



empenoso 14 минут назад Программирование на python ...

С бумаги на цифровую карту: генерация файла из таблицы для импорта на карту и геокодирование адресов с помощью Python

Сразу возникает вопрос - кому в 2024 году может понадобится переносить данные с бумажного носителя на цифровой, ведь большинство данных уже в цифровом виде. Тем не менее есть реальная задача. В исходных данных - растровая картинка проекта в виде таблицы с географическими координатами, выраженными в градусах, минутах и секундах, а на выходе должно получиться текстовое описание маршрутов с длинами и карта с точками и сегментами.

Предстоящие действия включают следующие шаги: из бумажного проекта взять таблицу с географическими координатами предстоящей застройки, оцифровать эти данные, а затем с помощью Python скрипта создать GPX-файл с точками и отрезками для нанесения на карту.

Затем, создав другой Python-скрипт, провести геокодирование координат для получения текстовых описаний с адресами и автоматически рассчитать расстояния между точками и сегментами.

Все эти действия гораздо быстрее ручного нанесения точек на карту и ручного подсчёта расстояний.

Исходные данные

82

Приложение 1

Географические координаты угловых точек участка предстоящей застройки.

Система координат WGS-84

№	СШ			ВД		
	Градусы	Минуты	Секунды	Градусы	Минуты	Секунды
1	56	53	40,309	54	32	41,3743
2	56	53	41,7976	54	32	46,1843
3	56	53	43,9318	54	32	54,6035
4	56	53	38,4788	54	33	4,3316
5	56	53	34,6804	54	33	10,4538
6	56	53	31,3372	54	33	17,4818
7	56	53	30,2328	54	33	25,2762
8	56	53	29,5115	54	33	29,2554

#Конкурс разработч...

Войти

ВойтиСоздать аккаунтЗабыли пароль?или продолжите с Войти с Яндекс ID Войти через VK ID Войти через Google Промокоды Работа Курсы Реклама Игры Пополнение SteamО сообществе сообщества

Лист проекта

В документе содержатся растровые изображения таблиц с географическими координатами планируемой застройки. Есть несколько проектов, причем все изображения вставлены одинаково небрежно – с наклоном только в разные стороны.

Поскольку страниц, содержащих точки не так много - всего по две страницы на проект, то выбрал использовать телефон с Google Lens (Гугл Объектив), вместо специализированной программы для оптического распознавания символов.

С помощью Google Объектив, закрывая двумя кусочками страницы соседние столбцы можно легко и корректно распознать полностью всю таблицу. Это быстро и является хорошим вариантом при отсутствии сканера.



Программирование...

694 поста • 11К по...

Подписаться



Сообщество любителей языка программирования python и просто неравнодушных к нему. Обучение программированию на языке python. Исследование его интересных возможностей и библиотек.

