Style Type

There are two style types in MigraDoc, character style and paragraph style.

Character Style

A character style can only be derived from another character style or from the 'DefaultParagraphFont' style. It controls only the font attributes of the text and is usually used to emphasize short parts of text within a paragraph.
Note that all parts not defined in the style definitions stay unchanged to the font used before.

```
\styles
{
  MyEmphasizeStyle : DefaultParagraphFont
  {
    Font.Underline = Dashes
    Font.Italic = True
  }
}
…
\paragraph[Format.Font.Color = blue]
{
  Use of the character style \font("MyEmphasizeStyle"){'MyEmphasizeStyle'}.
}
```

Use of the character style *'MyEmphasizeStyle'*.

If a character style is assigned to the style of a paragraph, only the font formatting of the paragraph will be changed. All further attributes are taken from the 'Normal' style, if not defined in the given paragraph's format itself.

Note that this procedure does **not** meet the intention of character styles!

```
\paragraph [Style = "MyEmphasizeStyle"]
{
  This paragraph uses 'MyEmphasizeStyle'.
}
```

This paragraph uses 'MyEmphasizeStyle'.

Paragraph Style

A paragraph style controls the whole formatting of a paragraph, that means both the layout of the paragraph including borders and shading, and the font of the text. It is intended to be assigned to paragraphs. A paragraph style can only be derived from another paragraph style or from the Normal style.

```
\styles
{
  MyParagraphStyle : Normal
  {
    Font.Name = "Courier New"
    ParagraphFormat.Borders.Top.Style = Dot
  }
}
…
\paragraph[Style = "MyParagraphStyle"]
{
  Use of the paragraph style 'MyParagraphStyle'.
}
```

`Use of the paragraph style 'MyParagraphStyle'.`

If a paragraph style is used within the \font keyword, only the style's font part is used. The layout stays unchanged.

Note that this procedure does **not** meet the intention of paragraph styles!

```
\paragraph[Format.Font.Color = blue]
{
  Use of the paragraph style \font("MyParagraphStyle"){'MyParagraphStyle'}
  within a text.
}
```

Use of the paragraph style `'MyParagraphStyle'` within a text.

Modifying Built-in Styles

Almost all of the built-in styles in MigraDoc can be modified to fit the needs of the document. The only exception is the character style 'DefaultParagraphFont'. It has no attributes set and must be kept as it is, it can serve as base for any character style.

Modifying the Style's Attributes

All of the paragraph styles in MigraDoc are derived from the 'Normal' style which uses the font 'Verdana' with a size of 10 pt. For example, if the document you create uses Arial with a size of 8 pt, you can simply change the font name und size attributes of the 'Normal' style to change the default font formatting of the whole document.

```
\styles
{
  Normal
  {
    Format.Font.Name = "Arial"
    Format.Font.Size = 8
  }
}
```

Modifying the Style's Base Style

If you want to keep the default formatting of the document but the headings should use a different font, you can do it in two ways. Either you can change the font of Heading1 which will change the font of all the headings. Or you can create a new style which serves as a default style for the headings, and then you change the base style of Heading1.

```
\styles
{
  StandardHeading
  {
    Format.Font.Name = "Times New Roman"
    Format.Font.Size = 16
  }

  Heading1 : StandardHeading
}
```

Creating new Styles

Complexer documents may need more styles than the built-in styles. In this case, you can define new styles that will suit the document's needs.

New Style

A style in MigraDoc has to be derived from a base style, otherwise the style will not be created and all texts that are defined to use this style will use InvalidStyleName, instead. The base style that is used for the new style must be defined before the new style, otherwise InvalidStyleName will be used as the base style, instead. Style names can contain blanks and special characters. If they do, the name must be surrounded by quotation marks.

```
\styles
{
  MyStyle1 : Normal
  {
    ParagraphFormat.Font.Name = "Times New Roman"
    ParagraphFormat.Font.Size = 12
  }

  MyStyle2 : MyStyle1
  {
    ParagraphFormat.Font.Bold = true
  }

  MyStyle3 : MyStyle2
  {
    ParagraphFormat.Font.Italic = true
  }
}
…
\paragraph [Style = "MyStyle1"]
{
  This is the MyStyle1 style.
}
\paragraph [Style = "MyStyle2"]
{
  This is the MyStyle2 style.
}
\paragraph [Style = "MyStyle3"]
{
  This is the MyStyle3 style.
}
```

This is the MyStyle1 style.

