

# Table of Contents

[Utility System](#)

[Data](#)

[ContextMenuItemButtonData](#)

[ContextMenuItemData](#)

[ContextMenuItemParentData](#)

[UIAnchorEnum](#)

[Extension](#)

[FloatExtension](#)

[IntExtension](#)

[TransformExtension](#)

[VectorExtension](#)

[VisualElementExtension](#)

[VisualElementStyleExtension](#)

[Script](#)

[Assets](#)

[CreateElements](#)

[RichText](#)

[Tool](#)

[Timer](#)

[UI](#)

[ContextMenuElement](#)

[ContextMenuElement.UxmlFactory](#)

[ContextMenuItemButtonElement](#)

[ContextMenuItemButtonElement.UxmlFactory](#)

[ContextMenuItemElement](#)

[ContextMenuItemElement.UxmlFactory](#)

[ContextMenuItemParentElement](#)

[ContextMenuItemParentElement.UxmlFactory](#)

[DialogBoxElement](#)

[DialogBoxElement.UxmlFactory](#)

[PopupBaseElement](#)

[SpriteElement](#)

[SpriteElement.UxmlFactory](#)

[ToolTipElement](#)

[ToolTipElement.UxmlFactory](#)

# Namespace UtilitySystem

This is The Namespace for the UtilitySystem, Assembly-CSharp

[UtilitySystem.Data](#)

---

[UtilitySystem.Extension](#)

---

[UtilitySystem.Script](#)

---

[UtilitySystem.Tool](#)

---

[UtilitySystem.UI](#)

---

# Namespace UtilitySystem.Data

## Classes

### [ContextMenuItemButtonData](#)

---

This class is the ContextMenuItemButtonData. It will be used to create a ContextMenuItemButtonElement. And is a child class of ContextMenuItemData.

### [ContextMenuItemData](#)

---

This class is the ContextMenuItemData. it will be used to create a ContextMenuItemElement.

### [ContextMenuItemParentData](#)

---

This class is the ContextMenuItemParentData. It will be used to create a ContextMenuItemParentElement. And is a child class of ContextMenuItemData.

## Enums

### [UIAnchorEnum](#)

---

This is used to determine how an Inventory will be sorted. To use this enum in UI Builder

# Class ContextMenuItemButtonData

This class is the ContextMenuItemButtonData. It will be used to create a ContextMenuItemButtonElement. And is a child class of ContextMenuItemData.

## Inheritance

System.Object

[ContextMenuItemData](#)

ContextMenuItemButtonData

Namespace: [UtilitySystem.Data](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
[Serializable]
public class ContextMenuItemButtonData : ContextMenuItemData
```

## Constructors

### ContextMenuItemButtonData(String, Action, Boolean)

This is a constructor and will create the Context Menu Item Data. (Button)

#### Declaration

```
public ContextMenuItemButtonData(string textView, Action action, bool isSectionEnd = false)
```

#### Parameters

| TYPE           | NAME         | DESCRIPTION                                      |
|----------------|--------------|--|
| System.String  | textValue    | This will be added to the text variable.         |
| System.Action  | action       | This will be added to the onClick variable.      |
| System.Boolean | isSectionEnd | This will be added to the isSectionEnd variable. |

### ContextMenuItemButtonData(String, Boolean, Action[])

This is a constructor and will create the Context Menu Item Data. (Button)

#### Declaration

```
public ContextMenuItemButtonData(string textView, bool isSectionEnd = false, params Action[] actions)
```

#### Parameters

| TYPE            | NAME         | DESCRIPTION                                      |
|-----------------|--------------|--|
| System.String   | textValue    | This will be added to the text variable.         |
| System.Boolean  | isSectionEnd | This will be added to the isSectionEnd variable. |
| System.Action[] | actions      | This will be added to the onClick variable.      |

## ContextMenuItemButtonData(String, List<Action>, Boolean)

This is a constructor and will create the Context Menu Item Data. (Button)

### Declaration

```
public ContextMenuItemButtonData(string textValue, List<Action> actions, bool isSectionEnd = false)
```

### Parameters

| TYPE   | NAME         | DESCRIPTION                                      |
|--|--------------|--|
| System.String                                  | textValue    | This will be added to the text variable.         |
| System.Collections.Generic.List<System.Action> | actions      | This will be added to the onClick variable.      |
| System.Boolean                                 | isSectionEnd | This will be added to the isSectionEnd variable. |

## Fields

### onClick

This will be invoked when the item's is clicked.

### Declaration

```
public Action onClick
```

### Field Value

| TYPE          | DESCRIPTION |
|---------------|-------------|
| System.Action |             |

## Extension Methods

```
RichText.UpperCaseText(Object)
RichText.LowerCaseText(Object)
RichText.BoldText(Object)
RichText.ItalicText(Object)
RichText.Text(Object)
RichText.UnderLineText(Object)
RichText.StrikeText(Object)
RichText.SupText(Object)
RichText.SubText(Object)
RichText.PositionText(Object, Single)
RichText.SizeType(Object, Int32)
RichText.ColoredText(Object, String)
RichText.ColoredText(Object, Color)
RichText.BooleanText(Object, Boolean)
RichText.BooleanText(Object, Boolean, Color)
```

# Class ContextMenuItemData

This class is the ContextMenuItemData. it will be used to create a ContextMenuItemElement.

## Inheritance

```
System.Object  
ContextMenuItemData  
ContextMenuItemButtonData  
ContextMenuItemParentData  
Namespace: UtilitySystem.Data  
Assembly: Assembly-CSharp.dll
```

## Syntax

```
[Serializable]  
public class ContextMenuItemData
```

## Constructors

### ContextMenuItemData(String, Boolean)

This is a constructor and will create the Context Menu Item Data.

#### Declaration

```
public ContextMenuItemData(string textView, bool isSectionEnd = false)
```

#### Parameters

| TYPE           | NAME         | DESCRIPTION                                      |
|----------------|--------------|--|
| System.String  | textValue    | This will be added to the text variable.         |
| System.Boolean | isSectionEnd | This will be added to the isSectionEnd variable. |

## Fields

### isSectionEnd

If true will add a line breaker after this menu item.

#### Declaration

```
public bool isSectionEnd
```

#### Field Value

| <b>TYPE</b>    | <b>DESCRIPTION</b> |
|----------------|--------------------|
| System.Boolean |                    |

## text

---

This is the text that will be displayed on the item's label.

### Declaration

```
public string text
```

### Field Value

| <b>TYPE</b>   | <b>DESCRIPTION</b> |
|---------------|--------------------|
| System.String |                    |

## Extension Methods

[RichText.UpperCaseText\(Object\)](#)  
[RichText.LowerCaseText\(Object\)](#)  
[RichText.BoldText\(Object\)](#)  
[RichText.ItalicText\(Object\)](#)  
[RichText.Text\(Object\)](#)  
[RichText.UnderLineText\(Object\)](#)  
[RichText.StrikeText\(Object\)](#)  
[RichText.SupText\(Object\)](#)  
[RichText.SubText\(Object\)](#)  
[RichText.PositionText\(Object, Single\)](#)  
[RichText.SizeType\(Object, Int32\)](#)  
[RichText.ColoredText\(Object, String\)](#)  
[RichText.ColoredText\(Object, Color\)](#)  
[RichText.BooleanText\(Object, Boolean\)](#)  
[RichText.BooleanText\(Object, Boolean, Color\)](#)

# Class ContextMenuItemParentData

This class is the ContextMenuItemParentData. It will be used to create a ContextMenuItemParentElement. And is a child class of ContextMenuItemData.

## Inheritance

System.Object

[ContextMenuItemData](#)

ContextMenuItemParentData

Namespace: [UtilitySystem.Data](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
[Serializable]
public class ContextMenuItemParentData : ContextMenuItemData
```

## Constructors

### ContextMenuItemParentData(String, ContextMenuItemData, Boolean)

This is a constructor and will create the Context Menu Item Data. (Menu)

#### Declaration

```
public ContextMenuItemParentData(string textView, ContextMenuItemData item, bool isSectionEnd = false)
```

#### Parameters

| TYPE                                | NAME         | DESCRIPTION                                      |
|-------------------------------------|--------------|--|
| System.String                       | textView     | This will be added to the text variable.         |
| <a href="#">ContextMenuItemData</a> | item         | This item will be added to the items variable.   |
| System.Boolean                      | isSectionEnd | This will be added to the isSectionEnd variable. |

### ContextMenuItemParentData(String, Boolean, ContextMenuItemData[])

This is a constructor and will create the Context Menu Item Data. (Menu)

#### Declaration

```
public ContextMenuItemParentData(string textView, bool isSectionEnd = false, params ContextMenuItemData[] items)
```

## Parameters

| Type                  | Name         | Description   |
|-----------------------|--------------|---|
| System.String         | textValue    | This will be added to the text variable.                |
| System.Boolean        | isSectionEnd | This will be added to the isSectionEnd variable.        |
| ContextMenuItemData[] | items        | These params items will be added to the items variable. |

## ContextMenuItemParentData(String, List<ContextMenuItemData>, Boolean)

This is a constructor and will create the Context Menu Item Data. (Menu)

## Declaration

```
public ContextMenuItemParentData(string textValue, List<ContextMenuItemData> items, bool isSectionEnd = false)
```

## Parameters

| Type   | Name         | Description                                      |
|--|--------------|--|
| System.String  | textValue    | This will be added to the text variable.         |
| System.Collections.Generic.List<ContextMenuItemData> | items        | These items will be added to the items variable. |
| System.Boolean                                       | isSectionEnd | This will be added to the isSectionEnd variable. |

## Fields

### items

This is the list of Context Menu Item Data that will be used to create a sub menu inside of the menu.

## Declaration

```
public List<ContextMenuItemData> items
```

## Field Value

| Type   | Description |
|--|-------------|
| System.Collections.Generic.List<ContextMenuItemData> |             |

## Extension Methods

RichText.UpperCaseText(Object)  
RichText.LowerCaseText(Object)  
RichText.BoldText(Object)  
RichText.ItalicText(Object)  
RichText.Text(Object)  
RichText.UnderLineText(Object)  
RichText.StrikeText(Object)  
RichText.SupText(Object)  
RichText.SubText(Object)  
RichText.PositionText(Object, Single)  
RichText.SizeTypeText(Object, Int32)  
RichText.ColoredText(Object, String)  
RichText.ColoredText(Object, Color)  
RichText.BooleanText(Object, Boolean)  
RichText.BooleanText(Object, Boolean, Color)

# Enum UIAnchorEnum

This is used to determine how an Inventory will be sorted. To use this enum in UI Builder

Namespace: [UtilitySystem.Data](#)

Assembly: Assembly-CSharp.dll

Syntax

```
public enum UIAnchorEnum
```

Remarks

[UtilitySystem.Data.SortByEnum](#), [Assembly-CSharp](#)

Fields

| NAME          | DESCRIPTION |
|---------------|-------------|
| Bottom_Center |             |
| Bottom_Left   |             |
| Bottom_Right  |             |
| Center_Center |             |
| Center_Left   |             |
| Center_Right  |             |
| None          |             |
| Top_Center    |             |
| Top_Left      |             |
| Top_Right     |             |

Extension Methods

[RichText.UpperCaseText\(\)](#)  
[RichText.LowerCaseText\(\)](#)  
[RichText.BoldText\(\)](#)  
[RichText.ItalicText\(\)](#)  
[RichText.Text\(\)](#)  
[RichText.UnderLineText\(\)](#)  
[RichText.StrikeText\(\)](#)  
[RichText.SupText\(\)](#)  
[RichText.SubText\(\)](#)  
[RichText.PositionText\(Single\)](#)

RichText.SizeType(Int32)  
RichText.ColoredText(String)  
RichText.ColoredText(Color)  
RichText.BooleanText(Boolean)  
RichText.BooleanText(Boolean, Color)

# Namespace UtilitySystem.Extension

## Classes

### [FloatExtension](#)

---

This static class extends the float Variable.

### [IntExtension](#)

---

This static class extends the integer Variable.

### [TransformExtension](#)

---

This static class extends the Transform type.

### [VectorExtension](#)

---

This static class extends the Vector2,3,4 types.

### [VisualElementExtension](#)

---

This static class is used to create short-hand extensions that will cut down on style boilerplate for the VisualElement(s)

### [VisualElementStyleExtension](#)

---

This static class is the extensions of the VisualElement. That deals with the boilder plate and simplify the styles.

# Class FloatExtension

This static class extends the float Variable.

## Inheritance

System.Object

FloatExtension

Namespace: [UtilitySystem.Extension](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public static class FloatExtension
```

## Methods

### FlipValue(Single)

This will change an negative number to positive and positive to negative.

#### Declaration

```
public static float FlipValue(this float value)
```

#### Parameters

| TYPE          | NAME  | DESCRIPTION                     |
|---------------|-------|---------------------------------|
| System.Single | value | the float that will be flipped. |

#### Returns

| TYPE          | DESCRIPTION               |
|---------------|---------------------------|
| System.Single | return the flipped float. |

### HighestValue(Single, Single)

This will take the highest of 2 floats.

#### Declaration

```
public static float HighestValue(this float value, float compare)
```

#### Parameters

| Type          | Name    | Description                     |
|---------------|---------|---------------------------------|
| System.Single | value   | The float that will check.      |
| System.Single | compare | The float that will be compare. |

Returns

| Type          | Description               |
|---------------|---------------------------|
| System.Single | Return the highest float. |

## LowestValue(Single, Single)

This will take the lowest of 2 floats.

Declaration

```
public static float LowestValue(this float value, float compare)
```

Parameters

| Type          | Name    | Description                     |
|---------------|---------|---------------------------------|
| System.Single | value   | The float that will check.      |
| System.Single | compare | The float that will be compare. |

Returns

| Type          | Description              |
|---------------|--------------------------|
| System.Single | Return the lowest float. |

## RoundDown(Single)

This will round down the float to an interger.

Declaration

```
public static int RoundDown(this float value)
```

## Parameters

| Type          | Name  | Description                        |
|---------------|-------|------------------------------------|
| System.Single | value | the float that will be round down. |

## Returns

| Type         | Description                              |
|--------------|--|
| System.Int32 | Return an float of the float round down. |

## RoundUp(Single)

This will round up the float to an interger.

## Declaration

```
public static int RoundUp(this float value)
```

## Parameters

| Type          | Name  | Description                      |
|---------------|-------|----------------------------------|
| System.Single | value | the float that will be round up. |

## Returns

| Type         | Description                            |
|--------------|--|
| System.Int32 | Return an float of the float round up. |

## WeightGramToKilogram(Single, Int32)

This will divide the value by 1000

## Declaration

```
public static float WeightGramToKilogram(this float value, int decimalPoint = 2)
```

## Parameters

| <b>TYPE</b>   | <b>NAME</b>  | <b>DESCRIPTION</b>                          |
|---------------|--------------|---|
| System.Single | value        | The Grams that will converted to Kilograms. |
| System.Int32  | decimalPoint | How many Decimal places.                    |

## Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b>    |
|---------------|-----------------------|
| System.Single | Return the Kilograms. |

## Remarks

1 Grams is the same as 0.001 Kilogram.

## WeightGramToOunce(Single, Int32)

---

This will divide the value by 28.35

## Declaration

```
public static float WeightGramToOunce(this float value, int decimalPoint = 2)
```

## Parameters

| <b>TYPE</b>   | <b>NAME</b>  | <b>DESCRIPTION</b>                      |
|---------------|--------------|---|
| System.Single | value        | The Pound that will converted to Ounce. |
| System.Int32  | decimalPoint | How many Decimal places.                |

## Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b> |
|---------------|--------------------|
| System.Single | Return the Ounce.  |

## Remarks

1 Grams is the same as 0.035274 Ounce.

## WeightKilogramToGram(Single, Int32)

This will multiple the value by 1000

#### Declaration

```
public static float WeightKilogramToGram(this float value, int decimalPoint = 2)
```

#### Parameters

| Type          | Name         | Description                                 |
|---------------|--------------|---|
| System.Single | value        | The kilograms that will converted to Grams. |
| System.Int32  | decimalPoint | How many Decimal places.                    |

#### Returns

| Type          | Description       |
|---------------|-------------------|
| System.Single | Return the Grams. |

#### Remarks

1 Kilogram is the same as 1000 Grams.

### WeightKiloGramToPound(Single, Int32)

This will divide the value by 2.205

#### Declaration

```
public static float WeightKiloGramToPound(this float value, int decimalPoint = 2)
```

#### Parameters

| Type          | Name         | Description                                |
|---------------|--------------|--|
| System.Single | value        | The Kilogram that will converted to Pound. |
| System.Int32  | decimalPoint | How many Decimal places.                   |

#### Returns

| TYPE          | DESCRIPTION       |
|---------------|-------------------|
| System.Single | Return the Pound. |

#### Remarks

1 Kilogram is the same as 2.20462 Pound.

#### WeightOunceToGram(Single, Int32)

This will multiple the value by 28.35

#### Declaration

```
public static float WeightOunceToGram(this float value, int decimalPoint = 2)
```

#### Parameters

| TYPE          | NAME         | DESCRIPTION                            |
|---------------|--------------|--|
| System.Single | value        | The Ounce that will converted to Gram. |
| System.Int32  | decimalPoint | How many Decimal places.               |

#### Returns

| TYPE          | DESCRIPTION      |
|---------------|------------------|
| System.Single | Return the Gram. |

#### Remarks

1 Ounce is the same as 28.3495 Gram.

#### WeightOunceToPound(Single, Int32)

This will divide the value by 16

#### Declaration

```
public static float WeightOunceToPound(this float value, int decimalPoint = 2)
```

#### Parameters

| Type          | Name         | Description                             |
|---------------|--------------|---|
| System.Single | value        | The Ounce that will converted to Pound. |
| System.Int32  | decimalPoint | How many Decimal places.                |

## Returns

| Type          | Description       |
|---------------|-------------------|
| System.Single | Return the Pound. |

## Remarks

1 Ounce is the same as 0.0625 Pound.

## WeightPoundToKiloGram(Single, Int32)

This will multiple the value by 2.205

## Declaration

```
public static float WeightPoundToKiloGram(this float value, int decimalPoint = 2)
```

## Parameters

| Type          | Name         | Description                                |
|---------------|--------------|--|
| System.Single | value        | The Pound that will converted to Kilogram. |
| System.Int32  | decimalPoint | How many Decimal places.                   |

## Returns

| Type          | Description          |
|---------------|----------------------|
| System.Single | Return the Kilogram. |

## Remarks

1 Pound is the same as 0.453592 Kilogram.

## WeightPoundToOunce(Single, Int32)

This will multiple the value by 16

#### Declaration

```
public static float WeightPoundToOunce(this float value, int decimalPoint = 2)
```

#### Parameters

| Type          | Name         | Description                             |
|---------------|--------------|---|
| System.Single | value        | The Pound that will converted to Ounce. |
| System.Int32  | decimalPoint | How many Decimal places.                |

#### Returns

| Type          | Description       |
|---------------|-------------------|
| System.Single | Return the Ounce. |

#### Remarks

1 Pound is the same as 16 Ounce.

# Class IntExtension

This static class extends the integer Variable.

## Inheritance

System.Object

IntExtension

Namespace: [UtilitySystem.Extension](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public static class IntExtension
```

## Methods

### CurrencyValue(Int32)

This will split the value into gold, silver & bronze.

#### Declaration

```
public static (int, int, int) CurrencyValue(this int value)
```

#### Parameters

| TYPE         | NAME  | DESCRIPTION          |
|--------------|-------|----------------------|
| System.Int32 | value | The currency amount. |

#### Returns

| TYPE  | DESCRIPTION                          |
|---|--------------------------------------|
| System.ValueTuple<System.Int32, System.Int32, System.Int32> | Return the gold,silver,bronze value. |

#### Remarks

1 gold is 100 silver, 1 silver is 100 copper.

### FlipValue(Int32)

This will change an negative number to positive and positive to negative.

#### Declaration

```
public static int FlipValue(this int value)
```

## Parameters

| Type         | Name  | Description                       |
|--------------|-------|-----------------------------------|
| System.Int32 | value | the integer that will be flipped. |

## Returns

| Type         | Description                 |
|--------------|-----------------------------|
| System.Int32 | return the flipped integer. |

## HighestValue(Int32, Int32)

This will take the highest of 2 integers.

## Declaration

```
public static int HighestValue(this int value, int compare)
```

## Parameters

| Type         | Name    | Description                       |
|--------------|---------|-----------------------------------|
| System.Int32 | value   | The integer that will check.      |
| System.Int32 | compare | The integer that will be compare. |

## Returns

| Type         | Description                 |
|--------------|-----------------------------|
| System.Int32 | Return the highest integer. |

## LowestValue(Int32, Int32)

This will take the lowest of 2 integers.

## Declaration

```
public static int LowestValue(this int value, int compare)
```

## Parameters

| Type         | Name    | Description                       |
|--------------|---------|-----------------------------------|
| System.Int32 | value   | The integer that will check.      |
| System.Int32 | compare | The integer that will be compare. |

## Returns

| Type         | Description                |
|--------------|----------------------------|
| System.Int32 | Return the lowest integer. |

# Class TransformExtension

This static class extends the Transform type.

## Inheritance

System.Object  
TransformExtension

Namespace: [UtilitySystem.Extension](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public static class TransformExtension
```

## Methods

### GetChildren(Transform, Boolean)

This will get a list of the children from a gameobject.

#### Declaration

```
public static List<GameObject> GetChildren(this Transform transform, bool includeInactive = false)
```

#### Parameters

| TYPE                  | NAME            | DESCRIPTION   |
|-----------------------|-----------------|---|
| UnityEngine.Transform | transform       | This is the transform that will get it's children.                          |
| System.Boolean        | includeInactive | If the param is true it will also includelinactive gameobject. else ignore. |

#### Returns

| TYPE  | DESCRIPTION                   |
|---|-------------------------------|
| System.Collections.Generic.List<UnityEngine.GameObject> | return a list of GameObjects. |

### ResetTransformation(Transform)

This will reset the transform.

#### Declaration

```
public static void ResetTransformation(this Transform transform)
```

## Parameters

| Type                  | Name      | Description                                |
|-----------------------|-----------|--|
| UnityEngine.Transform | transform | This is the transform that will get reset. |

# Class VectorExtension

This static class extends the Vector2,3,4 types.

## Inheritance

System.Object  
VectorExtension

Namespace: [UtilitySystem.Extension](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public static class VectorExtension
```

## Methods

### ClampVector2(Vector2, Vector2, Vector2)

This will clamp a Vector2 between 2 other Vector2s

#### Declaration

```
public static Vector2 ClampVector2(this Vector2 value, Vector2 min, Vector2 max)
```

#### Parameters

| Type                | Name  | Description                                |
|---------------------|-------|--|
| UnityEngine.Vector2 | value | This is the Vector2 that will be clamped.  |
| UnityEngine.Vector2 | min   | This is the Vector2 that is the min value. |
| UnityEngine.Vector2 | max   | This is the Vector2 that is the max value. |

#### Returns

| Type                | Description                        |
|---------------------|------------------------------------|
| UnityEngine.Vector2 | The return is the Vector2 clamped. |

### ClampVector3(Vector3, Vector3, Vector3)

This will clamp a Vector3 between 2 other Vector3s

#### Declaration

```
public static Vector3 ClampVector3(this Vector3 value, Vector3 min, Vector3 max)
```

## Parameters

| Type                | Name  | Description                                |
|---------------------|-------|--|
| UnityEngine.Vector3 | value | This is the Vector3 that will be clamped.  |
| UnityEngine.Vector3 | min   | This is the Vector3 that is the min value. |
| UnityEngine.Vector3 | max   | This is the Vector3 that is the max value. |

## Returns

| Type                | Description                        |
|---------------------|------------------------------------|
| UnityEngine.Vector3 | The return is the Vector3 clamped. |

## ClampVector4(Vector4, Vector4, Vector4)

This will clamp a Vector4 between 2 other Vector4s

## Declaration

```
public static Vector4 ClampVector4(this Vector4 value, Vector4 min, Vector4 max)
```

## Parameters

| Type                | Name  | Description                                |
|---------------------|-------|--|
| UnityEngine.Vector4 | value | This is the Vector4 that will be clamped.  |
| UnityEngine.Vector4 | min   | This is the Vector4 that is the min value. |
| UnityEngine.Vector4 | max   | This is the Vector4 that is the max value. |

## Returns

| Type | Description |
|------|-------------|
|      |             |

| TYPE                | DESCRIPTION                        |
|---------------------|------------------------------------|
| UnityEngine.Vector4 | The return is the Vector4 clamped. |

## HighestVector2(Vector2, Vector2)

---

This will get the highest Vector2 between 2 Vector2s

### Declaration

```
public static Vector2 HighestVector2(this Vector2 value, Vector2 compare)
```

### Parameters

| TYPE                | NAME    | DESCRIPTION                                   |
|---------------------|---------|---|
| UnityEngine.Vector2 | value   | This is the Vector2 that will be clamped.     |
| UnityEngine.Vector2 | compare | This is the Vector2 that will be compared to. |

### Returns

| TYPE                | DESCRIPTION                                    |
|---------------------|--|
| UnityEngine.Vector2 | The return is the Vector2 that is the highest. |

## HighestVector3(Vector3, Vector3)

---

This will get the highest Vector3 between 2 Vector3s

### Declaration

```
public static Vector2 HighestVector3(this Vector3 value, Vector3 compare)
```

### Parameters

| TYPE                | NAME    | DESCRIPTION                                   |
|---------------------|---------|---|
| UnityEngine.Vector3 | value   | This is the Vector3 that will be clamped.     |
| UnityEngine.Vector3 | compare | This is the Vector3 that will be compared to. |

## Returns

| Type                | Description                                    |
|---------------------|--|
| UnityEngine.Vector2 | The return is the Vector3 that is the highest. |

## HighestVector4(Vector4, Vector4)

This will get the highest Vector4 between 2 Vector4s

### Declaration

```
public static Vector4 HighestVector4(this Vector4 value, Vector4 compare)
```

### Parameters

| Type                | Name    | Description                                   |
|---------------------|---------|---|
| UnityEngine.Vector4 | value   | This is the Vector4 that will be clamped.     |
| UnityEngine.Vector4 | compare | This is the Vector4 that will be compared to. |

## Returns

| Type                | Description                                    |
|---------------------|--|
| UnityEngine.Vector4 | The return is the Vector4 that is the highest. |

## LowestVector2(Vector2, Vector2)

This will get the lowest Vector2 between 2 Vector2s

### Declaration

```
public static Vector2 LowestVector2(this Vector2 value, Vector2 compare)
```

### Parameters

| Type                | Name  | Description                               |
|---------------------|-------|---|
| UnityEngine.Vector2 | value | This is the Vector2 that will be clamped. |

| TYPE                | NAME    | DESCRIPTION                                   |
|---------------------|---------|---|
| UnityEngine.Vector2 | compare | This is the Vector2 that will be compared to. |

Returns

| TYPE                | DESCRIPTION                                   |
|---------------------|---|
| UnityEngine.Vector2 | The return is the Vector2 that is the lowest. |

### LowestVector3(Vector3, Vector3)

This will get the lowest Vector3 between 2 Vector3s

Declaration

```
public static Vector2 LowestVector3(this Vector3 value, Vector3 compare)
```

Parameters

| TYPE                | NAME    | DESCRIPTION                                   |
|---------------------|---------|---|
| UnityEngine.Vector3 | value   | This is the Vector3 that will be clamped.     |
| UnityEngine.Vector3 | compare | This is the Vector3 that will be compared to. |

Returns

| TYPE                | DESCRIPTION                                   |
|---------------------|---|
| UnityEngine.Vector2 | The return is the Vector3 that is the lowest. |

### LowestVector4(Vector4, Vector4)

This will get the lowest Vector4 between 2 Vector4s

Declaration

```
public static Vector2 LowestVector4(this Vector4 value, Vector4 compare)
```

Parameters

| Type                | Name    | Description                                   |
|---------------------|---------|---|
| UnityEngine.Vector4 | value   | This is the Vector4 that will be clamped.     |
| UnityEngine.Vector4 | compare | This is the Vector4 that will be compared to. |

## Returns

| Type                | Description                                   |
|---------------------|---|
| UnityEngine.Vector2 | The return is the Vector4 that is the lowest. |

# Class VisualElementExtension

This static class is used to create short-hand extensions that will cut down on style boilerplate for the VisualElement(s)

## Inheritance

System.Object

VisualElementExtension

Namespace: [UtilitySystem.Extension](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public static class VisualElementExtension
```

## Methods

### Child(VisualElement, VisualElement, Boolean)

This will add, remove an element in this parent or clear the children from the parent.

#### Declaration

```
public static void Child(this VisualElement parent, VisualElement content = null, bool isAdd = true)
```

#### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | parent  | The parent that will add, remove a child element or clear all children.  |
| UnityEngine.UIElements.VisualElement | content | This is an optional param if not null will allow for adding or remove of the content element. if null them clear all of the children from the element. |
| System.Boolean                       | isAdd   | this is an optional param the will determine of the stylesheet will be added or remove from the element.   |

### Focus(VisualElement, Boolean)

This will change the focusable from value.

#### Declaration

```
public static void Focus(this VisualElement element, bool value)
```

#### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                                |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | The element that will have it's pickmode changed. |
| System.Boolean                       | value       | The boolean that will change value.               |

## Focus\_OFF(VisualElement)

---

This will not allow this element to be interactable

### Declaration

```
public static void Focus_OFF(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                                      |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | The element that will have it's focusable set to false. |

## Focus\_ON(VisualElement)

---

This will allow this element to be interactable.

### Declaration

```
public static void Focus_ON(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                                     |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | The element that will have it's focusable set to true. |

## Focus\_Toggle(VisualElement)

---

This will toggle the focusable from true to false and vice versa.

### Declaration

```
public static void Focus_Toggle(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                                |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | The element that will have its focusable changed. |

## GetElementSize(VisualElement)

This will get the size of the element from the largest of the size.

