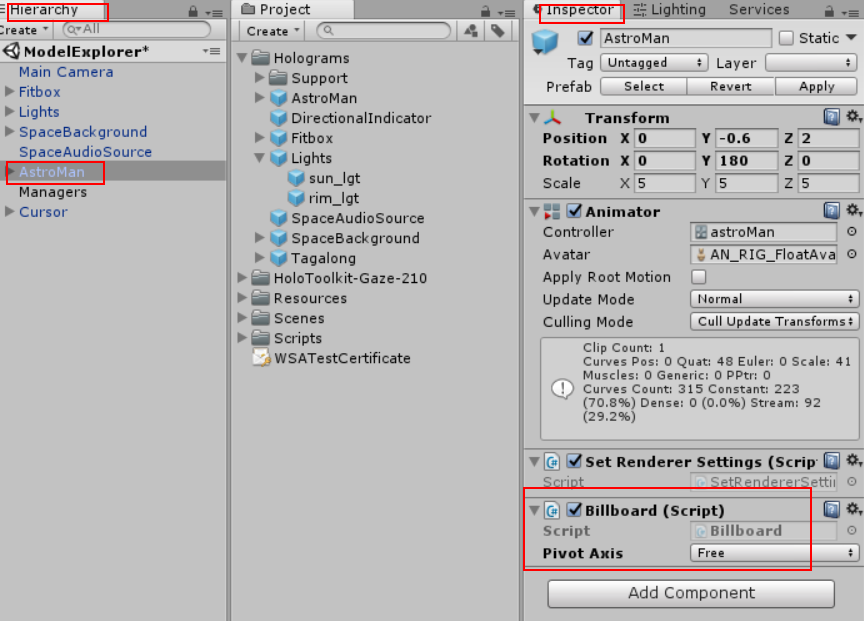
## 第五章 - 广告牌（Billboarding）

### 目标

使用广告牌让全息图总是面向你。

### 第一步：为全息物体添加Billboard组件

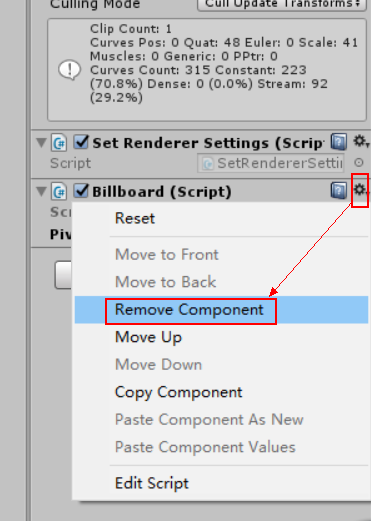
在Hierarchy面板选中AstroMan，到Inspector面改版点击Add Component,搜索Billboard,添加该组件。



### 第二步：部署至Hololens观察

如之前一般，构建并部署到Hololens,无论您怎么改变观测点，挂有Billboard组件的全息对象都会始终面朝你。

### 第三步：现在从AstroMan上删除Billboard脚本组件



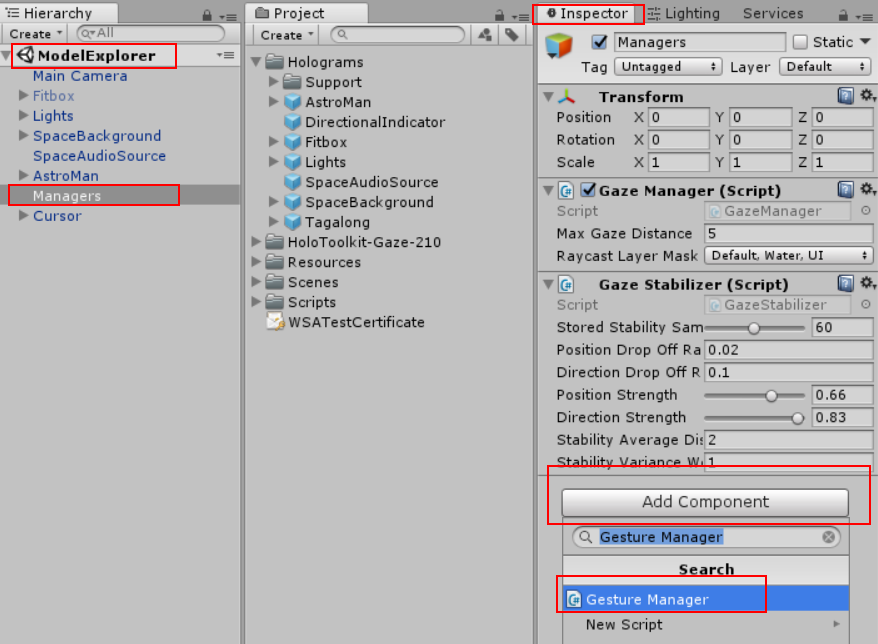
## 第六章 - 标签

### 目标

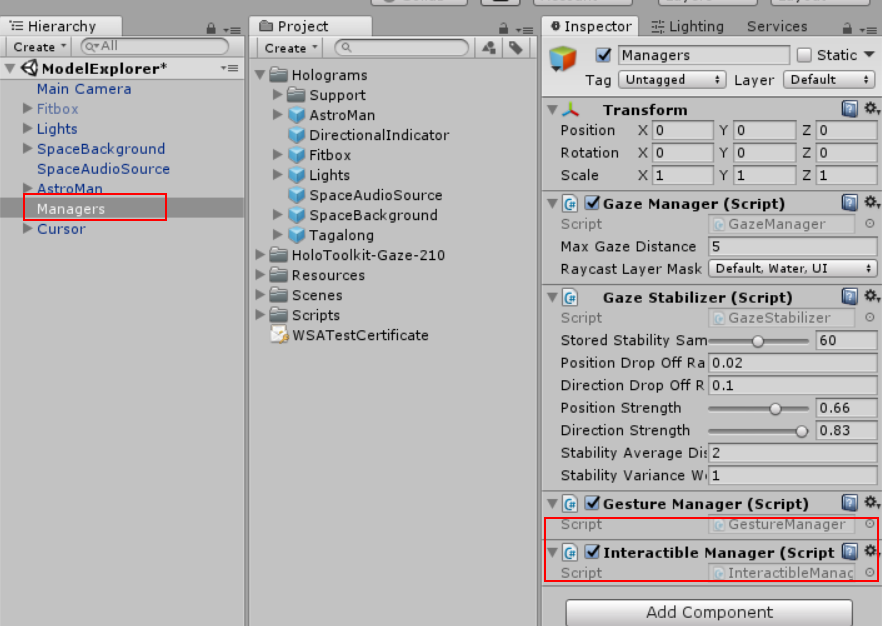
使用Tag-Along，让全息图在房间里跟随我们。

### 第一步：添加Gesture Manager

在Hierarchy面板选中Manager,到Inspector面板点击Add Component，搜索Gesture Manager,点击该组件添加。



### 第二步：添加Interactible Manager



### 第三步：Interactible和Interactible Action脚本

### 