

Schrankwand Anfangsprojekt

Schrankwand Anfangsprojekt

Bildershow

Schrankwand Anfangsprojekt

The screenshot shows a file explorer window with the following details:

- Path: LI > OO > 2021-LI-OO > Projekte > Schrankwand-Ausgangsprojekt
- Search bar: "Schrankwand-Ausgang..."
- Table headers: Name, Änderungsdatum, Typ, Größe
- Table data:

Name	Änderungsdatum	Typ	Größe
__pycache__	20.01.2021 17:03	Dateiordner	
grafikfenster.py	04.04.2020 18:45	Python File	9 KB
Kommentar.txt	06.01.2021 14:59	Textdokument	1 KB
moebel.py	06.01.2021 15:00	Python File	5 KB
raumplaner.py	10.02.2019 11:54	Python File	2 KB
schrank.py	10.02.2019 11:53	Python File	2 KB

Schrankwand Anfangsprojekt

The screenshot shows a file explorer window with the following details:

- Path: LI > OO > 2021-LI-OO > Projekte > Schrankwand-Ausgangsprojekt
- Search bar: "Schrankwand-Ausgangsprojekt"
- Table Headers: Name, Änderungsdatum, Typ, Größe
- Table Data:

Name	Änderungsdatum	Typ	Größe
__pycache__	20.01.2021 17:03	Dateiordner	
grafikfenster.py	04.04.2020 18:45	Python File	9 KB
Kommentar.txt	06.01.2021 14:59	Textdokument	1 KB
moebel.py	06.01.2021 15:00	Python File	5 KB
raumplaner.py	10.02.2019 11:54	Python File	2 KB
schrank - Kopie.py	10.02.2019 11:53	Python File	2 KB
schrank.py	10.02.2019 11:53	Python File	2 KB

