

**Hello world**

## 01\_Hello\_World.pdf

# Hello NZPUG

## 01\_Hello\_World.py

```
from reportlab.pdfgen.canvas import Canvas

# create a Canvas object. This is your PDF.
c = Canvas("01_Hello_World.pdf", pagesize="A4")

# The canvas is very stateful. Here we adjust the font size. This will affect
# all further text until a new font size is set. (If font size is not set it
# will default to 12pt. A bit small for this demo.)
c.setFontSize(60)

width, height = c._pagesize
c.drawCentredString(width / 2, height / 2, "Hello World")

c.showPage() # Finish page 1

# Any extra drawing commands entered here would appear on page 2!

c.save() # Finish the pdf and save to file. This Canvas can no longer be used.
```

## 02\_Text.pdf

Canvas.drawString	Once upon a midnight dreary, while I pondered, weak and weary, Over many a quaint and curious volume of forgotten lore—
Canvas.drawCentredString	While I nodded, nearly napping, suddenly there came a tapping, As of some one gently rapping, rapping at my chamber door.
Canvas.drawString	" 'Tis some visitor," I muttered, "tapping at my chamber door— Only this and nothing more."
Canvas.beginText (Text object)	<i>Ah, distinctly I remember it was in the bleak December, And each separate dying ember wrought its ghost upon the floor. Eagerly I wished the morrow;—vainly I had sought to borrow From my books surcease of sorrow—sorrow for the lost Lenore— For the rare and radiant maiden whom the angels name Lenore— Nameless here for evermore.</i>  <i>And the silken sad uncertain rustling of each purple curtain Thrilled me—filled me with fantastic terrors never felt before; So that now, to still the beating of my heart, I stood repeating " 'Tis some visiter entreating entrance at my chamber door— Some late visiter entreating entrance at my chamber door; This it is and nothing more."</i> ↑↑↑↑↑↑↑

## 02\_Text.py

```
from reportlab.pdfgen.canvas import Canvas
from reportlab.lib.units import mm, cm

c = Canvas("02_Text.pdf", pagesize=(210*mm, 297*mm))
text_file = open("02_The_Raven.txt")
text = (l.strip() for l in text_file if l.strip())

c.setFontSize(10)
leading = 14 # line spacing.

# translate halfway across page
c.translate(105*mm, 0)

# draw a light grey vertical line
c.setStrokeColorRGB(255, 0, 0)
c.line(0, 0, 0, 297*mm)

# move up the page
c.translate(0, 260*mm)

c.drawString(-100*mm, 0, "Canvas.drawString")
c.translate(0, -leading)
for i in range(2):
    c.translate(0, -leading)
    c.drawString(0, 0, text.next())

c.translate(0, -20*mm)
c.drawString(-100*mm, 0, "Canvas.drawCentredString")
c.translate(0, -leading)
for i in range(2):
    c.translate(0, -leading)
    c.drawCentredString(0, 0, text.next())

c.translate(0, -20*mm)
c.drawString(-100*mm, 0, "Canvas.drawRightString")
c.translate(0, -leading)
for i in range(2):
    c.translate(0, -leading)
    c.drawRightString(0, 0, text.next())

c.translate(0, -20*mm)
c.drawString(-100*mm, 0, "Canvas.beginText (Text object)")
c.translate(0, -leading*2)

# Text objects are more efficient for anything more than simple labels
textobject = c.beginText()
textobject.setTextOrigin(0, 0)
textobject.setFont("Times-Italic", 10)

# supply text one line at a time:
for i in range(6):
    textobject.textLine(text.next())
textobject.textLine()

# or select several lines of newline separated text
text_lines = '\n'.join(text.next() for i in range(5))
textobject.textLines(text_lines)

# or perhaps a word at a time...
text_words = text.next().split()
for word in text_words:
    textobject.textOut(word + " ")
    # Draw a red arrow to mark the cursor position
    x, y = textobject.getCursor()
    c.saveState()
    c.setFillColorRGB(255, 0, 0)
    c.drawCentredString(x, y-10, '|')
    c.drawCentredString(x, y-10, '^')
    c.restoreState()

# Finally render the text object
c.drawText(textobject)
```

```
text_file.close()
c.showPage()
c.save()

print "done"
```

## 02\_The\_Raven.txt

Once upon a midnight dreary, while I pondered, weak and weary,  
Over many a quaint and curious volume of forgotten lore—  
While I nodded, nearly napping, suddenly there came a tapping,  
As of some one gently rapping, rapping at my chamber door.  
" 'Tis some visitor," I muttered, "tapping at my chamber door—  
Only this and nothing more."

