comma after class not corresponding

Line 9: The properties specific to each individual visual are encapsulated in the corresponding class, such as a `Border` visual is defined in a `BorderVisual` class, a `Color` visual in a `ColorVisual` class, and so on. The properties that are common for all visual types are inherited from the [Tizen.NUI.VisualMap](https://samsung.github.io/TizenFX/latest/api/Tizen.NUI.VisualMap.html) class. To render a visual, it has to be added to a control. A container class [Tizen.NUI.BaseComponents.VisualView](https://samsung.github.io/TizenFX/latest/api/Tizen.NUI.BaseComponents.VisualView.html) controls any visual that you have added.

Line 20: 2. Specify the properties required for the visual type. The following section provides detailed information about the properties that are mandatory for a given visual:

Can we write like this?

2. Specify the properties required for the visual type. For detailed information about the properties that are mandatory for a given visual, see [properties](#..):

Line 127- 128

| `Origin` | `Visual.AlignType` | `Center` | Specifies the reference point within the parent's area. The possible values are `TopBegin`, `TopCenter`, `TopEnd`, `CenterBegin`, `Center`, `CenterEnd`, `BottomBegin`, `BottomCenter`, and `BottomEnd`. For more information, see the [explanation below](#ref\_point\_explanation). |

| `AnchorPoint` | `Visual.AlignType` | `Center` | Specifies the reference point within the visual's area. The possible values are as for the `Origin` property. For more information, see the [explanation below](#ref\_point\_explanation). |

We do not use directional language (below, above is not permitted)

Q1: see the [explanation below](#ref\_point\_explanation)

Where is this link going? I couldnt really figure it out the context here.

Q2: Can we remove below and write For more information, see [reference point](#ref\_point\_explanation)

Line 153: The position of the visual is defined as the `AnchorPoint` offset relative to the` Origin` in the reference frame, which is hooked in the upper left corner. The x and y-axis are directed to the right and to the bottom respectively. Examples of the arrangement of the visuals within the control realized in accordance with the code below as follows:

Not able to understand: Examples of the arrangement of the visuals within the control realized in accordance with the code below as follows:

Line 188: Suggestion: having same color as that of the frame

The `BorderVisual` renders a rectangular frame with a given thickness. The whole frame is plotted inside the area designated by the `BorderVisual` size as long as it is possible. When the `BorderVisual` size is 200x200 and the border width is 100, it appears as a square of the size of 200x200, having the color same as that of the border. When the `BorderVisual` size is 100x200 and the border width is 200, the frame does not fit inside the given size and it appears as a rectangle of the size of 300x200, having same color as that of the frame.

343: // Color limits in a relative coordinate system

407: The `ImageVisual` renders a raster image, such as `.jpg` or `.png`, into the control.

417: | `AuxiliaryImageURL` | `string` | No | The URL of the auxiliary image on top of a NPatch image. |

418-

419

| `FittingMode` | `FittingModeType` | No | Fitting options used when resizing the images to fit the specified dimensions. The possible values are `ShrinkToFit`, `ScaleToFill`, `FitWidth`, and `FitHeight`. |

| `SamplingMode` | `SamplingModeType` | No | Filtering options used when sampling the original pixels to resize images. The possible values are `Box`, `Nearest`, `Linear`, `BoxThenNearest`, `BoxThenLinear`, `NoFilter`, and `DontCare`. |

424: code styling on 0.0 and 0.1

| `WrapModeU`<br>`WrapModeV` | `WrapModeType` | No | Specifies the wrap mode for the U or V coordinate respectively, and how the texture should be sampled when the U or V coordinate exceeds the range of `0.0` to `1.0`. The possible values are `Default`, `ClampToEdge`, `Repeat`, and `MirroredRepeat`. |

464:The following example illustrates how to use `N-Patch`. The absolute path to the image is set as in case of the [`ImageVisual`](#imagevisual):

519:

The following example illustrates how to use the `SVGVisual`:

550:

The following example illustrates how to use the `AnimatedImageVisual`:

568-69

> [!NOTE]

> Due typographical mistake in source code, it is advised to use 'MaterialtURL' instead of 'MaterialURL' as property name.

596: The following example illustrates how to use the `MeshVisual`. The `.obj`, `.mtl` files, and directory with textures use the same path as used for the [ImageVisual](#imagevisual):

617:618:

> [!NOTE]

> Due typographical mistake in source code, it is advised to use 'ConicalFrustrum' instead of 'ConicalFrustum' as one of the 'Shape' names.

648-651,51

Do we need to keep 0 and 1 in code tag like this `0` `1`

730:

The following example illustrates how to set the `TextVisual`:

774:

Related information