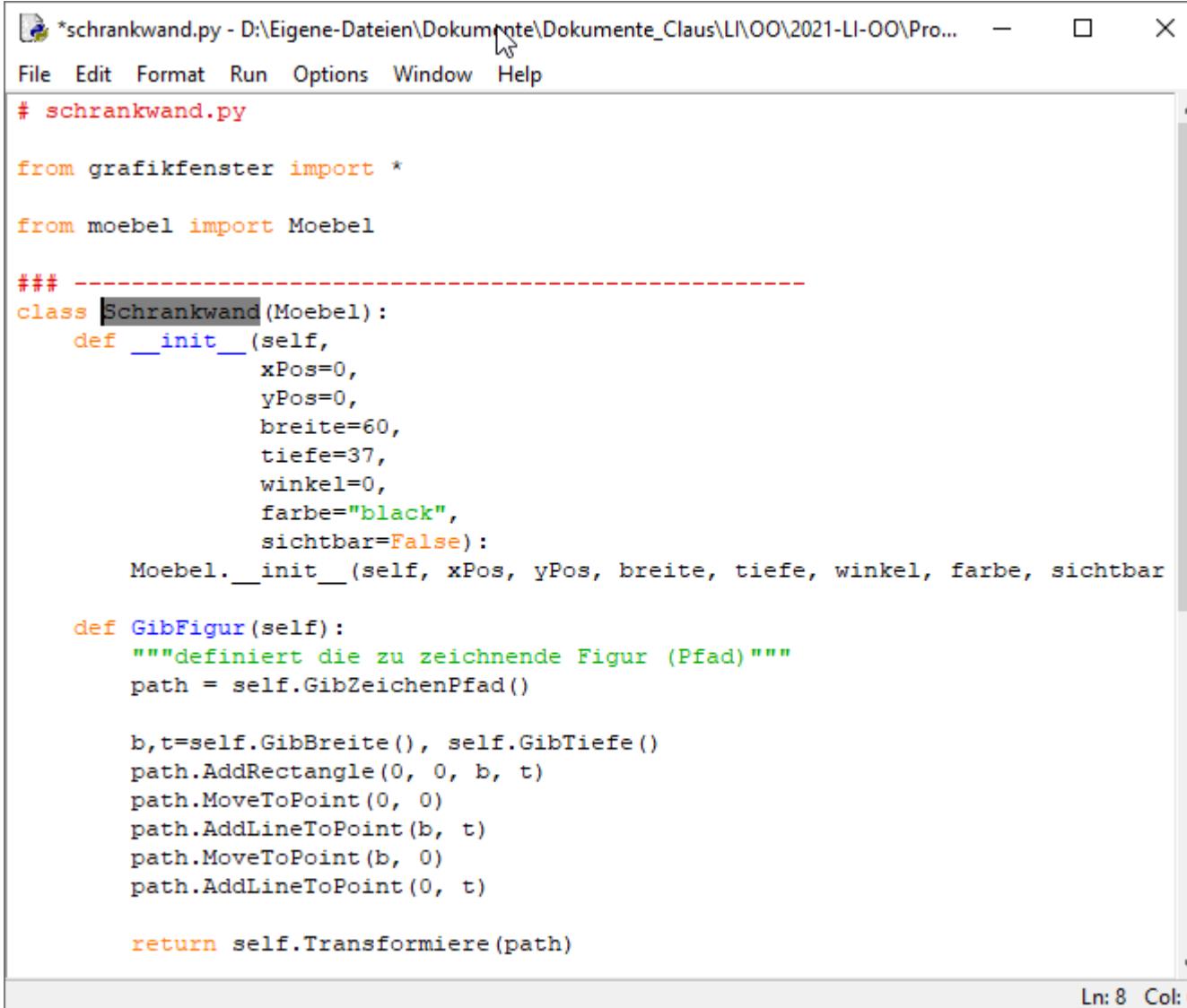
Schrankwand Anfangsprojekt

The screenshot shows a file explorer window with the following details:

- Path: LI > OO > 2021-LI-OO > Projekte > Schrankwand-Ausgangsprojekt
- Search bar: "Schrankwand-Ausgang..."
- Table headers: Name, Änderungsdatum, Typ, Größe
- Table data:

Name	Änderungsdatum	Typ	Größe
__pycache__	20.01.2021 17:03	Dateiordner	
grafikfenster.py	04.04.2020 18:45	Python File	9 KB
Kommentar.txt	06.01.2021 14:59	Textdokument	1 KB
moebel.py	06.01.2021 15:00	Python File	5 KB
raumplaner.py	10.02.2019 11:54	Python File	2 KB
schrank.py	10.02.2019 11:53	Python File	2 KB
schrankwand.py	10.02.2019 11:53	Python File	2 KB

