

Überarbeitung grafikfenster.py

Überarbeitung
der Datei
grafikfenster.py

Überarbeitung grafikfenster.py

- Auf Vorschlag eines Kollegen habe ich die Möglichkeit vorgesehen, in die Zeichenfläche ein Hintergrundbild einzubauen.
- Die Vorlage stammt aus dem Demoprogramm zu wx-Python (*Projektbeispiel zu GraphicsContext*), allerdings ist dort das Bild „*on top*“ eingefügt.
- Ergänzend habe ich die Möglichkeit eingebaut, für den *dark-modus* auf schwarze Hintergrundfarbe zu wechseln.

Überarbeitung grafikfenster.py

- Im Konstruktor Attribute hinzufügen:

```
self.__bgImage=None
```

```
self.__bgFaktor=1
```

```
self.__bgColor='WHITE'
```

Überarbeitung grafikfenster.py

- Set- und Get-Methoden

```
def SetBGImage(self, name, bgFaktor=1):  
    '''setzt ein Hintergrundbild'''  
    self.__bgImage=name  
    self.__bgFaktor=bgFaktor
```

```
def SetBGColor(self, colorName):  
    '''setzt die Hintergrundfarbe'''  
    self.__bgColor=colorName
```

```
def GetBGColor(self):  
    '''gibt die Hintergrundfarbe'''  
    return self.__bgColor
```

Überarbeitung grafikfenster.py

- In Draw erst das Hintergrundbild zeichnen:

```
gc.PushState() # save current...state
if self.__bgImage!=None:
    # Draw a bitmap ...
    bmp = wx.Bitmap(opj(self.__bgImage))
    bsz = bmp.GetSize()
    gc.DrawBitmap(bmp,
                  0, 0, ## xPos,yPos
                  bsz.width*s..., # breite
                  bsz.height*s...) # hoehe
gc.PopState()
```

Überarbeitung grafikfenster.py

- `bsz.width*self.__bgFaktor`, # breite
`bsz.height*self.__bgFaktor`) # breite ???

Die Anpassung der Breite und der Höhe mit einem Faktor ist möglich.

Überarbeitung grafikfenster.py

- Alternativ einfarbigen Hintergrund zeichnen:

else:

```
back = gc.CreatePath()
back.AddRectangle(0, 0,
                 self.GetClientSize().width,
                 self.GetClientSize().height)
gc.SetBrush(wx.Brush(self.__bgColor))
gc.FillPath(back)      # nur Füllung
gc.PopState()        # restore saved state
```

Überarbeitung grafikfenster.py

Randfarbe anpassen, damit bei schwarzem Hintergrund statt mit schwarzer Randfarbe weiss verwendet wird, dazu

- entweder die Methoden **GibFarbe()**
(*und GibFuellFarbe()* ?)
ändern
- oder in grafikfenster.py beim Zeichnen für die beiden Fälle verzweigen.
(*Das ist die aktuell gewählte Variante.*)