

Задача 2.1 | Отправка почты через smtplib

```
import smtplib
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart

def send_email(sender_email, sender_password, receiver_email,
subject, message):
    # Создаем объект MIMEMultipart
    msg = MIMEMultipart()
    msg['From'] = sender_email
    msg['To'] = receiver_email
    msg['Subject'] = subject

    # Добавляем текст сообщения в объект MIMEMultipart
    msg.attach(MIMEText(message, 'plain'))

    try:
        # Создаем SMTP-соединение с сервером отправки почты
        with smtplib.SMTP('smtp.gmail.com', 587) as server:
            server.starttls()

            # Авторизируемся на сервере отправки почты
            server.login(sender_email, sender_password)

            # Отправляем сообщение
            server.send_message(msg)

        print("Почта успешно отправлена!")
    except smtplib.SMTPException as e:
        print("Ошибка при отправке почты:", str(e))

# Пример использования
sender_email = "He*****@gmail.com"
sender_password = "***n* *q** *z** **f*"
receiver_email = "Sl*****@gmail.com"
subject = "Привет от Python!"
message = "Привет, это тестовое сообщение."

send_email(sender_email, sender_password, receiver_email, subject,
message)
```

```
C:\msys64\mingw64\bin\python.exe "C:\
Почта успешно отправлена!
|
Process finished with exit code 0
```

Задача 2.2 | Парсинг сайта погоды (wheather HTML parsing) на google.com и/или на простом сайте wttrin с помощью BeautifulSoup (v4).

```
import requests
from bs4 import BeautifulSoup

headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:91.0)
Gecko/20100101 Firefox/91.0'
}

response = requests.get('https://wttr.in/saint-petersburg',
headers=headers)
soup = BeautifulSoup(response.text, 'html.parser')

print(soup.pre.text)
```

Wed 25 Oct			
Morning	Noon	Evening	Night
Cloudy .--. 0(-1) °C .- (). ← 4-6 km/h (_____) 10 km 0.0 mm 0%	Cloudy ... +3(0) °C .- (). ← 8-10 km/h (_____) 10 km 0.0 mm 0%	\ / Clear .-. -1(-5) °C - () - ← 9-17 km/h `-' 10 km / \ 0.0 mm 0%	\ / Clear .-. -2(-5) °C - () - ← 9-17 km/h `-' 10 km / \ 0.0 mm 0%

Thu 26 Oct			
Morning	Noon	Evening	Night
Overcast .--. -1(-6) °C .- (). ← 14-19 km/h (_____) 10 km 0.0 mm 0%	Overcast ... +1(-3) °C .- (). ← 16-21 km/h (_____) 10 km 0.0 mm 0%	\ / Partly cloudy _ /""'-. 0(-5) °C _ (). ← 16-26 km/h /(_____) 10 km 0.0 mm 0%	Overcast .--. -1(-6) °C .- (). ← 15-23 km/h (_____) 10 km 0.0 mm 0%

Fri 27 Oct			
Morning	Noon	Evening	Night
Overcast .--. -1(-6) °C .- (). ← 16-22 km/h (_____) 10 km 0.0 mm 0%	Overcast ... +1(-4) °C .- (). ← 16-20 km/h (_____) 10 km 0.0 mm 0%	_ /""'-. Light snow ,_ (). -1(-5) °C /(_____) ← 13-19 km/h * * * 10 km * * * 0.1 mm 0%	Mist _ _ _ _ -2(-6) °C _ _ _ _ ← 12-19 km/h _ _ _ _ 2 km 0.0 mm 0%

Location: Санкт-Петербург, Центральный район, Санкт-Петербург, Северо-Западный федеральный округ, 190000, РФ [59.9387318,30.3162286]

Задача 2.3 | С помощью библиотеки matplotlib вывести два окна с графиками функций по личному выбору. В одном окне два графика двух разных функций. В другом окне - один график еще одной функции.

```
import matplotlib.pyplot as plt
import numpy as np

# Создаем данные для графиков
x = np.linspace(-10, 10, 100)
y1 = np.sin(x)
y2 = np.cos(x)
y3 = np.tan(x)

# Создаем первое окно с двумя графиками
plt.figure(1)

# Первый график
plt.subplot(211)
plt.plot(x, y1, label='sin(x)')
plt.legend()

# Второй график
plt.subplot(212)
plt.plot(x, y2, label='cos(x)')
plt.legend()

# Создаем второе окно с одним графиком
plt.figure(2)

# Третий график
plt.plot(x, y3, label='tan(x)')
plt.legend()

# Отображаем окна с графиками
plt.show()
```

