BIGDATA TEAM

#W1L2PA: Inverted Index Library

1. Описание задания

В этом задании вам нужно расширить библиотеку по работе с инвертированным индексом. Правила:

- TDD - сначала тесты, потом реализация;
- Рефакторинг не смешиваем с добавлением функциональности;
- Fuzz/Stress Testing для валидации эффективной реализации.

2. Инвертированный индекс (Inverted Index)

См. описание в W1L1PA.

3. Описание данных

3.1 Дамп Википедии

- Формат: текст
- В каждой строке находятся следующие поля, разделенные знаком табуляции:
    1. `INT` - id статьи,
    2. `STRING` - текст статьи,

Пример:

```
12    Anarchism         Anarchism is often defined as a political
philosophy which holds the state to be undesirable, unnecessary, or
harmful.
```

3.2 Стоп-слова

- Формат: одно стоп-слово на строчку

Пример:

```
...
wherein
whereupon
wherever
...
```

4. Задания

1. Добавьте все тесты, которые были представлены в видеоуроках, но не были реализованы вами ранее. Добейтесь уровня покрытия тестами 80%.
2. Рефакторинг: предоставьте возможность расширять функционал библиотеки инвертированного индекса с помощью Стратегии. Добавьте базовый класс стратегии реализации load/dump инвертированного индекса:
   - github:big-data-team/python-course/.../storage_policy.py

   Создайте класс JsonStoragePolicy и перенести необходимую функциональность туда. Убедитесь, что все тесты проходят. Следите за качеством кода с помощью pylint (см. https://github.com/big-data-team/python-course#howtos).
3. Реализуйте дополнительные стратегии хранения индекса с помощью библиотек pickle и zlib. Напишите тесты для проверки валидности реализации на основе JsonStoragePolicy.
4. Поделитесь в канале группы, какого сжатия удалось добиться с помощью выбранных библиотек по сравнению с json (без учета или с использованием стоп-слов). Добавляйте хеш-теги: #W1L2PA #inverted_index #compression_challenge #stop_words_removed / #stop_words_present.