Тестовое задание

## Алгоритм программы:

1. Получение ссылки
2. Проверка на корректность (status\_code == 200?)
3. Сбор заголовка и текста
4. Форматирование текста
5. Создание папки и файла
6. Сохранение текста в файл
7. Вывод пути к сохраненному файлу

## Инструкция к запуску программы:

Для начала, необходимо удостовериться, что все нужные библиотеки установлены или установить их. Сделать это можно так:

pip install -r requirements.txt

## Запуск происходит командой:

python3 main.py url , где url – ваша ссылка

## Исходный код и примеры работы:

<https://github.com/temakolpakov/tensor_test>

Ссылки, на которых проверялась работа:

<https://lenta.ru/news/2021/02/06/piter/>

<https://www.gazeta.ru/social/2021/02/06/13469606.shtml?updated>

<https://stackoverflow.com/questions/17911091/append-integer-to-beginning-of-list-in-python>

<https://tproger.ru/articles/zadanija-s-sobesedovanija-na-poziciju-middle-backend-razrabotchika-s-primerami-na-python/?utm_medium=messenger&utm_source=telegram>

Направление улучшения/развития:

В программе использован довольно простой парсер, который работает на большинстве сайтов, однако не на всех. Парсер вытаскивает все теги <p>, однако, бывают сайты, на которых не используются такие теги. Например, на https://habr.com/ru/, используется текст без тегов, разделенный тегами <br> (не самая хорошая практика). Если рассматривать такие сайты, то в дополнение к текущему парсеру, нужно встраивать распознаватель тегов <br> и определитель содержания между такими тегами. Также, можно отбрасывать теги footer итд, но не все сайты используют их, некоторые делают так <div class="layout\_\_row layout\_\_row\_footer-links">. Парсер - очень индивидуальный инструмент и требует настройки под определенный сайт.