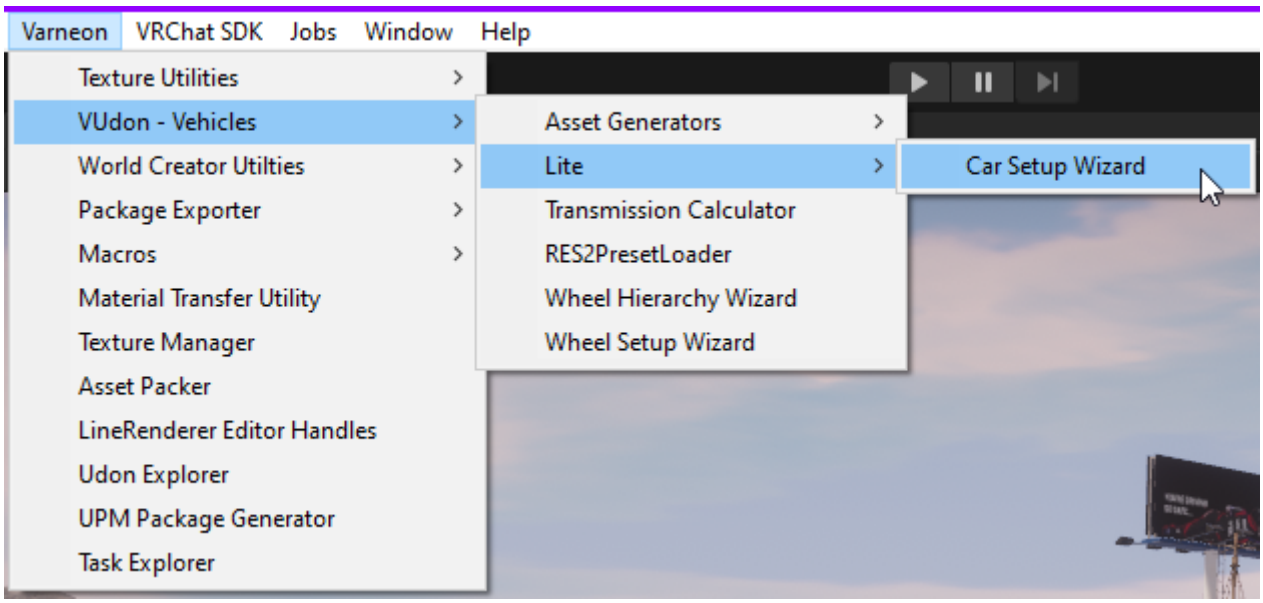
Custom Editor Menu Items

How to create different types of menu items for the Unity Editor.

Custom Toolbar Menu

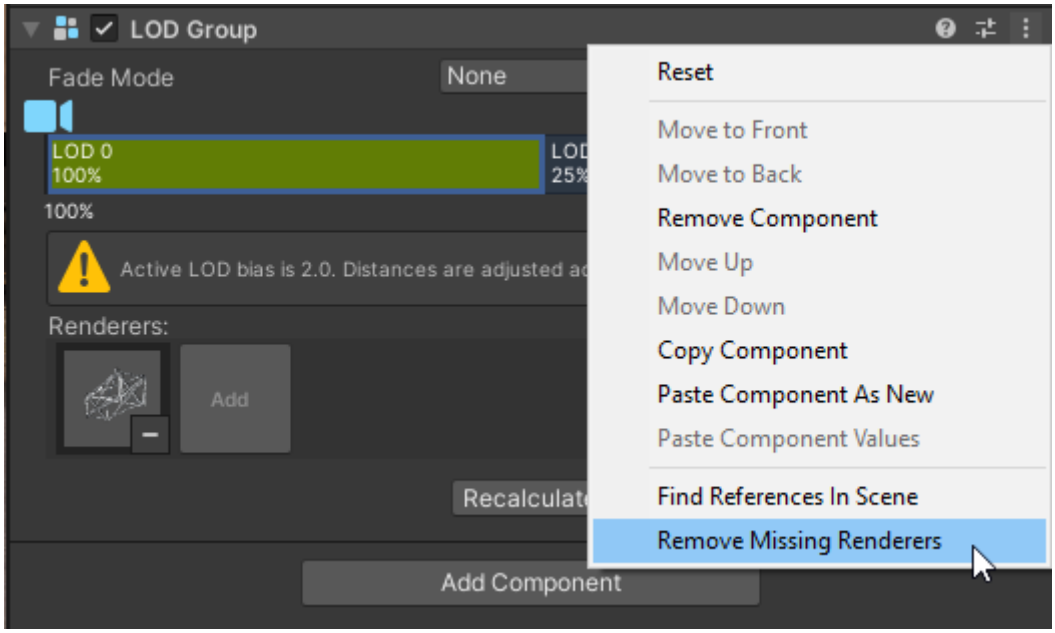
This is how you can make a custom toolbar menu for pointing into any static method.




```
namespace Varneon.VUdon.VehiclesLite.Editor
{
    public class CarSetupWizard : EditorWindow
    {
    // Every forward slash marks a new submenu
        [MenuItem("Varneon/VUdon - Vehicles/Lite/Car Setup Wizard")]
        private static void OpenWindow()
        {
            GetWindow<CarSetupWizard>("Car Setup Wizard");
        }
}
}
```