[Правила сообщества](#) ▾

Управление сообществом



rissy Администратор

Топ прошлой недели

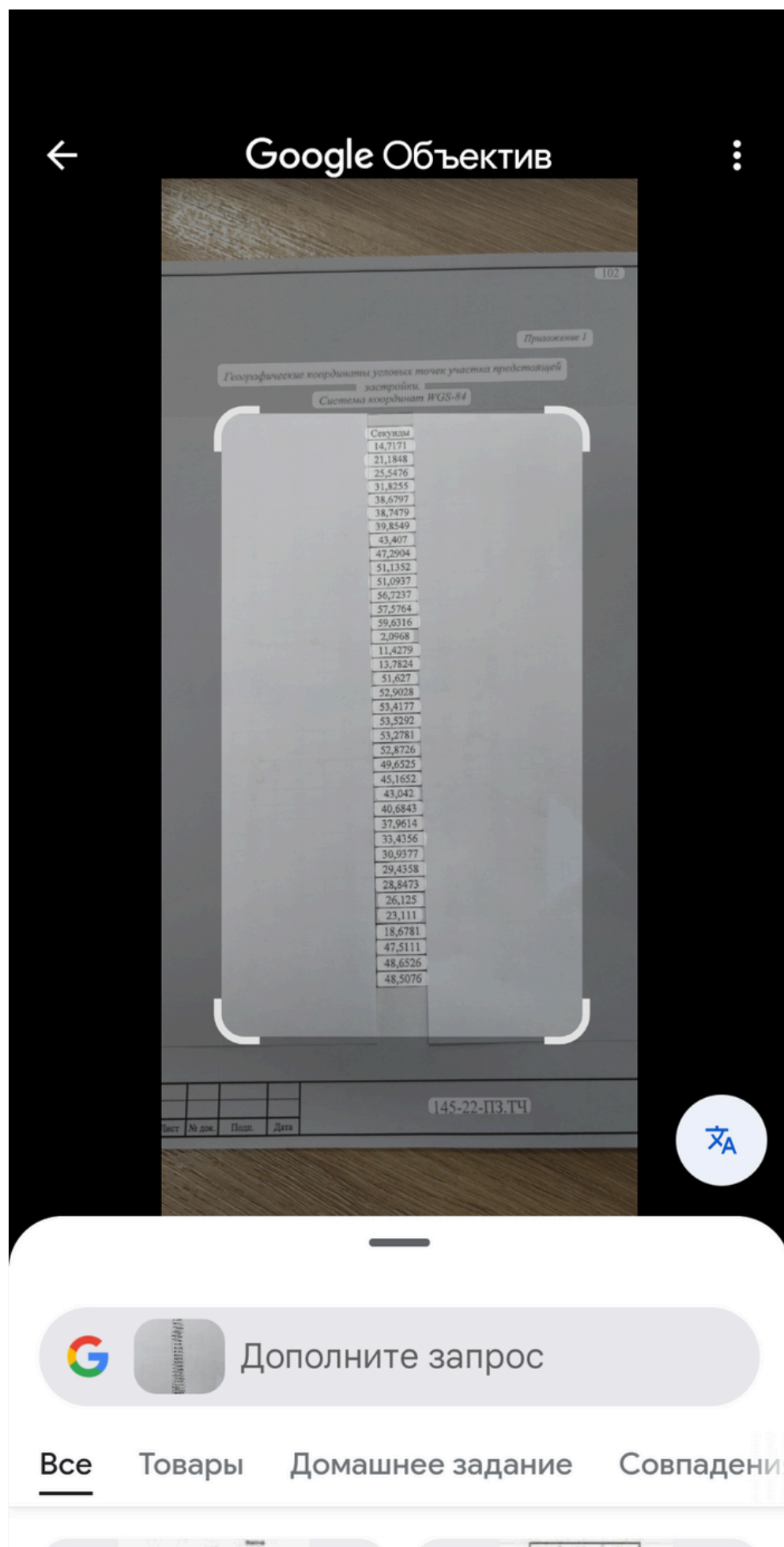


ZaTaS

4 поста



Kllirik.kir



2 поста

Animalrescued

32 поста

[Посмотреть весь топ](#)

Лучшие посты
недели



Рассылка Пикабу:
отправляем самые
рейтинговые материалы за 7
дней 🔥

Укажите

[Подписаться](#)

Нажимая кнопку
«Подписаться на рассылку»,
я соглашаюсь с [Правилами](#)
[Пикабу](#) и даю согласие на
обработку персональных
данных.

Новости Пикабу
Помощь
Кодекс Пикабу

Верификации
Награды
Контакты

№	№	№
61	226981	28
62	238328	26
63	250047	18
64	262144	19
65	274625	26
66	287496	34
67	300743	19
68	314432	17

No	№	№
27	-643 3178	
33	-775 3157	
26	-827 3231	
25	-889 2886	

70	343000	10
71	357911	24
72	372245	27
73	387057	26
74	402328	17
75	418076	27
76	434332	14

22	-957 3211	
23	-970 3287	
4	-1036 3055	
17	-1037 3079	
79	-1066 3148	

Google Объектив для распознавания таблицы

Подготовка данных

Мне показалось правильным перевести градусы, минуты и секунды в десятичные градусы следующим образом:

Десятичные градусы = градусы + (минуты / 60) + (секунды / 3600)

Провёл все вычисления в таблице:

Далее в Notepad++ при помощи макросов привёл данные к неизменяемому виду данных в Python, который используется для хранения упорядоченной последовательности элементов. Такая запись в Python называется кортежем (tuple). Кортеж представляет собой неизменяемый упорядоченный набор элементов, заключённых в круглые скобки. Каждый элемент кортежа отделяется запятой.

