Министерство НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

Национальный исследовательский университет ИТМО

**факультет безопасности информационных технологий**

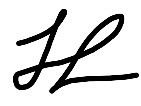
**Управление мобильными устройствами**

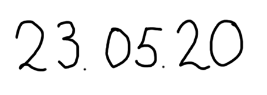
Лабораторная работа №3

«Формирование счета на оплату услуг»

Вариант№11

Работу выполнил:

Студент группы №3352   
Ласкус Е.О. \_\_ \_\_\_\_\_\_\_



Работу проверил:

Федоров И. Р . \_\_\_\_\_\_\_\_\_



Санкт-Петербург, 2020

**Цель работы:** разработка и реализация программного модуля создания и автоматического формирования счета на оплату по данным, полученным в предыдущих работах.

**Описание выбранных средств реализации:**

Разработанный программный модуль был реализован на языке Python с использованием библиотеки обработки и анализа данных Pandas.

Одна из причин выбора Python — более простое оформление, чем в других языках. Его проще изучить с нуля даже новичку в программировании. Это самый популярный язык общего назначения: он используется для машинного обучения, аналитике, разработке игр и в науке о данных.

За основу для создания PDF файла была взята библиотека PyFPDF, так как она позволяет собирать PDF файл из структур, схожих со структурами html.

**Исходный код:**

**MLW\_3.py**

# -\*- coding: utf-8 -\*-

"""Copy of MLW\_3.ipynb

Automatically generated by Colaboratory.

Original file is located at

https://colab.research.google.com/drive/13VkLgHfV6GbCeIK5hWqDS71dLMUyae4s

"""

from fpdf import FPDF

from num2words import num2words

import datetime

import math

import os

import sys

def banks (pdf: FPDF, \*\*kwargs):

MAXIMUM\_PAGE\_WIDTH = 210

PADDING = 20

height = 30

l\_col\_w = 95

m\_col\_w = 16

r\_col\_w = MAXIMUM\_PAGE\_WIDTH - PADDING \* 2 - l\_col\_w - m\_col\_w

pdf.line(PADDING, PADDING, MAXIMUM\_PAGE\_WIDTH - 20, PADDING)

pdf.line(PADDING, PADDING, PADDING, height + PADDING)

pdf.line(MAXIMUM\_PAGE\_WIDTH - PADDING, PADDING, MAXIMUM\_PAGE\_WIDTH - PADDING, height + PADDING)

pdf.line(PADDING, height + PADDING, MAXIMUM\_PAGE\_WIDTH - PADDING, height + PADDING)

pdf.line(PADDING, height \* 9 / 21 + PADDING, MAXIMUM\_PAGE\_WIDTH - PADDING, height \* 9 / 21 + PADDING)

pdf.line(PADDING, height \* 12 / 21 + PADDING, l\_col\_w + PADDING, height \* 12 / 21 + PADDING)

pdf.line(l\_col\_w + PADDING, PADDING, l\_col\_w + PADDING, height + PADDING)

pdf.line(l\_col\_w + PADDING + m\_col\_w, PADDING, l\_col\_w + PADDING + m\_col\_w, height + PADDING)

pdf.line(l\_col\_w + PADDING, PADDING + height \* 3 / 21, l\_col\_w + m\_col\_w + PADDING, PADDING + height \* 3 / 21)

pdf.line(PADDING + l\_col\_w / 2, PADDING + height \* 9 / 21, PADDING + l\_col\_w / 2, PADDING + height \* 12 / 21)

pdf.line(PADDING, height + 15, MAXIMUM\_PAGE\_WIDTH / 1.81 - 1, height + 15)

pdf.set\_font("TimesNewRoman", size=9)

pdf.set\_y(PADDING)

pdf.cell(10)

pdf.multi\_cell(95, 4, kwargs['payee\_bank'])

pdf.set\_y(PADDING + 12.5)

pdf.cell(10)

pdf.cell(l\_col\_w / 2, 5, f'ИНН {kwargs["INN"]}')

pdf.cell(l\_col\_w / 2, 5, f'КПП {kwargs["KPP"]}')

pdf.set\_y(PADDING + 17.5)

pdf.cell(10)

pdf.multi\_cell(95, 4, kwargs['payee'])

pdf.set\_y(PADDING)

pdf.cell(10 + l\_col\_w)

