**TreeView**

TreeView 是一个 IMGUI 控件，用于显示可以展开和折叠的层级数据。使用 TreeView 可为 Editor 窗口创建高度定制化的列表视图和多列表格，并可将其与其他 IMGUI 控件和组件结合使用。

为了处理 TreeView 的渲染，需要确定一个扩展项（称为行）列表。每一行代表一个 TreeViewItem。每个 TreeViewItem 都包含父子信息，此信息可供 TreeView 用来处理导航（按键和鼠标输入）。

TreeView 有一个单独的根 TreeViewItem，此项是隐藏的，不会出现在 Editor 中。此项是所有其他项的根。

重要的类和方法

除了 TreeView 本身之外，最重要的类是 TreeViewItem 和 TreeViewState。

TreeViewState (TreeViewState) 包含与 Editor 中的 TreeView 字段交互时更改的状态信息，例如选择状态、展开状态、导航状态和滚动状态。TreeViewState 是唯一可序列化的状态。TreeView 本身不可序列化，而是根据构造或重载时所代表的数据进行重建。将 TreeViewState 添加为 EditorWindow 派生类中的字段，可确保在重新加载脚本或进入播放模式时不会丢失用户更改的状态。

TreeViewItem (TreeViewItem) 包含有关单个 TreeView 项的数据，用于在 Editor 中构建树结构的表示。每个 TreeViewItem 必须以唯一的整数 ID（在 TreeView 中的所有项中是唯一的）进行构造。此 ID 用于在树中查找项、选择状态、展开状态和导航。如果树表示 Unity 对象，应为每个对象使用 GetInstanceID 作为 TreeViewItem 的 ID。在 Editor 中重新加载脚本或进入播放模式时，在 TreeViewState 中使用这些 ID 可以保持用户更改的状态（例如展开的项）。所有 TreeViewItems 都有一个 depth 属性，表示视觉缩进。

BuildRoot (BuildRoot) 是 TreeView 类的单个抽象方法，必须实现该方法才能创建 TreeView。使用此方法可以创建树的根项。每次对树调用 Reload 时都会调用此方法。对于使用小数据集的简单树，应在 BuildRoot 中的根项下创建 TreeViewItems 的整个树。对于非常大的树，在每次重新加载时创建整个树不是最佳选择。在这种情况下，应先创建根，然后重写 BuildRows 方法，仅创建当前行的项。如需查看 BuildRoot 的用法示例，请参阅下面的示例 1：简单 TreeView。

BuildRows (BuildRows) 是一种虚拟方法，此方法的默认实现基于 BuildRoot 中创建的完整树来构建行列表。如果只在 BuildRoot 中创建了根，则应重写此方法来处理展开的行。请参阅下面的初始化 TreeView 以了解更多信息。

初始化 TreeView

从 TreeView 对象调用 Reload 方法时，将初始化 TreeView。

可通过两种方法来设置 TreeView：

1.创建完整树 - 为树模型数据中的所有项创建 TreeViewItem。这是默认设置，只需较少的代码即可设置。从 TreeView 对象调用 BuildRoot 时，将构建完整树。

2.仅创建展开的项 - 这种方法要求重写 BuildRows 来手动控制显示的行，而 BuildRoot 仅用于创建根 TreeViewItem。这种方法最适合于大型数据集或经常更改的数据。

对于小型数据集或对于不经常更改的数据，请使用第一种方法。对大型数据集或经常更改的数据，请使用第二种方法，因为仅创建展开的项而非创建完整树的速度将更快。

可通过三种方式设置 TreeViewItem：

创建从头开始初始化的具有子项、父项和深度的 TreeViewItem。

创建具有父项和子项的 TreeViewItem，然后使用 SetupDepthsFromParentsAndChildren 设置深度。

创建只有深度信息的 TreeViewItem，然后使用 SetupDepthsFromParentsAndChildren 设置父项和子项引用。非

