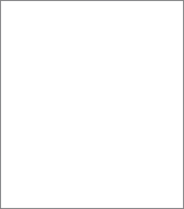
在本课的内容中，我们将使用另一个名为binding的方法来更新TimerText widget的内容。

Bindings

使用binding，可以自动更新特定的widget属性。但在此之前，widget的相关属性必须有Bind下拉列表。

我们可以将属性绑定在widget中的函数或变量,binding就会不断从该函数或变量中获取数值，然后将绑定的属性值设置为所获取的值。



可能你要问了，既然binding这么方便，为什么我们不干脆一直就使用binding呢？这是因为binding是比较低效的做法，因为它需要不断更新内容。这就意味着即便没有更新的信息，游戏也需要消耗资源更新相关属性。

话虽这么说，对于变化频率非常高的元素，比如timer这种，binding还是很好用的。接下来让我们为TimerText创建一个binding。

创建binding

打开WBP\_HUD，然后切换到Designer模式。

选中TimerText，然后在Details面板中找到Content部分。你会看到此处的Text属性是可绑定的，点击Bind的下拉列表，然后选择Create Binding。

此时会创建一个新的函数，并打开其graph。这里将其更名为UpdateTimerText。



可以看到，该函数有一个Return节点，同时还有一个Text类型的Return Type接口。TimerText中将显示所有连接到该接口的文本。

将GameManager拖动到graph中，然后从中获取TimeRemaining变量。

将TimeRemaiining变量连接到Return节点的Return Value接口，这样虚幻就会自动为我们添加一个转换节点。

小结一下:

1.binding会持续不断的调用UpdateTimerText函数

2.该函数从BP\_GameManager中获取TimeRemaining变量。

3.ToText(float)节点将会把TimeRemaining的值转换为Text类型

4.转换后的值将输出到Return节点

好了，HUD到此就完全完成了。点击编辑器上的Compile按钮，然后关闭WBP\_HUD.点击Play预览最终的游戏效果。



好了，到此为止,我们的UI部分教程就结束了，该部分的示例项目链接 :

链接:https://pan.baidu.com/s/1kVDxb75 密码:z4yn

在本部分内容中，我们了解了UMG的基础知识，在此基础上，我们可以创建更为复杂的交互界面。

如果你想了解UI Widget的更多知识，可以参考虚幻4的官方文档 :

[Widget Type Reference](https://docs.unrealengine.com/latest/INT/Engine/UMG/UserGuide/WidgetTypeReference/index.html)

从下一课开始，我们将使用已经学到的知识制作一个简单的游戏。

好了，我们下一课再见。