E:\Personal\Python\program\Reportlab\To Do List\todolist.py

```python
from reportlab.pdfgen import canvas
import webbrowser
import os
import json

with open('task.txt', 'r') as f:
    task = json.load(f)

c = canvas.Canvas('todolist.pdf')
c.setLineWidth(3)
c.line(0, 421, 595, 421)
c.line(298, 0, 298, 841)

c.drawCentredString(446, 841 - 30, 'Do')
c.drawCentredString(149, 841 - 30, 'Plan')
c.drawCentredString(149, 421 - 30, 'Delegate')
c.drawCentredString(446, 421 - 30, 'Eliminate')

for i in task:
    if i == 'Do':
        for j in range(len(task[i])):
            c.drawCentredString(446, 841 - 30 - 40*(j+1), task[i][j])
    if i == 'Plan':
        for j in range(len(task[i])):
            c.drawCentredString(149, 841 - 30 - 40*(j+1), task[i][j])
    if i == 'Delegate':
        for j in range(len(task[i])):
            c.drawCentredString(149, 421 - 30 - 40*(j+1), task[i][j])
    if i == 'Eliminate':
        for j in range(len(task[i])):
            c.drawCentredString(446, 421 - 30 - 40*(j+1), task[i][j])

c.save()
webbrowser.open('file://'+os.path.realpath('todolist.pdf'))
```

E:\Personal\Python\program\Reportlab\初识\初始reportlab.py

```python
from reportlab.pdfgen import canvas
import webbrowser
import os
from reportlab.pdfbase.ttfonts import TTFont
from reportlab.pdfbase.pdfmetrics import registerFont
registerFont(TTFont('fangzheng','fangzheng.TTF'))
c = canvas.Canvas('hello.pdf', pagesize=(500,500))
c.setFont('fangzheng',15)
c.drawString(250,250, "第一页")
c.showPage()
c.setFont('fangzheng', 20)
c.drawString(250, 250, "第二页")
c.save()
webbrowser.open('file://'+os.path.realpath('hello.pdf'))
```

E:\Personal\Python\program\Reportlab\图片转PDF\图片转PDF.py

```python
from reportlab.pdfgen import canvas
import webbrowser
import os
from PIL import Image


def imgtopdf(filename):
    img = Image.open(filename+'.png')
    pix = img.load()

    minX = 10000
    maxX = 0
    y = 0

    for x in range(0, img.width):
        if pix[x, y][3] != 0:
            if minX > x:
                minX = x
            if maxX < x:
                maxX = x

    width = maxX - minX
    img = img.crop((minX, 0, maxX, img.height))
    img.save(filename+'_裁剪.png')
    img.close()

    c = canvas.Canvas('test.pdf', pagesize=(width, img.height))
    c.drawImage(filename+'_裁剪.png', 0, 0, mask='auto')
    c.save()
    webbrowser.open('file://'+os.path.realpath('test.pdf'))

filename = '讲义'
imgtopdf(filename)
```

E:\Personal\Python\program\Reportlab\寄给源码世界的信\form.py

```python
from reportlab.pdfgen import canvas
import webbrowser
import os
from reportlab.lib import colors
from reportlab.pdfbase import pdfmetrics
from reportlab.pdfbase.ttfonts import TTFont

pdfmetrics.registerFont(TTFont('fangzheng', 'fangzheng.TTF'))

c = canvas.Canvas('form.pdf')
c.setFont('fangzheng', 15)
c.drawString(298, 800, '寄往源码世界的信')
form = c.acroForm
c.drawCentredString(200, 700, '昵称')
form.textfield(x=300, y=680, fillColor=colors.white)
c.drawCentredString(200, 600, '性别')
c.drawCentredString(300, 600, '男')
form.checkbox(x=320, y=595, buttonStyle='check')
c.drawCentredString(400, 600, '女')
form.checkbox(x=420, y=595, buttonStyle='check')
c.drawCentredString(200, 500, '年龄')
form.choice(x=300, y=480, value='6', options=[str(i) for i in range(6, 19)])
c.drawCentredString(200, 400, '寄语')
form.textfield(x=300, y=220, width=230, height=200, fillColor=colors.white)

c.save()
webbrowser.open('file://'+os.path.realpath('form.pdf'))
```

