|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

|  |  |
| --- | --- |
| **ФАКУЛЬТЕТ** | **ИУК «Информатика и управление»** |
| **КАФЕДРА** | **ИУК4 «Программное обеспечение ЭВМ,** |
| **информационные технологии»** | |

**Лабораторная работа №4**

**«Язык Pig Latin»**

**ДИСЦИПЛИНА: «Технологии обработки больших данных»**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил: студент гр. ИУК4-72Б | |  |  | ( | Сафронов Н.С. | ) |
|  |  |  | (подпись) |  | (Ф.И.О.) |  |
| Проверил: | |  |  | ( | Голубева С.Е. | ) |
|  |  |  | (подпись) |  | (Ф.И.О.) |  |

|  |  |
| --- | --- |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | |
|  | - Балльная оценка:  - Оценка: |

Калуга, 2023

**Цель работы:** формирование практических навыков реализации pig-скриптов для обработки больших данных.

**Постановка задачи**

**Вариант 4**

**Задание 1**

Построить индекс файла. Для каждого слова в файле результат должен содержать номера всех строка, в которых появляется данное слово. Индекс должен быть регистро-независимым. Результат должен быть сохранен в файле в виде: ((word1 (1 42 58)), (word2 (34, 55, 776, 3456), …).

**Задание 2**

База данных твитов состоит из двух файлов. Выполить задание по варианту, используя Pig Latin. Файл tweets.csv имеет формат: tweet\_id, tweet, login Файл users.csv имеет формат: login, user\_name, state Файлы: tweets.csv, users.csv

Вывести имена пользователей, опубликовавших хотя бы 2 твита. Отсортировать результат по активности пользователя (пользователи с наибольшим числом твитов должны быть вверху списка).

**Ход выполнения работы**

**Задание 1**

**Листинг программы**

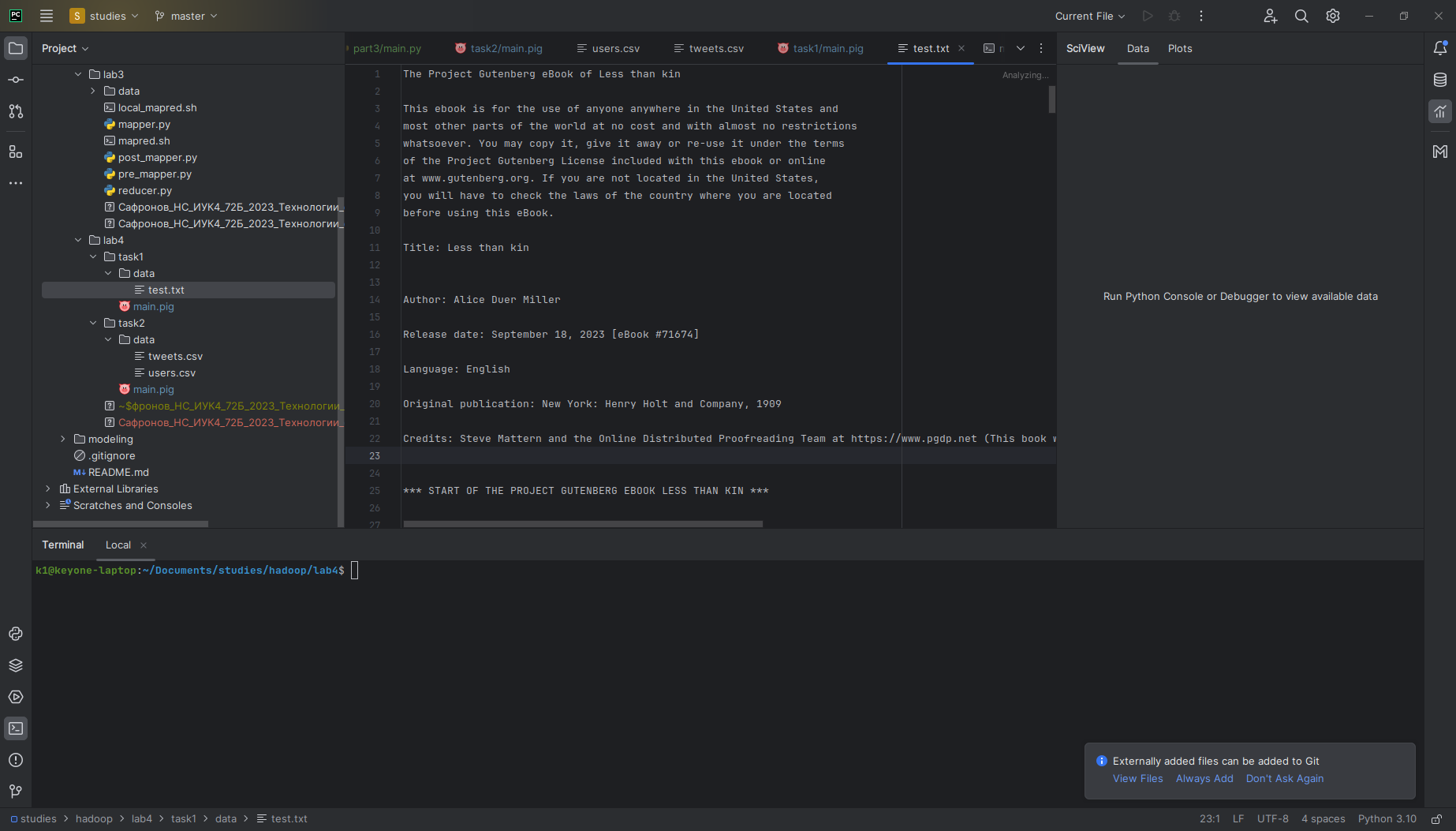
raw\_data = LOAD '/user/hduser/lab4/task1/test.txt' USING PigStorage('\n') AS (line:chararray);  
with\_indices = RANK raw\_data;  
words\_with\_indices = FOREACH with\_indices GENERATE $0, FLATTEN(TOKENIZE(line, ' '));  
grouped\_words = GROUP words\_with\_indices BY $1;  
result = FOREACH grouped\_words GENERATE $0 as word, BagToTuple($1.$0) as indices;