This is the MyStyle2 style.

This is the MyStyle3 style.

Style Name

Style names can contain blanks and special characters. If they do, the name must be surrounded by quotation marks.

```
\styles
{
  "Style name with blanks" : Normal
  {
    ParagraphFormat.Font.Color = Blue
  }

  "Dollar $ Style" : Normal
  {
    ParagraphFormat.Font.Color = Brown
  }
}
…
\paragraph [Style = "Style name with blanks"]
{
  This is the Style name with blanks style.
}
\paragraph [Style = "Dollar $ Style"]
{
  This is the Dollar $ Style style.
}
```

This is the Style name with blanks style.
This is the Dollar $ Style style.

Inheritance Behavior of Styles

A style contains several formatting definitions, both for layout formatting and text formatting. The layout formatting is only available for the paragraph styles. All of these formatting definitions are inherited if a style is derived from another style. That means if a new style is created without any additional formatting definition, then the new style will be identical with its base style except for the style name. If the derived style has any format defined, then the defined format will be used instead of the inherited format.

Font

The font is the only style format that is available for both, character and paragraph style, and defines how a text is formatted. The usage of each attribut of the font format is explained in the 'Paragraph-Font' sample.

```
\styles
{
  ParagraphStyle1_1 : Normal
  {
    Font.Name = "Arial"
    Font.Size = 8
    Font.Bold = true
    Font.Italic = false
    Font.Underline = Single
    Font.Color = Green
    Font.Subscript = false
    Font.Superscript = false
  }
  ParagraphStyle1_2 : ParagraphStyle1_1
  {
    Font.Bold = false
    Font.Italic = true
    Font.Color = Gold
  }
}
…
\paragraph [Style = "ParagraphStyle1_1"]
{
  This is ParagraphStyle1_1 style.
}
\paragraph [Style = "ParagraphStyle1_2"]
{
  This is ParagraphStyle1_2 style.
}
```

This is ParagraphStyle1_1 style.
This is ParagraphStyle1_2 style.

ParagraphFormat

ParagraphFormat is only available for the paragraph style and defines the layout of a paragraph. A detailed explanation of the attributes in ParagraphFormat is available in the 'Paragraph-Layout' sample.

```
\styles
{
  ParagraphStyle2 : Normal
  {
    ParagraphFormat.Alignment = Right
    ParagraphFormat.LeftIndent = 0
    ParagraphFormat.RightIndent = 0
    ParagraphFormat.FirstLineIndent = 0
    ParagraphFormat.LineSpacingRule = Multiple
    ParagraphFormat.LineSpacing = 1
    ParagraphFormat.OutLineLevel = Level1
    ParagraphFormat.PageBreakBefore = false
    ParagraphFormat.SpaceAfter = 0
    ParagraphFormat.SpaceBefore = 0
    ParagraphFormat.KeepTogether = false
    ParagraphFormat.KeepWithNext = false
    ParagraphFormat.WidowControl = false
  }
  ParagraphStyle2_2 : ParagraphStyle2_1
  {
    ParagraphFormat.Alignment = Center
  }
}
…
\paragraph [Style = "ParagraphStyle2_1"]
{
  This is ParagraphStyle2_1 style.
}
\paragraph [Style = "ParagraphStyle2_2"]
{
  This is ParagraphStyle2_2 style.
}
```

This is ParagraphStyle2_1 style.

This is ParagraphStyle2_2 style.

TabStops

TabStops defines where each tabulator stops and their behavior. TabStops are only available for paragraph style and explained in the 'Paragraph-Tabs' sample.