Реклама
О компании

О проекте
Зал славы

Промокоды
Скидки
Работа
Курсы
Блоги

Купоны Мегамаркет
Купоны Спортмастер
Купоны М.Видео
Купоны Aliexpress
Купоны Lamoda

Мобильное приложение



Генерация GPX файла

После того как точки получились в простом текстовом виде написал Python код, который генерирует GPX файл с этими точками. Рассматривал ещё KML файл, но в целом скорее всего без разницы какой формат выбрать для этой промежуточной стадии.

GPX (GPS eXchange Format) - это формат хранения и обмена данными устройств позиционирования GPS. Был создан в 2002, файл может содержать различные элементы, такие как треки `<rte>` и путевые точки `<trk>`.

Visual Studio Code

Python код генерации. Скрипт начинается с настройки среды для обработки выходных данных в кодировке UTF-8 и импортирует необходимую библиотеку XML (`xml.etree.ElementTree`). Это гарантирует, что выходной файл и любые сообщения терминала будут правильно обрабатывать специальные символы. В самом начале идёт определение данных:

- **Координаты:** определяется список пар широты и долготы. Каждая координата соответствует определенной географической точке. Скрипт начинает нумерацию

этих точек с 1, хотя в начале добавляется неиспользуемая точка-заполнитель для целей индексации.

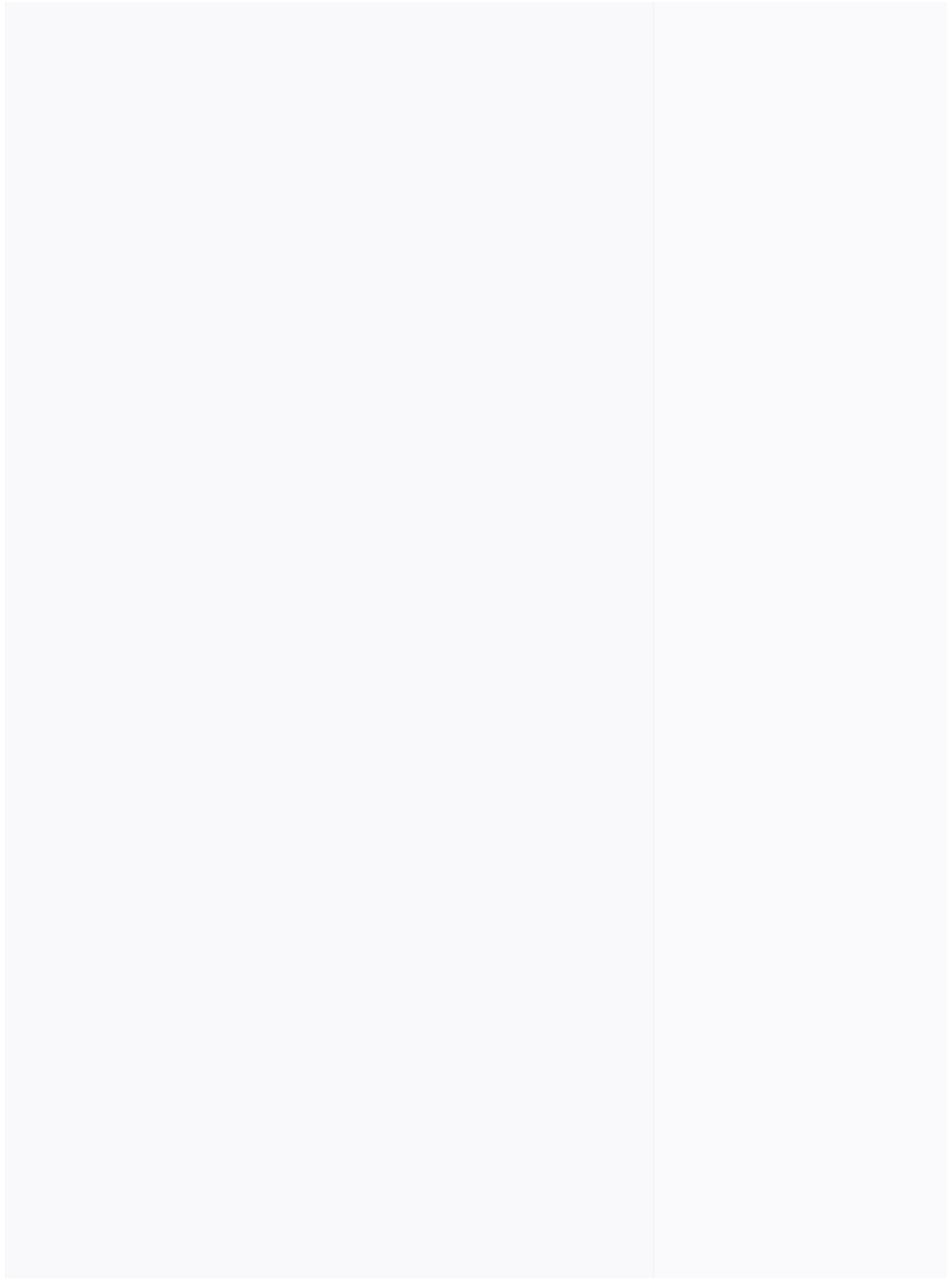
- **Сегменты:** Набор списков определяет «треки» или «маршруты», которые являются последовательностями точек, представленных их индексами в списке координат.