### Declaration

```
public static Vector2 GetElementSize(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                            |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will check its size. |

### Returns

| <b>TYPE</b>         | <b>DESCRIPTION</b>   |
|---------------------|--|
| UnityEngine.Vector2 | The return will be of the largest width and height of the element. |

### Remarks

this is used by ScreenBound method to keep the element within the screen.

## GetLayout(VisualElement)

This will set the element layout.

### Declaration

```
public static Rect GetLayout(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>   |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | The element that will set the width and height of the element. |

### Returns

| Type             | Description |
|------------------|-------------|
| UnityEngine.Rect |             |

## GetLayoutHeight(VisualElement)

---

This will set the size of the element layout.

### Declaration

```
public static float GetLayoutHeight(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will set the width and height of the element. |

### Returns

| Type          | Description |
|---------------|-------------|
| System.Single |             |

## GetLayoutPosition(VisualElement)

---

This will set the position of the element layout.

### Declaration

```
public static Vector2 GetLayoutPosition(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will set the width and height of the element. |

### Returns

| Type                | Description |
|---------------------|-------------|
| UnityEngine.Vector2 |             |

## GetLayoutPositionX(VisualElement)

---

This will set the position of the element layout.

#### Declaration

```
public static float GetLayoutPositionX(this VisualElement element)
```

#### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will set the width and height of the element. |

#### Returns

| Type          | Description |
|---------------|-------------|
| System.Single |             |

### GetLayoutPositionY(VisualElement)

This will set the position of the element layout.

#### Declaration

```
public static float GetLayoutPositionY(this VisualElement element)
```

#### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will set the width and height of the element. |

#### Returns

| Type          | Description |
|---------------|-------------|
| System.Single |             |

### GetLayoutSize(VisualElement)

This will set the size of the element layout.

#### Declaration

```
public static Vector2 GetLayoutSize(this VisualElement element)
```

#### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>   |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | The element that will set the width and height of the element. |

Returns

| <b>TYPE</b>         | <b>DESCRIPTION</b> |
|---------------------|--------------------|
| UnityEngine.Vector2 |                    |

## GetLayoutWidth(VisualElement)

---

This will set the size of the element layout.

Declaration

```
public static float GetLayoutWidth(this VisualElement element)
```

Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>   |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | The element that will set the width and height of the element. |

Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b> |
|---------------|--------------------|
| System.Single |                    |

## GetLocalBound(VisualElement)

---

This will get the element LocalBound.

Declaration

```
public static Rect GetLocalBound(this VisualElement element)
```

Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>   |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | The element that will set the width and height of the element. |

Returns

| Type             | Description |
|------------------|-------------|
| UnityEngine.Rect |             |

## GetLocalBoundPosition(VisualElement)

---

This will set the position of the element LocalBound.

### Declaration

```
public static Vector2 GetLocalBoundPosition(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will set the width and height of the element. |

### Returns

| Type                | Description |
|---------------------|-------------|
| UnityEngine.Vector2 |             |

## GetLocalBoundSize(VisualElement)

---

This will get the width and high of the element LocalBound.

### Declaration

```
public static Vector2 GetLocalBoundSize(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will set the width and height of the element. |

### Returns

| Type                | Description |
|---------------------|-------------|
| UnityEngine.Vector2 |             |

## GetTransform\_Position(VisualElement)

---

This will get the transform scale.

#### Declaration

```
public static Vector2 GetTransform_Position(this VisualElement element)
```

#### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will get it's transform position. |

#### Returns

| Type                | Description                           |
|---------------------|---------------------------------------|
| UnityEngine.Vector2 | The return is the transform position. |

### GetTransform\_Rotate(VisualElement)

#### Declaration

```
public static Quaternion GetTransform_Rotate(this VisualElement element)
```

#### Parameters

| Type                                 | Name    | Description |
|--------------------------------------|---------|-------------|
| UnityEngine.UIElements.VisualElement | element |             |

#### Returns

| Type                   | Description |
|------------------------|-------------|
| UnityEngine.Quaternion |             |

### GetTransform\_Scale(VisualElement)

This will get the transform scale.

#### Declaration

```
public static Vector2 GetTransform_Scale(this VisualElement element)
```

#### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                              |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | The element that will get it's transform scale. |

Returns

| <b>TYPE</b>         | <b>DESCRIPTION</b>                 |
|---------------------|------------------------------------|
| UnityEngine.Vector2 | The return is the transform scale. |

## GetVisible(VisualElement)

This will get this element's visible.

Declaration

```
public static bool GetVisible(this VisualElement element)
```

Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                               |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | The element that will have it's visible changed. |

Returns

| <b>TYPE</b>    | <b>DESCRIPTION</b> |
|----------------|--------------------|
| System.Boolean |                    |

## GetWorldBound(VisualElement)

This will get the element WorldBound.

Declaration

```
public static Rect GetWorldBound(this VisualElement element)
```

Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>   |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | The element that will set the width and height of the element. |

## Returns

| Type             | Description |
|------------------|-------------|
| UnityEngine.Rect |             |

## GetWorldBoundPosition(VisualElement)

This will set the position of the element WorldBound.

### Declaration

```
public static Vector2 GetWorldBoundPosition(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will set the width and height of the element. |

## Returns

| Type                | Description |
|---------------------|-------------|
| UnityEngine.Vector2 |             |

## GetWorldBoundSize(VisualElement)

This will get the width and high of the element WorldBound.

### Declaration

```
public static Vector2 GetWorldBoundSize(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will set the width and height of the element. |

## Returns

| Type                | Description |
|---------------------|-------------|
| UnityEngine.Vector2 |             |

## MouseGrab(VisualElement, Vector2)

This will prepare the element for it being grabbed.

### Declaration

```
public static Vector2 MouseGrab(this VisualElement element, Vector2 position)
```

### Parameters

| Type                                 | Name     | Description                                |
|--------------------------------------|----------|--|
| UnityEngine.UIElements.VisualElement | element  | The element that will be grabbed by mouse. |
| UnityEngine.Vector2                  | position | This is the position of the element.       |

### Returns

| Type                | Description   |
|---------------------|---|
| UnityEngine.Vector2 | The return is the offset of the mouse and element position. |

## MouseRelease(VisualElement)

This will release the element.

### Declaration

```
public static Vector2 MouseRelease(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description   |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | The element that will be used to get the screen bounds. |

### Returns

| Type                | Description                                    |
|---------------------|--|
| UnityEngine.Vector2 | The return is the new position of the element. |

## Name(VisualElement, String)

This will set the elements' name.

#### Declaration

```
public static void Name(this VisualElement element, string name)
```

#### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will have a name applied to it. |
| System.String                        | name    | This is the name that will be applied to the element.    |

#### Remarks

the name will be structured as "*Type\_name*" with *name* replacing spacing with '-'.

### Pick(VisualElement, PickingMode)

This will change the pickmode from value.

#### Declaration

```
public static void Pick(this VisualElement element, PickingMode value)
```

#### Parameters

| Type                                 | Name    | Description                                      |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have its pickmode changed. |
| UnityEngine.UIElements.PickingMode   | value   | The PickingMode that will change value.          |

### Pick\_OFF(VisualElement)

This will not allow this element to be interactable

#### Declaration

```
public static void Pick_OFF(this VisualElement element)
```

#### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                                      |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | The element that will have it's pickmode set to ignore. |

## Pick\_ON(VisualElement)

This will allow this element to be interactable.

### Declaration

```
public static void Pick_ON(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>  |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | The element that will have it's pickmode set to position. |

## Pick\_Toggle(VisualElement)

This will change the pickmode from position to ignore and vice versa.

### Declaration

```
public static void Pick_Toggle(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                               |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | The element that will have its pickmode changed. |

## ResetRotation(VisualElement)

This will reset the rotation of the UI element.

### Declaration

```
public static void ResetRotation(this VisualElement element)
```

### Parameters

| <b>TYPE</b> | <b>NAME</b> | <b>DESCRIPTION</b> |
|-------------|-------------|--------------------|
|             |             |                    |

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                              |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | The element that will have it's rotation reset. |

## Rich\_Text(TextElement, Boolean)

This will set the TextElement's enableRichText.

### Declaration

```
public static void Rich_Text(this TextElement element, bool value = true)
```

### Parameters

| <b>TYPE</b>                        | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|------------------------------------|-------------|--|
| UnityEngine.UIElements.TextElement | element     | This is the element that will be changed.  |
| System.Boolean                     | value       | This value will be applied to the element. |

## Rotate\_Clockwise(VisualElement, Vector3)

This will set the rotation of the UI element on z-axis Counter-Clockwise.

### Declaration

```
public static void Rotate_Clockwise(this VisualElement element, Vector3 value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                           |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | The element that will have its rotation set. |
| UnityEngine.Vector3                  | value       |  |

## Rotate\_Clockwise\_X(VisualElement, Single)

This will increase the rotation of the UI element on z-axis Clockwise.

### Declaration

```
public static void Rotate_Clockwise_X(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                                      |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have its rotation changed. |
| System.Single                        | value   |  |

## Rotate\_Clockwise\_Z(VisualElement, Single)

This will increase the rotation of the UI element on z-axis Clockwise.

## Declaration

```
public static void Rotate_Clockwise_Z(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                                      |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have its rotation changed. |
| System.Single                        | value   |  |

## Rotate\_Counter\_Clockwise(VisualElement, Vector3)

This will set the rotation of the UI element on z-axis Counter-Clockwise.

## Declaration

```
public static void Rotate_Counter_Clockwise(this VisualElement element, Vector3 value)
```

## Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have its rotation set. |
| UnityEngine.Vector3                  | value   |  |

## Rotate\_Counter\_Clockwise\_X(VisualElement, Single)

This will increase the rotation of the UI element on z-axis Counter-Clockwise.

## Declaration

```
public static void Rotate_Counter_Clockwise_X(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                                      |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have its rotation changed. |
| System.Single                        | value   |  |

## Rotate\_Counter\_Clockwise\_Y(VisualElement, Single)

This will increase the rotation of the UI element on z-axis Counter-Clockwise.

## Declaration

```
public static void Rotate_Counter_Clockwise_Y(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                                      |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have its rotation changed. |
| System.Single                        | value   |  |

## Rotate\_Counter\_Clockwise\_Z(VisualElement, Single)

This will increase the rotation of the UI element on z-axis Counter-Clockwise.

## Declaration

```
public static void Rotate_Counter_Clockwise_Z(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                                      |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have its rotation changed. |
| System.Single                        | value   |  |

## RotateY\_Clockwise\_Y(VisualElement, Single)

This will increase the rotation of the UI element on z-axis Clockwise.

## Declaration

```
public static void RotateY_Clockwise_Y(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                                      |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have its rotation changed. |
| System.Single                        | value   |  |

## ScreenBound(VisualElement)

This will return the screen bound for the element, Should be used to clamp the element to the screen so it doesn't go off screen.

## Declaration

```
public static Vector2 ScreenBound(this VisualElement element)
```

## Parameters

| Type                                 | Name    | Description   |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be used to determine the screen size - the element. |

## Returns

| Type                | Description                     |
|---------------------|---------------------------------|
| UnityEngine.Vector2 | The return of the screen bound. |

## SetTransform\_Position(VisualElement, Vector2)

This will set the transform position.

## Declaration

```
public static void SetTransform_Position(this VisualElement element, Vector2 value)
```

## Parameters

| Type | Name | Description |
|------|------|-------------|
|      |      |             |

| TYPE                                 | NAME    | DESCRIPTION   |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | The element that will have it's transform position changed.     |
| UnityEngine.Vector2                  | value   | this is the value that will be apply to the transform position. |

### SetTransform\_Rotate(VisualElement, Vector3)

This will set the rotation of the UI element on z-axis Counter-Clockwise.

#### Declaration

```
public static void SetTransform_Rotate(this VisualElement element, Vector3 value)
```

#### Parameters

| TYPE                                 | NAME    | DESCRIPTION                                   |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | The element that will have it's rotation set. |
| UnityEngine.Vector3                  | value   |   |

### SetTransform\_Scale(VisualElement, Vector3)

This will set the transform scale.

#### Declaration

```
public static void SetTransform_Scale(this VisualElement element, Vector3 value)
```

#### Parameters

| TYPE                                 | NAME    | DESCRIPTION  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have it's transform scale changed.     |
| UnityEngine.Vector3                  | value   | this is the value that will be apply to the transform scale. |

### SetVisible(VisualElement, Boolean)

This will set the visible from value.

#### Declaration

```
public static void SetVisible(this VisualElement element, bool value)
```

## Parameters

| Type                                 | Name    | Description                                     |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | The element that will have its visible changed. |
| System.Boolean                       | value   | The boolean that will change value.             |

## StyleSheet(VisualElement, StyleSheet, Boolean)

This will add, remove a stylesheet in this parent or clear the stylesheets in the element.

## Declaration

```
public static void StyleSheet(this VisualElement element, StyleSheet style = null, bool isAdd = true)
```

## Parameters

| Type                                 | Name    | Description   |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | The element that will have a style sheet added, removed or cleared.   |
| UnityEngine.UIElements.StyleSheet    | style   | This is an optional param if not null will allow for adding or removing of this stylesheet. If null then clear all of the stylesheets from the element.   |
| System.Boolean                       | isAdd   | This is an optional param that will determine if the stylesheet will be added or removed from the element. This is an optional param that will determine if the stylesheet will be added or removed from the element. |

## ToolTip(VisualElement, String)

This is the tooltip of the element.

## Declaration

```
public static void ToolTip(this VisualElement element, string value)
```

## Parameters

| Type | Name | Description |
|------|------|-------------|
|      |      |             |

| Type                                 | Name    | Description   |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | The element that will have its tooltip changed.           |
| System.String                        | value   | This is the string that will be displayed on the tooltip. |

## UpdatePosition(VisualElement, Vector2)

This will move the element to the mouse position - the offset.

### Declaration

```
public static void UpdatePosition(this VisualElement element, Vector2 offset)
```

### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will be used to get the screen bounds.                        |
| UnityEngine.Vector2                  | offset  | The offset is the distance of the initial mouse grab. to the element position. |

## UsageHints(VisualElement, UsageHints)

This is the usageHints of the element.

### Declaration

```
public static void UsageHints(this VisualElement element, UsageHints value)
```

### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have its usageHints changed. |
| UnityEngine.UIElements.UsageHints    | value   | This is the UsageHints for the element.            |

## ViewDataKey(VisualElement, String)

This is the ViewDataKey of the element.

## Declaration

```
public static void ViewDataKey(this VisualElement element, string value)
```

## Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have it's viewDataKey changed. |
| System.String                        | value   | This is the data for the viewDataKey.                |

## Visible\_Hide(VisualElement)

This will hide this element but keep it's effects on the UI.

## Declaration

```
public static void Visible_Hide(this VisualElement element)
```

## Parameters

| Type                                 | Name    | Description   |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | The element that will have it's visible set to false. |

## Visible\_IsInvisible(VisualElement)

This will get the element's visible

## Declaration

```
public static bool Visible_IsInvisible(this VisualElement element)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be checked. |

## Returns

| <b>TYPE</b>    | <b>DESCRIPTION</b>               |
|----------------|----------------------------------|
| System.Boolean | return true if display is false. |

## Visible\_IsVisible(VisualElement)

---

This will get the element's visible

Declaration

```
public static bool Visible_IsVisible(this VisualElement element)
```

Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be checked. |

Returns

| <b>TYPE</b>    | <b>DESCRIPTION</b>              |
|----------------|---------------------------------|
| System.Boolean | return true if display is true. |

## Visible\_Show(VisualElement)

---

This will show this element and keep it's effects on the UI.

Declaration

```
public static void Visible_Show(this VisualElement element)
```

Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                                   |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | The element that will have it's visible set to true. |

## Visible\_ToggleVisibility(VisualElement)

---

This will toggle the visible from true to false and vice versa.

Declaration

```
public static void Visible_ToggleVisibility(this VisualElement element)
```

## Parameters

| Type                                 | Name    | Description                                      |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have it's visible changed. |

# Class VisualElementStyleExtension

This static class is the extensions of the VisualElement. That deals with the boilder plate and simplify the styles.

## Inheritance

System.Object

VisualElementStyleExtension

Namespace: [UtilitySystem.Extension](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public static class VisualElementStyleExtension
```

## Methods

### Align\_Content(VisualElement, Align)

This will set the element's style alignContent

#### Declaration

```
public static void Align_Content(this VisualElement element, Align value)
```

#### Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.Align         | value   | This value will be applied to the element. |

### Align\_Content(VisualElement, StyleKeyword)

This will set the element's style alignContent

#### Declaration

```
public static void Align_Content(this VisualElement element, StyleKeyword value)
```

#### Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

| <b>TYPE</b>                         | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|-------------------------------------|-------------|--|
| UnityEngine.UIElements.StyleKeyword | value       | This value will be applied to the element. |

## Align\_Item(VisualElement, Align)

This will set the element's style alignItems

### Declaration

```
public static void Align_Item(this VisualElement element, Align value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.Align         | value       | This value will be applied to the element. |

## Align\_Item(VisualElement, StyleKeyword)

This will set the element's style alignItems

### Declaration

```
public static void Align_Item(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Align\_Self(VisualElement, Align)

This will set the element's style alignSelf

### Declaration

```
public static void Align_Self(this VisualElement element, Align value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.Align         | value   | This value will be applied to the element. |

## Align\_Self(VisualElement, StyleKeyword)

This will set the element's style alignSelf

### Declaration

```
public static void Align_Self(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Anchored\_Position\_Bottom\_Left(VisualElement, UIAnchorEnum)

This will set the element's style Bottom & Left.

### Declaration

```
public static void Anchored_Position_Bottom_Left(this VisualElement element, UIAnchorEnum value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UIAnchorEnum                         | value   | This value will be applied to the element. |

## Anchored\_Position\_Bottom\_Right(VisualElement, UIAnchorEnum)

This will set the element's style Bottom & Right.

## Declaration

```
public static void Anchored_Position_Bottom_Right(this VisualElement element, UIAnchorEnum value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UIAnchorEnum                         | value   | This value will be applied to the element. |

## Anchored\_Position\_Top\_Left(VisualElement, UIAnchorEnum)

This will set the element's style Top & Left.

## Declaration

```
public static void Anchored_Position_Top_Left(this VisualElement element, UIAnchorEnum value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UIAnchorEnum                         | value   | This value will be applied to the element. |

## Anchored\_Position\_Top\_Right(VisualElement, UIAnchorEnum)

This will set the element's style Top & Right.

## Declaration

```
public static void Anchored_Position_Top_Right(this VisualElement element, UIAnchorEnum value)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

| <b>TYPE</b>  | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------|-------------|--|
| UIAnchorEnum | value       | This value will be applied to the element. |

## BackGround\_Color(VisualElement, Color)

This will set the element's style

### Declaration

```
public static void BackGround_Color(this VisualElement element, Color value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.Color                    | value       | This value will be applied to the element. |

## Background\_Image(VisualElement, Sprite)

This will set the element's style

### Declaration

```
public static void Background_Image(this VisualElement element, Sprite value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.Sprite                   | value       | This value will be applied to the element. |

## Background\_Image(VisualElement, Texture2D)

This will set the element's style

### Declaration

```
public static void Background_Image(this VisualElement element, Texture2D value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.Texture2D                | value   | This value will be applied to the element. |

## Background\_Image(VisualElement, Background)

This will set the element's style

### Declaration

```
public static void Background_Image(this VisualElement element, Background value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.Background    | value   | This value will be applied to the element. |

## Background\_Image(VisualElement, StyleKeyword)

This will set the element's style

### Declaration

```
public static void Background_Image(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Border(VisualElement, Single[])

This will set the element's style Border Width, Color & Radius.

## Declaration

```
public static void Border(this VisualElement element, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                                 |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.   |
| System.Single[]                      | args    | These value will be applied to the element. |

## Border(VisualElement, Color, Single[])

This will set the element's style Border Width, Color & Radius.

## Declaration

```
public static void Border(this VisualElement element, Color color, params float[] args)
```

## Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.                |
| UnityEngine.Color                    | color   | This color that will be applied to the element's border. |
| System.Single[]                      | args    | These value will be applied to the element.              |

## Border(VisualElement, Color[], Single[])

This will set the element's style Border Width, Color & Radius.

## Declaration

```
public static void Border(this VisualElement element, Color[] color, params float[] args)
```

## Parameters

| Type | Name | Description |
|------|------|-------------|
|      |      |             |

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.                |
| UnityEngine.Color[]                  | color   | This color that will be applied to the element's border. |
| System.Single[]                      | args    | These values will be applied to the element.             |

## Border(VisualElement, LengthUnit, Color, Single[])

---

This will set the element's style Border Width, Color & Radius.

### Declaration

```
public static void Border(this VisualElement element, LengthUnit unit, Color color, params float[] args)
```

### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.                |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.                |
| UnityEngine.Color                    | color   | This color that will be applied to the element's border. |
| System.Single[]                      | args    | These values will be applied to the element.             |

## Border(VisualElement, LengthUnit, Color[], Single[])

---

This will set the element's style Border Width, Color & Radius.

### Declaration

```
public static void Border(this VisualElement element, LengthUnit unit, Color[] color, params float[] args)
```

### Parameters

| Type | Name | Description |
|------|------|-------------|
|      |      |             |

| TYPE                                 | NAME    | DESCRIPTION  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.                |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.                |
| UnityEngine.Color[]                  | color   | This color that will be applied to the element's border. |
| System.Single[]                      | args    | These values will be applied to the element.             |

## Border(VisualElement, LengthUnit[], Color, Single[])

This will set the element's style Border Width, Color & Radius.

### Declaration

```
public static void Border(this VisualElement element, LengthUnit[] unit, Color color, params float[] args)
```

### Parameters

| TYPE                                 | NAME    | DESCRIPTION  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.                |
| UnityEngine.UIElements.LengthUnit[]  | unit    | This unit will be applied to the element.                |
| UnityEngine.Color                    | color   | This color that will be applied to the element's border. |
| System.Single[]                      | args    | These values will be applied to the element.             |

## Border(VisualElement, LengthUnit[], Color[], Single[])

This will set the element's style Border Width, Color & Radius.

### Declaration

```
public static void Border(this VisualElement element, LengthUnit[] unit, Color[] color, params float[] args)
```

### Parameters

| TYPE                                 | NAME    | DESCRIPTION  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.                |
| UnityEngine.UIElements.LengthUnit[]  | unit    | This unit will be applied to the element.                |
| UnityEngine.Color[]                  | color   | This color that will be applied to the element's border. |
| System.Single[]                      | args    | These value will be applied to the element.              |

## Border(VisualElement, StyleKeyword[])

This will set the element's style Border Width & Radius.

### Declaration

```
public static void Border(this VisualElement element, params StyleKeyword[] args)
```

### Parameters

| TYPE                                  | NAME    | DESCRIPTION                                 |
|---------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement  | element | This is the element that will be changed.   |
| UnityEngine.UIElements.StyleKeyword[] | args    | These value will be applied to the element. |

## Border\_Color(VisualElement, Color[])

This will set the element's style borderTopColor, borderRightColor, borderBottomColor & borderLeftColor.

### Declaration

```
public static void Border_Color(this VisualElement element, params Color[] args)
```

### Parameters

| TYPE                                 | NAME    | DESCRIPTION                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

| <b>TYPE</b>         | <b>NAME</b> | <b>DESCRIPTION</b>                          |
|---------------------|-------------|---|
| UnityEngine.Color[] | args        | These value will be applied to the element. |

## Border\_Color(VisualElement, StyleKeyword[])

This will set the element's style borderTopColor, borderRightColor, borderBottomColor & borderLeftColor.

### Declaration

```
public static void Border_Color(this VisualElement element, params StyleKeyword[] args)
```

### Parameters

| <b>TYPE</b>                           | <b>NAME</b> | <b>DESCRIPTION</b>                          |
|---------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement  | element     | This is the element that will be changed.   |
| UnityEngine.UIElements.StyleKeyword[] | args        | These value will be applied to the element. |

## Border\_Color\_Bottom(VisualElement, Color)

This will set the element's style borderBottomColor.

### Declaration

```
public static void Border_Color_Bottom(this VisualElement element, Color value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.Color                    | value       | This value will be applied to the element. |

## Border\_Color\_Bottom(VisualElement, StyleKeyword)

This will set the element's style borderBottomColor.

### Declaration

```
public static void Border_Color_Bottom(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Border\_Color\_Left(VisualElement, Color)

This will set the element's style borderLeftColor.

### Declaration

```
public static void Border_Color_Left(this VisualElement element, Color value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.Color                    | value   | This value will be applied to the element. |

## Border\_Color\_Left(VisualElement, StyleKeyword)

This will set the element's style borderLeftColor.

### Declaration

```
public static void Border_Color_Left(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Border\_Color\_Right(VisualElement, Color)

This will set the element's style borderRightColor.

## Declaration

```
public static void Border_Color_Right(this VisualElement element, Color value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.Color                    | value   | This value will be applied to the element. |

## Border\_Color\_Right(VisualElement, StyleKeyword)

This will set the element's style borderRightColor.

## Declaration

```
public static void Border_Color_Right(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Border\_Color\_Top(VisualElement, Color)

This will set the element's style borderTopColor.

## Declaration

```
public static void Border_Color_Top(this VisualElement element, Color value)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

| Type              | Name  | Description                                |
|-------------------|-------|--|
| UnityEngine.Color | value | This value will be applied to the element. |

## Border\_Color\_Top(VisualElement, StyleKeyword)

This will set the element's style borderTopColor.

### Declaration

```
public static void Border_Color_Top(this VisualElement element, StyleKeyword value)
```

### Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Border\_Radius(VisualElement, Single[])

This will set the element's style borderTopLeftRadius, borderTopRightRadius, borderBottomLeftRadius & borderBottomRightRadius.

### Declaration

```
public static void Border_Radius(this VisualElement element, params float[] args)
```

### Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Border\_Radius(VisualElement, LengthUnit, Single[])

This will set the element's style borderTopLeftRadius, borderTopRightRadius, borderBottomLeftRadius & borderBottomRightRadius.

### Declaration

```
public static void Border_Radius(this VisualElement element, LengthUnit unit, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Border\_Radius(VisualElement, LengthUnit[], Single[])

This will set the element's style borderTopLeftRadius, borderTopRightRadius, borderBottomLeftRadius & borderBottomRightRadius.

## Declaration

```
public static void Border_Radius(this VisualElement element, LengthUnit[] unit, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| UnityEngine.UIElements.LengthUnit[]  | unit    | This unit will be applied to the element.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Border\_Radius(VisualElement, StyleKeyword[])

This will set the element's style borderTopLeftRadius, borderTopRightRadius, borderBottomLeftRadius & borderBottomRightRadius.

## Declaration

```
public static void Border_Radius(this VisualElement element, params StyleKeyword[] args)
```

## Parameters

| <b>TYPE</b>                           | <b>NAME</b> | <b>DESCRIPTION</b>                          |
|---------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement  | element     | This is the element that will be changed.   |
| UnityEngine.UIElements.StyleKeyword[] | args        | These value will be applied to the element. |

## Border\_Radius\_Bottom\_Left(VisualElement, Single, LengthUnit)

---

This will set the element's style borderBottomLeftRadius.

### Declaration

```
public static void Border_Radius_Bottom_Left(this VisualElement element, float radius, LengthUnit unit = LengthUnit.Pixel)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |
| System.Single                        | radius      |   |
| UnityEngine.UIElements.LengthUnit    | unit        | This unit will be applied to the element. |

## Border\_Radius\_Bottom\_Left(VisualElement, StyleKeyword)

---

This will set the element's style borderBottomLeftRadius.

### Declaration

```
public static void Border_Radius_Bottom_Left(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Border\_Radius\_Bottom\_Right(VisualElement, Single, LengthUnit)

---

This will set the element's style borderBottomRadius.

#### Declaration

```
public static void Border_Radius_Bottom_Right(this VisualElement element, float radius, LengthUnit unit = LengthUnit.Pixel)
```

#### Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |
| System.Single                        | radius  |   |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element. |

### Border\_Radius\_Bottom\_Right(VisualElement, StyleKeyword)

This will set the element's style borderBottomRadius.

#### Declaration

```
public static void Border_Radius_Bottom_Right(this VisualElement element, StyleKeyword value)
```

#### Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

### Border\_Radius\_Top\_Left(VisualElement, Single, LengthUnit)

This will set the element's style borderTopLeftRadius.

#### Declaration

```
public static void Border_Radius_Top_Left(this VisualElement element, float radius, LengthUnit unit = LengthUnit.Pixel)
```

#### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |
| System.Single                        | radius      |   |
| UnityEngine.UIElements.LengthUnit    | unit        | This unit will be applied to the element. |

## Border\_Radius\_Top\_Left(VisualElement, StyleKeyword)

---

This will set the element's style borderTopLeftRadius.

### Declaration

```
public static void Border_Radius_Top_Left(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Border\_Radius\_Top\_Right(VisualElement, Single, LengthUnit)

---

This will set the element's style borderTopRightRadius.

### Declaration

```
public static void Border_Radius_Top_Right(this VisualElement element, float radius, LengthUnit unit = LengthUnit.Pixel)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |
| System.Single                        | radius      |   |
| UnityEngine.UIElements.LengthUnit    | unit        | This unit will be applied to the element. |

## Border\_Radius\_Top\_Right(VisualElement, StyleKeyword)

This will set the element's style borderTopRadius.

### Declaration

```
public static void Border_Radius_Top_Right(this VisualElement element, StyleKeyword value)
```

### Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Border\_Width(VisualElement, Single[])

This will set the element's style borderTopWidth, borderRightWidth, borderBottomWidth & borderLeftWidth.

### Declaration

```
public static void Border_Width(this VisualElement element, params float[] args)
```

### Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Border\_Width(VisualElement, StyleKeyword[])

This will set the element's style borderTopWidth, borderRightWidth, borderBottomWidth & borderLeftWidth.

### Declaration

```
public static void Border_Width(this VisualElement element, params StyleKeyword[] args)
```

### Parameters

| Type | Name | Description |
|------|------|-------------|
|      |      |             |

| <b>TYPE</b>                           | <b>NAME</b> | <b>DESCRIPTION</b>                          |
|---------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement  | element     | This is the element that will be changed.   |
| UnityEngine.UIElements.StyleKeyword[] | args        | These value will be applied to the element. |

## Border\_Width\_Bottom(VisualElement, Single)

---

This will set the element's style borderBottomWidth.

### Declaration

```
public static void Border_Width_Bottom(this VisualElement element, float value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |

## Border\_Width\_Bottom(VisualElement, StyleKeyword)

---

This will set the element's style borderBottomWidth.