pdf.cell(m\_col\_w, 5, 'БИК')

pdf.cell(r\_col\_w, 5, kwargs['BIK'])

pdf.set\_y(PADDING + 5)

pdf.cell(10 + l\_col\_w)

pdf.cell(m\_col\_w, 5, 'Сч. №')

pdf.cell(r\_col\_w, 5, kwargs['account1'])

pdf.set\_y(PADDING + height \* 9 / 21)

pdf.cell(10 + l\_col\_w)

pdf.cell(m\_col\_w, 5, 'Сч. №')

pdf.cell(r\_col\_w, 5, kwargs['account2'])

pdf.set\_font("TimesNewRoman", size=8)

pdf.set\_y(PADDING + height \* 6 / 21)

pdf.cell(10)

pdf.cell(l\_col\_w, 4, 'Банк получателя')

pdf.set\_y(PADDING + height \* 18 / 21)

pdf.cell(10)

pdf.cell(l\_col\_w / 2, 5, f'Получатель {kwargs["addressee"]}')

return PADDING + height

def payment\_invoice(pdf: FPDF, \*\*kwargs):

MAXIMUM\_PAGE\_WIDTH = 210

PADDING = 20

pdf.set\_font("TimesNewRomanB", size=10)

pdf.set\_y(kwargs['height'] + 4)

pdf.cell(10)

pdf.cell(MAXIMUM\_PAGE\_WIDTH - PADDING, 9, f'Счет на оплату № {kwargs["account\_number"]} от {kwargs["day"]}.{kwargs["month"]}.20{kwargs["year"]} г.')

return kwargs['height'] + 14.1

def requisites(pdf: FPDF, \*\*kwargs):

MAXIMUM\_PAGE\_WIDTH = 210

PADDING = 20

l\_col\_w = 28

r\_col\_w = MAXIMUM\_PAGE\_WIDTH - 2 \* PADDING - l\_col\_w

line\_height = 5

pdf.set\_font("TimesNewRoman", size=9)

pdf.set\_y(kwargs['height'] + 2)

pdf.cell(10)

pdf.multi\_cell(l\_col\_w, line\_height, 'Поставщик:')

pdf.set\_font("TimesNewRomanB", size=9)

pdf.set\_y(kwargs['height'] + 2)

pdf.cell(10 + l\_col\_w)

executor = pdf.multi\_cell(r\_col\_w, line\_height, kwargs['executor'], split\_only=True)

pdf.multi\_cell(r\_col\_w, line\_height, kwargs['executor'])

height = kwargs['height'] + 6 + 5 \* len(executor)

pdf.set\_font("TimesNewRoman", size=9)

pdf.set\_y(height)

pdf.cell(10)

pdf.multi\_cell(l\_col\_w, line\_height, 'Покупатель:')

pdf.set\_font("TimesNewRomanB", size=9)

pdf.set\_y(height)

pdf.cell(10 + l\_col\_w)

buyer = pdf.multi\_cell(r\_col\_w, line\_height, kwargs['client'], split\_only=True)

pdf.multi\_cell(r\_col\_w, line\_height, kwargs['client'])

height += len(buyer) \* 5 + 6

pdf.set\_y(height)

pdf.cell(10)

pdf.set\_font("TimesNewRoman", size=9)

pdf.cell(l\_col\_w, line\_height, 'Основание:')

pdf.set\_font("TimesNewRomanB", size=9)

pdf.cell(r\_col\_w, line\_height, kwargs['cause'])

return height + line\_height

def services (pdf: FPDF, goods, \*\*kwargs):

MAXIMUM\_PAGE\_WIDTH = 210

PADDING = 20

col1\_w = 8

col3\_w = 20

col4\_w = 18

col5\_w = 18

col2\_w = MAXIMUM\_PAGE\_WIDTH - 2 \* PADDING - col1\_w - col3\_w - col4\_w - col5\_w

height = kwargs['height'] + 6

start = height

total = 0

pdf.set\_line\_width(0.2)

pdf.line(PADDING, height, MAXIMUM\_PAGE\_WIDTH - PADDING, height)

pdf.set\_line\_width(0.2)

pdf.line(PADDING, height + 5, MAXIMUM\_PAGE\_WIDTH - PADDING, height + 5)

pdf.set\_font("TimesNewRomanB", size=9)

pdf.set\_y(height)

