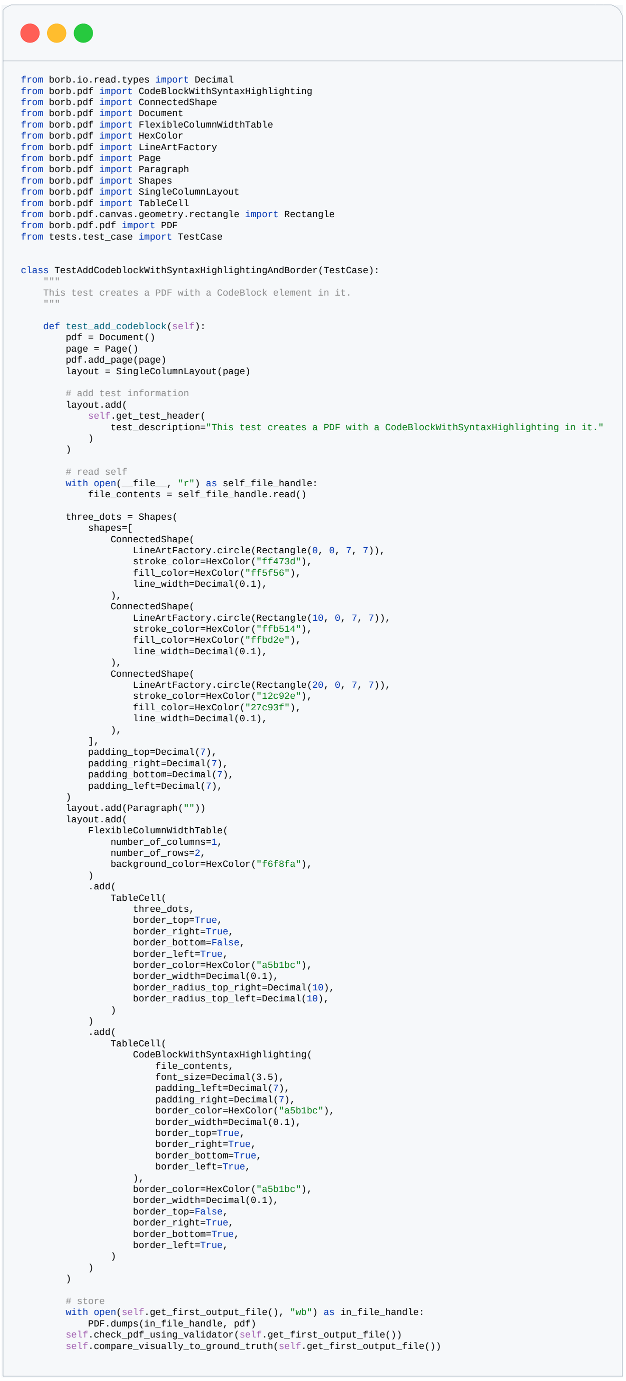
Date	17/12/2023, 07:41:13
Test	test_add_codebloc .. ghting_and_border
Description	This test creates a PDF with a CodeBlockWithSyntaxHighlighting in it.



```python
from borb.io.read.types import Decimal
from borb.pdf import CodeBlockWithSyntaxHighlighting
from borb.pdf import ConnectedShape
from borb.pdf import Document
from borb.pdf import FlexibleColumnWidthTable
from borb.pdf import HexColor
from borb.pdf import LineArtFactory
from borb.pdf import Page
from borb.pdf import Paragraph
from borb.pdf import Shapes
from borb.pdf import SingleColumnLayout
from borb.pdf import TableCell
from borb.pdf.canvas.geometry.rectangle import Rectangle
from borb.pdf.pdf import PDF
from tests.test_case import TestCase


class TestAddCodeblockWithSyntaxHighlightingAndBorder(TestCase):
    """
    This test creates a PDF with a CodeBlock element in it.
    """

    def test_add_codeblock(self):
        pdf = Document()
        page = Page()
        pdf.add_page(page)
        layout = SingleColumnLayout(page)

        # add test information
        layout.add(
            self.get_test_header(
                test_description="This test creates a PDF with a CodeBlockWithSyntaxHighlighting in it."
            )
        )

        # read self
        with open(__file__, "r") as self_file_handle:
            file_contents = self_file_handle.read()

        three_dots = Shapes(
            shapes=[
                ConnectedShape(
                    LineArtFactory.circle(Rectangle(0, 0, 7, 7)),
                    stroke_color=HexColor("ff473d"),
                    fill_color=HexColor("ff5f56"),
                    line_width=Decimal(0.1),
                ),
                ConnectedShape(
                    LineArtFactory.circle(Rectangle(10, 0, 7, 7)),
                    stroke_color=HexColor("ffb514"),
                    fill_color=HexColor("ffbd2e"),
                    line_width=Decimal(0.1),
                ),
                ConnectedShape(
                    LineArtFactory.circle(Rectangle(20, 0, 7, 7)),
                    stroke_color=HexColor("12c92e"),
                    fill_color=HexColor("27c93f"),
                    line_width=Decimal(0.1),
                ),
            ],
            padding_top=Decimal(7),
            padding_right=Decimal(7),
            padding_bottom=Decimal(7),
            padding_left=Decimal(7),
        )
        layout.add(Paragraph(""))
        layout.add(
            FlexibleColumnWidthTable(
                number_of_columns=1,
                number_of_rows=2,
                background_color=HexColor("f6f8fa"),
            )
            .add(
                TableCell(
                    three_dots,
                    border_top=True,
                    border_right=True,
                    border_bottom=False,
                    border_left=True,
                    border_color=HexColor("a5b1bc"),
                    border_width=Decimal(0.1),
                    border_radius_top_right=Decimal(10),
                    border_radius_top_left=Decimal(10),
                )
            )
            .add(
                TableCell(
                    CodeBlockWithSyntaxHighlighting(
                        file_contents,
                        font_size=Decimal(3.5),
                        padding_left=Decimal(7),
                        padding_right=Decimal(7),
                        border_color=HexColor("a5b1bc"),
                        border_width=Decimal(0.1),
                        border_top=True,
                        border_right=True,
                        border_bottom=True,
                        border_left=True,
                    ),
                    border_color=HexColor("a5b1bc"),
                    border_width=Decimal(0.1),
                    border_top=False,
                    border_right=True,
                    border_bottom=True,
                    border_left=True,
                )
            )
        )

        # store
        with open(self.get_first_output_file(), "wb") as in_file_handle:
            PDF.dumps(in_file_handle, pdf)
        self.check_pdf_using_validator(self.get_first_output_file())
        self.compare_visually_to_ground_truth(self.get_first_output_file())
```