Ah, distinctly I remember it was in the bleak December,  
And each separate dying ember wrought its ghost upon the floor.  
Eagerly I wished the morrow;—vainly I had sought to borrow  
From my books surcease of sorrow—sorrow for the lost Lenore—  
For the rare and radiant maiden whom the angels name Lenore—  
Nameless here for evermore.

And the silken sad uncertain rustling of each purple curtain  
Thrilled me—filled me with fantastic terrors never felt before;  
So that now, to still the beating of my heart, I stood repeating  
" 'Tis some visitor entreating entrance at my chamber door—  
Some late visitor entreating entrance at my chamber door;  
This it is and nothing more."

Presently my soul grew stronger; hesitating then no longer,  
"Sir," said I, "or Madam, truly your forgiveness I implore;  
But the fact is I was napping, and so gently you came rapping,  
And so faintly you came tapping, tapping at my chamber door,  
That I scarce was sure I heard you"—here I opened wide the  
door;  
Darkness there and nothing more.  
Deep into that darkness peering, long I stood there wondering,  
fearing,

----- SNIP -----

## The Tell-Tale Heart

TRUE!—NERVOUS—VERY, VERY dreadfully nervous I had been and am; but why will you say that I am mad? The disease had sharpened my senses—not destroyed—not dulled them. Above all was the sense of hearing acute. I heard all things in the heaven and in the earth. I heard many things in hell. How, then, am I mad? Hearken! and observe how healthily—how calmly I can tell you the whole story.

It is impossible to say how first the idea entered my brain; but once conceived, it haunted me day and night. Object there was none. Passion there was none. I loved the old man. He had never wronged me. He had never given me insult. For his gold I had no desire. I think it was his eye! yes, it was this! One of his eyes resembled that of a vulture—a pale blue eye, with a film over it. Whenever it fell upon me, my blood ran cold; and so by degrees—very gradually—I made up my mind to take the life of the old man, and thus rid myself of the eye for ever.



## 03\_Hello\_Again.py

```
from reportlab.platypus import SimpleDocTemplate, Paragraph, Spacer
from reportlab.lib.styles import import getSampleStyleSheet, ParagraphStyle
from reportlab.lib.units import mm
from reportlab.lib.enums import TA_JUSTIFY, TA_CENTER

# Create a couple of paragraph styles
body_style = ParagraphStyle(name="BodyStyle",
                             fontName="Times-Roman",
                             fontSize=14,
                             leading=17,
                             spaceAfter=20,
                             alignment=TA_JUSTIFY,
                             )
title_style = ParagraphStyle(name="TitleStyle",
                              fontName="Times-Bold",
                              fontSize=26,
                              alignment=TA_CENTER,
                              )

Title = "Hello world"
pageinfo = "platypus example"

doc = SimpleDocTemplate("03_Hello_Again.pdf",
                        leftMargin=40*mm,
                        rightMargin=40*mm,
                        topMargin=40*mm,
                        bottomMargin=40*mm)

story = [] # Fill this list with flowable objects

story.append(Paragraph("The Tell-Tale Heart",
                        title_style
                        )

story.append(Spacer(1, 25*mm))

story.append(Paragraph("""
TRUE!&#8212;NERVOUS&#8212;VERY, VERY dreadfully nervous I had been and am;
but why will you say that I am mad? The disease had sharpened my
senses&#8211;not destroyed&#8211;not dulled them. Above all was the sense of
hearing acute. I heard all things in the heaven and in the earth. I heard
many things in hell. How, then, am I mad? Hearken! and observe how
healthily&#8211;how calmly I can tell you the whole story.
""", body_style))

story.append(Paragraph("""
It is impossible to say how first the idea entered my brain; but once
conceived, it haunted me day and night. Object there was none. Passion there
was none. I loved the old man. He had never wronged me. He had never given
me insult. For his gold I had no desire. I think it was his eye! yes, it was
this! One of his eyes resembled that of a vulture&#8211;a pale blue eye,
with a film over it. Whenever it fell upon me, my blood ran cold; and so by
degrees&#8211;very gradually&#8211;I made up my mind to take the life of the
old man, and thus rid myself of the eye for ever.
""", body_style))

story.append(Spacer(1, 15*mm))

story.append(Paragraph("&#9829;", title_style))

doc.build(story)
```

