This short instruction describes how to add support for child Regions to your Custom Region created for SMath Studio program.

1) Mark your Custom Region class with `IRegionsContainer` interface.

2) `void OnCommandSend(string value)`

   Input **value** sent to your Region may contain something like **\[REGION[999999]]**. If this happened, then new child Region is added.
   Store it as a string in your data model like any other text entry supported by your plug-in. Normally no any additional handling required on this step.

3) `void PrePainter() and/or void PrePainter(IGraphics e)`

   During work with your data apply the following logic for all strings that may contain patterns like **\[REGION[999999]]**:

```
List<string> list = TermsConverter.GetManager(this.SessionProfile, false).GetList(this.text, out _);
```

   This will return parsed list of text entries including children Regions. If any of items in list contains more than 1 character it means that it is a Field or child Region. In this case use the following logic to detect how to handle current entry:

```
var encodedText = list[i];

int index;
Size size;
float middleLine;

if (base.Children.TryResolveRef(encodedText, out index, out size, out middleLine))
{
    // write your logic to handle child region here

    base.Children.UpdateByIndex(index, location);
    list[i] = this.Children.GetRef(regionIdx);
    regionIdx++;
}
else
{
    var decodedText = TermsConverter.DecodeText(encodedText, this.CurrentMetaContext, ref containsFields);

    // write logic to handle Field entry here
}
```

   Please note that list[i] = this.Children.GetRef(regionIdx); line is required and **regionIdx** must be a real index (in Regions appearing sort order, starting from 0) of the Region entry detected during work in **PrePainter** method and it may not be equal to **index** variable.
   Finally use String.Join(String.Empty, list.ToArray()); to update handled string in your data.

4) `bool Copy(IClipboardManager clipboard)`

   In order to perform copy operation it is required to replace internal references for child Regions with external references in text entries of your data. Absolutely the same approach can be used as described for void PrePainter() and/or void PrePainter(IGraphics e), but you do not need to handle Fields at all and instead of list[i] = base.Children.GetRef(regionIdx); you need to do the following:

```
string externalRef;
if (base.Children.TryGetExternalRef(list[i], out externalRef))
    list[i] = externalRef;
```

Normally you should not be interested to work with child Regions directly (however it is possible using `base.Children`), all you need is to specify sorting order and locations for child Regions and generate external references on data copy as described in instruction above.

Everything else will be handled by SMath Studio itself, f.e: child Regions rendering, interaction with mouse and keyboard, their context menu, save/open to Worksheet files, history for undo/redo operations, calculations for evaluable Regions, Dynamic Assistance and calculation errors.

Happy codding! 😊