

Санкт-Петербургский НИУ ИТМО

Факультет инфокоммуникационных технологий

Лабораторная работа №6

Вариант №4

Выполнил (и:)

Пономарев К.В.

Рашевский В.В.

Проверил

Мусаев А.А.

Санкт-Петербург,

2022

Задание.

1) Имеется документ в google-таблицах, содержащий информацию о курсе любой валюты, выбранной студентом. Оценить, завышен или занижен её курс, используя авторегрессионную модель. Допускается использовать любые библиотеки.

2) Имеется оформленный шаблон в таблицах google, содержащий описание и пустое поле для некоторых переменных (x , y , z , ...) и функции $f(x, y, z, \dots)$. Пользователь задает созданном интерфейсе название PDF-файла и переменные, после чего происходит расчет функций и все данные сохраняются в заранее подготовленной форме в гугл-таблицах. Аналогичная таблица сохраняется в виде PDF-файла, название которого содержит дату создания файла (ГГГГ-ММ-ДД) и название, заданное в интерфейсе.

3) Создать интерфейс, который предлагает выбрать и открыть ранее созданные PDF файлы (директория задается заранее в программе). Пользователь видит название файла без даты.

Решение.

Весь исходный код можно посмотреть в репозитории на GitHub по следующей ссылке (материалы находятся в разных ветках):

https://github.com/kpkpkpk/python_programming_homework/.

В листингах 1-3 приведены исходные коды заданий.

Листинг 1 – Решение задачи 1

```
from tkinter import *
from tkinter import messagebox
import gspread
import datetime

def alert(title, message, type_of_alert='info'):
    show_method = getattr(messagebox, 'show{}'.format(type_of_alert))
    show_method(title, message)

def finish():
    import pandas as pd
    import matplotlib.pyplot as plt
    global e_time, e_speed, e_name
    flag = False
    try:
        time_in_minutes = str(int(e_time.get()))
        speed = str(int(e_speed.get()))

        cell_c1 = gsheets.acell('B1')
        if str(cell_c1.value) != str(datetime.date.today()):
            gsheets.update('B1', str(datetime.date.today()))
        gsheets.update('B3', time_in_minutes)
        gsheets.update('B4', speed)
        alert('Success!', 'Data successfully updated!')
        flag = True
    except ValueError:
        alert('Error!', 'You should input 2 integers', 'error')
    if flag:
        name = str(datetime.date.today()) + '-' + e_name.get()
        fig, ax = plt.subplots()
        fig.patch.set_visible(False)
        ax.axis('off')
        ax.axis('tight')
        df = pd.DataFrame(gsheets.get_all_records())
        ax.table(cellText=df.values, colLabels=df.columns, loc='center')
        fig.tight_layout()
        plt.savefig(f'{name}.pdf')
        alert('Success!', 'Data successfully saved!')

gc = gspread.oauth()
gsheets = gc.open("MySheet").get_worksheet(0)

window = Tk()
window.title('Last lab ;)')
frame = Frame(master=window)
```

```

label_time = Label(master=frame, text='Time in minutes:')
label_time.grid(row=0, column=0)
e_time = Entry(master=frame)
e_time.grid(row=0, column=1)
label_speed = Label(master=frame, text='Speed in meters per minutes:')
label_speed.grid(row=1, column=0)
e_speed = Entry(master=frame)
e_speed.grid(row=1, column=1)
label_name = Label(master=frame, text='Name of PDF file:')
label_name.grid(row=2, column=0)
e_name = Entry(master=frame)
e_name.grid(row=2, column=1)
btn_confirm = Button(master=frame, text='Confirm')
btn_confirm.grid(row=3, column=0)
btn_confirm.bind('<Button-1>', finish)

frame.pack()
window.mainloop()

```

Листинг 2 – Решение задачи 2

```

import os
from tkinter import *
import re

files = []
file = ''

def finish():
    global file, var, regex
    d = os.path.dirname(os.path.abspath(var.get()))
    os.system(f'start chrome {d}\\{var.get()}')

def create_rbs():
    global var, row, column, regex
    for i in range(len(files)):
        f = files[i]
        text = re.sub(regex, '', f.replace('.pdf', ''), 0)
        rb = Radiobutton(master=frame, text=text, value=files[i],
variable=var)
        rb.select()
        rb.grid(row=row, column=column)
        row += 1

regex = r"\d{4}-\d{2}-\d{2}-"
for x in os.listdir():
    if x.endswith(".pdf"):
        files.append(x)

row = 0
column = 0

window = Tk()
window.title('Last lab ;)')
frame = Frame(master=window)
label = Label(master=frame, text='Which PDF file you want to open?')
label.grid(row=row, column=0)
row += 1