## 99\_Self\_Document.py

```
from reportlab.platypus.xpreformatted import PythonPreformatted, XPreformatted
from reportlab.platypus import SimpleDocTemplate, Spacer
from reportlab.platypus import Paragraph, KeepInFrame
from reportlab.platypus import Flowable, PageBreak
```

```

from reportlab.lib.styles import getSampleStyleSheet
from reportlab.lib.units import mm, inch
from reportlab.lib import colors
from reportlab.rl_config import defaultPageSize
from reportlab.lib.styles import ParagraphStyle as PS

from pdfwr import PdfReader
from pdfwr.buildxobj import pagexobj, ViewInfo
from pdfwr.toreportlab import makerl

import os

PAGE_HEIGHT=defaultPageSize[1]; PAGE_WIDTH=defaultPageSize[0]
styles = getSampleStyleSheet()

Title = "Hello world"
pageinfo = "platypus example"

class PdfFlowable(Flowable):
    """A custom flowable which draws the first page of a pdf file
    into the frame at 75% scale"""
    def __init__(self, pdf_filename, spaceBefore=12, spaceAfter=12):
        self.spaceBefore = spaceBefore
        self.spaceAfter = spaceAfter

        # Read the pdf file data
        pdf_page = PdfReader(pdf_filename, decompress=False).pages[0]
        print pdf_page["/MediaBox"]
        x, y, width, height = pdf_page["/MediaBox"]
        self.width = float(width)
        self.height = float(height)

        # Convert each page to a pagexobject. This is a special kind of
        # self contained pdf object that can be reused in other pdf files.
        self.xobj = pagexobj(pdf_page)

    def wrap(self, *args):
        # returns the height of the object
        return (0, float(self.height)*0.75 + self.spaceBefore + self.spaceAfter)

    def draw(self):
        c = self.canv
        c.saveState()
        c.translate(0, self.spaceBefore)
        c.scale(0.75, 0.75)
        c.saveState()
        c.translate(10, -10)
        c.setFillColorCMYK(0, 0, 0, 0.2)
        c.rect(0, 0, self.width, self.height, stroke=0, fill=1)
        c.restoreState()
        c.saveState()
        c.setFillColorCMYK(0, 0, 0, 0)
        c.rect(0, 0, self.width, self.height, stroke=0, fill=1)
        c.restoreState()
        c.doForm(makerl(c, self.xobj))
        c.rect(0, 0, self.width, self.height, stroke=1, fill=0)
        c.restoreState()

    def myFirstPage(canvas, doc):
        # draws the static elements of the first page
        canvas.saveState()
        canvas.setFont('Times-Bold',16)
        canvas.drawCentredString(PAGE_WIDTH/2.0, PAGE_HEIGHT-108, Title)
        canvas.setFont('Times-Roman',9)
        canvas.drawString(inch, 0.75 * inch, "First Page / %s" % pageinfo)
        canvas.restoreState()

    def myLaterPages(canvas, doc):
        # draws the static elements of all subsequent pages
        canvas.saveState()
        canvas.setFont('Times-Roman',9)
        canvas.drawString(inch, 0.75 * inch, "Page %d %s" % (doc.page, pageinfo))
        canvas.restoreState()

base_pre_style = PS('pre',
                    fontName="Courier",

```



```

        fontSize=9,
        leading=9,
        borderPadding=6,
        spaceBefore=12,
        spaceAfter=12,
    )

code_style = PS('code',
    parent=base_pre_style,
    backColor=colors.whitesmoke,
)

plain_style = PS('plaintext',
    parent=base_pre_style,
    backColor=colors.lightgreen,
)

title_style = PS('titletext',
    fontName="Helvetica-Bold",
    fontSize=14,
    leading=16,
    spaceBefore=24,
)

def pythonCode(text):
    text = open(text).read()
    return PythonPreformatted(text, code_style)

def plainText(text):
    text = open(text).read()
    # Snip long text files down to the first 30 lines
    text_lines = text.splitlines()
    if len(text_lines) > 30:
        text_lines = text_lines[:30]
        text_lines.append("-"*37 + " SNIP " + "-"*37)
        text = '\n'.join(text_lines)
    return XPreformatted(text, plain_style)

def pdfPage(file_data):
    # handler for pdf files
    return PdfFlowable(file_data)

def aCircle(self):
    canvas = self.canv
    canvas.saveState()
    canvas.rect(0, 0, 100*mm, 100*mm)
    canvas.restoreState()

def go():
    doc = SimpleDocTemplate("99_Self_Document.pdf")
    Story = []

    files = os.listdir('.')

    handlers = {"py":pythonCode,
                "txt":plainText,
                "pdf":pdfPage,
                }

    for file_name in sorted(files):
        if file_name in ("99_Self_Document.pdf", "pdfwr"):
            continue
        print file_name
        file_extension = file_name.split(".")[-1]
        if file_extension in handlers:
            if file_extension == "pdf":
                Story.append(PageBreak())
            heading = Paragraph(file_name, title_style)
            file_contents = handlers[file_extension](file_name)
            Story.extend([heading, file_contents])

    doc.build(Story, onFirstPage=myFirstPage, onLaterPages=myLaterPages)

go()

```