**Getting Started - Chapter 2 - The Viewer**

**@@启程-第二章-阅览组件@@**

**Changing the Viewer's Camera**

**@@改变阅览器标签的相机@@**

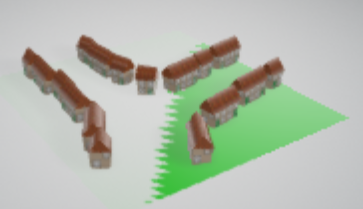
What happens when we put the **Village** as a model in the viewer?

@@如果我们把这个村庄场景作为模型放到阅览器组件（第一章的第三篇）里会发生什么？@@

[Example Viewer - Village](https://doc.babylonjs.com/webpages/page2.html) using the default Viewer.

@@示例阅览组件-村庄，使用默认的阅览组件@@

https://doc.babylonjs.com/webpages/page2



We see that the ground flickers. Why is that? This is because by default the Viewer already adds a ground and where they overlap they 'fight' for supremacy.

@@我们发现地面网格不断地闪烁，为什么会这样呢？这是因为默认情况下，这个阅览组件已经添加了一个地面网格，这两个地面网格重叠并争夺显示权。@@

How do we overcome this? We use the *extends* attribute in the ≤babylon> element and set it to minimal.

@@如何解决这一问题？我们修改<Babylon>标签的“*extends* ”属性，将它设为minimal即可@@

<babylon extends="minimal" model="path to model file"></babylon>

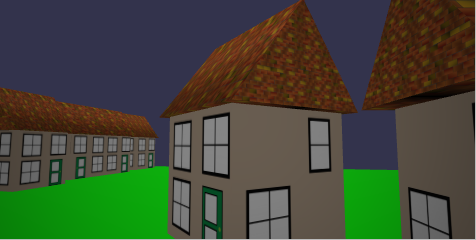
This removes the default ground along with other aspects such as the Babylon.js link and the full screen icon giving

@@这将移除默认的地面以及一些其他东西，比如Babylon.js网站的链接和全屏图标。@@

[Example Viewer - Village](https://doc.babylonjs.com/webpages/page3.html) using the minimal Viewer.

@@示例阅览组件-村庄，使用最小化的阅览组件@@

https://doc.babylonjs.com/webpages/page3



Removing the default ground has stopped the flickering. However the default Viewer calculates the extent of the model and adjusts the camera accordingly. By using *minimal* the camera just defaults to close to the center of the model village.

@@移除默认的地面网格阻止了闪烁。但是默认的阅览组件还要负责计算模型的范围，并相应地调整相机。将*extends* 属性设为“最小化”则默认相机将过于靠近村庄模型的中心。@@

When you want the camera further away you have to get your hands dirty with some code, which of course you can just copy and paste as needed.

@@如果你想让相机离远一些，你必须动手写一些代码，当然你可以根据需要复制粘贴。@@

To move the camera we have to adjust its *radius* property. This has to be done before the model is loaded. Properties cannot be changed once the model is loaded in the Viewer. We need to remove the *model* attribute from the <babylon> element to prevent the model loading before the camera radius can be changed. The <babylon> element must also be given an *id* which is referenced by the script that will alter the camera properties.

@@要移动相机我们必须调整它的“半径”属性。这必须在模型加载完毕之前完成。一旦阅览器组件中的模型加载完毕，标签的属性将不能再改变。因此我们需要移除<babylon>标签中的“model”属性，以阻止模型在相机半径调整完毕前加载。<babylon>标签还必须设置一个id属性，这个id将被调整相机属性的脚本引用。@@

<babylon id="myViewer" extends="minimal"></babylon>

The following code sets the camera radius (and in this case its angle of depression) and then load the model using

@@用下面的代码设置相机的半径（以及相机的俯角），并加载模型@@

<script>

BabylonViewer.viewerManager.getViewerPromiseById('myViewer').then((viewer) => {

viewer.onSceneInitObservable.add(() => {

viewer.sceneManager.camera.radius = 15; //set camera radius 设置相机的半径

viewer.sceneManager.camera.beta = Math.PI / 2.2; //angle of depression 俯角

});

viewer.onEngineInitObservable.add((scene) => {

viewer.loadModel({

url: "path to model file"

});

});

});

</script>

[Example Viewer - Village](https://doc.babylonjs.com/webpages/page4.html) adjusting the camera

@@示例阅览组件-村庄，调整相机@@

https://doc.babylonjs.com/webpages/page4

When you are when developing a web game or app you probably want more flexibility than the Viewer can give. Let's take another look at using the HTML template.

@@当你开发一个网页游戏或应用时，你可能需要比阅览器组件更高的灵活度。让我们看看如何使用HTML模板@@