

Namespace BLINK.Tools

Classes

[MaterialTilingOffset](#)

Class MaterialTilingOffset

Namespace: [BLINK.Tools](#)

Assembly: Assembly-CSharp-Editor.dll

```
public class MaterialTilingOffset : EditorWindow
```

Inheritance

[object](#)  ← [Object](#) ← [ScriptableObject](#) ← [EditorWindow](#) ← [MaterialTilingOffset](#)

Inherited Members

[EditorWindow.BeginWindows\(\)](#) , [EditorWindow.EndWindows\(\)](#) ,
[EditorWindow.ShowNotification\(GUIContent\)](#) , [EditorWindow.ShowNotification\(GUIContent, double\)](#)  ,
[EditorWindow.RemoveNotification\(\)](#) , [EditorWindow.ShowTab\(\)](#) , [EditorWindow.Focus\(\)](#) ,
[EditorWindow.ShowUtility\(\)](#) , [EditorWindow.ShowPopup\(\)](#) , [EditorWindow.ShowModalUtility\(\)](#) ,
[EditorWindow.ShowAsDropDown\(Rect, Vector2\)](#) , [EditorWindow.Show\(\)](#) , [EditorWindow.Show\(bool\)](#)  ,
[EditorWindow.ShowAuxWindow\(\)](#) , [EditorWindow.ShowModal\(\)](#) ,
[EditorWindow.GetWindow\(Type, bool, string, bool\)](#)  , [EditorWindow.GetWindow\(Type, bool, string\)](#)  ,
[EditorWindow.GetWindow\(Type, bool\)](#)  , [EditorWindow.GetWindow\(Type\)](#)  ,
[EditorWindow.GetWindowWithRect\(Type, Rect, bool, string\)](#)  ,
[EditorWindow.GetWindowWithRect\(Type, Rect, bool\)](#)  ,
[EditorWindow.GetWindowWithRect\(Type, Rect\)](#)  , [EditorWindow.GetWindow<T>\(\)](#) ,
[EditorWindow.GetWindow<T>\(bool\)](#)  , [EditorWindow.GetWindow<T>\(bool, string\)](#)  ,
[EditorWindow.GetWindow<T>\(string\)](#)  , [EditorWindow.GetWindow<T>\(string, bool\)](#)  ,
[EditorWindow.GetWindow<T>\(bool, string, bool\)](#)  , [EditorWindow.GetWindow<T>\(params Type\[\]\)](#)  ,
[EditorWindow.GetWindow<T>\(string, params Type\[\]\)](#)  ,
[EditorWindow.GetWindow<T>\(string, bool, params Type\[\]\)](#)  ,
[EditorWindow.CreateWindow<T>\(params Type\[\]\)](#)  ,
[EditorWindow.CreateWindow<T>\(string, params Type\[\]\)](#)  , [EditorWindow.HasOpenInstances<T>\(\)](#) ,
[EditorWindow.FocusWindowIfItsOpen\(Type\)](#)  , [EditorWindow.FocusWindowIfItsOpen<T>\(\)](#) ,
[EditorWindow.GetWindowWithRect<T>\(Rect\)](#) , [EditorWindow.GetWindowWithRect<T>\(Rect, bool\)](#)  ,
[EditorWindow.GetWindowWithRect<T>\(Rect, bool, string\)](#)  ,
[EditorWindow.GetWindowWithRect<T>\(Rect, bool, string, bool\)](#)  , [EditorWindow.SaveChanges\(\)](#) ,
[EditorWindow.DiscardChanges\(\)](#) , [EditorWindow.Close\(\)](#) , [EditorWindow.Repaint\(\)](#) ,
[EditorWindow.SendEvent\(Event\)](#) , [EditorWindow.GetExtraPaneTypes\(\)](#) ,
[EditorWindow.TryGetOverlay\(string, out Overlay\)](#)  , [EditorWindow.OnBackingScaleFactorChanged\(\)](#) ,
[EditorWindow.dataModeController](#) , [EditorWindow.rootVisualElement](#) , [EditorWindow.overlayCanvas](#) ,
[EditorWindow.wantsMouseMove](#) , [EditorWindow.wantsMouseEnterLeaveWindow](#) ,
[EditorWindow.wantsLessLayoutEvents](#) , [EditorWindow.autoRepaintOnSceneChange](#) ,

