# **Лабораторна робота №3**

**Створення геоінформаційної системи, що виконує найпростіші операції з аналізу географічних даних**

**Мета роботи:** створити ГІС, що реалізує в інтерактивному режимі функції аналізу географічних даних з попередньої лабораторної роботи.

## Хід роботи

У рамках даної лабораторної роботи для реалізації інтерактивної мапи з декількома шарами було обрано мову програмування Python і середовище розробки Jupyter Notebook.

### **Імпорт необхідних бібліотек**

Імпортували основні бібліотеки, класи та функції, які в подальшому були використані для реалізації завдання. Для кожного імпорту у коментарі поруч коротко описали їх необхідність (рисунок 3.1).

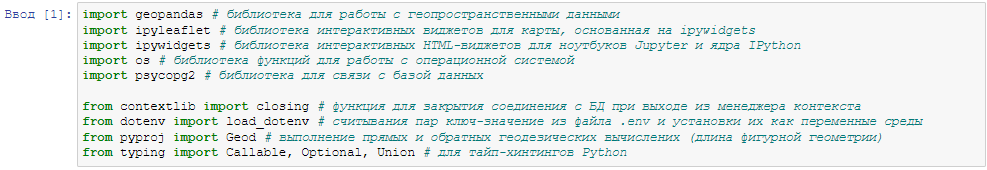


Рисунок 3.1 – Імпорт необхідних бібліотек

### **Завантаження даних про будівлі міста Познань**

В якості дослідження функціоналу бібліотеки *geopandas* виконали завантаження даних про будівлі міста Познань не з відповідної таблиці з бази даних, а напряму з shape-файлу (рисунок 3.2). Додатково було додано стовпчик id\_building, що подібний до первинного ключа таблиці БД. Оскільки отриманий набір даних містить більше 200 тисяч записів, то для оптимального навантаження на карту та тестування обрали 20 тисяч випадкових.

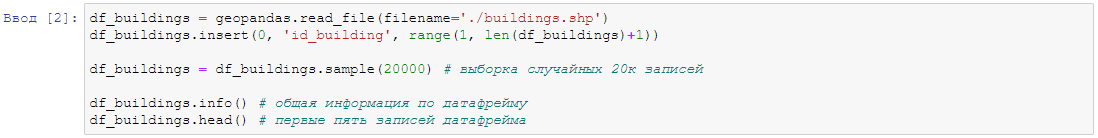


Рисунок 3.2 – Завантаження інформації напряму з shape-файлу

Окрім цього переглянули загальну інформацію по набору даних (кількість записів, тип даних у стовпцях тощо) та перші п’ять записів отриманого датасету (рисунок 3.3).

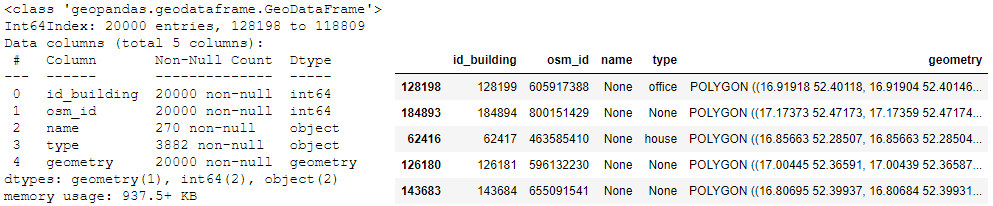


Рисунок 3.3 – Загальна інформація по отриманому датафрейму

### **Створення об’єкту інтерактивної мапи**

Далі перейшли до створення об’єкту інтерактивної мапи: в якості базової мапи визначили open-source мапу OpenStreetMap, встановили значення стартової до показу локації та максимального зуму. Також були додані панель контролю шарів, на якій можна «приховувати» певні шари, та можливість відкриття мапи у повноекранному режимі (рисунок 3.4).

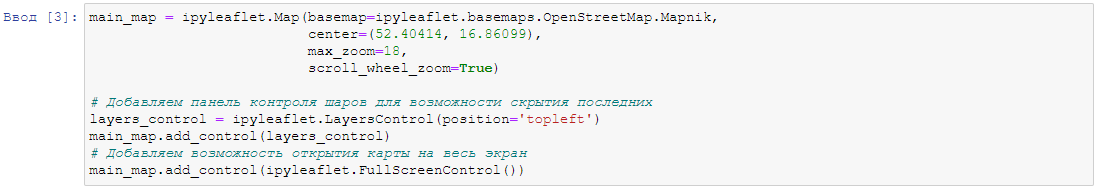


Рисунок 3.4 – Створення об’єкту мапи

### **Додавання шару з даними по будівлям міста Познань**

Одразу у якості першого користувацького шару додали шар з даними по будівлям міста Познань, які були отримані вище. Окрім цього було визначено назву та такі стильові характеристики шару як колір, прозорість, колір при наведенні миші тощо (рисунок 3.5).