E:\Personal\Python\program\Reportlab\文档魔法\文档魔法.py

```
import webbrowser
import os
from reportlab.platypus import SimpleDocTemplate, Paragraph, Spacer
from reportlab.lib.styles import getSampleStyleSheet, ParagraphStyle
from reportlab.pdfbase import pdfmetrics
from reportlab.pdfbase.ttfonts import TTFont

pdfmetrics.registerFont(TTFont('fangzheng', 'fangzheng.TTF'))

styles = getSampleStyleSheet()
styles.add(ParagraphStyle(name='cn', fontName='fangzheng'))

doc = SimpleDocTemplate('doc.pdf')
content = []
for i in range(100):
    p = Paragraph('句子一。句子二', styles['cn'])
    content.append(p)
    content.append(Spacer(20, 20))


def myOnFirstPage(canvas, doc):
    canvas.setFont('fangzheng', 10)
    canvas.drawCentredString(298, 30, '首页')


def myOnLaterPages(canvas, doc):
    canvas.setFont('fangzheng', 10)
    canvas.drawCentredString(298, 30, '第{}页'.format(doc.page))



doc.build(content, myOnFirstPage, myOnLaterPages)
webbrowser.open('file://'+os.path.realpath('doc.pdf'))
```

E:\Personal\Python\program\Reportlab\时光记录卡\时光记录卡.py

```
from reportlab.pdfgen import canvas
from reportlab.graphics.charts.piecharts import Pie
from reportlab.graphics import renderPDF
from reportlab.graphics.shapes import Drawing
from reportlab.lib.colors import red, yellow, green, blue, orange, purple
from reportlab.graphics.charts.legends import Legend
from reportlab.pdfbase.ttfonts import TTFont
from reportlab.pdfbase.pdfmetrics import registerFont
import webbrowser
import os

plan = ['睡觉', '上课', '写作业', '兴趣班', '吃饭']
data = [9, 8, 2, 2, 2, 1]

c = canvas.Canvas('文件名.pdf')
c.scale(0.5, 0.5)
c.drawImage('card.png', 0, 0, width=372, height=665, mask=None)
registerFont(TTFont('fangzheng', 'fangzheng.TTF'))
drawing = Drawing(372, 665)

pc = Pie()
pc.x = 115
pc.y = 350
pc.height = 150
pc.width = 150
pc.data = data
pc.labels = [str(i) for i in data]
pc.sideLabels = 1

colors = [red, orange, yellow, green, blue, purple]
count = 0
for i in colors:
    pc.slices[count].fillColor = i
    count += 1

pc.slices.popout = 5

legend = Legend()
legend.x = 133
legend.y = 260
legend.colorNamePairs = list(zip(colors, plan))
legend.fontName = 'fangzheng'
legend.alignment = 'right'

drawing.add(pc)
drawing.add(legend)
renderPDF.draw(drawing, c, 0, 0)

c.setFont('fangzheng', 16)
c.drawString(145, 300, '我的时光记录')

c.save()
webbrowser.open('file://'+os.path.realpath("文件名.pdf"))
```

E:\Personal\Python\program\Reportlab\源码奖状\源码奖状-1-1.py

```python
from reportlab.pdfgen import canvas
import webbrowser
import os
from PIL import Image

from reportlab.pdfbase.ttfonts import TTFont
from reportlab.pdfbase.pdfmetrics import registerFont
registerFont(TTFont('fangzheng', 'fangzheng.ttf'))

c = canvas.Canvas('源码奖状.pdf')

c.setFont('fangzheng', 40)
c.drawCentredString(298, 600, '课程证书')

c.drawImage('1.png', 0, 0, mask="auto", width=595.27, height=841.89)

img2 = Image.open('2.png')
img2.close()
c.drawImage('2.png', 298 - img2.width/8, 500, mask="auto", width=img2.width/4, height=img2.height/4)

img3 = Image.open('3.png')
img3.close()
c.drawImage('3.png', 298 - img3.width/8, 700, mask="auto", width=img3.width/4, height=img3.height/4)

img4 = Image.open('4.png')
img4.close()
c.drawImage('4.png', 298 - img4.width/8, 100, mask="auto", width=img4.width/4, height=img4.height/4)

c.setFontSize(20)
c.drawCentredString(298,400,'兹证明')

c.setFontSize(30)
c.drawCentredString(298,350,'我')

c.setFontSize(20)
c.drawCentredString(298,300,'完成了《源码奖状》课程')

c.save()
webbrowser.open('file://'+os.path.realpath("源码奖状.pdf"))
```