RMF /user/hduser/lab4/task2-output;

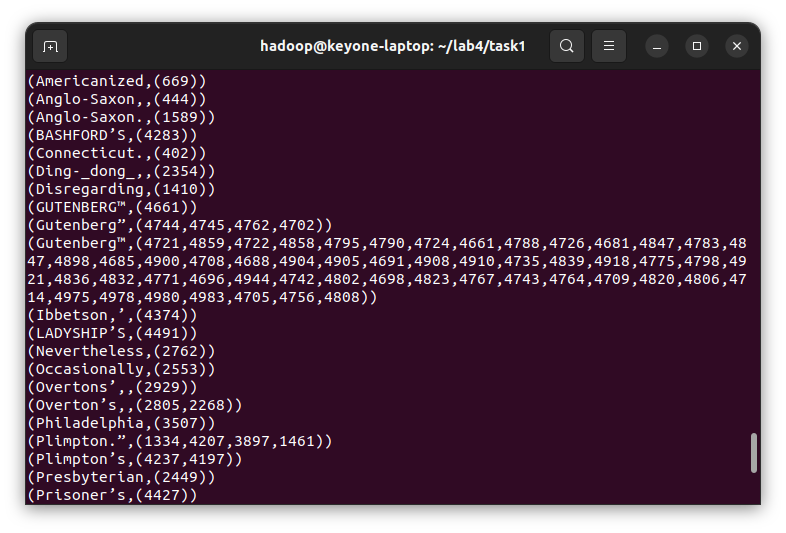
STORE result INTO '/user/hduser/lab4/task1-output.txt' using PigStorage(',');

DUMP result;