### Declaration

```
public static void Border_Width_Bottom(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Border\_Width\_Left(VisualElement, Single)

---

This will set the element's style borderLeftWidth.

## Declaration

```
public static void Border_Width_Left(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |

## Border\_Width\_Left(VisualElement, StyleKeyword)

This will set the element's style borderLeftWidth.

## Declaration

```
public static void Border_Width_Left(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Border\_Width\_Right(VisualElement, Single)

This will set the element's style borderRightWidth.

## Declaration

```
public static void Border_Width_Right(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

| <b>TYPE</b>   | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|---------------|-------------|--|
| System.Single | value       | This value will be applied to the element. |

## Border\_Width\_Right(VisualElement, StyleKeyword)

This will set the element's style borderRightWidth.

### Declaration

```
public static void Border_Width_Right(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Border\_Width\_Top(VisualElement, Single)

This will set the element's style borderTopWidth.

### Declaration

```
public static void Border_Width_Top(this VisualElement element, float value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |

## Border\_Width\_Top(VisualElement, StyleKeyword)

This will set the element's style borderTopWidth.

### Declaration

```
public static void Border_Width_Top(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Color(VisualElement, Color)

This will set the element's style

### Declaration

```
public static void Color(this VisualElement element, Color value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.Color                    | value   | This value will be applied to the element. |

## Color(VisualElement, StyleKeyword)

This will set the element's style

### Declaration

```
public static void Color(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Display\_Hide(VisualElement)

This will set the element's style display

## Declaration

```
public static void Display_Hide(this VisualElement element)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

## Display\_IsInvisible(VisualElement)

This will get the element's style display

## Declaration

```
public static bool Display_IsInvisible(this VisualElement element)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be checked. |

## Returns

| Type           | Description                     |
|----------------|---------------------------------|
| System.Boolean | return true if display is None. |

## Display\_IsVisible(VisualElement)

This will get the element's style display

## Declaration

```
public static bool Display_IsVisible(this VisualElement element)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be checked. |

## Returns

| Type           | Description                     |
|----------------|---------------------------------|
| System.Boolean | return true if display is Flex. |

## Display\_Show(VisualElement)

This will set the element's style display

### Declaration

```
public static void Display_Show(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

## Display\_ToggleVisibility(VisualElement)

This will set the element's style display

### Declaration

```
public static void Display_ToggleVisibility(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

## Flex\_Basis(VisualElement, Single)

This will set the element's style flexBasis

### Declaration

```
public static void Flex_Basis(this VisualElement element, float value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |

## Flex\_Basis(VisualElement, StyleKeyword)

---

This will set the element's style flexBasis

### Declaration

```
public static void Flex_Basis(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Flex\_Direction(VisualElement, FlexDirection)

---

This will set the element's style flexDirection

### Declaration

```
public static void Flex_Direction(this VisualElement element, FlexDirection value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.FlexDirection | value       | This value will be applied to the element. |

## Flex\_Direction(VisualElement, StyleKeyword)

---

This will set the element's style flexDirection

## Declaration

```
public static void Flex_Direction(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Flex\_Grow(VisualElement, Single)

This will set the element's style flexGrow

## Declaration

```
public static void Flex_Grow(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |

## Flex\_Grow(VisualElement, StyleKeyword)

This will set the element's style flexGrow

## Declaration

```
public static void Flex_Grow(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

| <b>TYPE</b>                         | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|-------------------------------------|-------------|--|
| UnityEngine.UIElements.StyleKeyword | value       | This value will be applied to the element. |

## Flex\_Shrink(VisualElement, Single)

This will set the element's style flexShrink

### Declaration

```
public static void Flex_Shrink(this VisualElement element, float value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |

## Flex\_Shrink(VisualElement, StyleKeyword)

This will set the element's style flexShrink

### Declaration

```
public static void Flex_Shrink(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Flex\_Wrap(VisualElement, StyleKeyword)

This will set the element's style flexWrap

### Declaration

```
public static void Flex_Wrap(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Flex\_Wrap(VisualElement, Wrap)

This will set the element's style flexWrap

### Declaration

```
public static void Flex_Wrap(this VisualElement element, Wrap value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.Wrap          | value   | This value will be applied to the element. |

## Font(VisualElement, Font)

This will set the element's style unityFontDefinition.

### Declaration

```
public static void Font(this VisualElement element, Font value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.Font                     | value   | This value will be applied to the element. |

## Font(VisualElement, FontAsset)

This will set the element's style unityFontDefinition.

## Declaration

```
public static void Font(this VisualElement element, FontAsset value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.TextCore.Text.FontAsset  | value   | This value will be applied to the element. |

## Font\_Size(VisualElement, Single)

This will set the element's style fontsize.

## Declaration

```
public static void Font_Size(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |

## Font\_Size(VisualElement, StyleKeyword)

This will set the element's style fontSize.

## Declaration

```
public static void Font_Size(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

| <b>TYPE</b>                         | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|-------------------------------------|-------------|--|
| UnityEngine.UIElements.StyleKeyword | value       | This value will be applied to the element. |

## GetBorder\_Height(VisualElement)

This will get the element's style borderTopWidth & borderBottomWidth.

### Declaration

```
public static float GetBorder_Height(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |

### Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b>                      |
|---------------|---|
| System.Single | The return is the height of the border. |

## GetBorder\_Size(VisualElement)

This will get the element's style borderTopWidth, borderRightWidth, borderBottomWidth & borderLeftWidth.

### Declaration

```
public static Vector2 GetBorder_Size(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |

### Returns

| <b>TYPE</b>         | <b>DESCRIPTION</b>                                 |
|---------------------|--|
| UnityEngine.Vector2 | The return is a Vector2 of the size of the border. |

## GetBorder\_Width(VisualElement)

This will get the element's style borderLeftWidth & borderRightWidth.

### Declaration

```
public static float GetBorder_Width(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

### Returns

| Type          | Description                            |
|---------------|--|
| System.Single | The return is the width of the border. |

## GetDisplay(VisualElement)

This will get this element's visible.

### Declaration

```
public static DisplayStyle GetDisplay(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description                                      |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | The element that will have it's visible changed. |

### Returns

| Type                                | Description |
|-------------------------------------|-------------|
| UnityEngine.UIElements.DisplayStyle |             |

## GetMargin\_Height(VisualElement)

This will get the element's style marginTop & marginBottom.

### Declaration

```
public static float GetMargin_Height(this VisualElement element)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

## Returns

| Type          | Description |
|---------------|-------------|
| System.Single |             |

## GetMargin\_Size(VisualElement)

This will get the element's style marginTop, marginRight, marginBottom & marginLeft.

## Declaration

```
public static Vector2 GetMargin_Size(this VisualElement element)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

## Returns

| Type                | Description |
|---------------------|-------------|
| UnityEngine.Vector2 |             |

## GetMargin\_Width(VisualElement)

This will get the element's style marginLeft & marginRight.

## Declaration

```
public static float GetMargin_Width(this VisualElement element)
```

## Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |

Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b> |
|---------------|--------------------|
| System.Single |                    |

### GetPadding\_Height(VisualElement)

This will get the element's style paddingTop & paddingBottom.

Declaration

```
public static float GetPadding_Height(this VisualElement element)
```

Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |

Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b> |
|---------------|--------------------|
| System.Single |                    |

### GetPadding\_Size(VisualElement)

This will get the element's style paddingTop, paddingRight, paddingBottom & paddingLeft.

Declaration

```
public static Vector2 GetPadding_Size(this VisualElement element)
```

Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |

Returns

| <b>TYPE</b>         | <b>DESCRIPTION</b>                                     |
|---------------------|--|
| UnityEngine.Vector2 | The return is a Vector2 of the padding width & Height. |

## GetPadding\_Width(VisualElement)

This will get the element's style paddingLeft & paddingRight.

### Declaration

```
public static float GetPadding_Width(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |

### Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b> |
|---------------|--------------------|
| System.Single |                    |

## GetRadius\_Bottom(VisualElement)

This will get the element's style borderBottomLeftRadius & borderBottomRightRadius.

### Declaration

```
public static float GetRadius_Bottom(this VisualElement element)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |

### Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b>                                  |
|---------------|---|
| System.Single | The return is the size of the border radius Bottom. |

## GetRadius\_Size(VisualElement)

This will get the element's style borderTopLeftRadius, borderTopRightRadius, borderBottomLeftRadius & borderBottomRightRadius.

### Declaration

```
public static Vector2 GetRadius_Size(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

### Returns

| Type                | Description  |
|---------------------|--|
| UnityEngine.Vector2 | The return is a Vector2 the size of the border radius. |

## GetRadius\_Top(VisualElement)

This will get the element's style borderTopLeftRadius & borderTopRightRadius.

### Declaration

```
public static float GetRadius_Top(this VisualElement element)
```

### Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

### Returns

| Type          | Description                                      |
|---------------|--|
| System.Single | The return is the size of the border radius Top. |

## GetSize(VisualElement, Boolean, Boolean)

### Declaration

```
public static Vector2 GetSize(this VisualElement element, bool margin = true, bool padding = true)
```

## Parameters

| Type                                 | Name    | Description |
|--------------------------------------|---------|-------------|
| UnityEngine.UIElements.VisualElement | element |             |
| System.Boolean                       | margin  |             |
| System.Boolean                       | padding |             |

## Returns

| Type                | Description |
|---------------------|-------------|
| UnityEngine.Vector2 |             |

## GetSize\_Max(VisualElement, Boolean, Boolean)

This will Get the element's style maxWidth, maxHeight, margin, & padding.

## Declaration

```
public static Vector2 GetSize_Max(this VisualElement element, bool margin = true, bool padding = true)
```

## Parameters

| Type                                 | Name    | Description                                   |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.     |
| System.Boolean                       | margin  | If true then add margin size to the element.  |
| System.Boolean                       | padding | If true then add padding size to the element. |

## Returns

| Type                | Description   |
|---------------------|---|
| UnityEngine.Vector2 | The return is the max size of the element, with margin & padding if true. |

## GetSize\_Min(VisualElement, Boolean, Boolean)

This will set the element's style minWidth, minHeight, margin, & padding.

## Declaration

```
public static Vector2 GetSize_Min(this VisualElement element, bool margin = true, bool padding = true)
```

## Parameters

| Type                                 | Name    | Description                                   |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.     |
| System.Boolean                       | margin  | If true then add margin size to the element.  |
| System.Boolean                       | padding | If true then add padding size to the element. |

## Returns

| Type                | Description   |
|---------------------|---|
| UnityEngine.Vector2 | The return is the max size of the element, with margin & padding if true. |

## Margin(VisualElement, Single[])

This will set the element's style marginTop, marginRight, marginBottom & marginLeft.

## Declaration

```
public static void Margin(this VisualElement element, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                                 |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.   |
| System.Single[]                      | args    | These value will be applied to the element. |

## Margin(VisualElement, LengthUnit, Single[])

This will set the element's style

## Declaration

```
public static void Margin(this VisualElement element, LengthUnit unit, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Margin(VisualElement, LengthUnit[], Single[])

This will set the element's style

## Declaration

```
public static void Margin(this VisualElement element, LengthUnit[] unit, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| UnityEngine.UIElements.LengthUnit[]  | unit    | This unit will be applied to the element.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Margin(VisualElement, StyleKeyword[])

This will set the element's style

## Declaration

```
public static void Margin(this VisualElement element, params StyleKeyword[] args)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

| <b>TYPE</b>                           | <b>NAME</b> | <b>DESCRIPTION</b>                          |
|---------------------------------------|-------------|---|
| UnityEngine.UIElements.StyleKeyword[] | args        | These value will be applied to the element. |

## Margin\_Bottom(VisualElement, Single, LengthUnit)

This will set the element's style marginBottom.

### Declaration

```
public static void Margin_Bottom(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit        | This unit will be applied to the element.  |

## Margin\_Bottom(VisualElement, StyleKeyword)

This will set the element's style marginBottom.

### Declaration

```
public static void Margin_Bottom(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Margin\_Left(VisualElement, Single, LengthUnit)

This will set the element's style marginLeft.

## Declaration

```
public static void Margin_Left(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.  |

## Margin\_Left(VisualElement, StyleKeyword)

This will set the element's style marginLeft.

## Declaration

```
public static void Margin_Left(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Margin\_Right(VisualElement, Single, LengthUnit)

This will set the element's style marginRight.

## Declaration

```
public static void Margin_Right(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

## Parameters

| Type | Name | Description |
|------|------|-------------|
|      |      |             |

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit        | This unit will be applied to the element.  |

## Margin\_Right(VisualElement, StyleKeyword)

This will set the element's style marginRight.

### Declaration

```
public static void Margin_Right(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Margin\_Top(VisualElement, Single, LengthUnit)

This will set the element's style marginTop.

### Declaration

```
public static void Margin_Top(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |

| <b>TYPE</b>                       | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|-----------------------------------|-------------|---|
| UnityEngine.UIElements.LengthUnit | unit        | This unit will be applied to the element. |

## Margin\_Top(VisualElement, StyleKeyword)

This will set the element's style marginTop.

### Declaration

```
public static void Margin_Top(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Overflow(VisualElement, Overflow)

This will set the element's style overflow.

### Declaration

```
public static void Overflow(this VisualElement element, Overflow value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.Overflow      | value       | This value will be applied to the element. |

## Overflow(VisualElement, StyleKeyword)

This will set the element's style overflow.

### Declaration

```
public static void Overflow(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Padding(VisualElement, Single[])

This will set the element's style paddingTop, paddingRight, paddingBottom & paddingLeft.

### Declaration

```
public static void Padding(this VisualElement element, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Padding(VisualElement, LengthUnit, Single[])

This will set the element's style paddingTop, paddingRight, paddingBottom & paddingLeft.

### Declaration

```
public static void Padding(this VisualElement element, LengthUnit unit = LengthUnit.Pixel, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Padding(VisualElement, LengthUnit[], Single[])

This will set the element's style paddingTop, paddingRight, paddingBottom & paddingLeft.

### Declaration

```
public static void Padding(this VisualElement element, LengthUnit[] unit, params float[] args)
```

### Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| UnityEngine.UIElements.LengthUnit[]  | unit    | This unit will be applied to the element.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Padding(VisualElement, StyleKeyword[])

This will set the element's style paddingTop, paddingRight, paddingBottom & paddingLeft.

### Declaration

```
public static void Padding(this VisualElement element, params StyleKeyword[] args)
```

### Parameters

| Type                                  | Name    | Description                                  |
|---------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement  | element | This is the element that will be changed.    |
| UnityEngine.UIElements.StyleKeyword[] | args    | These values will be applied to the element. |

## Padding\_Bottom(VisualElement, Single, LengthUnit)

This will set the element's style paddingBottom.

### Declaration

```
public static void Padding_Bottom(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit        | This unit will be applied to the element.  |

## Padding\_Bottom(VisualElement, StyleKeyword)

---

This will set the element's style paddingBottom.

### Declaration

```
public static void Padding_Bottom(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Padding\_Left(VisualElement, Single, LengthUnit)

---

This will set the element's style paddingLeft.

### Declaration

```
public static void Padding_Left(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |

| <b>TYPE</b>                       | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|-----------------------------------|-------------|---|
| UnityEngine.UIElements.LengthUnit | unit        | This unit will be applied to the element. |

## Padding\_Left(VisualElement, StyleKeyword)

This will set the element's style paddingLeft.

### Declaration

```
public static void Padding_Left(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Padding\_Right(VisualElement, Single, LengthUnit)

This will set the element's style paddingRight.

### Declaration

```
public static void Padding_Right(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit        | This unit will be applied to the element.  |

## Padding\_Right(VisualElement, StyleKeyword)

This will set the element's style paddingRight.

## Declaration

```
public static void Padding_Right(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Padding\_Top(VisualElement, Single, LengthUnit)

This will set the element's style paddingTop.

## Declaration

```
public static void Padding_Top(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.  |

## Padding\_Top(VisualElement, StyleKeyword)

## Declaration

```
public static void Padding_Top(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

| <b>TYPE</b>                         | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|-------------------------------------|-------------|--|
| UnityEngine.UIElements.StyleKeyword | value       | This value will be applied to the element. |

## Position\_Bottom\_Left(VisualElement, Single, Single)

This will set the element's style Bottom & Left.

### Declaration

```
public static void Position_Bottom_Left(this VisualElement element, float x, float y)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed. |
| System.Single                        | x           |   |
| System.Single                        | y           |   |

## Position\_Bottom\_Left(VisualElement, Vector2)

This will set the element's style Bottom & Left.

### Declaration

```
public static void Position_Bottom_Left(this VisualElement element, Vector2 value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.Vector2                  | value       | This value will be applied to the element. |

## Position\_Bottom\_Right(VisualElement, Single, Single)

This will set the element's style Bottom & Right.

### Declaration

```
public static void Position_Bottom_Right(this VisualElement element, float x, float y)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | x       | This value will be applied to the element. |
| System.Single                        | y       | This value will be applied to the element. |

## Position\_Bottom\_Right(VisualElement, Vector2)

This will set the element's style Bottom & Right.

## Declaration

```
public static void Position_Bottom_Right(this VisualElement element, Vector2 value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.Vector2                  | value   | This value will be applied to the element. |

## Position\_Top\_Left(VisualElement, Single, Single)

This will set the element's style Top & Left.

## Declaration

```
public static void Position_Top_Left(this VisualElement element, float x, float y)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |
| System.Single                        | x       |   |
| System.Single                        | y       |   |

## Position\_Top\_Left(VisualElement, Vector2)

---

This will set the element's style Top & Left.

### Declaration

```
public static void Position_Top_Left(this VisualElement element, Vector2 value)
```

### Parameters

| TYPE                                 | NAME    | DESCRIPTION                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.Vector2                  | value   | This value will be applied to the element. |

## Position\_Top\_Right(VisualElement, Single, Single)

---

This will set the element's style Top & Right.

### Declaration

```
public static void Position_Top_Right(this VisualElement element, float x, float y)
```

### Parameters

| TYPE                                 | NAME    | DESCRIPTION                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |
| System.Single                        | x       |   |
| System.Single                        | y       |   |

## Position\_Top\_Right(VisualElement, Vector2)

---

This will set the element's style Top & Right.

### Declaration

```
public static void Position_Top_Right(this VisualElement element, Vector2 value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.Vector2                  | value       | This value will be applied to the element. |

## SetDisplay(VisualElement, DisplayStyle)

---

This will set the visible from value.

### Declaration

```
public static void SetDisplay(this VisualElement element, DisplayStyle value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                              |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | element     | The element that will have its visible changed. |
| UnityEngine.UIElements.DisplayStyle  | value       | TheDisplayStyle that will change value.         |

## Size(VisualElement, Single[])

---

This will set the element's style width & height.

### Declaration

```
public static void Size(this VisualElement element, params float[] args)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                           |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.    |
| System.Single[]                      | args        | These values will be applied to the element. |

## Size(VisualElement, LengthUnit, Single[])

---

This will set the element's style width & height.

## Declaration

```
public static void Size(this VisualElement element, LengthUnit unit = LengthUnit.Pixel, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Size(VisualElement, LengthUnit, LengthUnit, Single[])

This will set the element's style width & height.

## Declaration

```
public static void Size(this VisualElement element, LengthUnit unitWidth, LengthUnit unitHeight, params float[] args)
```

## Parameters

| Type                                 | Name       | Description                                  |
|--------------------------------------|------------|--|
| UnityEngine.UIElements.VisualElement | element    | This is the element that will be changed.    |
| UnityEngine.UIElements.LengthUnit    | unitWidth  | This unit will be applied to the element.    |
| UnityEngine.UIElements.LengthUnit    | unitHeight | This unit will be applied to the element.    |
| System.Single[]                      | args       | These values will be applied to the element. |

## Size(VisualElement, StyleKeyword)

This will set the element's style width & height.

## Declaration

```
public static void Size(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Size(VisualElement, Vector2, LengthUnit[])

This will set the element's style width & height.

### Declaration

```
public static void Size(this VisualElement element, Vector2 value, params LengthUnit[] unit)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.Vector2                  | value   | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit[]  | unit    | This unit will be applied to the element.  |

## Size\_Height(VisualElement, Single, LengthUnit)

This will set the element's style height.

### Declaration

```
public static void Size_Height(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |

| <b>TYPE</b>                       | <b>NAME</b> | <b>DESCRIPTION</b>                        |
|-----------------------------------|-------------|---|
| UnityEngine.UIElements.LengthUnit | unit        | This unit will be applied to the element. |

## Size\_Height(VisualElement, StyleKeyword)

This will set the element's style height.

### Declaration

```
public static void Size_Height(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Size\_Height\_Max(VisualElement, Single, LengthUnit)

This will set the element's style maxHeight.

### Declaration

```
public static void Size_Height_Max(this VisualElement element, float value, LengthUnit unit =
LengthUnit.Pixel)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit        | This unit will be applied to the element.  |

## Size\_Height\_Max(VisualElement, StyleKeyword)

This will set the element's style maxHeight.

## Declaration

```
public static void Size_Height_Max(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Size\_Height\_Min(VisualElement, Single, LengthUnit)

This will set the element's style minHeight

## Declaration

```
public static void Size_Height_Min(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.  |

## Size\_Height\_Min(VisualElement, StyleKeyword)

This will set the element's style minHeight

## Declaration

```
public static void Size_Height_Min(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type | Name | Description |
|------|------|-------------|
|      |      |             |

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## Size\_Max(VisualElement, Single[])

This will set the element's style maxWidth & maxHeight.

### Declaration

```
public static void Size_Max(this VisualElement element, params float[] args)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                           |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.    |
| System.Single[]                      | args        | These values will be applied to the element. |

## Size\_Max(VisualElement, LengthUnit, Single[])

This will set the element's style maxWidth & maxHeight.

### Declaration

```
public static void Size_Max(this VisualElement element, LengthUnit unit = LengthUnit.Pixel, params float[] args)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                           |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.    |
| UnityEngine.UIElements.LengthUnit    | unit        | This unit will be applied to the element.    |
| System.Single[]                      | args        | These values will be applied to the element. |

## Size\_Max(VisualElement, LengthUnit, LengthUnit, Single[])

This will set the element's style maxWidth & maxHeight.

### Declaration

```
public static void Size_Max(this VisualElement element, LengthUnit unitWidth, LengthUnit unitHeight, params float[] args)
```

### Parameters

| Type                                 | Name       | Description                                  |
|--------------------------------------|------------|--|
| UnityEngine.UIElements.VisualElement | element    | This is the element that will be changed.    |
| UnityEngine.UIElements.LengthUnit    | unitWidth  | This unit will be applied to the element.    |
| UnityEngine.UIElements.LengthUnit    | unitHeight | This unit will be applied to the element.    |
| System.Single[]                      | args       | These values will be applied to the element. |

## Size\_Max(VisualElement, StyleKeyword)

This will set the element's style maxWidth & maxHeight.

### Declaration

```
public static void Size_Max(this VisualElement element, StyleKeyword value)
```

### Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Size\_Max(VisualElement, Vector2, LengthUnit[])

This will set the element's style maxWidth & maxHeight.

### Declaration

```
public static void Size_Max(this VisualElement element, Vector2 value, params LengthUnit[] unit)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.Vector2                  | value   | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit[]  | unit    | This unit will be applied to the element.  |

## Size\_Min(VisualElement, Single[])

This will set the element's style minWidth & minHeight.

### Declaration

```
public static void Size_Min(this VisualElement element, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                                  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.    |
| System.Single[]                      | args    | These values will be applied to the element. |

## Size\_Min(VisualElement, LengthUnit, Single[])

This will set the element's style minWidth & minHeight.

### Declaration

```
public static void Size_Min(this VisualElement element, LengthUnit unit = LengthUnit.Pixel, params float[] args)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element. |

| Type            | Name | Description                                 |
|-----------------|------|---|
| System.Single[] | args | These value will be applied to the element. |

## Size\_Min(VisualElement, LengthUnit, LengthUnit, Single[])

This will set the element's style minWidth & minHeight.

### Declaration

```
public static void Size_Min(this VisualElement element, LengthUnit unitWidth, LengthUnit unitHeight, params float[] args)
```

### Parameters

| Type                                 | Name       | Description                                 |
|--------------------------------------|------------|---|
| UnityEngine.UIElements.VisualElement | element    | This is the element that will be changed.   |
| UnityEngine.UIElements.LengthUnit    | unitWidth  | This unit will be applied to the element.   |
| UnityEngine.UIElements.LengthUnit    | unitHeight | This unit will be applied to the element.   |
| System.Single[]                      | args       | These value will be applied to the element. |

## Size\_Min(VisualElement, StyleKeyword)

This will set the element's style minWidth & minHeight.

### Declaration

```
public static void Size_Min(this VisualElement element, StyleKeyword value)
```

### Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Size\_Min(VisualElement, Vector2, LengthUnit[])

This will set the element's style minWidth & minHeight.

### Declaration

```
public static void Size_Min(this VisualElement element, Vector2 value, params LengthUnit[] unit)
```

### Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.Vector2                  | value   | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit[]  | unit    | This unit will be applied to the element.  |

## Size\_Width(VisualElement, Single, LengthUnit)

This will set the element's style width.

### Declaration

```
public static void Size_Width(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

### Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.  |

## Size\_Width(VisualElement, StyleKeyword)

This will set the element's style width.

### Declaration

```
public static void Size_Width(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Size\_Width\_Max(VisualElement, Single, LengthUnit)

This will set the element's style maxWidth.

### Declaration

```
public static void Size_Width_Max(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.  |

## Size\_Width\_Max(VisualElement, StyleKeyword)

This will set the element's style maxWidth.

### Declaration

```
public static void Size_Width_Max(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Size\_Width\_Min(VisualElement, Single, LengthUnit)

This will set the element's style minWidth

### Declaration

```
public static void Size_Width_Min(this VisualElement element, float value, LengthUnit unit = LengthUnit.Pixel)
```

### Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |
| UnityEngine.UIElements.LengthUnit    | unit    | This unit will be applied to the element.  |

## Size\_Width\_Min(VisualElement, StyleKeyword)

This will set the element's style minWidth

### Declaration

```
public static void Size_Width_Min(this VisualElement element, StyleKeyword value)
```

### Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## StylePosition(VisualElement, Position)

This will set the element's style position.

### Declaration

```
public static void StylePosition(this VisualElement element, Position value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.Position      | value       | This value will be applied to the element. |

## StylePosition(VisualElement, StyleKeyword)

---

This will set the element's style position.

### Declaration

```
public static void StylePosition(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## StylePosition\_Bottom(VisualElement, Single)

---

This will set the element's style bottom.

### Declaration

```
public static void StylePosition_Bottom(this VisualElement element, float value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |

## StylePosition\_Bottom(VisualElement, StyleKeyword)

---

This will set the element's style bottom.

## Declaration

```
public static void StylePosition_Bottom(this VisualElement element, StyleKeyword value)
```

## Parameters

| TYPE                                 | NAME    | DESCRIPTION                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## StylePosition\_Left(VisualElement, Single)

This will set the element's style left.

## Declaration

```
public static void StylePosition_Left(this VisualElement element, float value)
```

## Parameters

| TYPE                                 | NAME    | DESCRIPTION                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |

## StylePosition\_Left(VisualElement, StyleKeyword)

This will set the element's style left.

## Declaration

```
public static void StylePosition_Left(this VisualElement element, StyleKeyword value)
```

## Parameters

| TYPE                                 | NAME    | DESCRIPTION                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

| <b>TYPE</b>                         | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|-------------------------------------|-------------|--|
| UnityEngine.UIElements.StyleKeyword | value       | This value will be applied to the element. |

## StylePosition\_Right(VisualElement, Single)

This will set the element's style right.

### Declaration

```
public static void StylePosition_Right(this VisualElement element, float value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| System.Single                        | value       | This value will be applied to the element. |

## StylePosition\_Right(VisualElement, StyleKeyword)

This will set the element's style right.

### Declaration

```
public static void StylePosition_Right(this VisualElement element, StyleKeyword value)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | element     | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value       | This value will be applied to the element. |

## StylePosition\_Toggle(VisualElement)

This will set the element's style position.

### Declaration

```
public static void StylePosition_Toggle(this VisualElement element)
```

## Parameters

| Type                                 | Name    | Description                               |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed. |

## StylePosition\_Top(VisualElement, Single)

This will set the element's style top.

### Declaration

```
public static void StylePosition_Top(this VisualElement element, float value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| System.Single                        | value   | This value will be applied to the element. |

## StylePosition\_Top(VisualElement, StyleKeyword)

This will set the element's style top.

### Declaration

```
public static void StylePosition_Top(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## Text\_Position(VisualElement, TextAnchor)

This will set the element's style unityTextAlign.

### Declaration

```
public static void Text_Position(this VisualElement element, TextAnchor value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.TextAnchor               | value   | This value will be applied to the element. |

## White\_Space(VisualElement, StyleKeyword)

This will set the element's style whiteSpace

## Declaration

```
public static void White_Space(this VisualElement element, StyleKeyword value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.StyleKeyword  | value   | This value will be applied to the element. |

## White\_Space(VisualElement,WhiteSpace)

This will set the element's style whiteSpace

## Declaration

```
public static void White_Space(this VisualElement element, WhiteSpace value)
```

## Parameters

| Type                                 | Name    | Description                                |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | element | This is the element that will be changed.  |
| UnityEngine.UIElements.WhiteSpace    | value   | This value will be applied to the element. |

# Namespace UtilitySystem.Script

## Classes

### Assets

---

This static class contains boilderplate method(s) relating to AssetDatabase.

### CreateElements

---

This static class will allow the creation of UI Toolkit elements.

### RichText

---

This static class will simplify setting up rich text for "Text Mesh Pro" or "UI ToolKit"

# Class Assets

This static class contains boilderplate method(s) relating to AssetDatabase.

Inheritance

System.Object

Assets

Namespace: [UtilitySystem.Script](#)

Assembly: Assembly-CSharp.dll

Syntax

```
public static class Assets
```

# Class CreateElements

This static class will allow the creation of UI Toolkit elements.