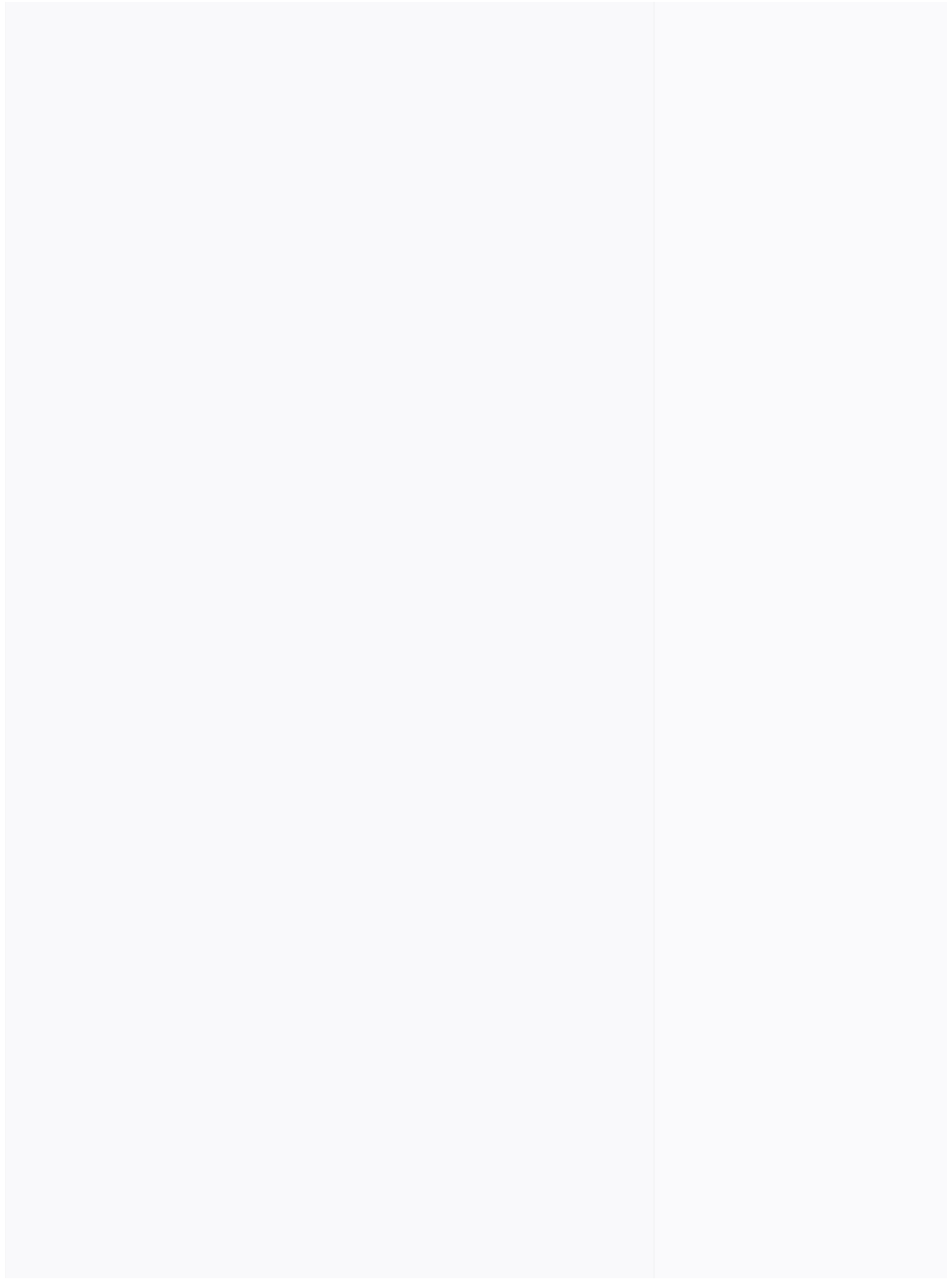
Дальше скрипт использует библиотеку XML для построения структурированного файла GPX