E:\Personal\Python\program\Reportlab\源码学生证\源码学生证-1-2.py

```python
from reportlab.pdfgen import canvas
import webbrowser
import os
from reportlab.graphics.shapes import Drawing
from reportlab.graphics.barcode import qr
from reportlab.graphics import renderPDF
from reportlab.pdfbase.ttfonts import TTFont
from reportlab.pdfbase.pdfmetrics import registerFont

def createQrCode(canvas, x, y):
    with open('aduan.txt', 'r', encoding='utf-8-sig') as document:
        content = document.read()
    Qr_code = qr.QrCodeWidget(content, barBorder=2, barWidth=113, barHeight=113)
    drawing = Drawing(50, 50)
    drawing.add(Qr_code)
    renderPDF.draw(drawing, canvas, x, y)


c = canvas.Canvas('学生证.pdf')
c.scale(0.5, 0.5)
c.drawImage('1.png', 0, 0, width=372, height=665, mask=None)
c.drawImage('2.png', 150, 420, width = 87, height = 95, mask = 'auto')
c.roundRect(145, 415, 97, 105, 6, stroke = 1, fill = 0)

registerFont(TTFont('fangzheng','fangzheng.TTF'))

c.setFont('fangzheng',35)
c.drawString(95, 550, '学　生　证')

c.setFont('fangzheng', 20)
c.drawString(105, 370, '姓　名：阿　　　短')
c.drawString(105, 330, '班　级：六 年 一 班')
c.drawString(105, 290, '学　号：１０００１')
c.drawString(105, 250, '性　别：　男')

c.drawString(200, 95, '想了解更多信息')
c.drawString(200, 65, '请扫一扫')

createQrCode(c, 51, 28)
c.showPage()
c.save()
webbrowser.open('file://'+os.path.realpath("学生证.pdf"))
```

E:\Personal\Python\program\Reportlab\源码手账\code.py

```
import random

def main():
    return random.randint(1, 10)

if __name__ == '__main__':
    main()
```

E:\Personal\Python\program\Reportlab\源码表格\源码表格.py

```python
from reportlab.platypus import SimpleDocTemplate, Table, TableStyle
from reportlab.lib import colors
import webbrowser
import os
from reportlab.pdfbase import pdfmetrics
from reportlab.pdfbase.ttfonts import TTFont

pdfmetrics.registerFont(TTFont('fangzheng', 'fangzheng.TTF'))

doc = SimpleDocTemplate('源码表格.pdf')
content = []
data = [['中', '文'], ['字', '体']]
t = Table(data, colWidths=100, rowHeights=100)
t.setStyle(TableStyle([
    ('TEXTCOLOR', (0, 0), (1, 1), colors.red),
    ('TEXTCOLOR', (0, 0), (0, 0), colors.blue),
    ('TEXTCOLOR', (1, 1), (1, 1), colors.blue),
    ('INNERGRID', (0, 0), (−1, −1), 0.25, colors.black),
    ('BOX', (0, 0), (−1, −1), 0.25, colors.black),
    ('ALIGN', (0, 0), (−1, −1), 'CENTER'),
    ('VALIGN', (0, 0), (−1, −1), 'MIDDLE'),
    ('FONTNAME', (0, 0), (−1, −1), 'fangzheng'),
    ('FONTSIZE', (0, 0), (−1, −1), 20)
]))

content.append(t)
doc.build(content)
webbrowser.open('file://'+os.path.realpath('源码表格.pdf'))
```