pdf.cell(10)

pdf.cell(col1\_w, 5, '№', align='C')

pdf.cell(col2\_w, 5, 'Товары (работы, услуги)', align='C')

pdf.cell(col3\_w, 5, 'Кол-во', align='C')

pdf.cell(col4\_w, 5, 'Цена', align='C')

pdf.cell(col5\_w, 5, 'Сумма', align='C')

pdf.set\_font("TimesNewRoman", size=9)

height += 6

index = 0

for id, good in enumerate(goods):

is\_last = id == len(goods) - 1

index += 1

pdf.set\_y(height)

pdf.cell(10)

pdf.cell(col1\_w, 5, str(index), align='C')

pdf.multi\_cell(col2\_w, 5, good['service'])

pdf.set\_y(height)

pdf.cell(10 + col1\_w + col2\_w)

pdf.cell(col3\_w, 5, f'{math.ceil(good["total"])}{" " + good["total\_unit"] if "total\_unit" in good else ""}', align='L')

pdf.cell(col4\_w, 5, f'{math.ceil(good["coefficient"])} {"p/" + good["total\_unit"] if "total\_unit" in good else "p"}', align='L')

if good['service'] == 'СМС':

pdf.cell(col5\_w, 5, f'{math.ceil(good["coefficient"]) \* math.ceil(good["total"]) - 5} р', align='R')

total += math.ceil(good["coefficient"]) \* math.ceil(good["total"]) - 5

height += 5 \* len(pdf.multi\_cell(col2\_w, 100, good['service'], split\_only=True))

pdf.line(PADDING, height + 1, MAXIMUM\_PAGE\_WIDTH - PADDING, height + 1)

height += 2

continue

pdf.cell(col5\_w, 5, f'{math.ceil(good["coefficient"]) \* math.ceil(good["total"])} р', align='R')

total += math.ceil(good["coefficient"]) \* math.ceil(good["total"])

height += 5 \* len(pdf.multi\_cell(col2\_w, 100, good['service'], split\_only=True))

if not is\_last:

pdf.line(PADDING, height + 1, MAXIMUM\_PAGE\_WIDTH - PADDING, height + 1)

height += 2

else:

height += 1

pdf.set\_line\_width(0.2)

pdf.line(PADDING, start, PADDING, height)

pdf.line(PADDING, height, MAXIMUM\_PAGE\_WIDTH - PADDING, height)

pdf.line(MAXIMUM\_PAGE\_WIDTH - PADDING, start, MAXIMUM\_PAGE\_WIDTH - PADDING, height)

pdf.set\_line\_width(0.2)

pdf.line(PADDING + col1\_w, start, PADDING + col1\_w, height)

pdf.line(PADDING + col1\_w + col2\_w, start, PADDING + col1\_w + col2\_w, height)

pdf.line(PADDING + col1\_w + col2\_w + col3\_w, start, PADDING + col1\_w + col2\_w + col3\_w, height)

pdf.line(PADDING + col1\_w + col2\_w + col3\_w + col4\_w, start, PADDING + col1\_w + col2\_w + col3\_w + col4\_w, height)

pdf.line(PADDING + col1\_w + col2\_w + col3\_w + col4\_w + col5\_w, start, PADDING + col1\_w + col2\_w + col3\_w + col4\_w + col5\_w, height)

height += 5

pdf.set\_font("TimesNewRomanB", size=9)

pdf.set\_y(height)

total\_price = f'{total:,.2f}'.replace(',', ' ')

NDS = f'{total \* 0.167:,.2f}'.replace(',', ' ')

pdf.cell(10)

pdf.multi\_cell(10 + MAXIMUM\_PAGE\_WIDTH - 2 \* PADDING, 5, f'Итого: {total\_price:>15} р.', align='L')

pdf.cell(10)

pdf.multi\_cell(10 + MAXIMUM\_PAGE\_WIDTH - 2 \* PADDING, 5, f'В том числе НДС: {NDS:>15} р.', align='L')

pdf.cell(10)

pdf.multi\_cell(10 + MAXIMUM\_PAGE\_WIDTH - 2 \* PADDING, 5, f'Всего к оплате: {total\_price:>15} р.', align='L')

pdf.set\_font("TimesNewRoman", size=9)

pdf.cell(10)

pdf.multi\_cell(MAXIMUM\_PAGE\_WIDTH - 2 \* PADDING, 5, f'Всего наименований {index} на сумму {total\_price} руб.')