## Inheritance

System.Object

CreateElements

Namespace: [UtilitySystem.Script](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public static class CreateElements
```

## Methods

### CreateElement\_Button(VisualElement, Action)

This will create a Button and add to the element that invokes this.

#### Declaration

```
public static Button CreateElement_Button(this VisualElement parent, Action action)
```

#### Parameters

| Type                                 | Name   | Description  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |
| System.Action                        | action | This is the action when the Button is clicked.     |

#### Returns

| Type                          | Description                               |
|-------------------------------|---|
| UnityEngine.UIElements.Button | The return is the Button that is created. |

### CreateElement\_ContextMenuItem(VisualElement, ContextMenuItemData[])

This will create a ContextMenuItemElement and add to the element that invokes this.

#### Declaration

```
public static ContextMenuItemElement CreateElement_ContextMenu(this VisualElement parent, params  
ContextMenuItemData[] data)
```

## Parameters

| Type                                 | Name   | Description  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |
| ContextMenuItemData[]                | data   | These params are the data of the Context Menu.     |

## Returns

| Type               | Description   |
|--------------------|---|
| ContextMenuElement | The return is the ContextMenuElement that is created. |

## CreateElement\_ContextMenu(VisualElement, List<ContextMenuItemData>)

This will create a ContextMenuElement and add to the element that invokes this.

## Declaration

```
public static ContextMenuElement CreateElement_ContextMenu(this VisualElement parent,  
List<ContextMenuItemData> data)
```

## Parameters

| Type   | Name   | Description  |
|--|--------|--|
| UnityEngine.UIElements.VisualElement                 | parent | This param is the element that invokes the method. |
| System.Collections.Generic.List<ContextMenuItemData> | data   | These are the data of the Context Menu.            |

## Returns

| Type               | Description   |
|--------------------|---|
| ContextMenuElement | The return is the ContextMenuElement that is created. |

## CreateElement\_ContextMenu\_Item(VisualElement, ContextMenuItemData)

This will create a ContextMenuItemElement and add to the element that invokes this.

## Declaration

```
public static ContextMenuItemElement CreateElement_ContextMenu_Item(this VisualElement parent,  
ContextMenuItemData data)
```

#### Parameters

| Type                                 | Name   | Description  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |
| ContextMenuItemData                  | data   | This is the data of the Context Menu Item.         |

#### Returns

| Type                   | Description   |
|------------------------|---|
| ContextMenuItemElement | The return is the ContextMenuItemElement that is created. |

### CreateElement\_ContextMenu\_Item\_Button(VisualElement, ContextMenuItemButtonData)

This will create a ContextMenuItemButtonElement and add to the element that invokes this.

#### Declaration

```
public static ContextMenuItemButtonElement CreateElement_ContextMenu_Item_Button(this VisualElement parent,  
ContextMenuItemButtonData data)
```

#### Parameters

| Type                                 | Name   | Description  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |
| ContextMenuItemButtonData            | data   | This is the data of the Context Menu button.       |

#### Returns

| Type                         | Description   |
|------------------------------|---|
| ContextMenuItemButtonElement | The return is the ContextMenuItemButtonElement that is created. |

### CreateElement\_ContextMenu\_Item\_Parent(VisualElement, ContextMenuItemParentData)

This will create a ContextMenuItemParentElement and add to the element that invokes this.

## Declaration

```
public static ContextMenuItemParentElement CreateElement_ContextMenu_Item_Parent(this VisualElement parent,  
ContextMenuItemParentData data)
```

## Parameters

| Type                                 | Name   | Description  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |
| ContextMenuItemParentData            | data   | This is the data of the Context Menu Parent.       |

## Returns

| Type                         | Description   |
|------------------------------|---|
| ContextMenuItemParentElement | The return is the ContextMenuItemParentElement that is created. |

## CreateElement\_ContextMenuItem(VisualElement, ContextMenuItemData)

This will create a ContextMenuItemElement and add to the element that invokes this.

## Declaration

```
public static ContextMenuItemElement CreateElement_ContextMenuItem(this VisualElement parent,  
ContextMenuItemData data)
```

## Parameters

| Type                                 | Name   | Description  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |
| ContextMenuItemData                  | data   | This is the data of the Context Menu.              |

## Returns

| Type                   | Description   |
|------------------------|---|
| ContextMenuItemElement | The return is the ContextMenuItemElement that is created. |

## CreateElement\_DialogBox(VisualElement, String, String, Boolean)

This will create a DialogBoxElement and add to the element that invokes this.

#### Declaration

```
public static DialogBoxElement CreateElement_DialogBox(this VisualElement parent, string UID, string  
titleText, bool isDisplay = false)
```

#### Parameters

| TYPE                                 | NAME      | DESCRIPTION  |
|--------------------------------------|-----------|--|
| UnityEngine.UIElements.VisualElement | parent    | This param is the element that invokes the method. |
| System.String                        | UID       | This is the Unique ID of the DialogBox.            |
| System.String                        | titleText | This is the title of the DialogBox.                |
| System.Boolean                       | isDisplay |  |

#### Returns

| TYPE             | DESCRIPTION   |
|------------------|---|
| DialogBoxElement | The return is the DialogBoxElement that is created. |

### CreateElement\_DialogBox(VisualElement, String, String, VisualElement, Boolean)

This will create a DialogBoxElement and add to the element that invokes this.

#### Declaration

```
public static DialogBoxElement CreateElement_DialogBox(this VisualElement parent, string UID, string  
titleText, VisualElement content, bool isDisplay = false)
```

#### Parameters

| TYPE                                 | NAME   | DESCRIPTION  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |
| System.String                        | UID    | This is the Unique ID of the DialogBox.            |

| TYPE                                 | NAME      | DESCRIPTION                         |
|--------------------------------------|-----------|-------------------------------------|
| System.String                        | titleText | This is the title of the DialogBox. |
| UnityEngine.UIElements.VisualElement | content   |                                     |
| System.Boolean                       | isDisplay |                                     |

## Returns

| TYPE             | DESCRIPTION   |
|------------------|---|
| DialogBoxElement | The return is the DialogBoxElement that is created. |

## CreateElement\_Dropdown(VisualElement, String, List<String>, Int32)

This will create a DropdownField and add to the element that invokes this.

## Declaration

```
public static DropdownField CreateElement_Dropdown(this VisualElement parent, string label = null,
List<string> choices = null, int defaultIndex = 0)
```

## Parameters

| TYPE   | NAME         | DESCRIPTION  |
|--|--------------|--|
| UnityEngine.UIElements.VisualElement           | parent       | This param is the element that invokes the method. |
| System.String                                  | label        | This is the label of the DropdownField.            |
| System.Collections.Generic.List<System.String> | choices      | This is the choices of the DropdownField.          |
| System.Int32                                   | defaultIndex | This is the defaultIndex of the DropdownField.     |

## Returns

| TYPE                                | DESCRIPTION                                     |
|-------------------------------------|---|
| UnityEngine.UIElements.DropoutField | The return is the DropoutField that is created. |

## CreateElement\_Foldout(VisualElement, String, Boolean)

This will create a Foldout and add to the element that invokes this.

#### Declaration

```
public static Foldout CreateElement_Foldout(this VisualElement parent, string label = null, bool isOpen = false)
```

#### Parameters

| Type                                 | Name   | Description  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |
| System.String                        | label  | This is the label of the Foldout.                  |
| System.Boolean                       | isOpen | This is the boolean that opens the Foldout.        |

#### Returns

| Type                           | Description                                |
|--------------------------------|--|
| UnityEngine.UIElements.Foldout | The return is the Foldout that is created. |

### CreateElement\_Label(VisualElement, String, Boolean)

This will create a Label and add to the element that invokes this.

#### Declaration

```
public static Label CreateElement_Label(this VisualElement parent, string text, bool enableRichText = true)
```

#### Parameters

| Type                                 | Name           | Description   |
|--------------------------------------|----------------|---|
| UnityEngine.UIElements.VisualElement | parent         | This param is the element that invokes the method.            |
| System.String                        | text           | This is the text of the Label.                                |
| System.Boolean                       | enableRichText | This is the boolean that will enable Rich Text for the Label. |

#### Returns

| TYPE                         | DESCRIPTION                              |
|------------------------------|--|
| UnityEngine.UIElements.Label | The return is the Label that is created. |

## CreateElement\_ListView< TValue >(VisualElement, List< TValue >, Single, SelectionType)

This will create a ListView and add to the element that invokes this.

### Declaration

```
public static ListView CreateElement_ListView< TValue >(this VisualElement parent, List< TValue > items, float itemHeight = -1F, SelectionType type = SelectionType.Single)
```

### Parameters

| TYPE                                      | NAME       | DESCRIPTION  |
|---|------------|--|
| UnityEngine.UIElements.VisualElement      | parent     | This param is the element that invokes the method.                   |
| System.Collections.Generic.List< TValue > | items      | This is the items of the ListView.                                   |
| System.Single                             | itemHeight | This is the height of the item in the list in pixel of the ListView. |
| UnityEngine.UIElements.SelectionType      | type       | This is the type of selection for the ListView.                      |

### Returns

| TYPE                            | DESCRIPTION                                 |
|---------------------------------|---|
| UnityEngine.UIElements.ListView | The return is the ListView that is created. |

### Type Parameters

| NAME   | DESCRIPTION |
|--------|-------------|
| TValue |             |

## CreateElement\_Progress\_Bar(VisualElement, String, Single, Single, Single)

This will create a ProgressBar and add to the element that invokes this.

### Declaration

```
public static ProgressBar CreateElement_Progress_Bar(this VisualElement parent, string title, float lowValue, float highValue, float value)
```

#### Parameters

| Type                                 | Name      | Description  |
|--------------------------------------|-----------|--|
| UnityEngine.UIElements.VisualElement | parent    | This param is the element that invokes the method. |
| System.String                        | title     | This is the title of the ProgressBar.              |
| System.Single                        | lowValue  | This is the lowValue of the ProgressBar.           |
| System.Single                        | highValue | This is the highValue of the ProgressBar.          |
| System.Single                        | value     | This is the value of the ProgressBar.              |

#### Returns

| Type                               | Description                                    |
|------------------------------------|--|
| UnityEngine.UIElements.ProgressBar | The return is the ProgressBar that is created. |

## CreateElement\_Radio(VisualElement, Boolean, String, Boolean)

This will create a RadioButton and add to the element that invokes this.

#### Declaration

```
public static RadioButton CreateElement_Radio(this VisualElement parent, bool value, string label = null, bool isSelected = false)
```

#### Parameters

| Type                                 | Name   | Description  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |
| System.Boolean                       | value  |  |
| System.String                        | label  | This is the label of the RadioButton.              |

| <b>TYPE</b>    | <b>NAME</b> | <b>DESCRIPTION</b>                         |
|----------------|-------------|--|
| System.Boolean | isSelected  | If true the radio button will be selected. |

Returns

| <b>TYPE</b>                        | <b>DESCRIPTION</b>                             |
|------------------------------------|--|
| UnityEngine.UIElements.RadioButton | The return is the RadioButton that is created. |

## CreateElement\_Radio\_Group(VisualElement, String, String[])

This will create a RadioButtonGroup and add to the element that invokes this.

Declaration

```
public static RadioButtonGroup CreateElement_Radio_Group(this VisualElement parent, string label = null,
params string[] choices)
```

Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                                 |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | parent      | This param is the element that invokes the method. |
| System.String                        | label       | This is the label of the RadioButtonGroup.         |
| System.String[]                      | choices     | This is the choices of the RadioButtonGroup.       |

Returns

| <b>TYPE</b>                             | <b>DESCRIPTION</b>                                  |
|---|---|
| UnityEngine.UIElements.RadioButtonGroup | The return is the RadioButtonGroup that is created. |

## CreateElement\_ScrollView(VisualElement, ScrollViewMode, VisualElement[])

This will create a ScrollView and add to the element that invokes this.

Declaration

```
public static ScrollView CreateElement_ScrollView(this VisualElement parent, ScrollViewMode scrollViewMode,
params VisualElement[] children)
```

## Parameters

| Type                                   | Name           | Description  |
|--|----------------|--|
| UnityEngine.UIElements.VisualElement   | parent         | This param is the element that invokes the method. |
| UnityEngine.UIElements.ScrollViewMode  | scrollViewMode | This is the scroll view of the ScrollView.         |
| UnityEngine.UIElements.VisualElement[] | children       | This is params the children of the ScrollView.     |

## Returns

| Type                             | Description                                   |
|----------------------------------|---|
| UnityEngine.UIElementsScrollView | The return is the ScrollView that is created. |

## CreateElement\_Slider(VisualElement, Single, Single, String)

This will create a Slider and add to the element that invokes this.

## Declaration

```
public static Slider CreateElement_Slider(this VisualElement parent, float minAmount, float MaxAmount, string label = null)
```

## Parameters

| Type                                 | Name      | Description  |
|--------------------------------------|-----------|--|
| UnityEngine.UIElements.VisualElement | parent    | This param is the element that invokes the method. |
| System.Single                        | minAmount | This is the minAmount of the Slider.               |
| System.Single                        | MaxAmount | This is the MaxAmount of the Slider.               |
| System.String                        | label     | This is the label of the Slider.                   |

## Returns

| TYPE                          | DESCRIPTION                               |
|-------------------------------|---|
| UnityEngine.UIElements.Slider | The return is the Slider that is created. |

## CreateElement\_Slider\_Int(VisualElement, Int32, Int32, String)

This will create a SliderInt and add to the element that invokes this.

### Declaration

```
public static SliderInt CreateElement_Slider_Int(this VisualElement parent, int minAmount, int MaxAmount,
string label = null)
```

### Parameters

| TYPE                                 | NAME      | DESCRIPTION  |
|--------------------------------------|-----------|--|
| UnityEngine.UIElements.VisualElement | parent    | This param is the element that invokes the method. |
| System.Int32                         | minAmount | This is the minAmount of the SliderInt.            |
| System.Int32                         | MaxAmount | This is the MaxAmount of the SliderInt.            |
| System.String                        | label     | This is the label of the SliderInt.                |

### Returns

| TYPE                             | DESCRIPTION                                  |
|----------------------------------|--|
| UnityEngine.UIElements.SliderInt | The return is the SliderInt that is created. |

## CreateElement\_Slider\_Min\_Max(VisualElement, String, Single, Single, Single, Single)

This will create a MinMaxSlider and add to the element that invokes this.

### Declaration

```
public static MinMaxSlider CreateElement_Slider_Min_Max(this VisualElement parent, string label = null, float
minValue = 0F, float maxValue = 10F, float minLimit = -3.40282347E+38F, float maxLimit = 3.40282347E+38F)
```

### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                                 |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | parent      | This param is the element that invokes the method. |
| System.String                        | label       | This is the label of the MinMaxSlider.             |
| System.Single                        | minValue    | This is the minValue of the MinMaxSlider.          |
| System.Single                        | maxValue    | This is the maxValue of the MinMaxSlider.          |
| System.Single                        | minLimit    | This is the minLimit of the MinMaxSlider.          |
| System.Single                        | maxLimit    | This is the maxLimit of the MinMaxSlider.          |

#### Returns

| <b>TYPE</b>                         | <b>DESCRIPTION</b>                              |
|-------------------------------------|---|
| UnityEngine.UIElements.MinMaxSlider | The return is the MinMaxSlider that is created. |

### CreateElement\_Sprite(VisualElement, Sprite)

This will create a SpriteElement and add to the element that invokes this.

#### Declaration

```
public static SpriteElement CreateElement_Sprite(this VisualElement parent, Sprite sprite)
```

#### Parameters

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                                 |
|--------------------------------------|-------------|--|
| UnityEngine.UIElements.VisualElement | parent      | This param is the element that invokes the method. |
| UnityEngine.Sprite                   | sprite      | This is the sprite of the element.                 |

#### Returns

| Type          | Description                                      |
|---------------|--|
| SpriteElement | The return is the SpriteElement that is created. |

## CreateElement\_TextField(VisualElement, String, Int32, Boolean, Boolean, Char)

This will create a TextField and add to the element that invokes this.

### Declaration

```
public static TextField CreateElement_TextField(this VisualElement parent, string label = null, int maxLength = 0, bool multiline = false, bool isPasswordField = false, char maskChar = '*')
```

### Parameters

| Type                                 | Name            | Description   |
|--------------------------------------|-----------------|---|
| UnityEngine.UIElements.VisualElement | parent          | This param is the element that invokes the method.        |
| System.String                        | label           | This is the label of the TextField.                       |
| System.Int32                         | maxLength       | This is the maxLength of the TextField.                   |
| System.Boolean                       | multiline       | This will determine of the TextField.                     |
| System.Boolean                       | isPasswordField | This is the boolean for passwords.                        |
| System.Char                          | maskChar        | This is the character that will be used for the password. |

### Returns

| Type                             | Description                                  |
|----------------------------------|--|
| UnityEngine.UIElements.TextField | The return is the TextField that is created. |

## CreateElement\_Toggle(VisualElement, String)

This will create a Toggle and add to the element that invokes this.

### Declaration

```
public static Toggle CreateElement_Toggle(this VisualElement parent, string label)
```

## Parameters

| Type                                 | Name   | Description  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |
| System.String                        | label  | This is the label of the Toggle.                   |

## Returns

| Type                          | Description                               |
|-------------------------------|---|
| UnityEngine.UIElements.Toggle | The return is the Toggle that is created. |

## CreateElement\_Tooltip(VisualElement, Font, Int32)

This will create a ToolTipElement and add to the element that invokes this.

## Declaration

```
public static ToolTipElement CreateElement_Tooltip(this VisualElement parent, Font font = null, int textSize = 0)
```

## Parameters

| Type                                 | Name     | Description  |
|--------------------------------------|----------|--|
| UnityEngine.UIElements.VisualElement | parent   | This param is the element that invokes the method. |
| UnityEngine.Font                     | font     | This is the font of the tooltip.                   |
| System.Int32                         | textSize | This is the size of the tooltip.                   |

## Returns

| Type           | Description                                       |
|----------------|---|
| ToolTipElement | The return is the ToolTipElement that is created. |

## CreateElement\_VisualElement(VisualElement)

---

This will create a VisualElement and add to the element that invokes this.

### Declaration

```
public static VisualElement CreateElement_VisualElement(this VisualElement parent)
```

### Parameters

| Type                                 | Name   | Description  |
|--------------------------------------|--------|--|
| UnityEngine.UIElements.VisualElement | parent | This param is the element that invokes the method. |

### Returns

| Type                                 | Description                                      |
|--------------------------------------|--|
| UnityEngine.UIElements.VisualElement | The return is the VisualElement that is created. |

# Class RichText

This static class will simplify setting up rich text for "Text Mesh Pro" or "UI ToolKit"

## Inheritance

System.Object

RichText

Namespace: [UtilitySystem.Script](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public static class RichText
```

## Properties

### NewLine

This will take an object and create a new line string.

#### Declaration

```
public static string NewLine { get; }
```

### Property Value

| TYPE          | DESCRIPTION |
|---------------|-------------|
| System.String |             |

### Tab

This will take an object and create a tab gap string.

#### Declaration

```
public static string Tab { get; }
```

### Property Value

| TYPE          | DESCRIPTION |
|---------------|-------------|
| System.String |             |

## Methods

### **BoldText(Object)**

This will take an object and create a string that is bold.

#### Declaration

```
public static string BoldText(this object arg)
```

#### Parameters

| TYPE          | NAME | DESCRIPTION  |
|---------------|------|--|
| System.Object | arg  | This is the object that will be passed in and will be added into a string. |

#### Returns

| TYPE          | DESCRIPTION                                    |
|---------------|--|
| System.String | This will return a string that is set to bold. |

### BooleanText(Boolean)

This will take an object and create a string that is either colored green for true or red for false.

#### Declaration

```
public static string BooleanText(this bool arg)
```

#### Parameters

| TYPE           | NAME | DESCRIPTION  |
|----------------|------|--|
| System.Boolean | arg  | This is the object that will be passed in and will be added into a string. |

#### Returns

| TYPE          | DESCRIPTION   |
|---------------|---|
| System.String | This will return a string that is either colored green for true or red for false. |

### BooleanText(Object, Boolean)

This will take an object and create a string that is either colored green for true or red for false.

#### Declaration

```
public static string BooleanText(this object arg, bool check)
```

## Parameters

| Type           | Name  | Description  |
|----------------|-------|--|
| System.Object  | arg   | This is the object that will be passed in and will be added into a string. |
| System.Boolean | check | This is the boolean that will determine if true or false.                  |

## Returns

| Type          | Description   |
|---------------|---|
| System.String | This will return a string that is either colored green for true or red for false. |

## BooleanText(Object, Boolean, Color)

This will take an object and create a string that is either colored green for true or red for false.

## Declaration

```
public static string BooleanText(this object arg, bool check, Color color)
```

## Parameters

| Type              | Name  | Description  |
|-------------------|-------|--|
| System.Object     | arg   | This is the object that will be passed in and will be added into a string. |
| System.Boolean    | check | This is the boolean that will determine if true or false.                  |
| UnityEngine.Color | color | if true set the text to the color else return without color.               |

## Returns

| Type          | Description   |
|---------------|---|
| System.String | This will return a string that is either colored green for true or red for false. |

## ColoredText(Object, String)

This will take an object and create a string that will color the text.

## Declaration

```
public static string ColoredText(this object arg, string color)
```

## Parameters

| Type          | Name  | Description   |
|---------------|-------|---|
| System.Object | arg   | This is the object that will be passed in and will be added into a string. This is the object that will be passed in and will be added into a string. |
| System.String | color | this is the color that will be used to color the text.  |

## Returns

| Type          | Description   |
|---------------|---|
| System.String | This will return a string that is colored from the color param. |

## ColoredText(Object, Color)

This will take an object and create a string that will color the text.

## Declaration

```
public static string ColoredText(this object arg, Color color)
```

## Parameters

| Type              | Name  | Description  |
|-------------------|-------|--|
| System.Object     | arg   | This is the object that will be passed in and will be added into a string. |
| UnityEngine.Color | color | this is the color that will be used to color the text.                     |

## Returns

| Type          | Description   |
|---------------|---|
| System.String | This will return a string that is colored from the color param. |

## CurrencyText(Int32)

This will take a 3 int value that represents gold,silver,copper or \$ amount.

#### Declaration

```
public static string CurrencyText(this int arg)
```

#### Parameters

| TYPE         | NAME | DESCRIPTION                             |
|--------------|------|---|
| System.Int32 | arg  | The 3 values to be converted to string. |

#### Returns

| TYPE          | DESCRIPTION                         |
|---------------|-------------------------------------|
| System.String | This will return a string currency. |

### **ItalicText(Object)**

This will take an object and create a string that is italic.

#### Declaration

```
public static string ItalicText(this object arg)
```

#### Parameters

| TYPE          | NAME | DESCRIPTION  |
|---------------|------|--|
| System.Object | arg  | This is the object that will be passed in and will be added into a string. |

#### Returns

| TYPE          | DESCRIPTION   |
|---------------|---|
| System.String | This will return a string that is given an italic lean. |

### **LowerCaseText(Object)**

This will return a string to an lowercase

#### Declaration

```
public static string LowerCaseText(this object arg)
```

## Parameters

| Type          | Name | Description  |
|---------------|------|--|
| System.Object | arg  | This is the object that will be passed in and will be added into a string. |

## Returns

| Type          | Description                                      |
|---------------|--|
| System.String | This will return a string that is all lowercase. |

## PositionText(Object, Single)

This will take an object and create a string that is a set position for the text on the line.

## Declaration

```
public static string PositionText(this object arg, float pos)
```

## Parameters

| Type          | Name | Description  |
|---------------|------|--|
| System.Object | arg  | This is the object that will be passed in and will be added into a string.     |
| System.Single | pos  | This is a float that will be used to set the position of the text in the line. |

## Returns

| Type          | Description  |
|---------------|--|
| System.String | This will return a string that is set at a position. |

## SetText(List<Func<String>>)

This will take an object and create a string that is either colored green for true or red for false.

## Declaration

```
public static string SetText(List<Func<string>> messages)
```

## Parameters

| <b>TYPE</b>   | <b>NAME</b> | <b>DESCRIPTION</b>  |
|---|-------------|---|
| System.Collections.Generic.List<System.Func<System.String>> | messages    | this is a list of Func strings that will be run and appended to a StringBuilder |

## Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b>                                |
|---------------|---|
| System.String | This will return a string from the StringBuilder. |

## SizeText(Object, Int32)

This will take an object and create a string that is the size for the text.

## Declaration

```
public static string SizeText(this object arg, int size)
```

## Parameters

| <b>TYPE</b>   | <b>NAME</b> | <b>DESCRIPTION</b>   |
|---------------|-------------|--|
| System.Object | arg         | This is the object that will be passed in and will be added into a string. |
| System.Int32  | size        | This is an integer for the size of the text.                               |

## Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b>  |
|---------------|---|
| System.String | This will return a string that has it's size changed to the size param. |

## StrikeText(Object)

This will take an object and create a string that is strikethrough.

## Declaration

```
public static string StrikeText(this object arg)
```

## Parameters

| <b>TYPE</b>   | <b>NAME</b> | <b>DESCRIPTION</b>   |
|---------------|-------------|--|
| System.Object | arg         | This is the object that will be passed in and will be added into a string. |

Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b>                                   |
|---------------|--|
| System.String | This will return a string that has a line though it. |

## SubText(Object)

---

This will take an object and create a string that is lowered.

Declaration

```
public static string SubText(this object arg)
```

Parameters

| <b>TYPE</b>   | <b>NAME</b> | <b>DESCRIPTION</b>   |
|---------------|-------------|--|
| System.Object | arg         | This is the object that will be passed in and will be added into a string. |

Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b>                                       |
|---------------|--|
| System.String | This will return a string that is lower then other text. |

## SupText(Object)

---

This will take an object and create a string that is raised.

Declaration

```
public static string SupText(this object arg)
```

Parameters

| <b>TYPE</b>   | <b>NAME</b> | <b>DESCRIPTION</b>   |
|---------------|-------------|--|
| System.Object | arg         | This is the object that will be passed in and will be added into a string. |

## Returns

| Type          | Description  |
|---------------|--|
| System.String | This will return a string that is higher then other text . |

## Text(Object)

---

This will take an object and create a string.

### Declaration

```
public static string Text(this object arg)
```

### Parameters

| Type          | Name | Description  |
|---------------|------|--|
| System.Object | arg  | This is the object that will be passed in and will be added into a string. |

## Returns

| Type          | Description  |
|---------------|--|
| System.String | This will return a string that is a generic text string. |

## UnderLineText(Object)

---

This will take an object and create a string that is underline.

### Declaration

```
public static string UnderLineText(this object arg)
```

### Parameters

| Type          | Name | Description  |
|---------------|------|--|
| System.Object | arg  | This is the object that will be passed in and will be added into a string. |

## Returns

| TYPE          | DESCRIPTION                                   |
|---------------|---|
| System.String | This will return a string that is underlined. |

## UpperCaseText(Object)

---

This will return a string to an uppercase

### Declaration

```
public static string UpperCaseText(this object arg)
```

### Parameters

| TYPE          | NAME | DESCRIPTION  |
|---------------|------|--|
| System.Object | arg  | This is the object that will be passed in and will be added into a string. |

### Returns

| TYPE          | DESCRIPTION                                      |
|---------------|--|
| System.String | This will return a string that is all uppercase. |

## WeightText(Single, Boolean)

---

This will take a float value that represents weight. And will convert to either metric or imperial. based on weight amount.

### Declaration

```
public static string WeightText(this float arg, bool IsMetric = true)
```

### Parameters

| TYPE           | NAME     | DESCRIPTION                                 |
|----------------|----------|---|
| System.Single  | arg      | The weight value to be converted to string. |
| System.Boolean | IsMetric |   |

### Returns

| <b>TYPE</b>   | <b>DESCRIPTION</b>                |
|---------------|-----------------------------------|
| System.String | This will return a string weight. |

# Namespace UtilitySystem.Tool

## Classes

### [Timer](#)

---

This class is a Timer that can be used to activate an action after an amount of time.

# Class Timer

This class is a Timer that can be used to activate an action after an amount of time.

## Inheritance

System.Object

Timer

Namespace: [UtilitySystem.Tool](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
[Serializable]
public class Timer
```

## Constructors

**Timer(Action, Single, Boolean)**

This is a constructor that will setup the timer.

### Declaration

```
public Timer(Action action, float timer, bool isRunning = true)
```

### Parameters

| Type           | Name      | Description   |
|----------------|-----------|---|
| System.Action  | action    | the Action that will be run when the timer finishes.                    |
| System.Single  | timer     | the amount of time it will take to finish.                              |
| System.Boolean | isRunning | This param if true will set the timer to start or stop. (default true). |

## Properties

**CurrentTime**

This property will return the currnet time round up.

### Declaration

```
public int CurrentTime { get; }
```

## Property Value

| Type         | Description |
|--------------|-------------|
| System.Int32 |             |

## Running

---

This will return true if the timer is running.

### Declaration

```
public bool Running { get; }
```

## Property Value

| Type           | Description |
|----------------|-------------|
| System.Boolean |             |

## Methods

### ResetTimer()

---

This will reset the timer.

### Declaration

```
public void ResetTimer()
```

### StartTimer()

---

This will start the timer.

### Declaration

```
public void StartTimer()
```

### StopTimer()

---

This will stop the timer.

### Declaration

```
public void StopTimer()
```

### UpdateTimer()

---

This will update the timer.

## Declaration

```
public void UpdateTimer()
```

## Extension Methods

[RichText.UpperCaseText\(Object\)](#)  
[RichText.LowerCaseText\(Object\)](#)  
[RichText.BoldText\(Object\)](#)  
[RichText.ItalicText\(Object\)](#)  
[RichText.Text\(Object\)](#)  
[RichText.UnderLineText\(Object\)](#)  
[RichText.StrikeText\(Object\)](#)  
[RichText.SupText\(Object\)](#)  
[RichText.SubText\(Object\)](#)  
[RichText.PositionText\(Object, Single\)](#)  
[RichText.SizeType\(Object, Int32\)](#)  
[RichText.ColoredText\(Object, String\)](#)  
[RichText.ColoredText\(Object, Color\)](#)  
[RichText.BooleanText\(Object, Boolean\)](#)  
[RichText.BooleanText\(Object, Boolean, Color\)](#)

# Namespace UtilitySystem.UI

## Classes

### [ContextMenuElement](#)

---

This class is the context menu for the runtime.

### [ContextMenuElement.UxmlFactory](#)

---

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

### [ContextMenuItemButtonElement](#)

---

This class is the context menu button for the runtime.

### [ContextMenuItemButtonElement.UxmlFactory](#)

---

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

### [ContextMenuItemElement](#)

---

This class is the context menu item for the runtime.

### [ContextMenuItemElement.UxmlFactory](#)

---

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

### [ContextMenuItemParentElement](#)

---

This class is the context sub-menu for the runtime.

### [ContextMenuItemParentElement.UxmlFactory](#)

---

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

### [DialogBoxElement](#)

---

This class is the dialog box that can be move around on the screen.

### [DialogBoxElement.UxmlFactory](#)

---

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

### [PopupBaseElement](#)

---

This abstract class that is the base of the popup elements.

## [SpriteElement](#)

---

This Class is an VisualElement that is created for the Inventory Editor.

## [SpriteElement.UxmlFactory](#)

---

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

## [ToolTipElement](#)

---

## [ToolTipElement.UxmlFactory](#)

---

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

# Class ContextMenuElement

This class is the context menu for the runtime.

## Inheritance

```
System.Object  
UnityEngine.UIElements.CallbackEventHandler  
UnityEngine.UIElements.Focusable  
UnityEngine.UIElements.VisualElement  
ContextMenuElement
```

Namespace: [Utility System . UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class ContextMenuElement : VisualElement, IEventHandler, ITransform, ITransitionAnimations,  
IExperimentalFeatures, IVisualElementScheduler, IResolvedStyle
```

## Constructors

### ContextMenuElement()

This is a constructor and will create the context menu.