**Результаты выполнения скрипта**



**Рисунок 1.** Входной файл



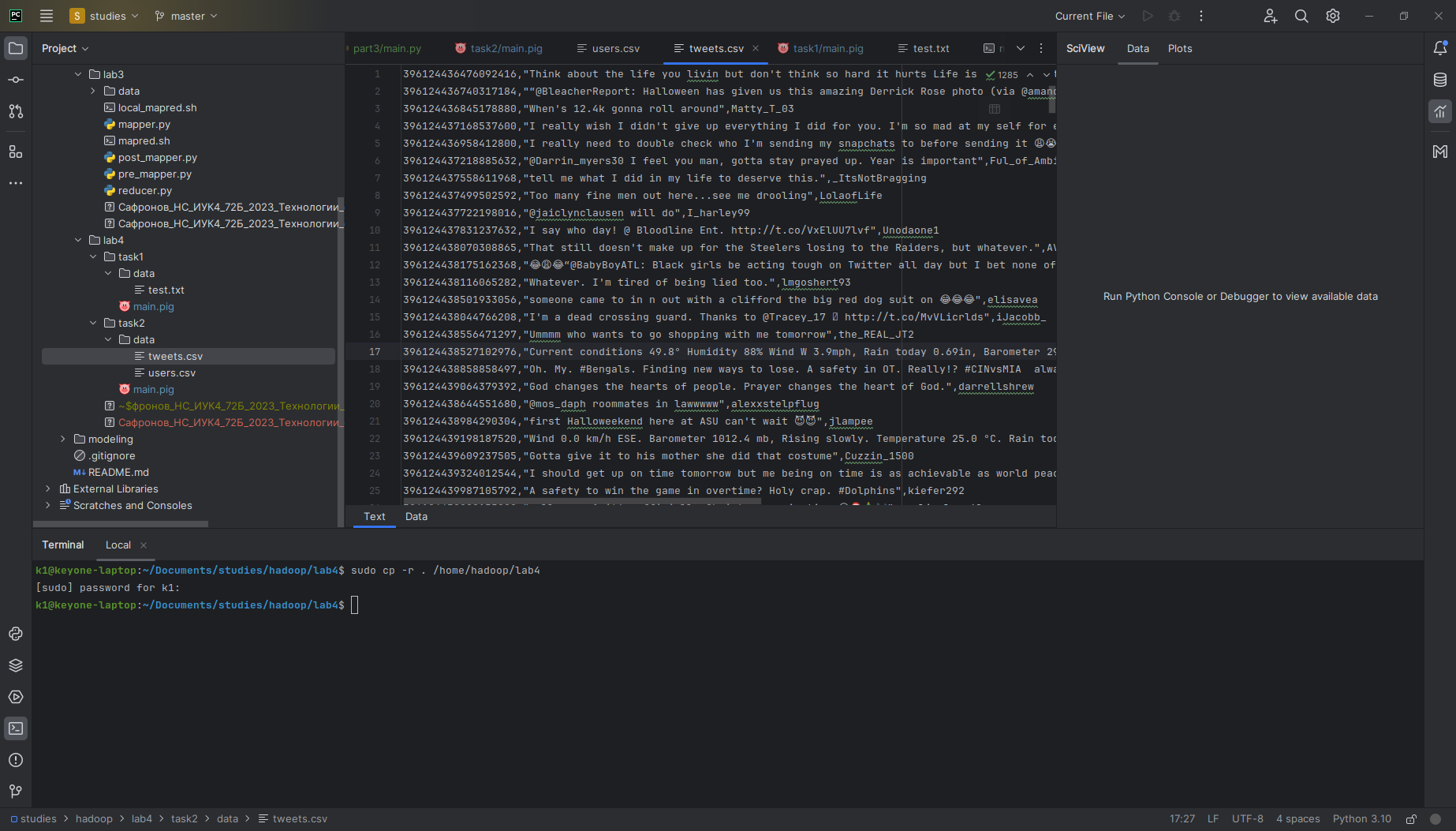
**Рисунок 2.** Демонстрация результата выполнения скрипта

