

**Министерство науки и высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**  
**(Университет ИТМО)**

**Инфокоммуникационные системы и технологии**

**Лабораторная работа №6**

**Выполнил:**

**Таякин Д.Р.**

**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.

На рисунке 1 представлен код программы для первого задания. В качестве модулей использовались statsmodels для авторегрессионной модели, sklearn для нахождения RMSE, gspread для чтения данных из Google Sheets и numpy для удобной работы с данными (массивами).

```
1  from statsmodels.tsa.ar_model import AutoReg
2  from sklearn.metrics import mean_squared_error
3  import gspread
4  import numpy as np
5
6  if __name__ == "__main__":
7      gc = gspread.service_account(filename="./python-college-project-406804075937.json")
8      sh = gc.open("Task 1").sheet1
9      data: list = sh.col_values(2)[1:][::-1]
10
11     train, test = np.array(data[:95]).astype("float"), np.array(data[95:]).astype("float")
12
13     model = AutoReg(train, lags=7)
14     model_fit = model.fit()
15
16     predictions = model_fit.predict(start=len(train), end=len(train) + len(test) - 1)
17
18     print(f"Predicted: {'{:0.2f}'.format(predictions[0])}, Actual: {test[0]}")
19
20     rmse = mean_squared_error(test, predictions) ** 0.5
21     print(f"RMSE is: {rmse}")
22
23     if test[0] > predictions[0]:
24         print("The price is overvalued.")
25     else:
26         print("The price is undervalued.")
27
```

Рисунок 1 – Код для первого задания

На рисунке 2 изображен вывод в командной строке программы.

```
taiakin@Daniels-MacBook-Air Lab-6 % python3 task_1.py
Predicted: 64.55, Actual: 56.41
RMSE is: 8.141309805298519
The price is undervalued.
```

Рисунок 2 – Вывод программы

## Задание 2

Имеется оформленный шаблон в таблицах google, содержащий описание и пустое поле для некоторых переменных (x, y, z, ...) и функции (f(x, y, z, ...)). 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 (пример: «2022-02-02 – кинетическая энергия первый расчет», «2022-02-03 – кинетическая энергия второй расчет»).

На рисунке 3 и 4 представлен код второго задания. В качестве создания интерфейса использовалась библиотека tkinter.

```
1  from tkinter import *
2  from datetime import date
3  import pandas as pd
4  import matplotlib.pyplot as plt
5  import gspread
6
7  gc = gspread.service_account(filename="./python-college-project-406804075937.json")
8  sh = gc.open("Task 2").sheet1
9
10 app_title = sh.get('A1')[0][0]
11
12 first_desc = sh.get('C4')[0][0]
13 second_desc = sh.get('C5')[0][0]
14
15 def App():
16     root = Tk()
17     root.title(app_title)
18     root.geometry("500x200")
19     root.resizable(width=False, height=False)
20
21     frame = Frame(root, bg="#171717")
22     frame.place(relwidth=1, relheight=1)
23
24     Label(frame, text=first_desc, bg="#171717").place(x=100, y=20)
25     first_entry = Entry(frame, width=5)
26     first_entry.place(x=40, y=20)
27
28     Label(frame, text=second_desc, bg="#171717").place(x=100, y=70)
29     second_entry = Entry(frame, width=5)
30     second_entry.place(x=40, y=70)
31
32     Label(frame, text="PDF Filename", bg="#171717").place(x=200, y=120)
33     filename_entry = Entry(frame, width=16)
34     filename_entry.place(x=40, y=120)
35
```

Рисунок 3 – Первая часть кода для второго задания

```

36     def save_click():
37         flag = False
38         sh.update('B4', int(first_entry.get()))
39         sh.update('B5', int(second_entry.get()))
40         sh.update('C1', date.today().strftime("%Y-%m-%d"))
41         flag = True
42
43         if flag:
44             data_frame = pd.DataFrame(sh.get_all_records())
45             filename = date.today().strftime("%Y-%m-%d") + " - " + filename_entry.get()
46
47             fig, axes = plt.subplots()
48             axes.axis("off")
49             axes.table(cellText=data_frame.values, colLabels=data_frame.columns, loc="center")
50             fig.tight_layout()
51             plt.savefig(f"{filename}.pdf")
52
53     Button(frame, text="Save", bg="#171717", borderwidth=0, command=save_click).place(x=40, y=170)
54
55     root.mainloop()
56
57 if __name__ == "__main__":
58     App()

```

Рисунок 4 – Вторая часть кода для второго задания

На рисунке 5 изображен интерфейс программы.