#### Declaration

```
public ContextMenuElement()
```

### ContextMenuElement(ContextMenuItemData[])

This is a constructor and will create the context menu.

#### Declaration

```
public ContextMenuElement(params ContextMenuItemData[] data)
```

#### Parameters

| TYPE                                  | NAME | DESCRIPTION                                 |
|---------------------------------------|------|---|
| <a href="#">ContextMenuItemData[]</a> | data | these is the list of Context Menu Item Data |

### ContextMenuElement(List<ContextMenuItemData>)

This is a constructor and will create the context menu.

#### Declaration

```
public ContextMenuElement(List<ContextMenuItemData> data)
```

## Parameters

| Type   | Name | Description                                |
|--|------|--|
| System.Collections.Generic.List<ContextMenuItemData> | data | this is the list of Context Menu Item Data |

## Methods

### GenerateElements(List<ContextMenuItemData>)

This will generate "ContextMenuItem" elements for this menu

#### Declaration

```
public void GenerateElements(List<ContextMenuItemData> data)
```

#### Parameters

| Type   | Name | Description   |
|--|------|---|
| System.Collections.Generic.List<ContextMenuItemData> | data | This is a list of ContextMenuItemData that will be used to create the menu items. |

### LineBreaker()

This will create a line breaker on the context menu.

#### Declaration

```
public VisualElement LineBreaker()
```

#### Returns

| Type                                 | Description                                |
|--------------------------------------|--|
| UnityEngine.UIElements.VisualElement | The return is the element that is created. |

### Style()

This will set the style of the context menu.

#### Declaration

```
public virtual void Style()
```

## Extension Methods

```
VisualElementStyleExtension.BackGround_Color(VisualElement, Color)
VisualElementStyleExtension.Background_Image(VisualElement, Background)
VisualElementStyleExtension.Background_Image(VisualElement, Texture2D)
VisualElementStyleExtension.Background_Image(VisualElement, Sprite)
VisualElementStyleExtension.Background_Image(VisualElement, StyleKeyword)
VisualElementStyleExtension.Color(VisualElement, Color)
VisualElementStyleExtension.Color(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Color_Top(VisualElement, Color)
VisualElementStyleExtension.Border_Color_Top(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Color_Right(VisualElement, Color)
VisualElementStyleExtension.Border_Color_Right(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Color_Bottom(VisualElement, Color)
VisualElementStyleExtension.Border_Color_Bottom(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Color_Left(VisualElement, Color)
VisualElementStyleExtension.Border_Color_Left(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Color(VisualElement, Color[])
VisualElementStyleExtension.Border_Color(VisualElement, StyleKeyword[])
VisualElementStyleExtension.GetRadius_Top(VisualElement)
VisualElementStyleExtension.GetRadius_Bottom(VisualElement)
VisualElementStyleExtension.GetRadius_Size(VisualElement)
VisualElementStyleExtension.Border_Radius_Top_Left(VisualElement, Single, LengthUnit)
VisualElementStyleExtension.Border_Radius_Top_Left(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Radius_Top_Right(VisualElement, Single, LengthUnit)
VisualElementStyleExtension.Border_Radius_Top_Right(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Radius_Bottom_Left(VisualElement, Single, LengthUnit)
VisualElementStyleExtension.Border_Radius_Bottom_Left(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Radius_Bottom_Right(VisualElement, Single, LengthUnit)
VisualElementStyleExtension.Border_Radius_Bottom_Right(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Radius(VisualElement, Single[])
VisualElementStyleExtension.Border_Radius(VisualElement, LengthUnit, Single[])
VisualElementStyleExtension.Border_Radius(VisualElement, LengthUnit[], Single[])
VisualElementStyleExtension.Border_Radius(VisualElement, StyleKeyword[])
VisualElementStyleExtension.GetBorder_Width(VisualElement)
VisualElementStyleExtension.GetBorder_Height(VisualElement)
VisualElementStyleExtension.GetBorder_Size(VisualElement)
VisualElementStyleExtension.Border_Width_Top(VisualElement, Single)
VisualElementStyleExtension.Border_Width_Top(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Width_Right(VisualElement, Single)
VisualElementStyleExtension.Border_Width_Right(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Width_Bottom(VisualElement, Single)
VisualElementStyleExtension.Border_Width_Bottom(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Width_Left(VisualElement, Single)
VisualElementStyleExtension.Border_Width_Left(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Width(VisualElement, Single[])
VisualElementStyleExtension.Border_Width(VisualElement, StyleKeyword[])
VisualElementStyleExtension.Border(VisualElement, Single[])
VisualElementStyleExtension.Border(VisualElement, Color, Single[])
VisualElementStyleExtension.Border(VisualElement, Color[], Single[])
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color, Single[])
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color[], Single[])
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color, Single[])
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color[], Single[])
```

VisualElementStyleExtension.Border(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetMargin\_Width(VisualElement)  
VisualElementStyleExtension.GetMargin\_Height(VisualElement)  
VisualElementStyleExtension.GetMargin\_Size(VisualElement)  
VisualElementStyleExtension.Margin\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin(VisualElement, Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Margin(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetPadding\_Width(VisualElement)  
VisualElementStyleExtension.GetPadding\_Height(VisualElement)  
VisualElementStyleExtension.GetPadding\_Size(VisualElement)  
VisualElementStyleExtension.Padding\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding(VisualElement, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Padding(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Size\_Width(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size(VisualElement, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Min(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Min(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, Single, LengthUnit)

VisualElementStyleExtension.Size\_Height\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Max(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Max(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.SetDisplay(VisualElement, DisplayStyle)  
VisualElementStyleExtension.GetDisplay(VisualElement)  
VisualElementStyleExtension.Display\_ToggleVisibility(VisualElement)  
VisualElementStyleExtension.Display\_Hide(VisualElement)  
VisualElementStyleExtension.Display\_Show(VisualElement)  
VisualElementStyleExtension.Display\_IsInvisible(VisualElement)  
VisualElementStyleExtension.Display\_IsVisible(VisualElement)  
VisualElementStyleExtension.StylePosition(VisualElement, Position)  
VisualElementStyleExtension.StylePosition(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Toggle(VisualElement)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Align\_Content(VisualElement, Align)  
VisualElementStyleExtension.Align\_Content(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Item(VisualElement, Align)  
VisualElementStyleExtension.Align\_Item(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Self(VisualElement, Align)  
VisualElementStyleExtension.Align\_Self(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, Wrap)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, FlexDirection)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, StyleKeyword)  
VisualElementStyleExtension.White\_Space(VisualElement,WhiteSpace)

VisualElementStyleExtension.White\_Space(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font\_Size(VisualElement, Single)  
VisualElementStyleExtension.Font\_Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font(VisualElement, Font)  
VisualElementStyleExtension.Font(VisualElement, FontAsset)  
VisualElementStyleExtension.Text\_Position(VisualElement, TextAnchor)  
VisualElementStyleExtension.Overflow(VisualElement, Overflow)  
VisualElementStyleExtension.Overflow(VisualElement, StyleKeyword)  
VisualElementExtension.ViewDataKey(VisualElement, String)  
VisualElementExtension.UsageHints(VisualElement, UsageHints)  
VisualElementExtension.ToolTip(VisualElement, String)  
VisualElementExtension.GetLayout(VisualElement)  
VisualElementExtension.GetLayoutSize(VisualElement)  
VisualElementExtension.GetLayoutWidth(VisualElement)  
VisualElementExtension.GetLayoutHeight(VisualElement)  
VisualElementExtension.GetLayoutPosition(VisualElement)  
VisualElementExtension.GetLayoutPositionX(VisualElement)  
VisualElementExtension.GetLayoutPositionY(VisualElement)  
VisualElementExtension.GetWorldBound(VisualElement)  
VisualElementExtension.GetWorldBoundSize(VisualElement)  
VisualElementExtension.GetWorldBoundPosition(VisualElement)  
VisualElementExtension.GetLocalBound(VisualElement)  
VisualElementExtension.GetLocalBoundSize(VisualElement)  
VisualElementExtension.GetLocalBoundPosition(VisualElement)  
VisualElementExtension.Pick(VisualElement, PickingMode)  
VisualElementExtension.Pick\_Toggle(VisualElement)  
VisualElementExtension.Pick\_OFF(VisualElement)  
VisualElementExtension.Pick\_ON(VisualElement)  
VisualElementExtension.Focus(VisualElement, Boolean)  
VisualElementExtension.Focus\_Toggle(VisualElement)  
VisualElementExtension.Focus\_OFF(VisualElement)  
VisualElementExtension.Focus\_ON(VisualElement)  
VisualElementExtension.SetVisible(VisualElement, Boolean)  
VisualElementExtension.GetVisible(VisualElement)  
VisualElementExtension.Visible\_ToggleVisibility(VisualElement)  
VisualElementExtension.Visible\_Hide(VisualElement)  
VisualElementExtension.Visible\_Show(VisualElement)  
VisualElementExtension.Visible\_IsInvisible(VisualElement)  
VisualElementExtension.Visible\_IsVisible(VisualElement)  
VisualElementExtension.ResetRotation(VisualElement)  
VisualElementExtension.SetTransform\_Rotate(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Rotate(VisualElement)  
VisualElementExtension.Rotate\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Counter\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.RotateY\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.SetTransform\_Position(VisualElement, Vector2)  
VisualElementExtension.GetTransform\_Position(VisualElement)  
VisualElementExtension.SetTransform\_Scale(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Scale(VisualElement)

VisualElementExtension.Child(VisualElement, VisualElement, Boolean)  
VisualElementExtension.StyleSheet(VisualElement, StyleSheet, Boolean)  
VisualElementExtension.Name(VisualElement, String)  
VisualElementExtension.GetElementSize(VisualElement)  
VisualElementExtension.ScreenBound(VisualElement)  
VisualElementExtension.MouseGrab(VisualElement, Vector2)  
VisualElementExtension.MouseRelease(VisualElement)  
VisualElementExtension.UpdatePosition(VisualElement, Vector2)  
RichText.UpperCaseText(Object)  
RichTextLowerCaseText(Object)  
RichText.BoldText(Object)  
RichText.ItalicText(Object)  
RichText.Text(Object)  
RichText.UnderLineText(Object)  
RichText.StrikeText(Object)  
RichText.SupText(Object)  
RichText.SubText(Object)  
RichText.PositionText(Object, Single)  
RichText.SizeTypeText(Object, Int32)  
RichText.ColoredText(Object, String)  
RichText.ColoredText(Object, Color)  
RichText.BooleanText(Object, Boolean)  
RichText.BooleanText(Object, Boolean, Color)  
CreateElements.CreateElement\_Label(VisualElement, String, Boolean)  
CreateElements.CreateElement\_VisualElement(VisualElement)  
CreateElements.CreateElement\_ScrollView(VisualElement, ScrollViewMode, VisualElement[])  
CreateElements.CreateElement\_ListView< TValue >(VisualElement, List< TValue >, Single, SelectionType)  
CreateElements.CreateElement\_Button(VisualElement, Action)  
CreateElements.CreateElement\_Toggle(VisualElement, String)  
CreateElements.CreateElement\_TextField(VisualElement, String, Int32, Boolean, Boolean, Char)  
CreateElements.CreateElement\_Foldout(VisualElement, String, Boolean)  
CreateElements.CreateElement\_Slider(VisualElement, Single, Single, String)  
CreateElements.CreateElement\_Slider\_Int(VisualElement, Int32, Int32, String)  
CreateElements.CreateElement\_Slider\_Min\_Max(VisualElement, String, Single, Single, Single, Single)  
CreateElements.CreateElement\_Progress\_Bar(VisualElement, String, Single, Single, Single)  
CreateElements.CreateElement\_Dropdown(VisualElement, String, List< String >, Int32)  
CreateElements.CreateElement\_Radio(VisualElement, Boolean, String, Boolean)  
CreateElements.CreateElement\_Radio\_Group(VisualElement, String, String[])  
CreateElements.CreateElement\_ContextMenuItem(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu(VisualElement, ContextMenuItemData[])  
CreateElements.CreateElement\_ContextMenu(VisualElement, List< ContextMenuItemData >)  
CreateElements.CreateElement\_ContextMenu\_Item\_Parent(VisualElement, ContextMenuItemParentData)  
CreateElements.CreateElement\_ContextMenu\_Item(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu\_Item\_Button(VisualElement, ContextMenuItemButtonData)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, Boolean)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, VisualElement, Boolean)  
CreateElements.CreateElement\_Tooltip(VisualElement, Font, Int32)  
CreateElements.CreateElement\_Sprite(VisualElement, Sprite)  
CreateElements.CreateElement\_Slot(VisualElement, Int32)  
CreateElements.CreateElement\_Inventory(VisualElement, Int32, Int32, Inventory)  
CreateElements.CreateElement\_Icon(VisualElement, InventoryItem)  
CreateElements.CreateElement\_SlotPopup(VisualElement, Vector2, Int32, InventoryElement, Boolean)  
CreateElements.CreateElement\_TransferPopup(VisualElement, Vector2, TransferItem, InventoryElement, InventoryElement)

# Class ContextMenuItemElement.UxmlFactory

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

## Inheritance

System.Object

UnityEngine.UIElements.UxmlFactory<[ContextMenuItemElement](#), UnityEngine.UIElements.VisualElement.UxmlTraits>

ContextMenuElement.UxmlFactory

Namespace: [Utility System.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class UxmlFactory : UxmlFactory<ContextMenuItemElement, VisualElement.UxmlTraits>, IUxmlFactory
```

## Extension Methods

[RichText.UpperCaseText\(Object\)](#)

[RichText.LowerCaseText\(Object\)](#)

[RichText.BoldText\(Object\)](#)

[RichText.ItalicText\(Object\)](#)

[RichText.Text\(Object\)](#)

[RichText.UnderLineText\(Object\)](#)

[RichText.StrikeText\(Object\)](#)

[RichText.SupText\(Object\)](#)

[RichText.SubText\(Object\)](#)

[RichText.PositionText\(Object, Single\)](#)

[RichText.SizeType\(Object, Int32\)](#)

[RichText.ColoredText\(Object, String\)](#)

[RichText.ColoredText\(Object, Color\)](#)

[RichText.BooleanText\(Object, Boolean\)](#)

[RichText.BooleanText\(Object, Boolean, Color\)](#)

# Class ContextMenuItemButtonElement

This class is the context menu button for the runtime.

## Inheritance

```
System.Object  
UnityEngine.UIElements.CallbackEventHandler  
UnityEngine.UIElements.Focusable  
UnityEngine.UIElements.VisualElement  
ContextMenuItemElement  
ContextMenuItemButtonElement
```

Namespace: [UtilitySystem.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class ContextMenuItemButtonElement : ContextMenuItemElement, IEventHandler, ITransform,  
ITransitionAnimations, IExperimentalFeatures, IVisualElementScheduler, IResolvedStyle
```

## Constructors

### ContextMenuItemButtonElement()

This is a constructor and will create the context menu item.

#### Declaration

```
public ContextMenuItemButtonElement()
```

### ContextMenuItemButtonElement(ContextMenuItemButtonData)

This is a constructor and will create the context menu item.

#### Declaration

```
public ContextMenuItemButtonElement(ContextMenuItemButtonData data)
```

#### Parameters

| TYPE                                      | NAME | DESCRIPTION                                |
|---|------|--|
| <a href="#">ContextMenuItemButtonData</a> | data | this is the list of Context Menu Item Data |

## Fields

### action

This is the action will be invoke when the item is clicked.

## Declaration

```
public Action action
```

## Field Value

| TYPE          | DESCRIPTION |
|---------------|-------------|
| System.Action |             |

## Methods

### GenerateElements(ContextMenuItemButtonData)

This will generate "ContextMenuItem" elements for this menu

## Declaration

```
public void GenerateElements(ContextMenuItemButtonData data)
```

## Parameters

| TYPE                      | NAME | DESCRIPTION   |
|---------------------------|------|---|
| ContextMenuItemButtonData | data | This is a ContextMenuItemData that will be used to create the menu items. |

### Interaction()

This will setup the Interaction when the mouse enters and leave the hover of this element.

## Declaration

```
public override void Interaction()
```

## Overrides

[ContextMenuItemElement.Interaction\(\)](#)

### Style()

This will set the style of the context menu.

## Declaration

```
public override void Style()
```

## Overrides

ContextMenuItemElement.Style()

## Extension Methods

VisualElementStyleExtension.BackGround\_Color(VisualElement, Color)  
VisualElementStyleExtension.Background\_Image(VisualElement, Background)  
VisualElementStyleExtension.Background\_Image(VisualElement, Texture2D)  
VisualElementStyleExtension.Background\_Image(VisualElement, Sprite)  
VisualElementStyleExtension.Background\_Image(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Color(VisualElement, Color)  
VisualElementStyleExtension.Color(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Top(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Right(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Bottom(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Left(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color(VisualElement, Color[])  
VisualElementStyleExtension.Border\_Color(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetRadius\_Top(VisualElement)  
VisualElementStyleExtension.GetRadius\_Bottom(VisualElement)  
VisualElementStyleExtension.GetRadius\_Size(VisualElement)  
VisualElementStyleExtension.Border\_Radius\_Top\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Top\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius\_Top\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Top\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius(VisualElement, Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetBorder\_Width(VisualElement)  
VisualElementStyleExtension.GetBorder\_Height(VisualElement)  
VisualElementStyleExtension.GetBorder\_Size(VisualElement)  
VisualElementStyleExtension.Border\_Width\_Top(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width(VisualElement, Single[])  
VisualElementStyleExtension.Border\_Width(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Border(VisualElement, Single[])  
VisualElementStyleExtension.Border(VisualElement, Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color, Single[])

VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetMargin\_Width(VisualElement)  
VisualElementStyleExtension.GetMargin\_Height(VisualElement)  
VisualElementStyleExtension.GetMargin\_Size(VisualElement)  
VisualElementStyleExtension.Margin\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin(VisualElement, Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Margin(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetPadding\_Width(VisualElement)  
VisualElementStyleExtension.GetPadding\_Height(VisualElement)  
VisualElementStyleExtension.GetPadding\_Size(VisualElement)  
VisualElementStyleExtension.Padding\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding(VisualElement, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Padding(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Size\_Width(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size(VisualElement, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Min(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Min(VisualElement, Boolean, Boolean)

VisualElementStyleExtension.Size\_Width\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Max(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Max(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.SetDisplay(VisualElement, DisplayStyle)  
VisualElementStyleExtension.GetDisplay(VisualElement)  
VisualElementStyleExtension.Display\_ToggleVisibility(VisualElement)  
VisualElementStyleExtension.Display\_Hide(VisualElement)  
VisualElementStyleExtension.Display\_Show(VisualElement)  
VisualElementStyleExtension.Display\_IsInvisible(VisualElement)  
VisualElementStyleExtension.Display\_IsVisible(VisualElement)  
VisualElementStyleExtension.StylePosition(VisualElement, Position)  
VisualElementStyleExtension.StylePosition(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Toggle(VisualElement)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Align\_Content(VisualElement, Align)  
VisualElementStyleExtension.Align\_Content(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Item(VisualElement, Align)  
VisualElementStyleExtension.Align\_Item(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Self(VisualElement, Align)  
VisualElementStyleExtension.Align\_Self(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, Wrap)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, FlexDirection)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, StyleKeyword)

VisualElementStyleExtension.Flex\_Basis(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, StyleKeyword)  
VisualElementStyleExtension.White\_Space(VisualElement,WhiteSpace)  
VisualElementStyleExtension.White\_Space(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font\_Size(VisualElement, Single)  
VisualElementStyleExtension.Font\_Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font(VisualElement, Font)  
VisualElementStyleExtension.Font(VisualElement, FontAsset)  
VisualElementStyleExtension.Text\_Position(VisualElement, TextAnchor)  
VisualElementStyleExtension.Overflow(VisualElement, Overflow)  
VisualElementStyleExtension.Overflow(VisualElement, StyleKeyword)  
VisualElementExtension.ViewDataKey(VisualElement, String)  
VisualElementExtension.UsageHints(VisualElement, UsageHints)  
VisualElementExtension.ToolTip(VisualElement, String)  
VisualElementExtension.GetLayout(VisualElement)  
VisualElementExtension.GetLayoutSize(VisualElement)  
VisualElementExtension.GetLayoutWidth(VisualElement)  
VisualElementExtension.GetLayoutHeight(VisualElement)  
VisualElementExtension.GetLayoutPosition(VisualElement)  
VisualElementExtension.GetLayoutPositionX(VisualElement)  
VisualElementExtension.GetLayoutPositionY(VisualElement)  
VisualElementExtension.GetWorldBound(VisualElement)  
VisualElementExtension.GetWorldBoundSize(VisualElement)  
VisualElementExtension.GetWorldBoundPosition(VisualElement)  
VisualElementExtension.GetLocalBound(VisualElement)  
VisualElementExtension.GetLocalBoundSize(VisualElement)  
VisualElementExtension.GetLocalBoundPosition(VisualElement)  
VisualElementExtension.Pick(VisualElement, PickingMode)  
VisualElementExtension.Pick\_Toggle(VisualElement)  
VisualElementExtension.Pick\_OFF(VisualElement)  
VisualElementExtension.Pick\_ON(VisualElement)  
VisualElementExtension.Focus(VisualElement, Boolean)  
VisualElementExtension.Focus\_Toggle(VisualElement)  
VisualElementExtension.Focus\_OFF(VisualElement)  
VisualElementExtension.Focus\_ON(VisualElement)  
VisualElementExtension.SetVisible(VisualElement, Boolean)  
VisualElementExtension.GetVisible(VisualElement)  
VisualElementExtension.Visible\_ToggleVisibility(VisualElement)  
VisualElementExtension.Visible\_Hide(VisualElement)  
VisualElementExtension.Visible\_Show(VisualElement)  
VisualElementExtension.Visible\_IsInvisible(VisualElement)  
VisualElementExtension.Visible\_IsVisible(VisualElement)  
VisualElementExtension.ResetRotation(VisualElement)  
VisualElementExtension.SetTransform\_Rotate(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Rotate(VisualElement)  
VisualElementExtension.Rotate\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Counter\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.RotateY\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.SetTransform\_Position(VisualElement, Vector2)

VisualElementExtension.GetTransform\_Position(VisualElement)  
VisualElementExtension.SetTransform\_Scale(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Scale(VisualElement)  
VisualElementExtension.Child(VisualElement, VisualElement, Boolean)  
VisualElementExtension.StyleSheet(VisualElement, StyleSheet, Boolean)  
VisualElementExtension.Name(VisualElement, String)  
VisualElementExtension.GetElementSize(VisualElement)  
VisualElementExtension.ScreenBound(VisualElement)  
VisualElementExtension.MouseGrab(VisualElement, Vector2)  
VisualElementExtension.MouseRelease(VisualElement)  
VisualElementExtension.UpdatePosition(VisualElement, Vector2)  
RichText.UpperCaseText(Object)  
RichText.LowerCaseText(Object)  
RichText.BoldText(Object)  
RichText.ItalicText(Object)  
RichText.Text(Object)  
RichText.UnderLineText(Object)  
RichText.StrikeText(Object)  
RichText.SupText(Object)  
RichText.SubText(Object)  
RichText.PositionText(Object, Single)  
RichText.SizeTypeText(Object, Int32)  
RichText.ColoredText(Object, String)  
RichText.ColoredText(Object, Color)  
RichText.BooleanText(Object, Boolean)  
RichText.BooleanText(Object, Boolean, Color)  
CreateElements.CreateElement\_Label(VisualElement, String, Boolean)  
CreateElements.CreateElement\_VisualElement(VisualElement)  
CreateElements.CreateElement\_ScrollView(VisualElement, ScrollViewMode, VisualElement[])  
CreateElements.CreateElement\_ListView< TValue >(VisualElement, List< TValue >, Single, SelectionType)  
CreateElements.CreateElement\_Button(VisualElement, Action)  
CreateElements.CreateElement\_Toggle(VisualElement, String)  
CreateElements.CreateElement\_TextField(VisualElement, String, Int32, Boolean, Boolean, Char)  
CreateElements.CreateElement\_Foldout(VisualElement, String, Boolean)  
CreateElements.CreateElement\_Slider(VisualElement, Single, Single, String)  
CreateElements.CreateElement\_Slider\_Int(VisualElement, Int32, Int32, String)  
CreateElements.CreateElement\_Slider\_Min\_Max(VisualElement, String, Single, Single, Single, Single)  
CreateElements.CreateElement\_Progress\_Bar(VisualElement, String, Single, Single, Single)  
CreateElements.CreateElement\_Dropdown(VisualElement, String, List< String >, Int32)  
CreateElements.CreateElement\_Radio(VisualElement, Boolean, String, Boolean)  
CreateElements.CreateElement\_Radio\_Group(VisualElement, String, String[])  
CreateElements.CreateElement\_ContextMenuItem(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu(VisualElement, ContextMenuItemData[])  
CreateElements.CreateElement\_ContextMenu(VisualElement, List< ContextMenuItemData >)  
CreateElements.CreateElement\_ContextMenu\_Item\_Parent(VisualElement, ContextMenuItemParentData)  
CreateElements.CreateElement\_ContextMenu\_Item(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu\_Item\_Button(VisualElement, ContextMenuItemButtonData)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, Boolean)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, VisualElement, Boolean)  
CreateElements.CreateElement\_Tooltip(VisualElement, Font, Int32)  
CreateElements.CreateElement\_Sprite(VisualElement, Sprite)  
CreateElements.CreateElement\_Slot(VisualElement, Int32)  
CreateElements.CreateElement\_Inventory(VisualElement, Int32, Int32, Inventory)  
CreateElements.CreateElement\_Icon(VisualElement, InventoryItem)

CreateElements.CreateElement\_SlotPopup(VisualElement, Vector2, Int32, InventoryElement, Boolean)

CreateElements.CreateElement\_TransferPopup(VisualElement, Vector2, TransferItem, InventoryElement, InventoryElement)

# Class ContextMenuItemButtonElement.UxmlFactory

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

## Inheritance

System.Object

UnityEngine.UIElements.UxmlFactory<[ContextMenuItemButtonElement](#), UnityEngine.UIElements.VisualElement.UxmlTraits>

ContextMenuItemButtonElement.UxmlFactory

Namespace: [Utility System.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class UxmlFactory : UxmlFactory<ContextMenuItemButtonElement, VisualElement.UxmlTraits>, IUxmlFactory
```

## Extension Methods

[RichText.UpperCaseText\(Object\)](#)

[RichText.LowerCaseText\(Object\)](#)

[RichText.BoldText\(Object\)](#)

[RichText.ItalicText\(Object\)](#)

[RichText.Text\(Object\)](#)

[RichText.UnderLineText\(Object\)](#)

[RichText.StrikeText\(Object\)](#)

[RichText.SupText\(Object\)](#)

[RichText.SubText\(Object\)](#)

[RichText.PositionText\(Object, Single\)](#)

[RichText.SizeTypeText\(Object, Int32\)](#)

[RichText.ColoredText\(Object, String\)](#)

[RichText.ColoredText\(Object, Color\)](#)

[RichText.BooleanText\(Object, Boolean\)](#)

[RichText.BooleanText\(Object, Boolean, Color\)](#)

# Class ContextMenuItemElement

This class is the context menu item for the runtime.

## Inheritance

System.Object  
UnityEngine.UIElements.CallbackEventHandler  
UnityEngine.UIElements.Focusable  
UnityEngine.UIElements.VisualElement  
ContextMenuItemElement  
[ContextMenuItemButtonElement](#)  
[ContextMenuItemParentElement](#)

Namespace: [UtilitySystem.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class ContextMenuItemElement : VisualElement, IEventHandler, ITransform, ITransitionAnimations,  
IExperimentalFeatures, IVisualElementScheduler, IResolvedStyle
```

## Constructors

### ContextMenuItemElement()

This is a constructor and will create the context menu item.

#### Declaration

```
public ContextMenuItemElement()
```

### ContextMenuItemElement(ContextMenuItemData)

This is a constructor and will create the context menu item.

#### Declaration

```
public ContextMenuItemElement(ContextMenuItemData data)
```

#### Parameters

| Type                                | Name | Description                                |
|-------------------------------------|------|--|
| <a href="#">ContextMenuItemData</a> | data | this is the list of Context Menu Item Data |

## Fields

### text

This is the label that will display the text of the item.