Schrankwand Anfangsprojekt



The screenshot shows a Windows Notepad window with the file name "schrankwand.py". The code defines a class Schrankwand that inherits from Moebel. It includes methods for initializing attributes and drawing a cabinet wall shape using a path object.

```
# schrankwand.py

from grafikfenster import *
from moebel import Moebel

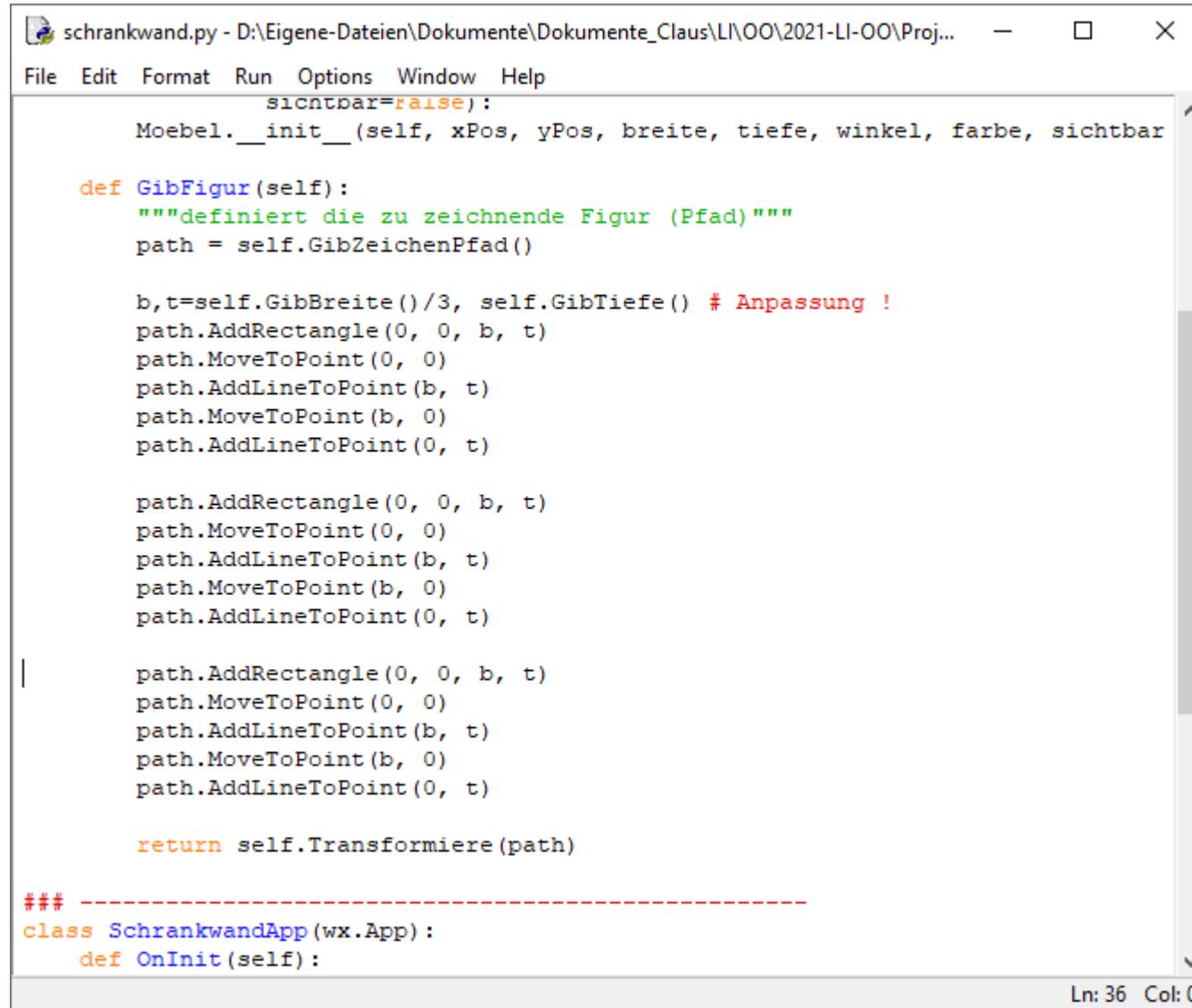
### -
class Schrankwand(Moebel):
    def __init__(self,
                 xPos=0,
                 yPos=0,
                 breite=60,
                 tiefe=37,
                 winkel=0,
                 farbe="black",
                 sichtbar=False):
        Moebel.__init__(self, xPos, yPos, breite, tiefe, winkel, farbe, sichtbar)

    def GibFigur(self):
        """definiert die zu zeichnende Figur (Pfad)"""
        path = self.GibZeichenPfad()

        b,t=self.GibBreite(), self.GibTiefe()
        path.AddRectangle(0, 0, b, t)
        path.MoveToPoint(0, 0)
        path.AddLineToPoint(b, t)
        path.MoveToPoint(b, 0)
        path.AddLineToPoint(0, t)

        return self.Transformiere(path)
```