Component Context Menu

This is how you can make context menu items for different types of components in the inspector.




```
using System.Linq;
using UnityEditor;
using UnityEngine;

namespace Varneon.EditorUtilities.ComponentExtensions
{
    /// <summary>
    /// Collection of context menu actions for LODGroups
    /// </summary>
    public static class LODGroupContextMenuActions
    {
        /// <summary>
        /// Validate the method for removing missing renderer references from LODGroup
        /// </summary>
        /// <param name="command"></param>
        [MenuItem("CONTEXT/LODGroup/Remove Missing Renderers", validate = true)]
        private static bool ValidateRemoveMissingRenderers(MenuCommand command)
        {
            LODGroup lodGroup = (LODGroup)command.context;

            // Check if any of the renderer references is null
            return lodGroup.GetLODs().SelectMany(l => l.renderers).Any(r => r == null);
        }
```

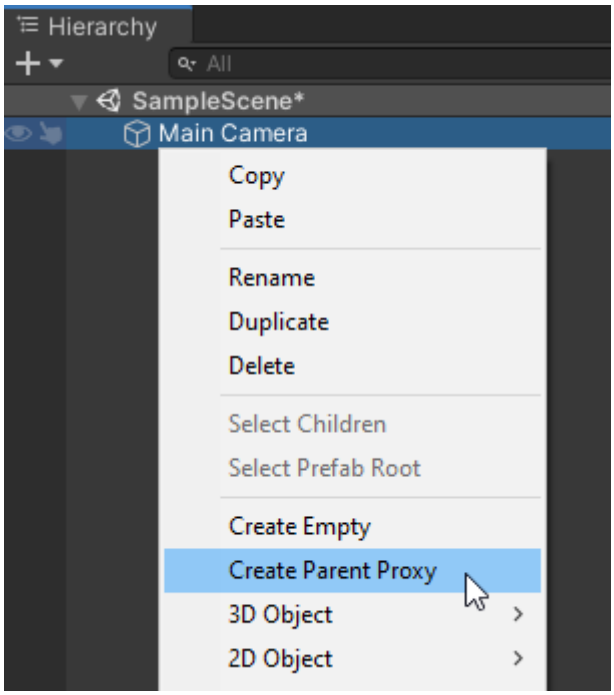
```
        /// <summary>
        /// Remove missing renderer references from LODGroup
        /// </summary>
        /// <param name="command"></param>
        [MenuItem("CONTEXT/LODGroup/Remove Missing Renderers")]
        private static void RemoveMissingRenderers(MenuCommand command)
        {
            LODGroup lodGroup = (LODGroup)command.context;

            Undo.RecordObject(lodGroup, "Remove missing LODGroup renderers");

            lodGroup.SetLODs(lodGroup.GetLODs().Select(l => new
LOD(l.screenRelativeTransitionHeight, l.renderers.Where(r => r !=
null).ToArray())).ToArray());
        }
    }
}
```

Hierarchy Menu

This is how you can make menu items for the hierarchy window.



```
using UnityEditor;
using UnityEngine;

namespace Varneon.EditorUtilities.HierarchyActions
{
    /// <summary>
    /// Collection of hierarchy context menu actions for creating proxy objects
    /// </summary>
    public static class ProxyObjectActions
    {
        /// <summary>
        /// Creates a new proxy parent object for the selected object(s)
        /// </summary>
        /// <param name="command"></param>
        [MenuItem("GameObject/Create Parent Proxy", false, 0)]
        private static void CreateProxyParent(MenuCommand command)
        {
            // Get the selected object's transform
            Transform contextTransform = ((GameObject)command.context).transform;

            // Create a new object
            Transform newProxyTransform = new GameObject(string.Format("{0}_Proxy",
contextTransform.name)).transform;

            // Match the parent
            newProxyTransform.SetParent(contextTransform.parent, false);

            // Match the sibling index
            newProxyTransform.SetSiblingIndex(contextTransform.GetSiblingIndex());

            // Register undo for the new proxy
            Undo.RegisterCreatedObjectUndo(newProxyTransform.gameObject, "Create proxy
parent");

            // Parent the original object to the new proxy with undo
            Undo.SetTransformParent(contextTransform, newProxyTransform, "Parent object to
proxy");
        }
    }
```

```
}
```

Revision #1
Created 2022-12-09 20:08:20 UTC by Varneon
Updated 2022-12-09 20:21:20 UTC by Varneon