


Luthetus.TextEditor.Razor
Lib.TextEditors.Models.ITextEditor
ModelApi.AddPresentationModel



```
graph LR; A["Luthetus.TextEditor.Razor<br/>Lib.TextEditors.Models.ITextEditor<br/>ModelApi.AddPresentationModel"] --> B["Luthetus.TextEditor.Razor<br/>Lib.Decorations.Models.TextEditor<br/>PresentationModel"]; B --> C["Luthetus.TextEditor.Razor<br/>Lib.Decorations.Models.TextEditor<br/>PresentationModelCalculation"]
```

The diagram illustrates a three-step process for adding a presentation model. It begins with a grey box containing the initial state: 'Luthetus.TextEditor.Razor', 'Lib.TextEditors.Models.ITextEditor', and 'ModelApi.AddPresentationModel'. A blue arrow points to a white box representing the intermediate state: 'Luthetus.TextEditor.Razor', 'Lib.Decorations.Models.TextEditor', and 'PresentationModel'. A second blue arrow points to a final white box: 'Luthetus.TextEditor.Razor', 'Lib.Decorations.Models.TextEditor', and 'PresentationModelCalculation'.

Luthetus.TextEditor.Razor
Lib.Decorations.Models.TextEditor
PresentationModel

Luthetus.TextEditor.Razor
Lib.Decorations.Models.TextEditor
PresentationModelCalculation