#### Declaration

```
public Label text
```

#### Field Value

| TYPE                         | DESCRIPTION |
|------------------------------|-------------|
| UnityEngine.UIElements.Label |             |

## Methods

### GenerateElements(ContextMenuItemData)

This will generate "ContextMenuItem" elements for this menu

#### Declaration

```
public void GenerateElements(ContextMenuItemData data)
```

#### Parameters

| TYPE                | NAME | DESCRIPTION   |
|---------------------|------|---|
| ContextMenuItemData | data | This is a ContextMenuItemData that will be used to create the menu items. |

### Interaction()

This will setup the Interaction when the mouse enters and leave the hover of this element.

#### Declaration

```
public virtual void Interaction()
```

### Style()

This will the style of the context menu.

#### Declaration

```
public virtual void Style()
```

## Extension Methods

### VisualElementStyleExtension.BackGround\_Color(VisualElement, Color)

VisualElementStyleExtension.Background\_Image(VisualElement, Background)  
VisualElementStyleExtension.Background\_Image(VisualElement, Texture2D)  
VisualElementStyleExtension.Background\_Image(VisualElement, Sprite)  
VisualElementStyleExtension.Background\_Image(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Color(VisualElement, Color)  
VisualElementStyleExtension.Color(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Top(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Right(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Bottom(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Left(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color(VisualElement, Color[])  
VisualElementStyleExtension.Border\_Color(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetRadius\_Top(VisualElement)  
VisualElementStyleExtension.GetRadius\_Bottom(VisualElement)  
VisualElementStyleExtension.GetRadius\_Size(VisualElement)  
VisualElementStyleExtension.Border\_Radius\_Top\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Top\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius\_Top\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Top\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius(VisualElement, Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetBorder\_Width(VisualElement)  
VisualElementStyleExtension.GetBorder\_Height(VisualElement)  
VisualElementStyleExtension.GetBorder\_Size(VisualElement)  
VisualElementStyleExtension.Border\_Width\_Top(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width(VisualElement, Single[])  
VisualElementStyleExtension.Border\_Width(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Border(VisualElement, Single[])  
VisualElementStyleExtension.Border(VisualElement, Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetMargin\_Width(VisualElement)  
VisualElementStyleExtension.GetMargin\_Height(VisualElement)

VisualElementStyleExtension.GetMargin\_Size(VisualElement)  
VisualElementStyleExtension.Margin\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin(VisualElement, Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Margin(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetPadding\_Width(VisualElement)  
VisualElementStyleExtension.GetPadding\_Height(VisualElement)  
VisualElementStyleExtension.GetPadding\_Size(VisualElement)  
VisualElementStyleExtension.Padding\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding(VisualElement, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Padding(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Size\_Width(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size(VisualElement, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Min(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Min(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Max(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, Single[])

VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, LengthUnit[], Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Max(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.SetDisplay(VisualElement, DisplayStyle)  
VisualElementStyleExtension.GetDisplay(VisualElement)  
VisualElementStyleExtension.Display\_ToggleVisibility(VisualElement)  
VisualElementStyleExtension.Display\_Hide(VisualElement)  
VisualElementStyleExtension.Display\_Show(VisualElement)  
VisualElementStyleExtension.Display\_IsInvisible(VisualElement)  
VisualElementStyleExtension.Display\_IsVisible(VisualElement)  
VisualElementStyleExtension.StylePosition(VisualElement, Position)  
VisualElementStyleExtension.StylePosition(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Toggle(VisualElement)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Align\_Content(VisualElement, Align)  
VisualElementStyleExtension.Align\_Content(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Item(VisualElement, Align)  
VisualElementStyleExtension.Align\_Item(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Self(VisualElement, Align)  
VisualElementStyleExtension.Align\_Self(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, Wrap)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Shink(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, FlexDirection)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, StyleKeyword)  
VisualElementStyleExtension.White\_Space(VisualElement,WhiteSpace)  
VisualElementStyleExtension.White\_Space(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font\_Size(VisualElement, Single)  
VisualElementStyleExtension.Font\_Size(VisualElement, StyleKeyword)

VisualElementStyleExtension.Font(VisualElement, Font)  
VisualElementStyleExtension.Font(VisualElement, FontAsset)  
VisualElementStyleExtension.Text\_Position(VisualElement, TextAnchor)  
VisualElementStyleExtension.Overflow(VisualElement, Overflow)  
VisualElementStyleExtension.Overflow(VisualElement, StyleKeyword)  
VisualElementExtension.ViewDataKey(VisualElement, String)  
VisualElementExtension.UsageHints(VisualElement, UsageHints)  
VisualElementExtension.ToolTip(VisualElement, String)  
VisualElementExtension.GetLayout(VisualElement)  
VisualElementExtension.GetLayoutSize(VisualElement)  
VisualElementExtension.GetLayoutWidth(VisualElement)  
VisualElementExtension.GetLayoutHeight(VisualElement)  
VisualElementExtension.GetLayoutPosition(VisualElement)  
VisualElementExtension.GetLayoutPositionX(VisualElement)  
VisualElementExtension.GetLayoutPositionY(VisualElement)  
VisualElementExtension.GetWorldBound(VisualElement)  
VisualElementExtension.GetWorldBoundSize(VisualElement)  
VisualElementExtension.GetWorldBoundPosition(VisualElement)  
VisualElementExtension.GetLocalBound(VisualElement)  
VisualElementExtension.GetLocalBoundSize(VisualElement)  
VisualElementExtension.GetLocalBoundPosition(VisualElement)  
VisualElementExtension.Pick(VisualElement, PickingMode)  
VisualElementExtension.Pick\_Toggle(VisualElement)  
VisualElementExtension.Pick\_OFF(VisualElement)  
VisualElementExtension.Pick\_ON(VisualElement)  
VisualElementExtension.Focus(VisualElement, Boolean)  
VisualElementExtension.Focus\_Toggle(VisualElement)  
VisualElementExtension.Focus\_OFF(VisualElement)  
VisualElementExtension.Focus\_ON(VisualElement)  
VisualElementExtension.SetVisible(VisualElement, Boolean)  
VisualElementExtension.GetVisible(VisualElement)  
VisualElementExtension.Visible\_ToggleVisibility(VisualElement)  
VisualElementExtension.Visible\_Hide(VisualElement)  
VisualElementExtension.Visible\_Show(VisualElement)  
VisualElementExtension.Visible\_IsInvisible(VisualElement)  
VisualElementExtension.Visible\_IsVisible(VisualElement)  
VisualElementExtension.ResetRotation(VisualElement)  
VisualElementExtension.SetTransform\_Rotate(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Rotate(VisualElement)  
VisualElementExtension.Rotate\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Counter\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.RotateY\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.SetTransform\_Position(VisualElement, Vector2)  
VisualElementExtension.GetTransform\_Position(VisualElement)  
VisualElementExtension.SetTransform\_Scale(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Scale(VisualElement)  
VisualElementExtension.Child(VisualElement, VisualElement, Boolean)  
VisualElementExtension.StyleSheet(VisualElement, StyleSheet, Boolean)  
VisualElementExtension.Name(VisualElement, String)

VisualElementExtension.GetElementSize(VisualElement)  
VisualElementExtension.ScreenBound(VisualElement)  
VisualElementExtension.MouseGrab(VisualElement, Vector2)  
VisualElementExtension.MouseRelease(VisualElement)  
VisualElementExtension.UpdatePosition(VisualElement, Vector2)  
RichText.UpperCaseText(Object)  
RichTextLowerCaseText(Object)  
RichText.BoldText(Object)  
RichText.ItalicText(Object)  
RichText.Text(Object)  
RichText.UnderLineText(Object)  
RichText.StrikeText(Object)  
RichText.SupText(Object)  
RichText.SubText(Object)  
RichText.PositionText(Object, Single)  
RichText.SizeType(Object, Int32)  
RichText.ColoredText(Object, String)  
RichText.ColoredText(Object, Color)  
RichText.BooleanText(Object, Boolean)  
RichText.BooleanText(Object, Boolean, Color)  
CreateElements.CreateElement\_Label(VisualElement, String, Boolean)  
CreateElements.CreateElement\_VisualElement(VisualElement)  
CreateElements.CreateElement\_ScrollView(VisualElement, ScrollViewMode, VisualElement[])  
CreateElements.CreateElement\_ListView< TValue >(VisualElement, List< TValue >, Single, SelectionType)  
CreateElements.CreateElement\_Button(VisualElement, Action)  
CreateElements.CreateElement\_Toggle(VisualElement, String)  
CreateElements.CreateElement\_TextField(VisualElement, String, Int32, Boolean, Boolean, Char)  
CreateElements.CreateElement\_Foldout(VisualElement, String, Boolean)  
CreateElements.CreateElement\_Slider(VisualElement, Single, Single, String)  
CreateElements.CreateElement\_Slider\_Int(VisualElement, Int32, Int32, String)  
CreateElements.CreateElement\_Slider\_Min\_Max(VisualElement, String, Single, Single, Single, Single)  
CreateElements.CreateElement\_Progress\_Bar(VisualElement, String, Single, Single, Single)  
CreateElements.CreateElement\_Dropdown(VisualElement, String, List< String >, Int32)  
CreateElements.CreateElement\_Radio(VisualElement, Boolean, String, Boolean)  
CreateElements.CreateElement\_Radio\_Group(VisualElement, String, String[])  
CreateElements.CreateElement\_ContextMenuItem(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu(VisualElement, ContextMenuItemData[])  
CreateElements.CreateElement\_ContextMenu(VisualElement, List< ContextMenuItemData >)  
CreateElements.CreateElement\_ContextMenu\_Item\_Parent(VisualElement, ContextMenuItemParentData)  
CreateElements.CreateElement\_ContextMenu\_Item(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu\_Item\_Button(VisualElement, ContextMenuItemButtonData)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, Boolean)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, VisualElement, Boolean)  
CreateElements.CreateElement\_Tooltip(VisualElement, Font, Int32)  
CreateElements.CreateElement\_Sprite(VisualElement, Sprite)  
CreateElements.CreateElement\_Slot(VisualElement, Int32)  
CreateElements.CreateElement\_Inventory(VisualElement, Int32, Int32, Inventory)  
CreateElements.CreateElement\_Icon(VisualElement, InventoryItem)  
CreateElements.CreateElement\_SlotPopup(VisualElement, Vector2, Int32, InventoryElement, Boolean)  
CreateElements.CreateElement\_TransferPopup(VisualElement, Vector2, TransferItem, InventoryElement, InventoryElement)

# Class ContextMenuItemElement.UxmlFactory

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

## Inheritance

System.Object

UnityEngine.UIElements.UxmlFactory<ContextMenuItemElement, UnityEngine.UIElements.VisualElement.UxmlTraits>  
ContextMenuItemElement.UxmlFactory

Namespace: [Utility System.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class UxmlFactory : UxmlFactory<ContextMenuItemElement, VisualElement.UxmlTraits>, IUxmlFactory
```

## Extension Methods

[RichText.UpperCaseText\(Object\)](#)

[RichText.LowerCaseText\(Object\)](#)

[RichText.BoldText\(Object\)](#)

[RichText.ItalicText\(Object\)](#)

[RichText.Text\(Object\)](#)

[RichText.UnderLineText\(Object\)](#)

[RichText.StrikeText\(Object\)](#)

[RichText.SupText\(Object\)](#)

[RichText.SubText\(Object\)](#)

[RichText.PositionText\(Object, Single\)](#)

[RichText.SizeTypeText\(Object, Int32\)](#)

[RichText.ColoredText\(Object, String\)](#)

[RichText.ColoredText\(Object, Color\)](#)

[RichText.BooleanText\(Object, Boolean\)](#)

[RichText.BooleanText\(Object, Boolean, Color\)](#)

# Class ContextMenuItemParentElement

This class is the context sub-menu for the runtime.

## Inheritance

```
System.Object  
UnityEngine.UIElements.CallbackEventHandler  
UnityEngine.UIElements.Focusable  
UnityEngine.UIElements.VisualElement  
ContextMenuItemElement  
ContextMenuItemParentElement
```

Namespace: [UtilitySystem.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class ContextMenuItemParentElement : ContextMenuItemElement, IEventHandler, ITransform,  
ITransitionAnimations, IExperimentalFeatures, IVisualElementScheduler, IResolvedStyle
```

## Constructors

### ContextMenuItemParentElement()

This is a constructor and will create the context menu item.

#### Declaration

```
public ContextMenuItemParentElement()
```

### ContextMenuItemParentElement(ContextMenuItemParentData)

This is a constructor and will create the context menu item.

#### Declaration

```
public ContextMenuItemParentElement(ContextMenuItemParentData data)
```

#### Parameters

| TYPE                                      | NAME | DESCRIPTION                                       |
|---|------|---|
| <a href="#">ContextMenuItemParentData</a> | data | this is the list of Context Menu Item Parent Data |

## Fields

### arrow

This is the element that contains '>' that rotates when hovered.

## Declaration

```
public VisualElement arrow
```

## Field Value

| TYPE                                 | DESCRIPTION |
|--------------------------------------|-------------|
| UnityEngine.UIElements.VisualElement |             |

## menu

This is the sub menu attached to this item.

## Declaration

```
public ContextMenuItemElement menu
```

## Field Value

| TYPE               | DESCRIPTION |
|--------------------|-------------|
| ContextMenuElement |             |

## Methods

### GenerateElements(ContextMenuItemParentData)

This will generate "ContextMenuItem" elements for this menu

## Declaration

```
public void GenerateElements(ContextMenuItemParentData data)
```

## Parameters

| TYPE                      | NAME | DESCRIPTION   |
|---------------------------|------|---|
| ContextMenuItemParentData | data | This is a ContextMenuItemData that will be used to create the menu items. |

## Style()

This will set the style of the context menu.

## Declaration

```
public override void Style()
```

## Overrides

ContextMenuItemElement.Style()

## Extension Methods

VisualElementStyleExtension.BackGround\_Color(VisualElement, Color)  
VisualElementStyleExtension.Background\_Image(VisualElement, Background)  
VisualElementStyleExtension.Background\_Image(VisualElement, Texture2D)  
VisualElementStyleExtension.Background\_Image(VisualElement, Sprite)  
VisualElementStyleExtension.Background\_Image(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Color(VisualElement, Color)  
VisualElementStyleExtension.Color(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Top(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Right(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Bottom(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color\_Left(VisualElement, Color)  
VisualElementStyleExtension.Border\_Color\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Color(VisualElement, Color[])  
VisualElementStyleExtension.Border\_Color(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetRadius\_Top(VisualElement)  
VisualElementStyleExtension.GetRadius\_Bottom(VisualElement)  
VisualElementStyleExtension.GetRadius\_Size(VisualElement)  
VisualElementStyleExtension.Border\_Radius\_Top\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Top\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius\_Top\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Top\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius(VisualElement, Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetBorder\_Width(VisualElement)  
VisualElementStyleExtension.GetBorder\_Height(VisualElement)  
VisualElementStyleExtension.GetBorder\_Size(VisualElement)  
VisualElementStyleExtension.Border\_Width\_Top(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width(VisualElement, Single[])  
VisualElementStyleExtension.Border\_Width(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Border(VisualElement, Single[])  
VisualElementStyleExtension.Border(VisualElement, Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, Color[], Single[])

VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetMargin\_Width(VisualElement)  
VisualElementStyleExtension.GetMargin\_Height(VisualElement)  
VisualElementStyleExtension.GetMargin\_Size(VisualElement)  
VisualElementStyleExtension.Margin\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin(VisualElement, Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Margin(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetPadding\_Width(VisualElement)  
VisualElementStyleExtension.GetPadding\_Height(VisualElement)  
VisualElementStyleExtension.GetPadding\_Size(VisualElement)  
VisualElementStyleExtension.Padding\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding(VisualElement, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Padding(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Size\_Width(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size(VisualElement, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Min(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Min(VisualElement, StyleKeyword)

VisualElementStyleExtension.GetSize\_Min(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Max(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Max(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.SetDisplay(VisualElement, DisplayStyle)  
VisualElementStyleExtension.GetDisplay(VisualElement)  
VisualElementStyleExtension.Display\_ToggleVisibility(VisualElement)  
VisualElementStyleExtension.Display\_Hide(VisualElement)  
VisualElementStyleExtension.Display\_Show(VisualElement)  
VisualElementStyleExtension.Display\_IsInvisible(VisualElement)  
VisualElementStyleExtension.Display\_IsVisible(VisualElement)  
VisualElementStyleExtension.StylePosition(VisualElement, Position)  
VisualElementStyleExtension.StylePosition(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Toggle(VisualElement)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Align\_Content(VisualElement, Align)  
VisualElementStyleExtension.Align\_Content(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Item(VisualElement, Align)  
VisualElementStyleExtension.Align\_Item(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Self(VisualElement, Align)  
VisualElementStyleExtension.Align\_Self(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, Wrap)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, FlexDirection)

VisualElementStyleExtension.Flex\_Direction(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, StyleKeyword)  
VisualElementStyleExtension.White\_Space(VisualElement,WhiteSpace)  
VisualElementStyleExtension.White\_Space(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font\_Size(VisualElement, Single)  
VisualElementStyleExtension.Font\_Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font(VisualElement, Font)  
VisualElementStyleExtension.Font(VisualElement, FontAsset)  
VisualElementStyleExtension.Text\_Position(VisualElement, TextAnchor)  
VisualElementStyleExtension.Overflow(VisualElement, Overflow)  
VisualElementStyleExtension.Overflow(VisualElement, StyleKeyword)  
VisualElementExtension.ViewDataKey(VisualElement, String)  
VisualElementExtension.UsageHints(VisualElement, UsageHints)  
VisualElementExtension.ToolTip(VisualElement, String)  
VisualElementExtension.GetLayout(VisualElement)  
VisualElementExtension.GetLayoutSize(VisualElement)  
VisualElementExtension.GetLayoutWidth(VisualElement)  
VisualElementExtension.GetLayoutHeight(VisualElement)  
VisualElementExtension.GetLayoutPosition(VisualElement)  
VisualElementExtension.GetLayoutPositionX(VisualElement)  
VisualElementExtension.GetLayoutPositionY(VisualElement)  
VisualElementExtension.GetWorldBound(VisualElement)  
VisualElementExtension.GetWorldBoundSize(VisualElement)  
VisualElementExtension.GetWorldBoundPosition(VisualElement)  
VisualElementExtension.GetLocalBound(VisualElement)  
VisualElementExtension.GetLocalBoundSize(VisualElement)  
VisualElementExtension.GetLocalBoundPosition(VisualElement)  
VisualElementExtension.Pick(VisualElement, PickingMode)  
VisualElementExtension.Pick\_Toggle(VisualElement)  
VisualElementExtension.Pick\_OFF(VisualElement)  
VisualElementExtension.Pick\_ON(VisualElement)  
VisualElementExtension.Focus(VisualElement, Boolean)  
VisualElementExtension.Focus\_Toggle(VisualElement)  
VisualElementExtension.Focus\_OFF(VisualElement)  
VisualElementExtension.Focus\_ON(VisualElement)  
VisualElementExtension.SetVisible(VisualElement, Boolean)  
VisualElementExtension.GetVisible(VisualElement)  
VisualElementExtension.Visible\_ToggleVisibility(VisualElement)  
VisualElementExtension.Visible\_Hide(VisualElement)  
VisualElementExtension.Visible\_Show(VisualElement)  
VisualElementExtension.Visible\_IsInvisible(VisualElement)  
VisualElementExtension.Visible\_IsVisible(VisualElement)  
VisualElementExtension.ResetRotation(VisualElement)  
VisualElementExtension.SetTransform\_Rotate(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Rotate(VisualElement)  
VisualElementExtension.Rotate\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Counter\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.RotateY\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Z(VisualElement, Single)

VisualElementExtension.SetTransform\_Position(VisualElement, Vector2)  
VisualElementExtension.GetTransform\_Position(VisualElement)  
VisualElementExtension.SetTransform\_Scale(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Scale(VisualElement)  
VisualElementExtension.Child(VisualElement, VisualElement, Boolean)  
VisualElementExtension.StyleSheet(VisualElement, StyleSheet, Boolean)  
VisualElementExtension.Name(VisualElement, String)  
VisualElementExtension.GetElementSize(VisualElement)  
VisualElementExtension.ScreenBound(VisualElement)  
VisualElementExtension.MouseGrab(VisualElement, Vector2)  
VisualElementExtension.MouseRelease(VisualElement)  
VisualElementExtension.UpdatePosition(VisualElement, Vector2)  
RichText.UpperCaseText(Object)  
RichTextLowerCaseText(Object)  
RichText.BoldText(Object)  
RichText.ItalicText(Object)  
RichText.Text(Object)  
RichText.UnderLineText(Object)  
RichText.StrikeText(Object)  
RichText.SupText(Object)  
RichText.SubText(Object)  
RichText.PositionText(Object, Single)  
RichText.SizeTypeText(Object, Int32)  
RichText.ColoredText(Object, String)  
RichText.ColoredText(Object, Color)  
RichText.BooleanText(Object, Boolean)  
RichText.BooleanText(Object, Boolean, Color)  
CreateElements.CreateElement\_Label(VisualElement, String, Boolean)  
CreateElements.CreateElement\_VisualElement(VisualElement)  
CreateElements.CreateElement\_ScrollView(VisualElement, ScrollViewMode, VisualElement[])  
CreateElements.CreateElement\_ListView < TValue >(VisualElement, List< TValue >, Single, SelectionType)  
CreateElements.CreateElement\_Button(VisualElement, Action)  
CreateElements.CreateElement\_Toggle(VisualElement, String)  
CreateElements.CreateElement\_TextField(VisualElement, String, Int32, Boolean, Boolean, Char)  
CreateElements.CreateElement\_Foldout(VisualElement, String, Boolean)  
CreateElements.CreateElement\_Slider(VisualElement, Single, Single, String)  
CreateElements.CreateElement\_Slider\_Int(VisualElement, Int32, Int32, String)  
CreateElements.CreateElement\_Slider\_Min\_Max(VisualElement, String, Single, Single, Single, Single)  
CreateElements.CreateElement\_Progress\_Bar(VisualElement, String, Single, Single, Single)  
CreateElements.CreateElement\_Dropdown(VisualElement, String, List< String >, Int32)  
CreateElements.CreateElement\_Radio(VisualElement, Boolean, String, Boolean)  
CreateElements.CreateElement\_Radio\_Group(VisualElement, String, String[])  
CreateElements.CreateElement\_ContextMenuItem(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu(VisualElement, ContextMenuItemData[])  
CreateElements.CreateElement\_ContextMenu(VisualElement, List< ContextMenuItemData >)  
CreateElements.CreateElement\_ContextMenu\_Item\_Parent(VisualElement, ContextMenuItemParentData)  
CreateElements.CreateElement\_ContextMenu\_Item(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu\_Item\_Button(VisualElement, ContextMenuItemButtonData)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, Boolean)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, VisualElement, Boolean)  
CreateElements.CreateElement\_Tooltip(VisualElement, Font, Int32)  
CreateElements.CreateElement\_Sprite(VisualElement, Sprite)  
CreateElements.CreateElement\_Slot(VisualElement, Int32)  
CreateElements.CreateElement\_Inventory(VisualElement, Int32, Int32, Inventory)

CreateElements.CreateElement\_Icon(VisualElement, InventoryItem)

CreateElements.CreateElement\_SlotPopup(VisualElement, Vector2, Int32, InventoryElement, Boolean)

CreateElements.CreateElement\_TransferPopup(VisualElement, Vector2, TransferItem, InventoryElement, InventoryElement)

# Class ContextMenuItemParentElement.UxmlFactory

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

## Inheritance

System.Object

UnityEngine.UIElements.UxmlFactory<ContextMenuItemParentElement, UnityEngine.UIElements.VisualElement.UxmlTraits>

ContextMenuItemParentElement.UxmlFactory

Namespace: [Utility System.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class UxmlFactory : UxmlFactory<ContextMenuItemParentElement, VisualElement.UxmlTraits>, IUxmlFactory
```

## Extension Methods

[RichText.UpperCaseText\(Object\)](#)

[RichText.LowerCaseText\(Object\)](#)

[RichText.BoldText\(Object\)](#)

[RichText.ItalicText\(Object\)](#)

[RichText.Text\(Object\)](#)

[RichText.UnderLineText\(Object\)](#)

[RichText.StrikeText\(Object\)](#)

[RichText.SupText\(Object\)](#)

[RichText.SubText\(Object\)](#)

[RichText.PositionText\(Object, Single\)](#)

[RichText.SizeTypeText\(Object, Int32\)](#)

[RichText.ColoredText\(Object, String\)](#)

[RichText.ColoredText\(Object, Color\)](#)

[RichText.BooleanText\(Object, Boolean\)](#)

[RichText.BooleanText\(Object, Boolean, Color\)](#)

# Class DialogBoxElement

This class is the dialog box that can be move around on the screen.

## Inheritance

System.Object  
UnityEngine.UIElements.CallbackEventHandler  
UnityEngine.UIElements.Focusable  
UnityEngine.UIElements.VisualElement  
DialogBoxElement

Namespace: [Utility System .UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class DialogBoxElement : VisualElement, IEventHandler, ITransform, ITransitionAnimations,  
IExperimentalFeatures, IVisualElementScheduler, IResolvedStyle
```

## Constructors

### DialogBoxElement()

This is a constructor and will create the dialog box.

#### Declaration

```
public DialogBoxElement()
```

### DialogBoxElement(String, String, Boolean, StyleSheet)

This is a constructor and will create the dialog box.

#### Declaration

```
public DialogBoxElement(string UID, string titleText, bool isDisplay = false, StyleSheet style = null)
```

#### Parameters

| Type           | Name      | Description  |
|----------------|-----------|--|
| System.String  | UID       | This is the Unique ID for this DialogBox.                              |
| System.String  | titleText | The text for the label that will be the title of the element.          |
| System.Boolean | isDisplay | This boolean if true will add Visual to the onClose or Close if false. |

| Type                              | Name  | Description                     |
|-----------------------------------|-------|---------------------------------|
| UnityEngine.UIElements.StyleSheet | style | The stylesheet for the element. |

## DialogBoxElement(String, String, VisualElement, Boolean, StyleSheet)

This is a constructor and will create the dialog box.

### Declaration

```
public DialogBoxElement(string UID, string titleText, VisualElement content, bool isDisplay = false,
StyleSheet style = null)
```

### Parameters

| Type                                 | Name      | Description  |
|--------------------------------------|-----------|--|
| System.String                        | UID       | This is the Unique ID for this DialogBox.                              |
| System.String                        | titleText | The text for the label that will be the title of the element.          |
| UnityEngine.UIElements.VisualElement | content   | This element will be added to the content box.                         |
| System.Boolean                       | isDisplay | This boolean if true will add Visual to the onClose or Close if false. |
| UnityEngine.UIElements.StyleSheet    | style     | The stylesheet for the element.  |

## Properties

### close

This is the button that will be on the title bar.

### Declaration

```
public Button close { get; }
```

### Property Value

| Type                          | Description |
|-------------------------------|-------------|
| UnityEngine.UIElements.Button |             |

## GetContentBox

---

This will get the contentBox.

### Declaration

```
public VisualElement GetContentBox { get; }
```

### Property Value

| TYPE                                 | DESCRIPTION             |
|--------------------------------------|-------------------------|
| UnityEngine.UIElements.VisualElement | returns the contentBox. |

## GetFootBar

---

This will get the footBar.

### Declaration

```
public VisualElement GetFootBar { get; }
```

### Property Value

| TYPE                                 | DESCRIPTION          |
|--------------------------------------|----------------------|
| UnityEngine.UIElements.VisualElement | returns the footBar. |

## GetTitleBar

---

This will get the titlebar.

### Declaration

```
public VisualElement GetTitleBar { get; }
```

### Property Value

| TYPE                                 | DESCRIPTION           |
|--------------------------------------|-----------------------|
| UnityEngine.UIElements.VisualElement | returns the titleBar. |

## maximise

---

This is the button that will be on the title bar.

## Declaration

```
public Button maximise { get; }
```

## Property Value

| TYPE                          | DESCRIPTION |
|-------------------------------|-------------|
| UnityEngine.UIElements.Button |             |

## minimise

This is the button that will be on the title bar.

## Declaration

```
public Button minimise { get; }
```

## Property Value

| TYPE                          | DESCRIPTION |
|-------------------------------|-------------|
| UnityEngine.UIElements.Button |             |

## UID

This is the unique id for this DialogBox

## Declaration

```
public string UID { get; }
```

## Property Value

| TYPE          | DESCRIPTION |
|---------------|-------------|
| System.String |             |

## Methods

### Close()

This will remove this element from it's parent.

## Declaration

```
public void Close()
```

## CloseAction(Action, Boolean)

---

This will add or remove an action for onClose

### Declaration

```
public void CloseAction(Action action, bool isAdd = true)
```

### Parameters

| Type           | Name   | Description                                       |
|----------------|--------|---|
| System.Action  | action | This is the action that will be added or removed. |
| System.Boolean | isAdd  | If this is true then add else remove.             |

## Content(VisualElement, Boolean)

---

This will add, remove an element or clear the whole content box.

### Declaration

```
public void Content(VisualElement content = null, bool isAdd = true)
```

### Parameters

| Type                                 | Name    | Description  |
|--------------------------------------|---------|--|
| UnityEngine.UIElements.VisualElement | content | Is an optional element to be added to the content box. |
| System.Boolean                       | isAdd   | Is an optional bool to determine of added or removed.  |

## Footer(VisualElement, Boolean)

---

This will add, remove an element or clear the whole foot bar.

### Declaration

```
public void Footer(VisualElement content = null, bool isAdd = true)
```

### Parameters

| Type | Name | Description |
|------|------|-------------|
|      |      |             |

| <b>TYPE</b>                          | <b>NAME</b> | <b>DESCRIPTION</b>                                    |
|--------------------------------------|-------------|---|
| UnityEngine.UIElements.VisualElement | content     | Is an optional element to be added to the foot bar.   |
| System.Boolean                       | isAdd       | Is an optional bool to determine of added or removed. |

## GenerateButton(Action, String, String)

---

This will create the buttons for the titlebar.