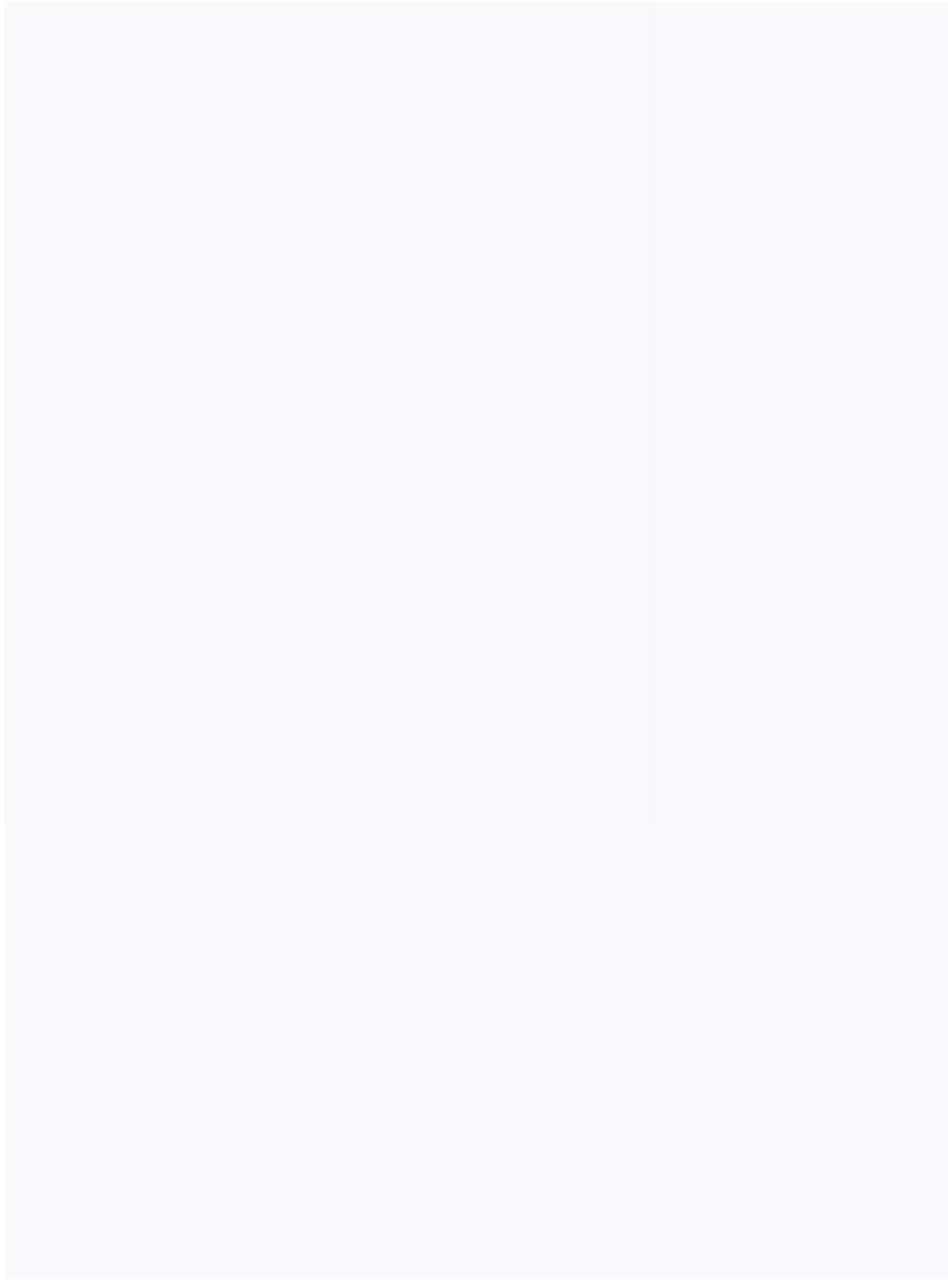
- **Путевые точки:** каждая координата добавляется как элемент (путевая точка). Вложенный элемент назначает метку, например «Точка 1», «Точка 2» и т. д.
- **Треки:** список segments используется для определения элементов (трек). Каждый трек имеет для идентификации (например, «Сегмент 1») и содержит последовательность элементов (точка трека), соответствующих индексам в сегменте. Они также включают элементы для маркировки.