EditorWindow.maximized , EditorWindow.hasFocus , EditorWindow.docked ,
EditorWindow.focusedWindow , EditorWindow.mouseOverWindow ,
EditorWindow.hasUnsavedChanges , EditorWindow.saveChangesMessage , EditorWindow.minSize ,
EditorWindow.maxSize , EditorWindow.title , EditorWindow.titleContent , EditorWindow.depthBufferBits ,
EditorWindow.antiAlias , EditorWindow.position , ScriptableObject.SetDirty() ,
[ScriptableObject.CreateInstance\(string\)](#) , [ScriptableObject.CreateInstance\(Type\)](#) ,
ScriptableObject.CreateInstance<T>() , Object.GetInstanceID() , Object.GetHashCode() ,
[Object.Equals\(object\)](#) , Object.InstantiateAsync<T>(T) , Object.InstantiateAsync<T>(T, Transform) ,
Object.InstantiateAsync<T>(T, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, Transform, Vector3, Quaternion) , [Object.InstantiateAsync<T>\(T, int\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
Object.Instantiate(Object, Vector3, Quaternion) ,
Object.Instantiate(Object, Vector3, Quaternion, Transform) , Object.Instantiate(Object) ,
Object.Instantiate(Object, Scene) , Object.Instantiate(Object, Transform) ,
[Object.Instantiate\(Object, Transform, bool\)](#) , Object.Instantiate<T>(T) ,
Object.Instantiate<T>(T, Vector3, Quaternion) ,
Object.Instantiate<T>(T, Vector3, Quaternion, Transform) , Object.Instantiate<T>(T, Transform) ,
[Object.Instantiate<T>\(T, Transform, bool\)](#) , [Object.Destroy\(Object, float\)](#) , Object.Destroy(Object) ,
[Object.DestroyImmediate\(Object, bool\)](#) , Object.DestroyImmediate(Object) ,
[Object.FindObjectsOfType\(Type\)](#) , [Object.FindObjectsOfType\(Type, bool\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsSortMode\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsInactive, FindObjectsSortMode\)](#) ,
Object.DontDestroyOnLoad(Object) , [Object.DestroyObject\(Object, float\)](#) ,
Object.DestroyObject(Object) , [Object.FindSceneObjectsOfType\(Type\)](#) ,
[Object.FindObjectsOfTypeIncludingAssets\(Type\)](#) , Object.FindObjectsOfType<T>() ,
Object.FindObjectsByType<T>(FindObjectsSortMode) , [Object.FindObjectsOfType<T>\(bool\)](#) ,
Object.FindObjectsByType<T>(FindObjectsInactive, FindObjectsSortMode) ,
Object.FindObjectOfType<T>() , [Object.FindObjectOfType<T>\(bool\)](#) ,
Object.FindFirstObjectByType<T>() , Object.FindAnyObjectByType<T>() ,
Object.FindFirstObjectByType<T>(FindObjectsInactive) ,
Object.FindAnyObjectByType<T>(FindObjectsInactive) , [Object.FindObjectsOfTypeAll\(Type\)](#) ,
[Object.FindObjectOfType\(Type\)](#) , [Object.FindFirstObjectByType\(Type\)](#) ,
[Object.FindAnyObjectByType\(Type\)](#) , [Object.FindObjectOfType\(Type, bool\)](#) ,
[Object.FindFirstObjectByType\(Type, FindObjectsInactive\)](#) ,
[Object.FindAnyObjectByType\(Type, FindObjectsInactive\)](#) , Object.ToString() , Object.name ,

Object.hideFlags , [object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

gameObjectList

```
public GameObject[] gameObjectList
```

Field Value

GameObject[]