МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

**«Московский технический университет связи и информатики» (МТУСИ)**

Кафедра «Системное программирование»

**ОТЧЕТ**

по лабораторной работе № 2 дисциплина **«Введение в информационные технологии»**

Выполнил: Студент гр.БВТ2207

Кузьмин И.А.

“ ” 2023г.

Проверил:

“ ” 2023г.

Москва, 2023

# Введение

# Цель работы:

# Задание:

Создать приложение с метеоинформацией

.

# Реализация

В процессе выполнения лабораторной работы использованы следующие информационные источники [1-2].

**Практическая реализация**

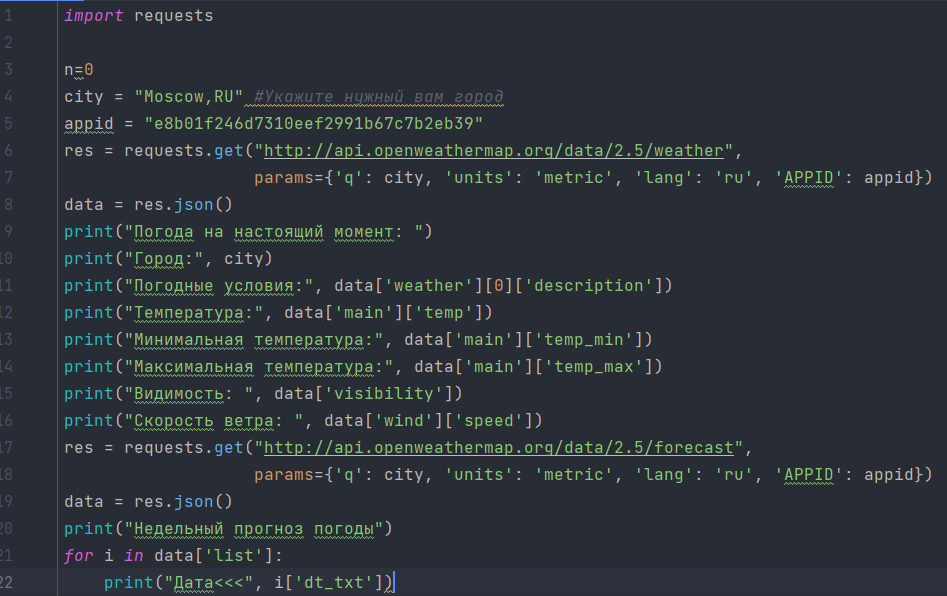


Рис. 1 – код программы

Данный код выполняет запросы к API погоды с использованием библиотеки `requests` и выводит информацию о погоде для указанного города.

1. В начале кода импортируется модуль `requests`, который позволяет отправлять HTTP-запросы к API.

2. Затем определяются переменные: `n` (неиспользуемая переменная), `city` (название города, для которого будет запрашиваться погода), и `appid` (API-ключ, который необходим для доступа к сервису погоды).

3. С помощью функции `requests.get()` выполняется GET-запрос к API погоды. Первый запрос отправляется на URL "http://api.openweathermap.org/data/2.5/weather", с указанием параметров запроса: `q` (город), `units` (единицы измерения температуры), `lang` (язык ответа) и `APPID` (API-ключ). Результат запроса сохраняется в переменную `res`.

4. После получения ответа, функция `res.json()` преобразует ответ в формате JSON в структуру данных Python. Результат сохраняется в переменную `data`.

5. Далее выводится информация о текущей погоде: название города, описание погодных условий, текущая температура, минимальная и максимальная температуры, видимость и скорость ветра.

6. Затем выполняется второй запрос к API с помощью функции `requests.get()`, на URL "http://api.openweathermap.org/data/2.5/forecast" для получения недельного прогноза погоды.

7. Результат запроса преобразуется в формат JSON и сохраняется в переменную `data`.

8. Далее происходит вывод даты и времени прогноза погоды для каждого элемента в списке `data['list']`.

Обратите внимание, что для работы кода необходимо иметь действующий API-ключ (`appid`) и указать нужный город (`city`).

# Заключение

# Список использованных источников