Сконструированное дерево XML сохраняется в файле с именем output.gpx с кодировкой UTF-8 и декларацией XML. Подтверждающее сообщение выводится на консоль. Точки приведены просто как пример:

На выходе GPX файл:







Отображение GPX-файла с точками и отрезками на карте

GPX файл импортировал в [SAS.Planet.Release.241111](#) для отображения на нужных слоях карты.

SAS.Planet - это программа с открытым программным кодом для скачивания спутниковых снимков и карт из различных онлайн-сервисов, таких как Google Maps, Яндекс.Карты и другие. Она позволяет сохранять карты и снимки высокого разрешения на локальный компьютер для последующего использования без доступа к интернету.

Яндекс Карта и [Rosreestr.ru](https://rosreestr.ru) кадастровые границы

На карте выбраны слои Яндекс Карта и [Rosreestr.ru](https://rosreestr.ru) кадастровые границы - на них наложены точки и получившиеся сегменты пути.

Из SAS.Planet можно можно сохранить и распечатать слои с наложенными на них точками в любом формате включая A0 и A1.

Геокодирование координат для получения текстовых описаний с адресами и автоматический расчет расстояний между точками и внутри сегментов

Написал Python код, который производит геокодирование координат для получения текстовых описаний с адресами и делает автоматический расчет расстояний между точками и внутри сегментов. Использовал две библиотеки:

Shapely - это библиотека Python для создания, анализа и манипулирования геометрическими фигурами, такими как точки, линии и многоугольники. Она отлично справляется с пространственными операциями, такими как расчет расстояний, поиск пересечений и проверка взаимосвязей между геометриями.

Geopy, с другой стороны, ориентирована на геокодирование и геопространственные вычисления. Она преобразует адреса в географические координаты и наоборот, а также может вычислять расстояния между местоположениями, используя различные геодезические методы.

Вместе эти библиотеки предоставляют мощный набор инструментов для обработки и анализа геопространственных данных.