### Declaration

```
public Button GenerateButton(Action action, string name, string text)
```

### Parameters

| <b>TYPE</b>   | <b>NAME</b> | <b>DESCRIPTION</b>                           |
|---------------|-------------|--|
| System.Action | action      | This is the action that the button will use. |
| System.String | name        |  |
| System.String | text        |  |

### Returns

| <b>TYPE</b>                   | <b>DESCRIPTION</b>                         |
|-------------------------------|--|
| UnityEngine.UIElements.Button | The Return is the button that was created. |

## HideAction(Action, Boolean)

---

This will add or remove an action for onHide

### Declaration

```
public void HideAction(Action action, bool isAdd = true)
```

### Parameters

| <b>TYPE</b>   | <b>NAME</b> | <b>DESCRIPTION</b>                                |
|---------------|-------------|---|
| System.Action | action      | This is the action that will be added or removed. |

| TYPE           | NAME  | DESCRIPTION                           |
|----------------|-------|---------------------------------------|
| System.Boolean | isAdd | If this is true then add else remove. |

## MaximiseAction(Action, Boolean)

---

This will add or remove an action for onMaximise

### Declaration

```
public void MaximiseAction(Action action, bool isAdd = true)
```

### Parameters

| TYPE           | NAME   | DESCRIPTION                                       |
|----------------|--------|---|
| System.Action  | action | This is the action that will be added or removed. |
| System.Boolean | isAdd  | If this is true then add else remove.             |

## MinimiseAction(Action, Boolean)

---

This will add or remove an action for onMinimise

### Declaration

```
public void MinimiseAction(Action action, bool isAdd = true)
```

### Parameters

| TYPE           | NAME   | DESCRIPTION                                       |
|----------------|--------|---|
| System.Action  | action | This is the action that will be added or removed. |
| System.Boolean | isAdd  | If this is true then add else remove.             |

## ShowAction(Action, Boolean)

---

This will add or remove an action for onShow

### Declaration

```
public void ShowAction(Action action, bool isAdd = true)
```

## Parameters

| Type           | Name   | Description                                       |
|----------------|--------|---|
| System.Action  | action | This is the action that will be added or removed. |
| System.Boolean | isAdd  | If this is true then add else remove.             |

## Title(VisualElement, Boolean)

This will add, remove an element or clear the whole title bar.

### Declaration

```
public void Title(VisualElement content = null, bool isAdd = true)
```

## Parameters

| Type                                 | Name    | Description   |
|--------------------------------------|---------|---|
| UnityEngine.UIElements.VisualElement | content | Is an optional element to be added to the title bar.  |
| System.Boolean                       | isAdd   | Is an optional bool to determine of added or removed. |

## Visual(Boolean)

This allows for the changing of the display of the dialogbox so it is kept instead of removing the UI and can be turned on if needed.

### Declaration

```
public void Visual(bool show = true)
```

## Parameters

| Type           | Name | Description  |
|----------------|------|--|
| System.Boolean | show | This boolean if true will invoke the onShow() action or onHide() if false. |

## WindowDragInit(Action<Object>)

This when invoked will allow for the element to be moved by the player.

### Declaration

```
public void WindowDragInit(Action<object> action = null)
```

#### Parameters

| Type                         | Name   | Description |
|------------------------------|--------|-------------|
| System.Action<System.Object> | action |             |

#### Remarks

Parent Element needed and will cover the whole screen on mouse down event.

## Extension Methods

[VisualElementStyleExtension.BackGround\\_Color\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, Background\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, Texture2D\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, Sprite\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Color\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Color\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Top\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Top\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Right\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Right\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Bottom\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Bottom\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Left\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Left\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\(VisualElement, Color\[\]\)](#)  
[VisualElementStyleExtension.Border\\_Color\(VisualElement, StyleKeyword\[\]\)](#)  
[VisualElementStyleExtension.GetRadius\\_Top\(VisualElement\)](#)  
[VisualElementStyleExtension.GetRadius\\_Bottom\(VisualElement\)](#)  
[VisualElementStyleExtension.GetRadius\\_Size\(VisualElement\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Left\(VisualElement, Single, LengthUnit\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Left\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Right\(VisualElement, Single, LengthUnit\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Right\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Bottom\\_Left\(VisualElement, Single, LengthUnit\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Bottom\\_Left\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Bottom\\_Right\(VisualElement, Single, LengthUnit\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Bottom\\_Right\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Radius\(VisualElement, Single\[\]\)](#)  
[VisualElementStyleExtension.Border\\_Radius\(VisualElement, LengthUnit, Single\[\]\)](#)  
[VisualElementStyleExtension.Border\\_Radius\(VisualElement, LengthUnit\[\], Single\[\]\)](#)  
[VisualElementStyleExtension.Border\\_Radius\(VisualElement, StyleKeyword\[\]\)](#)  
[VisualElementStyleExtension.GetBorder\\_Width\(VisualElement\)](#)  
[VisualElementStyleExtension.GetBorder\\_Height\(VisualElement\)](#)  
[VisualElementStyleExtension.GetBorder\\_Size\(VisualElement\)](#)  
[VisualElementStyleExtension.Border\\_Width\\_Top\(VisualElement, Single\)](#)  
[VisualElementStyleExtension.Border\\_Width\\_Top\(VisualElement, StyleKeyword\)](#)

VisualElementStyleExtension.Border\_Width\_Right(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width(VisualElement, Single[])  
VisualElementStyleExtension.Border\_Width(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Border(VisualElement, Single[])  
VisualElementStyleExtension.Border(VisualElement, Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetMargin\_Width(VisualElement)  
VisualElementStyleExtension.GetMargin\_Height(VisualElement)  
VisualElementStyleExtension.GetMargin\_Size(VisualElement)  
VisualElementStyleExtension.Margin\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin(VisualElement, Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Margin(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetPadding\_Width(VisualElement)  
VisualElementStyleExtension.GetPadding\_Height(VisualElement)  
VisualElementStyleExtension.GetPadding\_Size(VisualElement)  
VisualElementStyleExtension.Padding\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding(VisualElement, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Padding(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Size\_Width(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size(VisualElement, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, Vector2, LengthUnit[])

VisualElementStyleExtension.Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Min(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Min(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Max(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Max(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.SetDisplay(VisualElement, DisplayStyle)  
VisualElementStyleExtension.GetDisplay(VisualElement)  
VisualElementStyleExtension.Display\_ToggleVisibility(VisualElement)  
VisualElementStyleExtension.Display\_Hide(VisualElement)  
VisualElementStyleExtension.Display\_Show(VisualElement)  
VisualElementStyleExtension.Display\_IsInvisible(VisualElement)  
VisualElementStyleExtension.Display\_IsVisible(VisualElement)  
VisualElementStyleExtension.StylePosition(VisualElement, Position)  
VisualElementStyleExtension.StylePosition(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Toggle(VisualElement)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Align\_Content(VisualElement, Align)  
VisualElementStyleExtension.Align\_Content(VisualElement, StyleKeyword)

VisualElementStyleExtension.Align\_Item(VisualElement, Align)  
VisualElementStyleExtension.Align\_Item(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Self(VisualElement, Align)  
VisualElementStyleExtension.Align\_Self(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, Wrap)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, FlexDirection)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, StyleKeyword)  
VisualElementStyleExtension.White\_Space(VisualElement,WhiteSpace)  
VisualElementStyleExtension.White\_Space(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font\_Size(VisualElement, Single)  
VisualElementStyleExtension.Font\_Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font(VisualElement, Font)  
VisualElementStyleExtension.Font(VisualElement, FontAsset)  
VisualElementStyleExtension.Text\_Position(VisualElement, TextAnchor)  
VisualElementStyleExtension.Overflow(VisualElement, Overflow)  
VisualElementStyleExtension.Overflow(VisualElement, StyleKeyword)  
VisualElementExtension.ViewDataKey(VisualElement, String)  
VisualElementExtension.UsageHints(VisualElement, UsageHints)  
VisualElementExtension.ToolTip(VisualElement, String)  
VisualElementExtension.GetLayout(VisualElement)  
VisualElementExtension.GetLayoutSize(VisualElement)  
VisualElementExtension.GetLayoutWidth(VisualElement)  
VisualElementExtension.GetLayoutHeight(VisualElement)  
VisualElementExtension.GetLayoutPosition(VisualElement)  
VisualElementExtension.GetLayoutPositionX(VisualElement)  
VisualElementExtension.GetLayoutPositionY(VisualElement)  
VisualElementExtension.GetWorldBound(VisualElement)  
VisualElementExtension.GetWorldBoundSize(VisualElement)  
VisualElementExtension.GetWorldBoundPosition(VisualElement)  
VisualElementExtension.GetLocalBound(VisualElement)  
VisualElementExtension.GetLocalBoundSize(VisualElement)  
VisualElementExtension.GetLocalBoundPosition(VisualElement)  
VisualElementExtension.Pick(VisualElement, PickingMode)  
VisualElementExtension.Pick\_Toggle(VisualElement)  
VisualElementExtension.Pick\_OFF(VisualElement)  
VisualElementExtension.Pick\_ON(VisualElement)  
VisualElementExtension.Focus(VisualElement, Boolean)  
VisualElementExtension.Focus\_Toggle(VisualElement)  
VisualElementExtension.Focus\_OFF(VisualElement)  
VisualElementExtension.Focus\_ON(VisualElement)  
VisualElementExtension.SetVisible(VisualElement, Boolean)  
VisualElementExtension.GetVisible(VisualElement)  
VisualElementExtension.Visible\_ToggleVisibility(VisualElement)  
VisualElementExtension.Visible\_Hide(VisualElement)  
VisualElementExtension.Visible\_Show(VisualElement)  
VisualElementExtension.Visible\_IsInvisible(VisualElement)  
VisualElementExtension.Visible\_IsVisible(VisualElement)

```
VisualElementExtension.ResetRotation(VisualElement)
VisualElementExtension.SetTransform_Rotate(VisualElement, Vector3)
VisualElementExtension.GetTransform_Rotate(VisualElement)
VisualElementExtension.Rotate_Clockwise(VisualElement, Vector3)
VisualElementExtension.Rotate_Counter_Clockwise(VisualElement, Vector3)
VisualElementExtension.Rotate_Clockwise_X(VisualElement, Single)
VisualElementExtension.Rotate_Counter_Clockwise_X(VisualElement, Single)
VisualElementExtension.RotateY_Clockwise_Y(VisualElement, Single)
VisualElementExtension.Rotate_Counter_Clockwise_Y(VisualElement, Single)
VisualElementExtension.Rotate_Clockwise_Z(VisualElement, Single)
VisualElementExtension.Rotate_Counter_Clockwise_Z(VisualElement, Single)
VisualElementExtension.SetTransform_Position(VisualElement, Vector2)
VisualElementExtension.GetTransform_Position(VisualElement)
VisualElementExtension.SetTransform_Scale(VisualElement, Vector3)
VisualElementExtension.GetTransform_Scale(VisualElement)
VisualElementExtension.Child(VisualElement, VisualElement, Boolean)
VisualElementExtension.StyleSheet(VisualElement, StyleSheet, Boolean)
VisualElementExtension.Name(VisualElement, String)
VisualElementExtension.GetElementSize(VisualElement)
VisualElementExtension.ScreenBound(VisualElement)
VisualElementExtension.MouseGrab(VisualElement, Vector2)
VisualElementExtension.MouseRelease(VisualElement)
VisualElementExtension.UpdatePosition(VisualElement, Vector2)
RichText.UpperCaseText(Object)
RichText.LowerCaseText(Object)
RichText.BoldText(Object)
RichText.ItalicText(Object)
RichText.Text(Object)
RichText.UnderLineText(Object)
RichText.StrikeText(Object)
RichText.SupText(Object)
RichText.SubText(Object)
RichText.PositionText(Object, Single)
RichText.SizeType(Object, Int32)
RichText.ColoredText(Object, String)
RichText.ColoredText(Object, Color)
RichText.BooleanText(Object, Boolean)
RichText.BooleanText(Object, Boolean, Color)
CreateElements.CreateElement_Label(VisualElement, String, Boolean)
CreateElements.CreateElement_VisualElement(VisualElement)
CreateElements.CreateElement_ScrollView(VisualElement, ScrollViewMode, VisualElement[])
CreateElements.CreateElement_ListView< TValue >(VisualElement, List< TValue >, Single, SelectionType)
CreateElements.CreateElement_Button(VisualElement, Action)
CreateElements.CreateElement_Toggle(VisualElement, String)
CreateElements.CreateElement_TextField(VisualElement, String, Int32, Boolean, Boolean, Char)
CreateElements.CreateElement_Foldout(VisualElement, String, Boolean)
CreateElements.CreateElement_Slider(VisualElement, Single, Single, String)
CreateElements.CreateElement_Slider_Int(VisualElement, Int32, Int32, String)
CreateElements.CreateElement_Slider_Min_Max(VisualElement, String, Single, Single, Single, Single)
CreateElements.CreateElement_Progress_Bar(VisualElement, String, Single, Single, Single)
CreateElements.CreateElement_Dropdown(VisualElement, String, List< String >, Int32)
CreateElements.CreateElement_Radio(VisualElement, Boolean, String, Boolean)
CreateElements.CreateElement_Radio_Group(VisualElement, String, String[])
CreateElements.CreateElement_ContextMenuItem(VisualElement, ContextMenuItemData)
```

CreateElements.CreateElement\_ContextMenu(VisualElement, ContextMenuItemData[])  
CreateElements.CreateElement\_ContextMenu(VisualElement, List<ContextMenuItemData>)  
CreateElements.CreateElement\_ContextMenu\_Item\_Parent(VisualElement, ContextMenuItemParentData)  
CreateElements.CreateElement\_ContextMenu\_Item(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu\_Item\_Button(VisualElement, ContextMenuItemButtonData)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, Boolean)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, VisualElement, Boolean)  
CreateElements.CreateElement\_Tooltip(VisualElement, Font, Int32)  
CreateElements.CreateElement\_Sprite(VisualElement, Sprite)  
CreateElements.CreateElement\_Slot(VisualElement, Int32)  
CreateElements.CreateElement\_Inventory(VisualElement, Int32, Int32, Inventory)  
CreateElements.CreateElement\_Icon(VisualElement, InventoryItem)  
CreateElements.CreateElement\_SlotPopup(VisualElement, Vector2, Int32, InventoryElement, Boolean)  
CreateElements.CreateElement\_TransferPopup(VisualElement, Vector2, TransferItem, InventoryElement, InventoryElement)

# Class DialogBoxElement.UxmlFactory

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

## Inheritance

System.Object

UnityEngine.UIElements.UxmlFactory<[DialogBoxElement](#), UnityEngine.UIElements.VisualElement.UxmlTraits>

DialogBoxElement.UxmlFactory

Namespace: [UtilitySystem.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class UxmlFactory : UxmlFactory<DialogBoxElement, VisualElement.UxmlTraits>, IUxmlFactory
```

## Extension Methods

[RichText.UpperCaseText\(Object\)](#)

[RichText.LowerCaseText\(Object\)](#)

[RichText.BoldText\(Object\)](#)

[RichText.ItalicText\(Object\)](#)

[RichText.Text\(Object\)](#)

[RichText.UnderLineText\(Object\)](#)

[RichText.StrikeText\(Object\)](#)

[RichText.SupText\(Object\)](#)

[RichText.SubText\(Object\)](#)

[RichText.PositionText\(Object, Single\)](#)

[RichText.SizeType\(Object, Int32\)](#)

[RichText.ColoredText\(Object, String\)](#)

[RichText.ColoredText\(Object, Color\)](#)

[RichText.BooleanText\(Object, Boolean\)](#)

[RichText.BooleanText\(Object, Boolean, Color\)](#)

# Class PopupBaseElement

This abstract class that is the base of the popup elements.

## Inheritance

System.Object  
UnityEngine.UIElements.CallbackEventHandler  
UnityEngine.UIElements.Focusable  
UnityEngine.UIElements.VisualElement  
PopupBaseElement  
[SlotPopupElement](#)  
[TransferPopupElement](#)

Namespace: [UtilitySystem.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public abstract class PopupBaseElement : VisualElement, IEventHandler, ITransform, ITransitionAnimations,  
IExperimentalFeatures, IVisualElementScheduler, IResolvedStyle
```

## Constructors

### [PopupBaseElement\(\)](#)

This is the constructor

#### Declaration

```
public PopupBaseElement()
```

### [PopupBaseElement\(Vector2\)](#)

This is the constructor

#### Declaration

```
public PopupBaseElement(Vector2 size)
```

#### Parameters

| TYPE                | NAME | DESCRIPTION                    |
|---------------------|------|--------------------------------|
| UnityEngine.Vector2 | size | This is the size of the popup. |

## Fields

### [background](#)

This is the background of the popup.

Declaration

```
protected VisualElement background
```

Field Value

| TYPE                                 | DESCRIPTION |
|--------------------------------------|-------------|
| UnityEngine.UIElements.VisualElement |             |

**buttonParent**

This is the element that holds the button.

Declaration

```
protected VisualElement buttonParent
```

Field Value

| TYPE                                 | DESCRIPTION |
|--------------------------------------|-------------|
| UnityEngine.UIElements.VisualElement |             |

**cancelButton**

This is the cancel button.

Declaration

```
protected Button cancelButton
```

Field Value

| TYPE                          | DESCRIPTION |
|-------------------------------|-------------|
| UnityEngine.UIElements.Button |             |

**contentParent**

This is the element that will hold the content.

Declaration

```
protected VisualElement contentParent
```

Field Value

| TYPE                                 | DESCRIPTION |
|--------------------------------------|-------------|
| UnityEngine.UIElements.VisualElement |             |

## submitButton

---

This is the submit button.

### Declaration

```
protected Button submitButton
```

## Field Value

| TYPE                          | DESCRIPTION |
|-------------------------------|-------------|
| UnityEngine.UIElements.Button |             |

## titleText

---

This is the title of the popup.

### Declaration

```
protected Label titleText
```

## Field Value

| TYPE                         | DESCRIPTION |
|------------------------------|-------------|
| UnityEngine.UIElements.Label |             |

## Methods

### Cancel()

---

This will be invoke by the cancel button and will close the popup.

### Declaration

```
public virtual void Cancel()
```

### GenerateElements()

---

This will generate the elements for the popup

### Declaration

```
public virtual void GenerateElements()
```

## Style(Vector2)

This will style the element.

### Declaration

```
public virtual void Style(Vector2 size)
```

### Parameters

| Type                | Name | Description |
|---------------------|------|-------------|
| UnityEngine.Vector2 | size |             |

## Submit()

This will be invoke by the submit button and will either drop or split the item.

### Declaration

```
public abstract void Submit()
```

## Extension Methods

[VisualElementStyleExtension.BackGround\\_Color\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, Background\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, Texture2D\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, Sprite\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Color\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Color\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Top\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Top\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Right\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Right\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Bottom\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Bottom\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Left\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Left\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\(VisualElement, Color\[\]\)](#)  
[VisualElementStyleExtension.Border\\_Color\(VisualElement, StyleKeyword\[\]\)](#)  
[VisualElementStyleExtension.GetRadius\\_Top\(VisualElement\)](#)  
[VisualElementStyleExtension.GetRadius\\_Bottom\(VisualElement\)](#)  
[VisualElementStyleExtension.GetRadius\\_Size\(VisualElement\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Left\(VisualElement, Single, LengthUnit\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Left\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Right\(VisualElement, Single, LengthUnit\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Right\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Bottom\\_Left\(VisualElement, Single, LengthUnit\)](#)

VisualElementStyleExtension.Border\_Radius\_Bottom\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Border\_Radius\_Bottom\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Radius(VisualElement, Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Border\_Radius(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetBorder\_Width(VisualElement)  
VisualElementStyleExtension.GetBorder\_Height(VisualElement)  
VisualElementStyleExtension.GetBorder\_Size(VisualElement)  
VisualElementStyleExtension.Border\_Width\_Top(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width(VisualElement, Single[])  
VisualElementStyleExtension.Border\_Width(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Border(VisualElement, Single[])  
VisualElementStyleExtension.Border(VisualElement, Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color, Single[])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color[], Single[])  
VisualElementStyleExtension.Border(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetMargin\_Width(VisualElement)  
VisualElementStyleExtension.GetMargin\_Height(VisualElement)  
VisualElementStyleExtension.GetMargin\_Size(VisualElement)  
VisualElementStyleExtension.Margin\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin(VisualElement, Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Margin(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetPadding\_Width(VisualElement)  
VisualElementStyleExtension.GetPadding\_Height(VisualElement)  
VisualElementStyleExtension.GetPadding\_Size(VisualElement)  
VisualElementStyleExtension.Padding\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Left(VisualElement, StyleKeyword)

VisualElementStyleExtension.Padding(VisualElement, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Padding(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Size\_Width(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size(VisualElement, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Min(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Min(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Max(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Max(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.SetDisplay(VisualElement, DisplayStyle)  
VisualElementStyleExtension.GetDisplay(VisualElement)  
VisualElementStyleExtension.Display\_ToggleVisibility(VisualElement)  
VisualElementStyleExtension.Display\_Hide(VisualElement)  
VisualElementStyleExtension.Display\_Show(VisualElement)  
VisualElementStyleExtension.Display\_IsInvisible(VisualElement)  
VisualElementStyleExtension.Display\_IsVisible(VisualElement)  
VisualElementStyleExtension.StylePosition(VisualElement, Position)  
VisualElementStyleExtension.StylePosition(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Toggle(VisualElement)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Single, Single)

VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Align\_Content(VisualElement, Align)  
VisualElementStyleExtension.Align\_Content(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Item(VisualElement, Align)  
VisualElementStyleExtension.Align\_Item(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Self(VisualElement, Align)  
VisualElementStyleExtension.Align\_Self(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, Wrap)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Shink(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Shink(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, FlexDirection)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, StyleKeyword)  
VisualElementStyleExtension.White\_Space(VisualElement,WhiteSpace)  
VisualElementStyleExtension.White\_Space(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font\_Size(VisualElement, Single)  
VisualElementStyleExtension.Font\_Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font(VisualElement, Font)  
VisualElementStyleExtension.Font(VisualElement, FontAsset)  
VisualElementStyleExtension.Text\_Position(VisualElement, TextAnchor)  
VisualElementStyleExtension.Overflow(VisualElement, Overflow)  
VisualElementStyleExtension.Overflow(VisualElement, StyleKeyword)  
VisualElementExtension.ViewDataKey(VisualElement, String)  
VisualElementExtension.UsageHints(VisualElement, UsageHints)  
VisualElementExtension.ToolTip(VisualElement, String)  
VisualElementExtension.GetLayout(VisualElement)  
VisualElementExtension.GetLayoutSize(VisualElement)  
VisualElementExtension.GetLayoutWidth(VisualElement)  
VisualElementExtension.GetLayoutHeight(VisualElement)  
VisualElementExtension.GetLayoutPosition(VisualElement)  
VisualElementExtension.GetLayoutPositionX(VisualElement)  
VisualElementExtension.GetLayoutPositionY(VisualElement)  
VisualElementExtension.GetWorldBound(VisualElement)  
VisualElementExtension.GetWorldBoundSize(VisualElement)  
VisualElementExtension.GetWorldBoundPosition(VisualElement)  
VisualElementExtension.GetLocalBound(VisualElement)  
VisualElementExtension.GetLocalBoundSize(VisualElement)  
VisualElementExtension.GetLocalBoundPosition(VisualElement)  
VisualElementExtension.Pick(VisualElement, PickingMode)  
VisualElementExtension.Pick\_Toggle(VisualElement)  
VisualElementExtension.Pick\_OFF(VisualElement)

VisualElementExtension.Pick\_ON(VisualElement)  
VisualElementExtension.Focus(VisualElement, Boolean)  
VisualElementExtension.Focus\_Toggle(VisualElement)  
VisualElementExtension.Focus\_OFF(VisualElement)  
VisualElementExtension.Focus\_ON(VisualElement)  
VisualElementExtension.SetVisible(VisualElement, Boolean)  
VisualElementExtension.GetVisible(VisualElement)  
VisualElementExtension.Visible\_ToggleVisibility(VisualElement)  
VisualElementExtension.Visible\_Hide(VisualElement)  
VisualElementExtension.Visible\_Show(VisualElement)  
VisualElementExtension.Visible\_IsInvisible(VisualElement)  
VisualElementExtension.Visible\_IsVisible(VisualElement)  
VisualElementExtension.ResetRotation(VisualElement)  
VisualElementExtension.SetTransform\_Rotate(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Rotate(VisualElement)  
VisualElementExtension.Rotate\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Counter\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.RotateY\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.SetTransform\_Position(VisualElement, Vector2)  
VisualElementExtension.GetTransform\_Position(VisualElement)  
VisualElementExtension.SetTransform\_Scale(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Scale(VisualElement)  
VisualElementExtension.Child(VisualElement, VisualElement, Boolean)  
VisualElementExtension.StyleSheet(VisualElement, StyleSheet, Boolean)  
VisualElementExtension.Name(VisualElement, String)  
VisualElementExtension.GetElementSize(VisualElement)  
VisualElementExtension.ScreenBound(VisualElement)  
VisualElementExtension.MouseGrab(VisualElement, Vector2)  
VisualElementExtension.MouseRelease(VisualElement)  
VisualElementExtension.UpdatePosition(VisualElement, Vector2)  
RichText.UpperCaseText(Object)  
RichText.LowerCaseText(Object)  
RichText.BoldText(Object)  
RichText.ItalicText(Object)  
RichText.Text(Object)  
RichText.UnderLineText(Object)  
RichText.StrikeText(Object)  
RichText.SupText(Object)  
RichText.SubText(Object)  
RichText.PositionText(Object, Single)  
RichText.SizeType(Object, Int32)  
RichText.ColoredText(Object, String)  
RichText.ColoredText(Object, Color)  
RichText.BooleanText(Object, Boolean)  
RichText.BooleanText(Object, Boolean, Color)  
CreateElements.CreateElement\_Label(VisualElement, String, Boolean)  
CreateElements.CreateElement\_VisualElement(VisualElement)  
CreateElements.CreateElement\_ScrollView(VisualElement, ScrollViewMode, VisualElement[])  
CreateElements.CreateElement\_ListView<TValue>(VisualElement, List<TValue>, Single, SelectionType)

CreateElements.CreateElement\_Button(VisualElement, Action)  
CreateElements.CreateElement\_Toggle(VisualElement, String)  
CreateElements.CreateElement\_TextField(VisualElement, String, Int32, Boolean, Boolean, Char)  
CreateElements.CreateElement\_Foldout(VisualElement, String, Boolean)  
CreateElements.CreateElement\_Slider(VisualElement, Single, Single, String)  
CreateElements.CreateElement\_Slider\_Int(VisualElement, Int32, Int32, String)  
CreateElements.CreateElement\_Slider\_Min\_Max(VisualElement, String, Single, Single, Single, Single)  
CreateElements.CreateElement\_Progress\_Bar(VisualElement, String, Single, Single, Single)  
CreateElements.CreateElement\_Dropdown(VisualElement, String, List<String>, Int32)  
CreateElements.CreateElement\_Radio(VisualElement, Boolean, String, Boolean)  
CreateElements.CreateElement\_Radio\_Group(VisualElement, String, String[])  
CreateElements.CreateElement\_ContextMenuItem(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu(VisualElement, ContextMenuItemData[])  
CreateElements.CreateElement\_ContextMenu(VisualElement, List<ContextMenuItemData>)  
CreateElements.CreateElement\_ContextMenu\_Item\_Parent(VisualElement, ContextMenuItemParentData)  
CreateElements.CreateElement\_ContextMenu\_Item(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu\_Item\_Button(VisualElement, ContextMenuItemButtonData)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, Boolean)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, VisualElement, Boolean)  
CreateElements.CreateElement\_Tooltip(VisualElement, Font, Int32)  
CreateElements.CreateElement\_Sprite(VisualElement, Sprite)  
CreateElements.CreateElement\_Slot(VisualElement, Int32)  
CreateElements.CreateElement\_Inventory(VisualElement, Int32, Int32, Inventory)  
CreateElements.CreateElement\_Icon(VisualElement, InventoryItem)  
CreateElements.CreateElement\_SlotPopup(VisualElement, Vector2, Int32, InventoryElement, Boolean)  
CreateElements.CreateElement\_TransferPopup(VisualElement, Vector2, TransferItem, InventoryElement, InventoryElement)

# Class SpriteElement

This Class is an VisualElement that is created for the Inventory Editor.