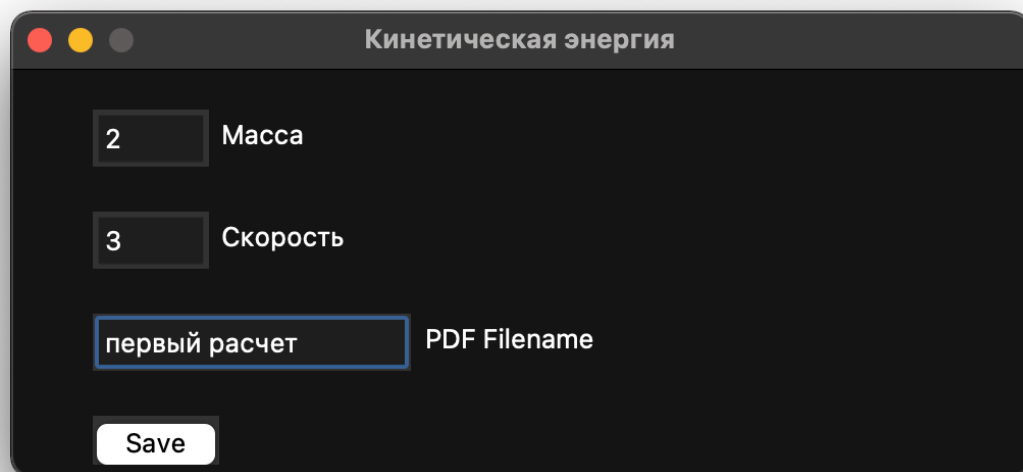


Рисунок 5 – Интерфейс программы для второго задания

### Задание 3

Создать интерфейс, который предлагает выбрать и открыть ранее созданные PDF-файлы (директория задается заранее в программе). Пользователь видит название файла без даты (пример: «кинетическая энергия первый расчет», «кинетическая энергия второй расчет»).

На рисунке 6 представлен код программы для задания 3.

```
1  import os, glob
2  from tkinter import *
3
4  def App():
5      root = Tk()
6      root.title("Открыватель PDF")
7      root.geometry("300x140")
8      root.resizable(width=False, height=False)
9
10     frame = Frame(root, bg="#171717")
11     frame.place(relwidth=1, relheight=1)
12
13     Label(frame, text="Select PDF file that you want to open.", bg="#171717").pack()
14
15     def option_menu(data: list, default_value) -> StringVar:
16         variable = StringVar(frame)
17         variable.set(default_value)
18         OptionMenu(frame, variable, *data).pack()
19         return variable
20
21     file_variable = option_menu(files, "Select file")
22
23     def open_click():
24         value = file_variable.get()
25         filename = glob.glob(f"*{value}.pdf")[0].replace(" ", "\\ ")
26         os.system(f"open {filename}")
27
28     open_btn = Button(frame, text="Open", bg="#171717", borderwidth=0, command=open_click)
29     open_btn.pack()
30
31     root.mainloop()
32
33 if __name__ == "__main__":
34     files = [f[13:-4] for f in glob.glob("*.pdf")]
35     App()
```

Рисунок 6 – Код для третьего задания

На рисунке 7 представлен графический интерфейс данной программы.

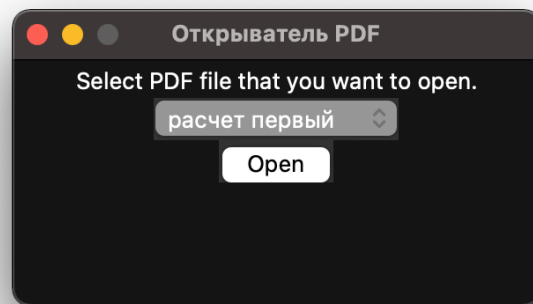


Рисунок 7 – Графический интерфейс программы 3

## **Вывод**

В лабораторной работе №6 были написаны три программы, две из которых имеют графический интерфейс. Данные программы позволяют осуществлять работу с таблицами Google Sheets, оценивать курс соотношения валют USD/RUB при помощи авторегрессионной модели данных.