Schrankwand Anfangsprojekt



The screenshot shows a Windows Notepad window titled "schrankwand.py". The code is written in Python and defines a class named "SchrankwandApp" which inherits from "wx.App". The class has a method "OnInit" which contains a large block of code for creating a "Moebel" object. This object's constructor ("__init__") takes parameters for position (xPos, yPos), dimensions (breite, tiefen), orientation (winkel), color (farbe), and visibility (sichtbar). The constructor then calls a "GibFigur" method, which defines a path for drawing a cabinet. The path consists of several rectangle and line segments. The code uses comments to explain its purpose and includes some specific adjustments like "Anpassung !". The Notepad window also shows the file path "D:\Eigene-Dateien\Dokumente\Dokumente_Claus\LI\OO\2021-LI-OO\Proj..." at the top.

```
schrankwand.py - D:\Eigene-Dateien\Dokumente\Dokumente_Claus\LI\OO\2021-LI-OO\Proj...
File Edit Format Run Options Window Help
    sichtbar=raise):
        Moebel.__init__(self, xPos, yPos, breite, tiefen, winkel, farbe, sichtbar

    def GibFigur(self):
        """definiert die zu zeichnende Figur (Pfad)"""
        path = self.GibZeichenPfad()

        b,t=self.GibBreite()/3, self.GibTiefe() # Anpassung !
        path.AddRectangle(0, 0, b, t)
        path.MoveToPoint(0, 0)
        path.AddLineToPoint(b, t)
        path.MoveToPoint(b, 0)
        path.AddLineToPoint(0, t)

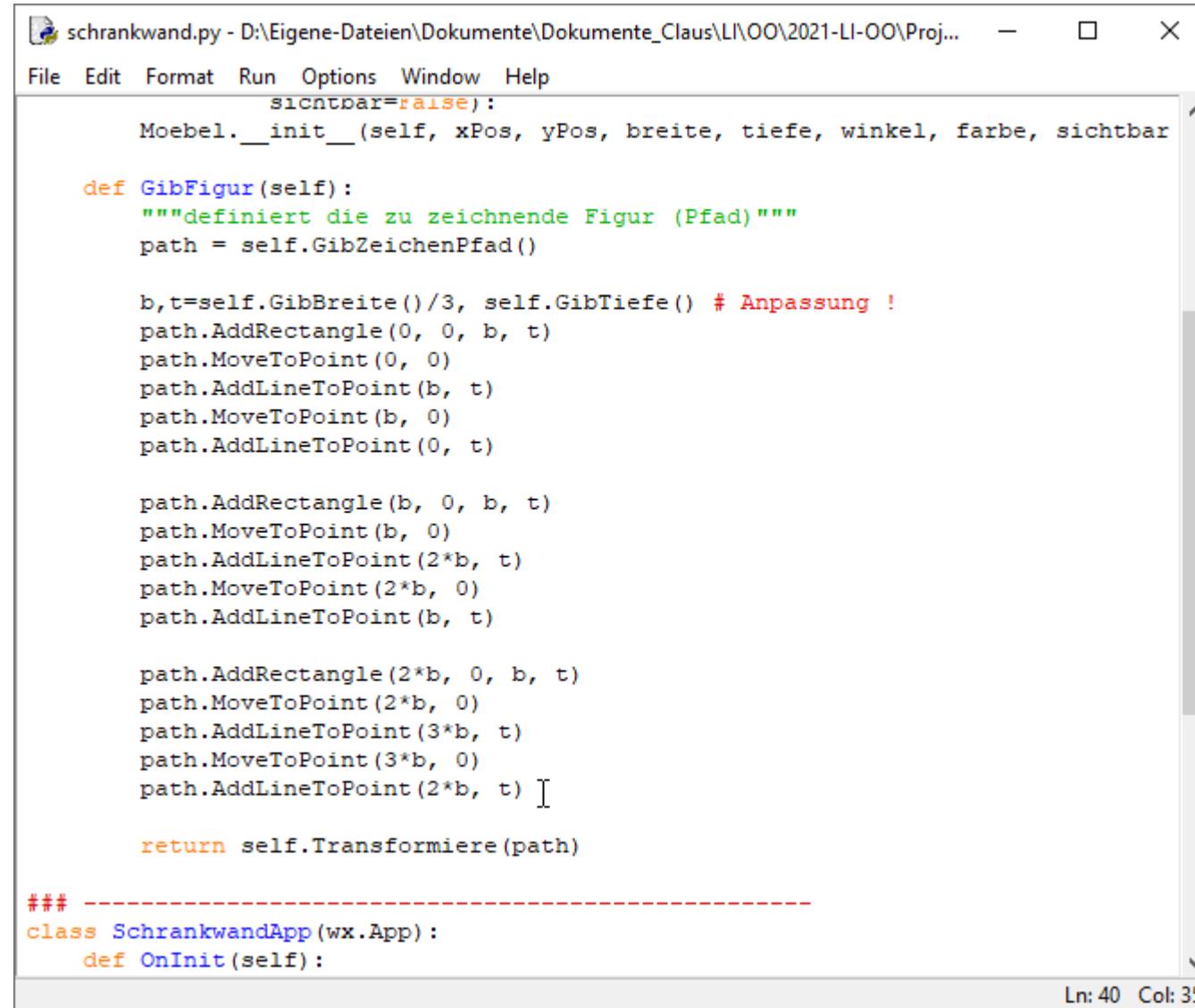
        path.AddRectangle(0, 0, b, t)
        path.MoveToPoint(0, 0)
        path.AddLineToPoint(b, t)
        path.MoveToPoint(b, 0)
        path.AddLineToPoint(0, t)

        path.AddRectangle(0, 0, b, t)
        path.MoveToPoint(0, 0)
        path.AddLineToPoint(b, t)
        path.MoveToPoint(b, 0)
        path.AddLineToPoint(0, t)

    return self.Transformiere(path)

### -----
class SchrankwandApp(wx.App):
    def OnInit(self):
Ln: 36 Col: 0
```