pdf.set\_font("TimesNewRomanB", size=9)

return height + 25

def conditions(pdf: FPDF, \*\*kwargs):

def add\_text(text, height):

MAXIMUM\_PAGE\_WIDTH = 210

PADDING = 20

pdf.cell(10)

height += 4 \* len(pdf.multi\_cell(MAXIMUM\_PAGE\_WIDTH - 2 \* PADDING, 4, text, split\_only=True))

pdf.multi\_cell(MAXIMUM\_PAGE\_WIDTH - 2 \* PADDING, 4, text)

return height

height = kwargs['height'] + 10

pdf.set\_font("TimesNewRoman", size=8)

pdf.set\_y(height)

height = add\_text('Внимание!', height)

height = add\_text('Оплата данного счета означает согласие с условиями поставки товара.', height)

height = add\_text('Уведомление об оплате обязательно, в противном случае не гарантируется наличие товара на складе.', height)

height = add\_text('Товар отпускается по факту прихода денег на р/с Поставщика, самовывозом, при наличии доверенности и паспорта.', height)

MAXIMUM\_PAGE\_WIDTH = 210

PADDING = 20

pdf.set\_y(height)

pdf.set\_font("TimesNewRomanB", size=9)

pdf.cell(10)

pdf.cell(30, 5, 'Руководитель')

pdf.set\_font("TimesNewRoman", size=9)

pdf.cell(60, 5, kwargs['supervisor'], align='R')

pdf.set\_font("TimesNewRomanB", size=9)

pdf.cell(30, 5, 'Бухгалтер', align='C')

pdf.set\_font("TimesNewRoman", size=9)

pdf.cell(MAXIMUM\_PAGE\_WIDTH - 2 \* PADDING - 120, 5, kwargs['accountant'], align='R')

pdf.set\_line\_width(0.2)

pdf.line(PADDING + 35, height + 5, PADDING + 90, height + 5)

pdf.line(PADDING + 120, height + 5, MAXIMUM\_PAGE\_WIDTH - PADDING, height + 5)

return height

pdf = FPDF()

pdf.add\_page()

pdf.add\_font('TimesNewRoman', '', 'fonts/TNR.ttf', uni=True)

pdf.add\_font('TimesNewRomanB', '', 'fonts/TNRB.ttf', uni=True)

now = datetime.datetime.today()

height = banks(pdf,

payee\_bank = 'ООО "Айрон Банк"',

INN = '7722737766',

KPP = '772201001',

payee = 'ОOО "Харвестер"',

BIK = '044525700',

account1 = '30101810200000000700',

account2 = '40702810900000002453',

addressee = 'ООО "Шай-Хулуд"')

height = payment\_invoice(pdf,

account\_number = 42,

day = now.day,

month = '{:02d}'.format(now.month),

year = 20,

height = height)

height = requisites(pdf,

height=height,

executor='ОOО "Смерть за Долги", '

'Вольные Города, г. Браавос, домовладение 10, стр.1',

client='ООО "Шай-Хулуд", ИНН 7714037378, КПП 777550001, 119361, '

'Арракис, равнина Лето II, плато Гейрат',

cause='№ 11223344 от 01.01.2019')

height = services(pdf, [

{

'service': 'Входящие вызовы',

'total': 9.2,

'total\_unit': 'мин.',

'coefficient': 1,

}, {

'service': 'Исходящие вызовы',

'total': 36.23,

'total\_unit': 'мин.',

'coefficient': 1,

}, {

'service': 'СМС',

'total': 5,

'total\_unit': 'шт.',

'coefficient': 1,

}, {

'service': 'Исходящий трафик',

'total': 0.0392,

'total\_unit': 'Мб',

'coefficient': 0.5,

}, {

'service': 'Входящий трафик',

'total': 0.0108,

'total\_unit': 'Мб',

'coefficient': 0.5,

},

], height=height)

height = conditions(pdf,

height=height,

supervisor='Семенов Д.А',

accountant='Семенов Д.А.')

pdf.output('Payment.pdf')

**Вывод:**

В данной Лабораторной работе были изучены принципы и технологии разработки и реализации программного модуля создания и автоматического формирования счета на оплату по данным, полученным в предыдущих работах.