Visual Studio Code

Код Python скрипта. В самом начале задаются:

- **Координаты:** список пар широты и долготы представляет различные географические точки. Первая запись — это заполнитель для выравнивания индексации с удобной для восприятия нумерацией.
- **Сегменты:** это группы точек, идентифицированных по их индексам, которые образуют непрерывные линии или пути.

Дальше библиотеки:

- `shapely.geometry`: используется для создания геометрических представлений, таких как линии, образованные путем соединения координат.
- `геору`: предоставляет инструменты для расчета расстояний и геокодирования (преобразования координат в адреса).
- `Nominatim`: геокодер из `OpenStreetMap`, используемый для обратного геокодирования координат в удобные для восприятия адреса.

Основные функции

- Обратное геокодирование: функция `reverse_geocode` преобразует широту и долготу в адреса. Она корректно обрабатывает ошибки, возвращая соответствующее сообщение, если адрес не может быть найден или если есть исключение.
- Расчет расстояния: функция `geodesic` из `геору` вычисляет расстояние между последовательными точками в метрах.

Для каждого сегмента создаётся отчёт:

1. Информация о пути:

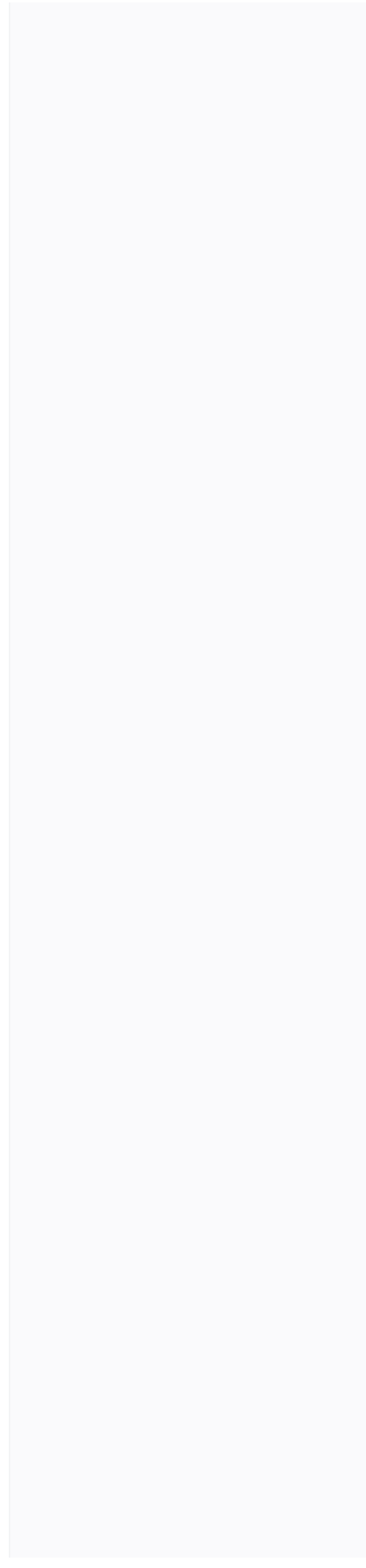
- Точки, образующие сегмент, соединяются в линию (`LineString`), а общая длина пути вычисляется путем суммирования расстояний между последовательными точками.
- Эта информация форматируется в виде описания.