```

```

var = StringVar()
create_rbs()

btn = Button(master=frame, text='Open', command=finish)
btn.grid(row=row, column=0)

window.minsize(250, 100)
frame.pack()
window.mainloop()

```

Листинг 3 – Решение задачи 3

```

import gspread
from statsmodels.tsa.ar_model import AutoReg
from sklearn.metrics import mean_squared_error
from math import sqrt

gc = gspread.oauth()
gsheet = gc.open("MySheet").get_worksheet(1)
data_all = gsheet.get_all_records()
series = []
for i in range(len(data_all)):
    series.append(data_all[i].get('price'))

train, test = series[1:len(series) - 1], series[len(series) - 1:]
# train autoregression
model = AutoReg(train, lags=6) # 6 - 63.89, 7 (max) - 58.82, expected -
62,25
model_fit = model.fit()
print('Coefficients: %s' % model_fit.params)
# make predictions
predictions = model_fit.predict(start=len(train), end=len(train) + len(test)
- 1, dynamic=False)
for i in range(len(predictions)):
    if predictions[i] > test[i]:
        print('predicted=%f, expected=%f' % (predictions[i], test[i]) + ' -
overvalued')
    else:
        print('predicted=%f, expected=%f' % (predictions[i], test[i]) + ' -
undervalued')
rmse = sqrt(mean_squared_error(test, predictions))
print('Test RMSE: %.3f' % rmse)

```

Результат выполнения программы.

Результат работы программ приведены на рисунках 1-4.