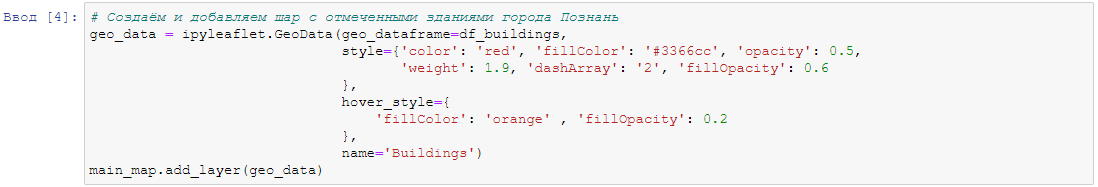


Рисунок 3.5 – Додавання шару з даними по будівлям міста Познань

### **Створення та налаштування шару маркерів**

Для зручності користувача визначати точки для побудови маршрутів між будівлями, створили кластер маркерів. Окрім цього визначили наступні функції: додавання та видалення маркерів, реакції на клік по карті, на переміщення маркера по карті та на подвійний клік по маркеру (рисунок 3.6).



Рисунок 3.6 – Створення та налаштування шару маркерів

### **Створення кнопок керування та поля виводу**

Для того, щоб користувачу було легко згенерувати чи видалити маршрут за заданими маркерами, а також переглянути його довжину у метрах, додали відповідні кнопки “Build Route!” і “Remove Route!” і недоступне для заповнення користувачем поле виводу (рисунок 3.7)

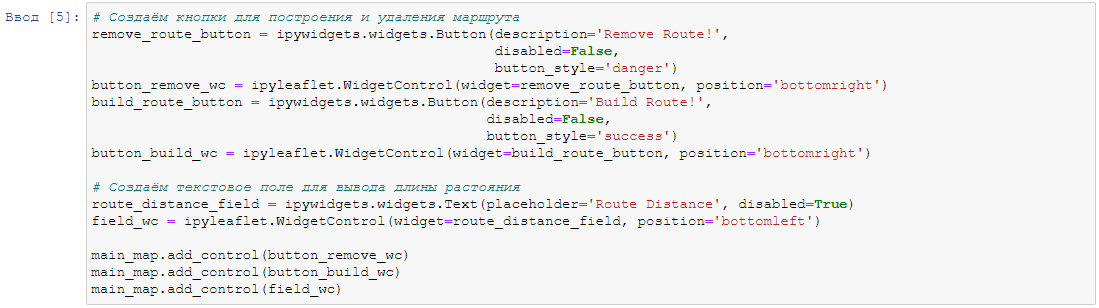


Рисунок 3.7 – Створення кнопок керування та поля виводу

### **Реалізація шару побудованого маршруту**

Для того, щоб побудувати маршрут, вирішили використати збережену функцію бази даних, яка повертає маршрут у форматі *geometry* для подальшої обробки. В рамках даного пункту створили функції-реакції при натисканні на кожну з двох кнопок “Build Route!” і “Remove Route!”, а також функції, які генерують параметр для збереженої функції, викликають її та відображають отриманий маршрут на мапі (рисунок 3.8).



Рисунок 3.8 – Реалізація шару побудованого маршруту

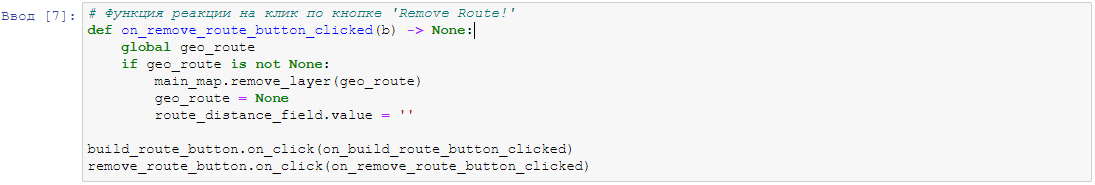


Рисунок 3.8 – Реалізація шару побудованого маршруту (продовження)

### **Тестування інтерактивної мапи**

Останнім етапом стало тестування розробленої інтерактивної мапи. Нижче на рисунку 3.9 наведено її початковий вигляд.