Schrankwand Anfangsprojekt



The screenshot shows a Windows Notepad window titled "schrankwand.py". The code is written in Python and defines a class named "SchrankwandApp" which inherits from "wx.App". The class has a method "OnInit" that creates a "Moebel" object with specific parameters. The "Moebel" class has an "__init__" method that takes parameters like xPos, yPos, breite, tiefen, winkel, farbe, and sichtbar. It also contains a "GibFigur" method that returns a path object with various line and rectangle commands to draw a cabinet door shape.

```
schrankwand.py - D:\Eigene-Dateien\Dokumente\Dokumente_Claus\LI\OO\2021-LI-OO\Proj...
File Edit Format Run Options Window Help
    sichtbar=raise):
        Moebel.__init__(self, xPos, yPos, breite, tiefen, winkel, farbe, sichtbar

    def GibFigur(self):
        """definiert die zu zeichnende Figur (Pfad)"""
        path = self.GibZeichenPfad()

        b,t=self.GibBreite()/3, self.GibTiefe() # Anpassung !
        path.AddRectangle(0, 0, b, t)
        path.MoveToPoint(0, 0)
        path.AddLineToPoint(b, t)
        path.MoveToPoint(b, 0)
        path.AddLineToPoint(0, t)

        path.AddRectangle(b, 0, b, t)
        path.MoveToPoint(b, 0)
        path.AddLineToPoint(2*b, t)
        path.MoveToPoint(2*b, 0)
        path.AddLineToPoint(b, t)

        path.AddRectangle(2*b, 0, b, t)
        path.MoveToPoint(2*b, 0)
        path.AddLineToPoint(3*b, t)
        path.MoveToPoint(3*b, 0)
        path.AddLineToPoint(2*b, t) []

    return self.Transformiere(path)

### -----
class SchrankwandApp(wx.App):
    def OnInit(self):
Ln: 40 Col: 35
```

Schrankwand Anfangsprojekt