The following example demonstrates how one tabstop is added to and another one is removed from an inherited tabstop list.

```
\styles
{
  ParagraphStyle3_1 : Normal
  {
    ParagraphFormat.TabStops +=
    {
      Position = "2cm"
      Alignment = Center
      Leader = Spaces
    }
    ParagraphFormat.TabStops +=
    {
      Position = "7cm"
      Alignment = Right
    }
    ParagraphFormat.TabStops +=
    {
      Position = "5cm"
    }
  }
  ParagraphStyle3_2 : ParagraphStyle3_1
  {
    ParagraphFormat.TabStops -= "5cm"
    ParagraphFormat.TabStops +=
    {
      Position = "4cm"
      Alignment = Right
    }
  }
}
…
\paragraph [Style = "ParagraphStyle3_1"]
{
  TT \tab TT \tab TT \tab TT
}
\paragraph [Style = "ParagraphStyle3_2"]
{
  TT \tab TT \tab TT \tab TT
}
```

```
TT        TT                    TT        TT
TT        TT        TT                    TT
```

Borders

The borders that surround the paragraph can be defined individually for each side: top, bottom, left, and right. Each attribute is explained in the 'Paragraph-Layout' sample.

The following example demonstrates how a single border, inherited from a base style, can be made invisible.

```
\styles
{
  ParagraphStyle4_1 : Normal
  {
    ParagraphFormat.Borders
    {
      Visible = true
      Style = Single
      Color = Blue
      DistanceFromTop = 1
      DistanceFromLeft = 1
      DistanceFromRight = 1
      DistanceFromBottom = 1
    }
  }
  ParagraphStyle4_2 : ParagraphStyle4_1
  {
    ParagraphFormat.Borders.Top.Visible = false
  }
}
…
\paragraph [Style = "ParagraphStyle4_1"]
{
  This is a paragraph with a complete border.
}
\paragraph
\paragraph [Style = "ParagraphStyle4_2"]
{
  This is a paragraph without the top border.
}
```

This is a paragraph with a complete border.

This is a paragraph without the top border.

Shading

Shading defines the background color of the paragraph and is only available for the paragraph style. A complete explanation can be found in the 'Paragraph-Layout' sample.

The following example demonstrates how a shading can be switched off and on using the 'Visible' attribute. Note that the shading keeps its color within the inheritance succession.

```
\styles
{
  ParagraphStyle5_1 : Normal
  {
    ParagraphFormat.Shading.Color = LightBlue
  }
  ParagraphStyle5_2 : ParagraphStyle5_1
  {
    ParagraphFormat.Shading.Visible = false
  }
  ParagraphStyle5_3 : ParagraphStyle5_2
  {
    ParagraphFormat.Shading.Visible = true
  }
}
…
\paragraph [Style = "ParagraphStyle5_1"]
{
  This paragraph's background color is set to LightBlue.
}
\paragraph [Style = "ParagraphStyle5_2"]
{
  This paragraph's background color is switched off.
}
\paragraph [Style = "ParagraphStyle5_3"]
{
  The background color is switched on again, and keeps its original color.
}
```

This paragraph's background color is set to LightBlue.

This paragraph's background color is switched off.

The background color is switched on again, and keeps its original color.

ListInfo

ListInfo defines the layout of a list and is available only for paragraph styles. A complete explanation can be found in the 'Paragraph-Layout' sample.

The following example demonstrates how single list attributes can be kept or overriden when inheriting from a base style.

```
\styles
{
  ParagraphStyle6_1 : Normal
  {
    ParagraphFormat.ListInfo
    {
      ListType = BulletList1
      NumberPosition = 12
    }
  }
  ParagraphStyle6_2 : ParagraphStyle6_1
  {
    ParagraphFormat.ListInfo
    {
      NumberPosition = 0
      ContinuePreviousList = false
    }
  }
  ParagraphStyle6_3 : Normal
  {
    ParagraphFormat.LeftIndent = 50
    ParagraphFormat.ListInfo
    {
      ListType = NumberList2
      ContinuePreviousList = true
      NumberPosition = 30
    }
  }
}
…
\paragraph [Style = "ParagraphStyle6_1"]
{
  Line1

  Line2

  Line3
}
\paragraph [Style = "ParagraphStyle6_2"]
{
  Line1

  Line2

  Line3
}
\paragraph [Style = "ParagraphStyle6_3"]
{
  Line1

  Line2

  Line3
}
```

- Line1
- Line2
- Line3

- Line1
- Line2
- Line3

1) Line2
2) Line2
3) Line3