## Inheritance

System.Object  
UnityEngine.UIElements.CallbackEventHandler  
UnityEngine.UIElements.Focusable  
UnityEngine.UIElements.VisualElement  
UnityEngine.UIElements.Image  
SpriteElement

Namespace: [UtilitySystem.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class SpriteElement : Image, IEventHandler, ITransform, ITransitionAnimations, IExperimentalFeatures,  
IVisualElementScheduler, IResolvedStyle
```

## Constructors

### SpriteElement()

This is a constructor and will set the style for the SpriteElement.

#### Declaration

```
public SpriteElement()
```

### SpriteElement(Sprite, StyleSheet)

This is a constructor and will set the style for the SpriteElement.

#### Declaration

```
public SpriteElement(Sprite sprite, StyleSheet style = null)
```

#### Parameters

| TYPE                              | NAME   | DESCRIPTION |
|-----------------------------------|--------|-------------|
| UnityEngine.Sprite                | sprite |             |
| UnityEngine.UIElements.StyleSheet | style  |             |

## Methods

### RemoveSprite()

This will remove the sprite for this element.

#### Declaration

```
public void RemoveSprite()
```

### SetSprite(Sprite)

This will set the sprite for this element.

#### Declaration

```
public void SetSprite(Sprite sprite)
```

#### Parameters

| TYPE               | NAME   | DESCRIPTION |
|--------------------|--------|-------------|
| UnityEngine.Sprite | sprite |             |

## Extension Methods

[VisualElementStyleExtension.BackGround\\_Color\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, Background\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, Texture2D\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, Sprite\)](#)  
[VisualElementStyleExtension.Background\\_Image\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Color\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Color\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Top\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Top\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Right\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Right\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Bottom\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Bottom\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Left\(VisualElement, Color\)](#)  
[VisualElementStyleExtension.Border\\_Color\\_Left\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Color\(VisualElement, Color\[\]\)](#)  
[VisualElementStyleExtension.Border\\_Color\(VisualElement, StyleKeyword\[\]\)](#)  
[VisualElementStyleExtension.GetRadius\\_Top\(VisualElement\)](#)  
[VisualElementStyleExtension.GetRadius\\_Bottom\(VisualElement\)](#)  
[VisualElementStyleExtension.GetRadius\\_Size\(VisualElement\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Left\(VisualElement, Single, LengthUnit\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Left\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Right\(VisualElement, Single, LengthUnit\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Top\\_Right\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Bottom\\_Left\(VisualElement, Single, LengthUnit\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Bottom\\_Left\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Bottom\\_Right\(VisualElement, Single, LengthUnit\)](#)  
[VisualElementStyleExtension.Border\\_Radius\\_Bottom\\_Right\(VisualElement, StyleKeyword\)](#)  
[VisualElementStyleExtension.Border\\_Radius\(VisualElement, Single\[\]\)](#)  
[VisualElementStyleExtension.Border\\_Radius\(VisualElement, LengthUnit, Single\[\]\)](#)

VisualElementStyleExtension.Border\_Radius(VisualElement, LengthUnit[], Single[]])  
VisualElementStyleExtension.Border\_Radius(VisualElement, StyleKeyword[]))  
VisualElementStyleExtension.GetBorder\_Width(VisualElement)  
VisualElementStyleExtension.GetBorder\_Height(VisualElement)  
VisualElementStyleExtension.GetBorder\_Size(VisualElement)  
VisualElementStyleExtension.Border\_Width\_Top(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, Single)  
VisualElementStyleExtension.Border\_Width\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Border(VisualElement, Single[]])  
VisualElementStyleExtension.Border(VisualElement, Color, Single[]])  
VisualElementStyleExtension.Border(VisualElement, Color[], Single[]])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color, Single[]])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color[], Single[]])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color, Single[]])  
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color[], Single[]])  
VisualElementStyleExtension.Border(VisualElement, StyleKeyword[]])  
VisualElementStyleExtension.GetMargin\_Width(VisualElement)  
VisualElementStyleExtension.GetMargin\_Height(VisualElement)  
VisualElementStyleExtension.GetMargin\_Size(VisualElement)  
VisualElementStyleExtension.Margin\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin(VisualElement, Single[]])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit[], Single[]])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit, Single[]])  
VisualElementStyleExtension.Margin(VisualElement, StyleKeyword[]])  
VisualElementStyleExtension.GetPadding\_Width(VisualElement)  
VisualElementStyleExtension.GetPadding\_Height(VisualElement)  
VisualElementStyleExtension.GetPadding\_Size(VisualElement)  
VisualElementStyleExtension.Padding\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding(VisualElement, Single[]])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit, Single[]])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit[], Single[]])  
VisualElementStyleExtension.Padding(VisualElement, StyleKeyword[]])  
VisualElementStyleExtension.Size\_Width(VisualElement, Single, LengthUnit)

VisualElementStyleExtension.Size\_Width(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size(VisualElement, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Min(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Min(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Max(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Max(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.SetDisplay(VisualElement, DisplayStyle)  
VisualElementStyleExtension.GetDisplay(VisualElement)  
VisualElementStyleExtension.Display\_ToggleVisibility(VisualElement)  
VisualElementStyleExtension.Display\_Hide(VisualElement)  
VisualElementStyleExtension.Display\_Show(VisualElement)  
VisualElementStyleExtension.Display\_IsInvisible(VisualElement)  
VisualElementStyleExtension.Display\_IsVisible(VisualElement)  
VisualElementStyleExtension.StylePosition(VisualElement, Position)  
VisualElementStyleExtension.StylePosition(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Toggle(VisualElement)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Vector2)

VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Align\_Content(VisualElement, Align)  
VisualElementStyleExtension.Align\_Content(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Item(VisualElement, Align)  
VisualElementStyleExtension.Align\_Item(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Self(VisualElement, Align)  
VisualElementStyleExtension.Align\_Self(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, Wrap)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, FlexDirection)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, StyleKeyword)  
VisualElementStyleExtension.White\_Space(VisualElement,WhiteSpace)  
VisualElementStyleExtension.White\_Space(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font\_Size(VisualElement, Single)  
VisualElementStyleExtension.Font\_Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font(VisualElement, Font)  
VisualElementStyleExtension.Font(VisualElement, FontAsset)  
VisualElementStyleExtension.Text\_Position(VisualElement, TextAnchor)  
VisualElementStyleExtension.Overflow(VisualElement, Overflow)  
VisualElementStyleExtension.Overflow(VisualElement, StyleKeyword)  
VisualElementExtension.ViewDataKey(VisualElement, String)  
VisualElementExtension.UsageHints(VisualElement, UsageHints)  
VisualElementExtension.ToolTip(VisualElement, String)  
VisualElementExtension.GetLayout(VisualElement)  
VisualElementExtension.GetLayoutSize(VisualElement)  
VisualElementExtension.GetLayoutWidth(VisualElement)  
VisualElementExtension.GetLayoutHeight(VisualElement)  
VisualElementExtension.GetLayoutPosition(VisualElement)  
VisualElementExtension.GetLayoutPositionX(VisualElement)  
VisualElementExtension.GetLayoutPositionY(VisualElement)  
VisualElementExtension.GetWorldBound(VisualElement)  
VisualElementExtension.GetWorldBoundSize(VisualElement)  
VisualElementExtension.GetWorldBoundPosition(VisualElement)  
VisualElementExtension.GetLocalBound(VisualElement)  
VisualElementExtension.GetLocalBoundSize(VisualElement)  
VisualElementExtension.GetLocalBoundPosition(VisualElement)  
VisualElementExtension.Pick(VisualElement, PickingMode)  
VisualElementExtension.Pick\_Toggle(VisualElement)  
VisualElementExtension.Pick\_OFF(VisualElement)  
VisualElementExtension.Pick\_ON(VisualElement)  
VisualElementExtension.Focus(VisualElement, Boolean)  
VisualElementExtension.Focus\_Toggle(VisualElement)  
VisualElementExtension.Focus\_OFF(VisualElement)  
VisualElementExtension.Focus\_ON(VisualElement)

```
VisualElementExtension.SetVisible(VisualElement, Boolean)
VisualElementExtension.GetVisible(VisualElement)
VisualElementExtension.Visible_ToggleVisibility(VisualElement)
VisualElementExtension.Visible_Hide(VisualElement)
VisualElementExtension.Visible_Show(VisualElement)
VisualElementExtension.Visible_IsInvisible(VisualElement)
VisualElementExtension.Visible_IsVisible(VisualElement)
VisualElementExtension.ResetRotation(VisualElement)
VisualElementExtension.SetTransform_Rotate(VisualElement, Vector3)
VisualElementExtension.GetTransform_Rotate(VisualElement)
VisualElementExtension.Rotate_Clockwise(VisualElement, Vector3)
VisualElementExtension.Rotate_Counter_Clockwise(VisualElement, Vector3)
VisualElementExtension.Rotate_Clockwise_X(VisualElement, Single)
VisualElementExtension.Rotate_Counter_Clockwise_X(VisualElement, Single)
VisualElementExtension.RotateY_Clockwise_Y(VisualElement, Single)
VisualElementExtension.Rotate_Counter_Clockwise_Y(VisualElement, Single)
VisualElementExtension.Rotate_Clockwise_Z(VisualElement, Single)
VisualElementExtension.Rotate_Counter_Clockwise_Z(VisualElement, Single)
VisualElementExtension.SetTransform_Position(VisualElement, Vector2)
VisualElementExtension.GetTransform_Position(VisualElement)
VisualElementExtension.SetTransform_Scale(VisualElement, Vector3)
VisualElementExtension.GetTransform_Scale(VisualElement)
VisualElementExtension.Child(VisualElement, VisualElement, Boolean)
VisualElementExtension.StyleSheet(VisualElement, StyleSheet, Boolean)
VisualElementExtension.Name(VisualElement, String)
VisualElementExtension.GetElementSize(VisualElement)
VisualElementExtension.ScreenBound(VisualElement)
VisualElementExtension.MouseGrab(VisualElement, Vector2)
VisualElementExtension.MouseRelease(VisualElement)
VisualElementExtension.UpdatePosition(VisualElement, Vector2)
RichText.UpperCaseText(Object)
RichText.LowerCaseText(Object)
RichText.BoldText(Object)
RichText.ItalicText(Object)
RichText.Text(Object)
RichText.UnderLineText(Object)
RichText.StrikeText(Object)
RichText.SupText(Object)
RichText.SubText(Object)
RichText.PositionText(Object, Single)
RichText.SizeType(Object, Int32)
RichText.ColoredText(Object, String)
RichText.ColoredText(Object, Color)
RichText.BooleanText(Object, Boolean)
RichText.BooleanText(Object, Boolean, Color)
CreateElements.CreateElement_Label(VisualElement, String, Boolean)
CreateElements.CreateElement_VisualElement(VisualElement)
CreateElements.CreateElement_ScrollView(VisualElement, ScrollViewMode, VisualElement[])
CreateElements.CreateElement_ListView< TValue >(VisualElement, List< TValue >, Single, SelectionType)
CreateElements.CreateElement_Button(VisualElement, Action)
CreateElements.CreateElement_Toggle(VisualElement, String)
CreateElements.CreateElement_TextField(VisualElement, String, Int32, Boolean, Boolean, Char)
CreateElements.CreateElement_Foldout(VisualElement, String, Boolean)
CreateElements.CreateElement_Slider(VisualElement, Single, Single, String)
```

```
CreateElements.CreateElement_Slider_Int(VisualElement, Int32, Int32, String)
CreateElements.CreateElement_Slider_Min_Max(VisualElement, String, Single, Single, Single, Single)
CreateElements.CreateElement_Progress_Bar(VisualElement, String, Single, Single, Single)
CreateElements.CreateElement_Dropdown(VisualElement, String, List<String>, Int32)
CreateElements.CreateElement_Radio(VisualElement, Boolean, String, Boolean)
CreateElements.CreateElement_Radio_Group(VisualElement, String, String[])
CreateElements.CreateElement_ContextMenuItem(VisualElement, ContextMenuItemData)
CreateElements.CreateElement_ContextMenu(VisualElement, ContextMenuItemData[])
CreateElements.CreateElement_ContextMenu(VisualElement, List<ContextMenuItemData>)
CreateElements.CreateElement_ContextMenu_Item_Parent(VisualElement, ContextMenuItemParentData)
CreateElements.CreateElement_ContextMenu_Item(VisualElement, ContextMenuItemData)
CreateElements.CreateElement_ContextMenu_Item_Button(VisualElement, ContextMenuItemButtonData)
CreateElements.CreateElement_DialogBox(VisualElement, String, String, Boolean)
CreateElements.CreateElement_DialogBox(VisualElement, String, String, VisualElement, Boolean)
CreateElements.CreateElement_Tooltip(VisualElement, Font, Int32)
CreateElements.CreateElement_Sprite(VisualElement, Sprite)
CreateElements.CreateElement_Slot(VisualElement, Int32)
CreateElements.CreateElement_Inventory(VisualElement, Int32, Int32, Inventory)
CreateElements.CreateElement_Icon(VisualElement, InventoryItem)
CreateElements.CreateElement_SlotPopup(VisualElement, Vector2, Int32, InventoryElement, Boolean)
CreateElements.CreateElement_TransferPopup(VisualElement, Vector2, TransferItem, InventoryElement, InventoryElement)
```

# Class SpriteElement.UxmlFactory

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

## Inheritance

System.Object

UnityEngine.UIElements.UxmlFactory<[SpriteElement](#), UnityEngine.UIElements.Image.UxmlTraits>

SpriteElement.UxmlFactory

Namespace: [UtilitySystem.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class UxmlFactory : UxmlFactory<SpriteElement, Image.UxmlTraits>, IUxmlFactory
```

## Extension Methods

[RichText.UpperCaseText\(Object\)](#)

[RichText.LowerCaseText\(Object\)](#)

[RichText.BoldText\(Object\)](#)

[RichText.ItalicText\(Object\)](#)

[RichText.Text\(Object\)](#)

[RichText.UnderLineText\(Object\)](#)

[RichText.StrikeText\(Object\)](#)

[RichText.SupText\(Object\)](#)

[RichText.SubText\(Object\)](#)

[RichText.PositionText\(Object, Single\)](#)

[RichText.SizeType\(Object, Int32\)](#)

[RichText.ColoredText\(Object, String\)](#)

[RichText.ColoredText\(Object, Color\)](#)

[RichText.BooleanText\(Object, Boolean\)](#)

[RichText.BooleanText\(Object, Boolean, Color\)](#)

# Class ToolTipElement

## Inheritance

System.Object  
UnityEngine.UIElements.CallbackEventHandler  
UnityEngine.UIElements.Focusable  
UnityEngine.UIElements.VisualElement  
ToolTipElement

Namespace: [UtilitySystem.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class ToolTipElement : VisualElement, IEventHandler, ITransform, ITransitionAnimations,  
IExperimentalFeatures, IVisualElementScheduler, IResolvedStyle
```

## Constructors

### ToolTipElement()

This is a constructor for the tool tip.

#### Declaration

```
public ToolTipElement()
```

### ToolTipElement(Font, Int32, StyleSheet)

This is a constructor for the tool tip.

#### Declaration

```
public ToolTipElement(Font font = null, int textSize = 0, StyleSheet style = null)
```

## Parameters

| Type                              | Name     | Description  |
|-----------------------------------|----------|--|
| UnityEngine.Font                  | font     | Is an optional font that will be added to the label. |
| System.Int32                      | textSize | Is an optional size of the text in the label.        |
| UnityEngine.UIElements.StyleSheet | style    | Is an optional stylesheet for the element.           |

## Methods

## AddText(List<Func<String>>)

---

This will add a new message to the Func string list.

### Declaration

```
public void AddText(List<Func<string>> messages)
```

### Parameters

| TYPE  | NAME     | DESCRIPTION  |
|---|----------|--|
| System.Collections.Generic.List<System.Func<System.String>> | messages | The list of func strings that will be added to the func string list for the message. |

## AddText(Func<String>)

---

This will add a new message to the Func string list.

### Declaration

```
public void AddText(Func<string> message)
```

### Parameters

| TYPE                       | NAME    | DESCRIPTION   |
|----------------------------|---------|---|
| System.Func<System.String> | message | The func string that will be added to the func string list for the message. |

## AddText(String)

---

This will add a new message to the Func string list.

### Declaration

```
public void AddText(string message)
```

### Parameters

| TYPE          | NAME    | DESCRIPTION  |
|---------------|---------|--|
| System.String | message | The string that will be added to the func string list for the message. |

## RemoveText()

---

this will clear the message.

## Declaration

```
public void RemoveText()
```

## Style()

This will set the style of the Icon.

## Declaration

```
public void Style()
```

## TextSize(Int32)

This will allow for the text size to be changed.

## Declaration

```
public void TextSize(int textSize)
```

## Parameters

| TYPE         | NAME     | DESCRIPTION                           |
|--------------|----------|---------------------------------------|
| System.Int32 | textSize | the int used to update the text size. |

## UpdatePosition(Vector2)

This will update the position of the element on screen and maintain its bound within the screens bounds.

## Declaration

```
public void UpdatePosition(Vector2 mousePosition)
```

## Parameters

| TYPE                | NAME          | DESCRIPTION   |
|---------------------|---------------|---|
| UnityEngine.Vector2 | mousePosition | This is the vector2 position it is trying to move to. |

## UpdateText()

This will update the message that is displayed.

## Declaration

```
public void UpdateText()
```

## Extension Methods

```
VisualElementStyleExtension.BackGround_Color(VisualElement, Color)
VisualElementStyleExtension.Background_Image(VisualElement, Background)
VisualElementStyleExtension.Background_Image(VisualElement, Texture2D)
VisualElementStyleExtension.Background_Image(VisualElement, Sprite)
VisualElementStyleExtension.Background_Image(VisualElement, StyleKeyword)
VisualElementStyleExtension.Color(VisualElement, Color)
VisualElementStyleExtension.Color(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Color_Top(VisualElement, Color)
VisualElementStyleExtension.Border_Color_Top(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Color_Right(VisualElement, Color)
VisualElementStyleExtension.Border_Color_Right(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Color_Bottom(VisualElement, Color)
VisualElementStyleExtension.Border_Color_Bottom(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Color_Left(VisualElement, Color)
VisualElementStyleExtension.Border_Color_Left(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Color(VisualElement, Color[])
VisualElementStyleExtension.Border_Color(VisualElement, StyleKeyword[])
VisualElementStyleExtension.GetRadius_Top(VisualElement)
VisualElementStyleExtension.GetRadius_Bottom(VisualElement)
VisualElementStyleExtension.GetRadius_Size(VisualElement)
VisualElementStyleExtension.Border_Radius_Top_Left(VisualElement, Single, LengthUnit)
VisualElementStyleExtension.Border_Radius_Top_Left(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Radius_Top_Right(VisualElement, Single, LengthUnit)
VisualElementStyleExtension.Border_Radius_Top_Right(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Radius_Bottom_Left(VisualElement, Single, LengthUnit)
VisualElementStyleExtension.Border_Radius_Bottom_Left(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Radius_Bottom_Right(VisualElement, Single, LengthUnit)
VisualElementStyleExtension.Border_Radius_Bottom_Right(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Radius(VisualElement, Single[])
VisualElementStyleExtension.Border_Radius(VisualElement, LengthUnit, Single[])
VisualElementStyleExtension.Border_Radius(VisualElement, LengthUnit[], Single[])
VisualElementStyleExtension.Border_Radius(VisualElement, StyleKeyword[])
VisualElementStyleExtension.GetBorder_Width(VisualElement)
VisualElementStyleExtension.GetBorder_Height(VisualElement)
VisualElementStyleExtension.GetBorder_Size(VisualElement)
VisualElementStyleExtension.Border_Width_Top(VisualElement, Single)
VisualElementStyleExtension.Border_Width_Top(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Width_Right(VisualElement, Single)
VisualElementStyleExtension.Border_Width_Right(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Width_Bottom(VisualElement, Single)
VisualElementStyleExtension.Border_Width_Bottom(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Width_Left(VisualElement, Single)
VisualElementStyleExtension.Border_Width_Left(VisualElement, StyleKeyword)
VisualElementStyleExtension.Border_Width(VisualElement, Single[])
VisualElementStyleExtension.Border_Width(VisualElement, StyleKeyword[])
VisualElementStyleExtension.Border(VisualElement, Single[])
VisualElementStyleExtension.Border(VisualElement, Color, Single[])
VisualElementStyleExtension.Border(VisualElement, Color[], Single[])
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color, Single[])
VisualElementStyleExtension.Border(VisualElement, LengthUnit, Color[], Single[])
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color, Single[])
VisualElementStyleExtension.Border(VisualElement, LengthUnit[], Color[], Single[])
```

VisualElementStyleExtension.Border(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetMargin\_Width(VisualElement)  
VisualElementStyleExtension.GetMargin\_Height(VisualElement)  
VisualElementStyleExtension.GetMargin\_Size(VisualElement)  
VisualElementStyleExtension.Margin\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Margin\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Margin(VisualElement, Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Margin(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Margin(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.GetPadding\_Width(VisualElement)  
VisualElementStyleExtension.GetPadding\_Height(VisualElement)  
VisualElementStyleExtension.GetPadding\_Size(VisualElement)  
VisualElementStyleExtension.Padding\_Top(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Right(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding\_Left(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Padding\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Padding(VisualElement, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Padding(VisualElement, LengthUnit[], Single[])  
VisualElementStyleExtension.Padding(VisualElement, StyleKeyword[])  
VisualElementStyleExtension.Size\_Width(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size(VisualElement, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Height\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Min(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Min(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Min(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Min(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, Single, LengthUnit)  
VisualElementStyleExtension.Size\_Width\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Height\_Max(VisualElement, Single, LengthUnit)

VisualElementStyleExtension.Size\_Height\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Size\_Max(VisualElement, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, LengthUnit, LengthUnit, Single[])  
VisualElementStyleExtension.Size\_Max(VisualElement, Vector2, LengthUnit[])  
VisualElementStyleExtension.Size\_Max(VisualElement, StyleKeyword)  
VisualElementStyleExtension.GetSize\_Max(VisualElement, Boolean, Boolean)  
VisualElementStyleExtension.SetDisplay(VisualElement, DisplayStyle)  
VisualElementStyleExtension.GetDisplay(VisualElement)  
VisualElementStyleExtension.Display\_ToggleVisibility(VisualElement)  
VisualElementStyleExtension.Display\_Hide(VisualElement)  
VisualElementStyleExtension.Display\_Show(VisualElement)  
VisualElementStyleExtension.Display\_IsInvisible(VisualElement)  
VisualElementStyleExtension.Display\_IsVisible(VisualElement)  
VisualElementStyleExtension.StylePosition(VisualElement, Position)  
VisualElementStyleExtension.StylePosition(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Toggle(VisualElement)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Top(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Right(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Bottom(VisualElement, StyleKeyword)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, Single)  
VisualElementStyleExtension.StylePosition\_Left(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Top\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Left(VisualElement, Single, Single)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Vector2)  
VisualElementStyleExtension.Position\_Bottom\_Right(VisualElement, Single, Single)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Top\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Left(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Anchored\_Position\_Bottom\_Right(VisualElement, UIAnchorEnum)  
VisualElementStyleExtension.Align\_Content(VisualElement, Align)  
VisualElementStyleExtension.Align\_Content(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Item(VisualElement, Align)  
VisualElementStyleExtension.Align\_Item(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Align\_Self(VisualElement, Align)  
VisualElementStyleExtension.Align\_Self(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, Wrap)  
VisualElementStyleExtension.Flex\_Wrap(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Shrink(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Grow(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, FlexDirection)  
VisualElementStyleExtension.Flex\_Direction(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, Single)  
VisualElementStyleExtension.Flex\_Basis(VisualElement, StyleKeyword)  
VisualElementStyleExtension.White\_Space(VisualElement,WhiteSpace)

VisualElementStyleExtension.White\_Space(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font\_Size(VisualElement, Single)  
VisualElementStyleExtension.Font\_Size(VisualElement, StyleKeyword)  
VisualElementStyleExtension.Font(VisualElement, Font)  
VisualElementStyleExtension.Font(VisualElement, FontAsset)  
VisualElementStyleExtension.Text\_Position(VisualElement, TextAnchor)  
VisualElementStyleExtension.Overflow(VisualElement, Overflow)  
VisualElementStyleExtension.Overflow(VisualElement, StyleKeyword)  
VisualElementExtension.ViewDataKey(VisualElement, String)  
VisualElementExtension.UsageHints(VisualElement, UsageHints)  
VisualElementExtension.ToolTip(VisualElement, String)  
VisualElementExtension.GetLayout(VisualElement)  
VisualElementExtension.GetLayoutSize(VisualElement)  
VisualElementExtension.GetLayoutWidth(VisualElement)  
VisualElementExtension.GetLayoutHeight(VisualElement)  
VisualElementExtension.GetLayoutPosition(VisualElement)  
VisualElementExtension.GetLayoutPositionX(VisualElement)  
VisualElementExtension.GetLayoutPositionY(VisualElement)  
VisualElementExtension.GetWorldBound(VisualElement)  
VisualElementExtension.GetWorldBoundSize(VisualElement)  
VisualElementExtension.GetWorldBoundPosition(VisualElement)  
VisualElementExtension.GetLocalBound(VisualElement)  
VisualElementExtension.GetLocalBoundSize(VisualElement)  
VisualElementExtension.GetLocalBoundPosition(VisualElement)  
VisualElementExtension.Pick(VisualElement, PickingMode)  
VisualElementExtension.Pick\_Toggle(VisualElement)  
VisualElementExtension.Pick\_OFF(VisualElement)  
VisualElementExtension.Pick\_ON(VisualElement)  
VisualElementExtension.Focus(VisualElement, Boolean)  
VisualElementExtension.Focus\_Toggle(VisualElement)  
VisualElementExtension.Focus\_OFF(VisualElement)  
VisualElementExtension.Focus\_ON(VisualElement)  
VisualElementExtension.SetVisible(VisualElement, Boolean)  
VisualElementExtension.GetVisible(VisualElement)  
VisualElementExtension.Visible\_ToggleVisibility(VisualElement)  
VisualElementExtension.Visible\_Hide(VisualElement)  
VisualElementExtension.Visible\_Show(VisualElement)  
VisualElementExtension.Visible\_IsInvisible(VisualElement)  
VisualElementExtension.Visible\_IsVisible(VisualElement)  
VisualElementExtension.ResetRotation(VisualElement)  
VisualElementExtension.SetTransform\_Rotate(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Rotate(VisualElement)  
VisualElementExtension.Rotate\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Counter\_Clockwise(VisualElement, Vector3)  
VisualElementExtension.Rotate\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_X(VisualElement, Single)  
VisualElementExtension.RotateY\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Y(VisualElement, Single)  
VisualElementExtension.Rotate\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.Rotate\_Counter\_Clockwise\_Z(VisualElement, Single)  
VisualElementExtension.SetTransform\_Position(VisualElement, Vector2)  
VisualElementExtension.GetTransform\_Position(VisualElement)  
VisualElementExtension.SetTransform\_Scale(VisualElement, Vector3)  
VisualElementExtension.GetTransform\_Scale(VisualElement)

VisualElementExtension.Child(VisualElement, VisualElement, Boolean)  
VisualElementExtension.StyleSheet(VisualElement, StyleSheet, Boolean)  
VisualElementExtension.Name(VisualElement, String)  
VisualElementExtension.GetElementSize(VisualElement)  
VisualElementExtension.ScreenBound(VisualElement)  
VisualElementExtension.MouseGrab(VisualElement, Vector2)  
VisualElementExtension.MouseRelease(VisualElement)  
VisualElementExtension.UpdatePosition(VisualElement, Vector2)  
RichText.UpperCaseText(Object)  
RichTextLowerCaseText(Object)  
RichText.BoldText(Object)  
RichText.ItalicText(Object)  
RichText.Text(Object)  
RichText.UnderLineText(Object)  
RichText.StrikeText(Object)  
RichText.SupText(Object)  
RichText.SubText(Object)  
RichText.PositionText(Object, Single)  
RichText.SizeTypeText(Object, Int32)  
RichText.ColoredText(Object, String)  
RichText.ColoredText(Object, Color)  
RichText.BooleanText(Object, Boolean)  
RichText.BooleanText(Object, Boolean, Color)  
CreateElements.CreateElement\_Label(VisualElement, String, Boolean)  
CreateElements.CreateElement\_VisualElement(VisualElement)  
CreateElements.CreateElement\_ScrollView(VisualElement, ScrollViewMode, VisualElement[])  
CreateElements.CreateElement\_ListView< TValue >(VisualElement, List< TValue >, Single, SelectionType)  
CreateElements.CreateElement\_Button(VisualElement, Action)  
CreateElements.CreateElement\_Toggle(VisualElement, String)  
CreateElements.CreateElement\_TextField(VisualElement, String, Int32, Boolean, Boolean, Char)  
CreateElements.CreateElement\_Foldout(VisualElement, String, Boolean)  
CreateElements.CreateElement\_Slider(VisualElement, Single, Single, String)  
CreateElements.CreateElement\_Slider\_Int(VisualElement, Int32, Int32, String)  
CreateElements.CreateElement\_Slider\_Min\_Max(VisualElement, String, Single, Single, Single, Single)  
CreateElements.CreateElement\_Progress\_Bar(VisualElement, String, Single, Single, Single)  
CreateElements.CreateElement\_Dropdown(VisualElement, String, List< String >, Int32)  
CreateElements.CreateElement\_Radio(VisualElement, Boolean, String, Boolean)  
CreateElements.CreateElement\_Radio\_Group(VisualElement, String, String[])  
CreateElements.CreateElement\_ContextMenuItem(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu(VisualElement, ContextMenuItemData[])  
CreateElements.CreateElement\_ContextMenu(VisualElement, List< ContextMenuItemData >)  
CreateElements.CreateElement\_ContextMenu\_Item\_Parent(VisualElement, ContextMenuItemParentData)  
CreateElements.CreateElement\_ContextMenu\_Item(VisualElement, ContextMenuItemData)  
CreateElements.CreateElement\_ContextMenu\_Item\_Button(VisualElement, ContextMenuItemButtonData)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, Boolean)  
CreateElements.CreateElement\_DialogBox(VisualElement, String, String, VisualElement, Boolean)  
CreateElements.CreateElement\_Tooltip(VisualElement, Font, Int32)  
CreateElements.CreateElement\_Sprite(VisualElement, Sprite)  
CreateElements.CreateElement\_Slot(VisualElement, Int32)  
CreateElements.CreateElement\_Inventory(VisualElement, Int32, Int32, Inventory)  
CreateElements.CreateElement\_Icon(VisualElement, InventoryItem)  
CreateElements.CreateElement\_SlotPopup(VisualElement, Vector2, Int32, InventoryElement, Boolean)  
CreateElements.CreateElement\_TransferPopup(VisualElement, Vector2, TransferItem, InventoryElement, InventoryElement)

# Class ToolTipElement.UxmlFactory

This new class is UxmlFactory and is needed to create a uxml tag for the UIDocument to read.

## Inheritance

System.Object

UnityEngine.UIElements.UxmlFactory<ToolTipElement, UnityEngine.UIElements.VisualElement.UxmlTraits>

ToolTipElement.UxmlFactory

Namespace: [Utility System.UI](#)

Assembly: Assembly-CSharp.dll

## Syntax

```
public class UxmlFactory : UxmlFactory<ToolTipElement, VisualElement.UxmlTraits>, IUxmlFactory
```

## Extension Methods

[RichText.UpperCaseText\(Object\)](#)

[RichText.LowerCaseText\(Object\)](#)

[RichText.BoldText\(Object\)](#)

[RichText.ItalicText\(Object\)](#)

[RichText.Text\(Object\)](#)

[RichText.UnderLineText\(Object\)](#)

[RichText.StrikeText\(Object\)](#)

[RichText.SupText\(Object\)](#)

[RichText.SubText\(Object\)](#)

[RichText.PositionText\(Object, Single\)](#)

[RichText.SizeType\(Object, Int32\)](#)

[RichText.ColoredText\(Object, String\)](#)

[RichText.ColoredText\(Object, Color\)](#)

[RichText.BooleanText\(Object, Boolean\)](#)

[RichText.BooleanText\(Object, Boolean, Color\)](#)