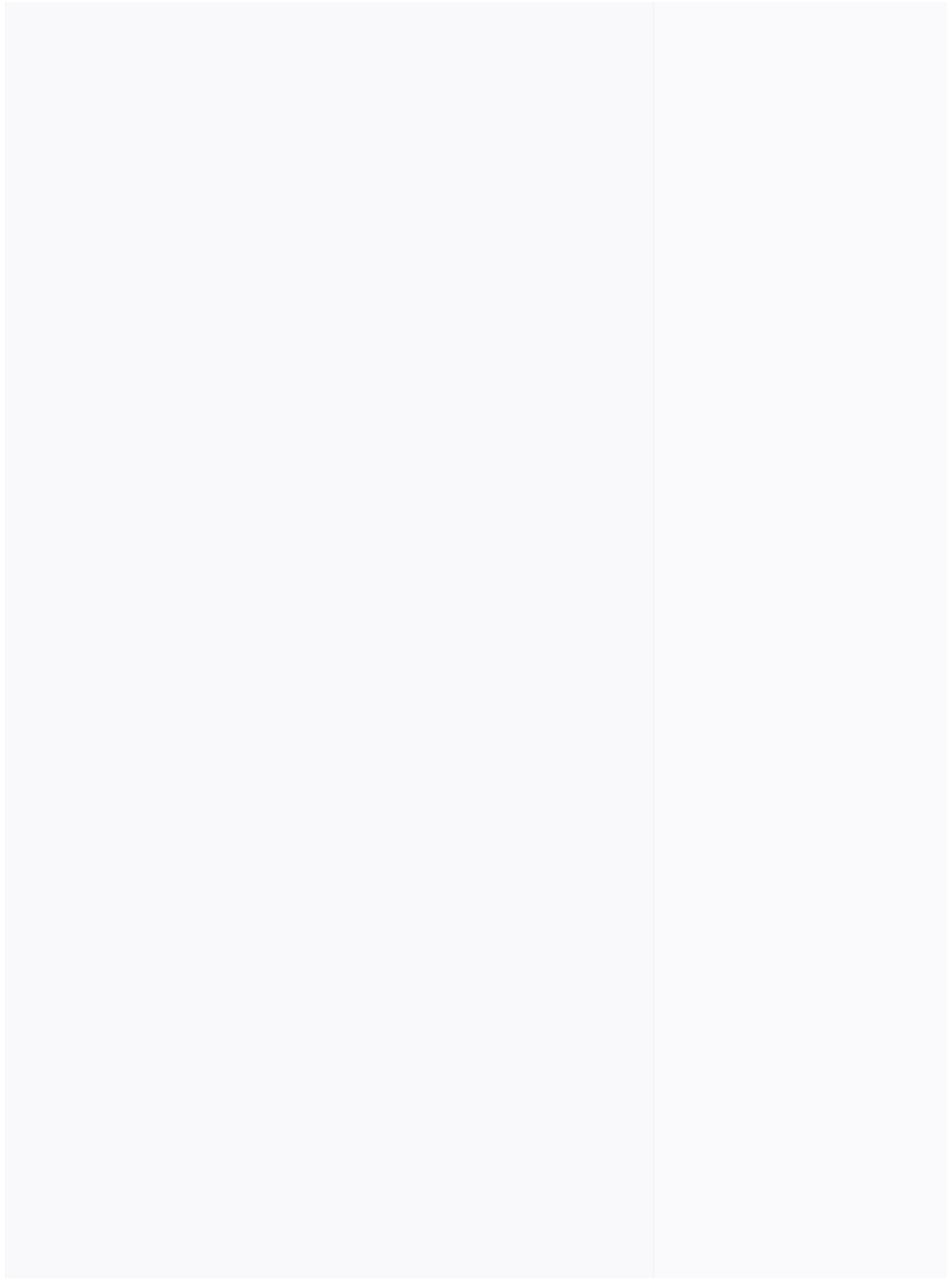
1. Сведения о точке:

- Каждая точка в сегменте подвергается обратному геокодированию для предоставления адреса, удобного для восприятия человеком.
- Расстояние между каждой парой последовательных точек вычисляется и включается в отчет.

Точки приведены просто как пример:

Результат выполнения скрипта:





Такой текстовый результат полностью устраивал.

Итоги

Гораздо быстрее получилось создать автоматизацию для получения текстовое описания маршрутов с длинами и отображения карты с точками и сегментами, чем вручную наносить точки на карту и вручную делать расчёт расстояний.

Автор: [Михаил Шардин](#),

12 декабря 2024 г.



0 316



₽ Поддержать

Эмоции



Программирование на python

694 поста • 11.9K подписчика

[Добавить пост](#)[Подписаться](#)

Правила сообщества

Публиковать могут пользователи с любым рейтингом. Однако!

Приветствуется:...

[Подробнее](#) ✓

Чтобы оставить комментарий, необходимо [зарегистрироваться](#) или [войти](#)