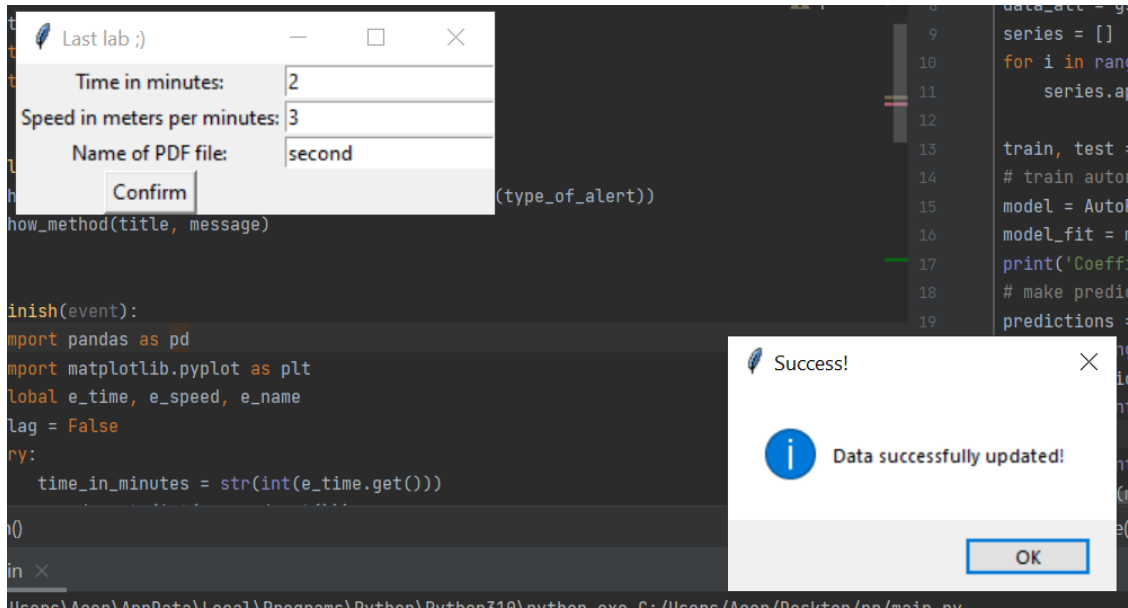


Рисунок 1 – Результат работы программы 1

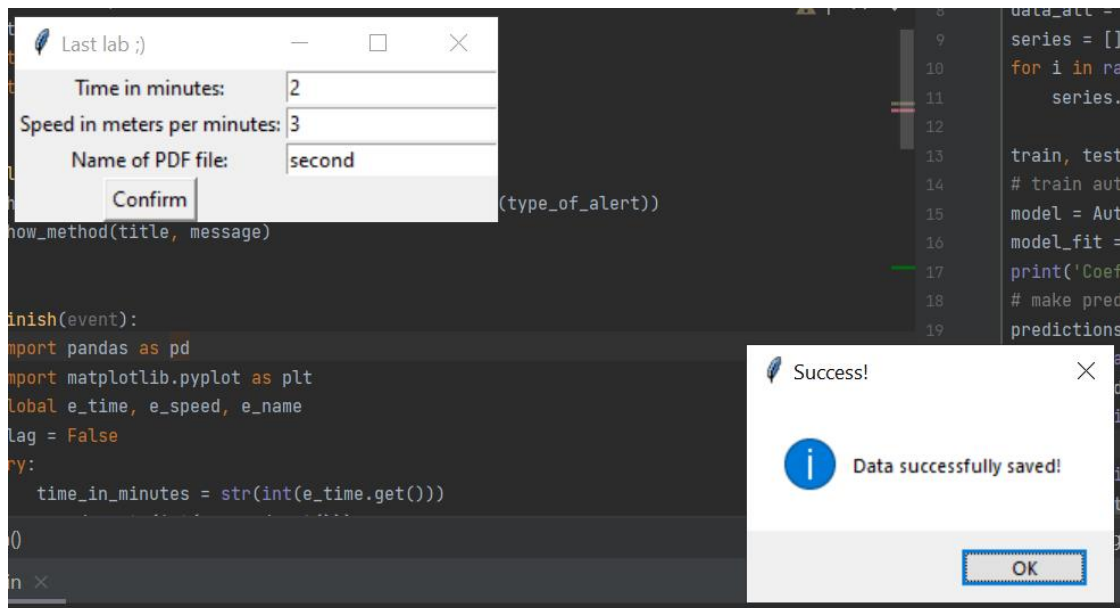


Рисунок 2 – Результат работы программы 1, продолжение 1

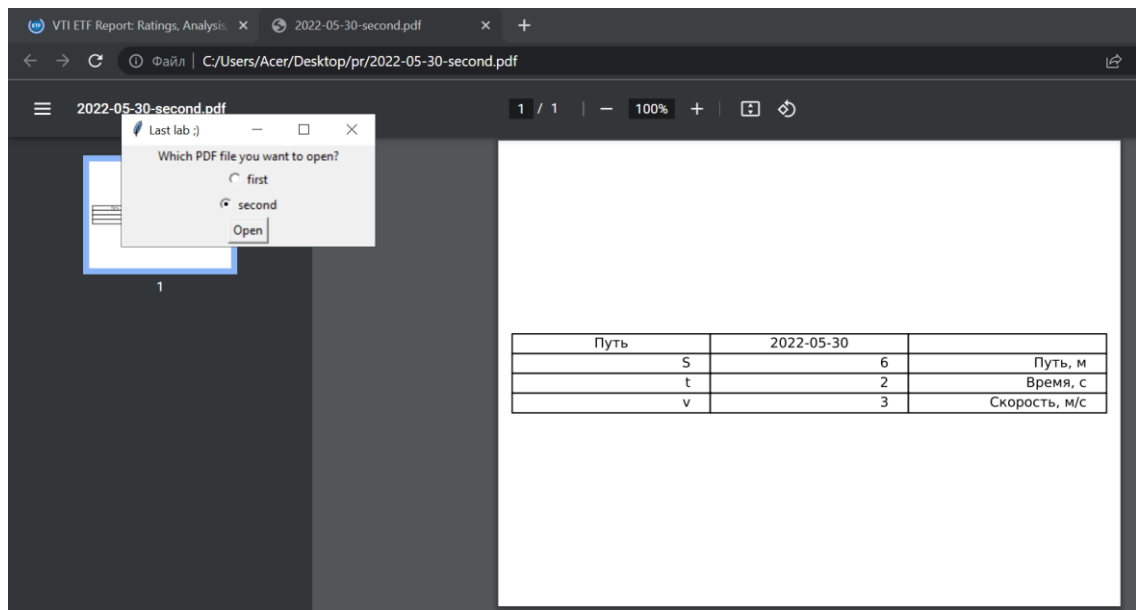


Рисунок 3 – Результат работы программы 2

```

Coefficients: [19.57341115  1.10155803 -0.58937838 -0.19332638  0.27303742 -0.0324394
 0.11865539]
predicted=63.892316, expected=62.250000 - overvalued
Test RMSE: 1.642

Process finished with exit code 0

```

Рисунок 4 – Результат работы программы 3

Выводы.

В ходе выполнения лабораторной работы были написаны 3 программы с графическими интерфейсами, позволяющая работать с google-таблицами, а также определять оценку курса валютной пары USD/RUB. Для построения графических интерфейсов использовалась библиотека tkinter, для работы с данными были использованы библиотеки math и gspread.

Все 3 программы являются полностью рабочими. Все поставленные задачи выполнены в полном объеме.