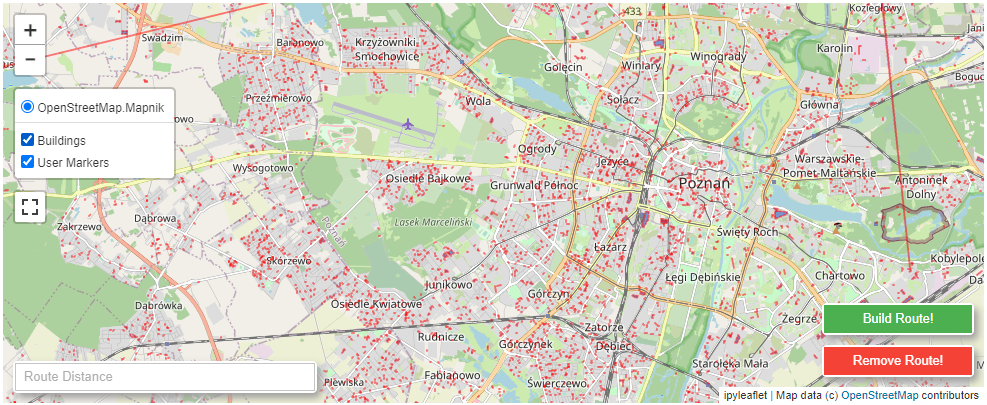


Рисунок 3.9 – Початковий вигляд інтерактивної мапи

Додали декілька маркерів на будівлі (рисунок 3.10). При одноразовому натисканні на маркер над ним з’являється поп-ап, який містить довготу та широту позиції маркера.

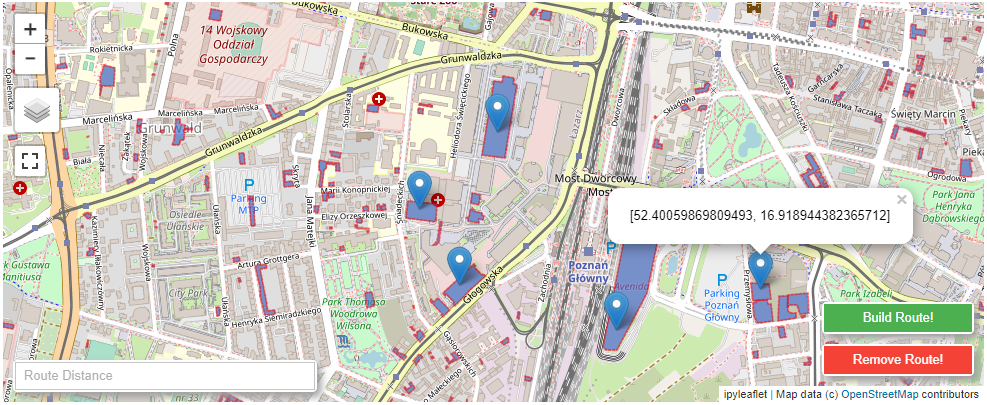


Рисунок 3.10 – Додавання маркерів на інтерактивну мапу

Після встановлення маркерів натиснули кнопку “Build Route!”, після чого на мапі з’являється зелена ламана – розрахований маршрут через центри будівель, а у текстовому полі – його довжина (рисунок 3.11).

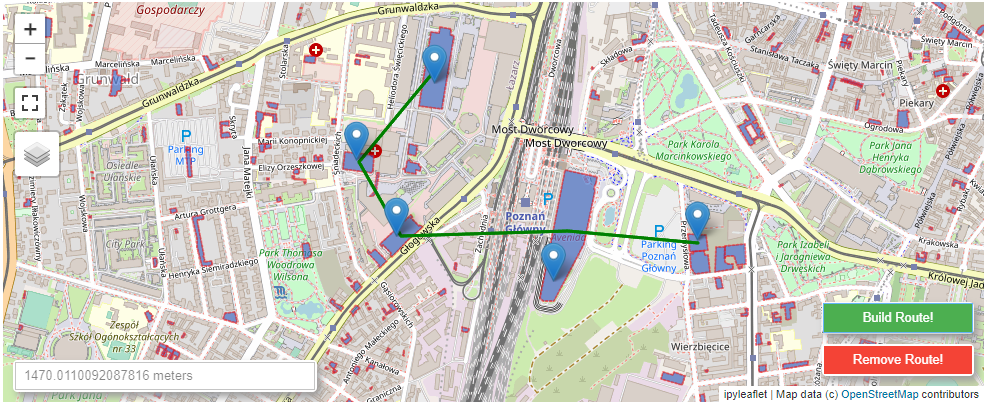


Рисунок 3.11 – Побудова маршруту на основі маркерів користувача

## Висновки

У ході лабораторної роботи №3 ми засобами мови програмування Python у середовищі розробки Jupyter Notebook розробили інтерактивну мапу з декількома шарами, використавши функції аналізу географічних даних з попередньої лабораторної роботи.