**Задание 2**

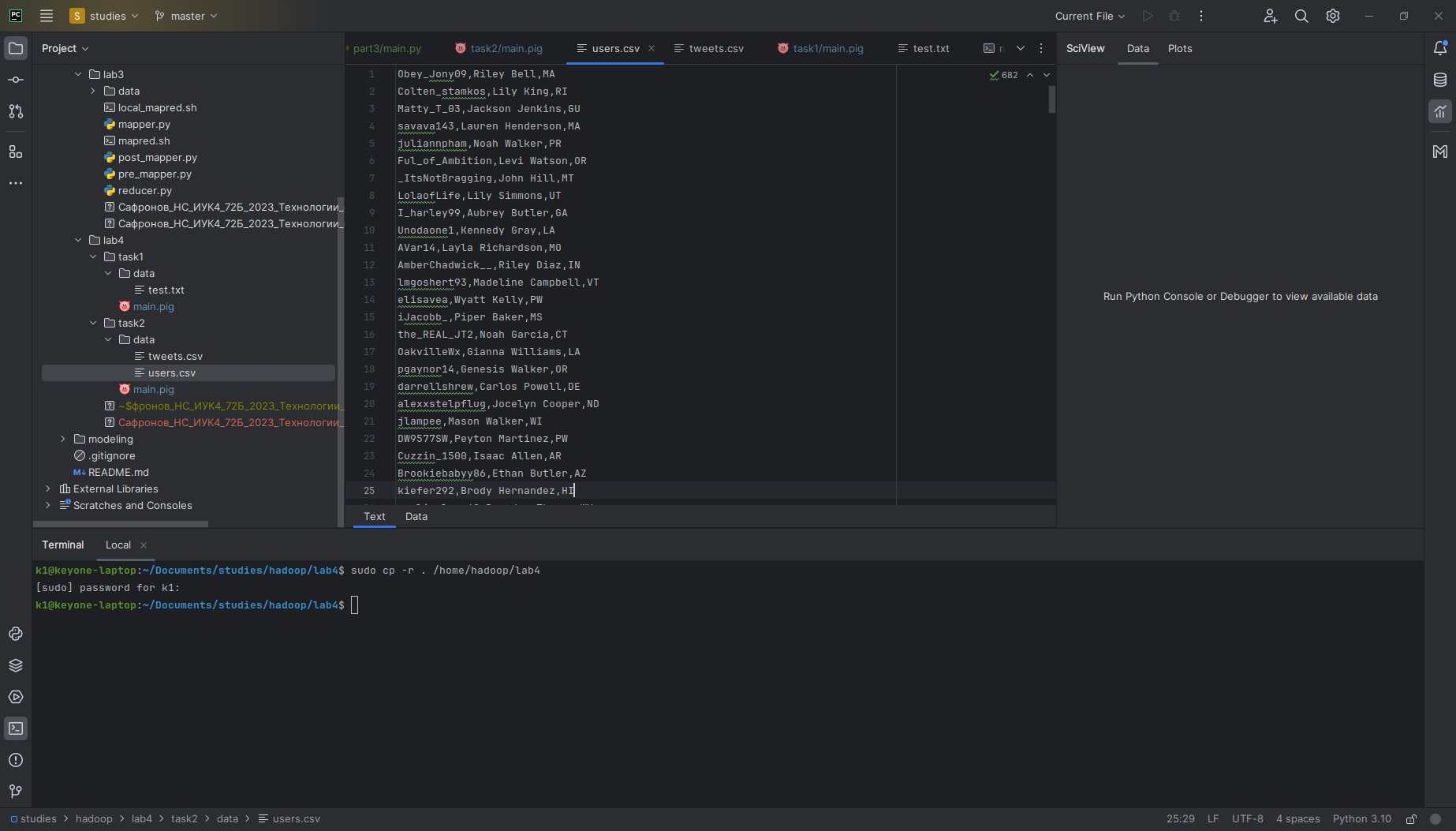
**Листинг программы**

REGISTER hdfs:///tmp/piggybank.jar;  
DEFINE CSVLoader org.apache.pig.piggybank.storage.CSVLoader();  
  
tweets = LOAD '/user/hduser/lab4/task2/tweets.csv' using CSVLoader() AS (tweet\_id: long, tweet: chararray, login: chararray);  
users = LOAD '/user/hduser/lab4/task2/users.csv' using CSVLoader() AS (login: chararray, user\_name: chararray, state: chararray);  
  
grouped\_tweets = GROUP tweets BY login;  
login\_with\_tweet\_count = FOREACH grouped\_tweets GENERATE $0 as login, COUNT($1) as tweet\_count;  
filtered\_logins = FILTER login\_with\_tweet\_count BY tweet\_count >= 2;  
joined\_logins = JOIN filtered\_logins BY login, users BY login;  
user\_tweet\_count = FOREACH joined\_logins GENERATE users::user\_name, filtered\_logins::tweet\_count;  
result = ORDER user\_tweet\_count BY $1 DESC;  
  
RMF /user/hduser/lab4/task2-output;  
STORE result INTO '/user/hduser/lab4/task2-output' using PigStorage(',');  
DUMP result;

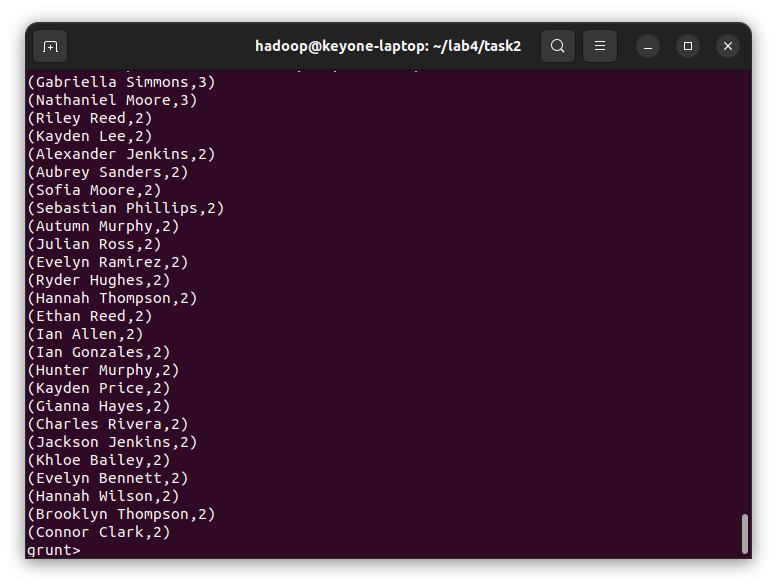
**Результаты выполнения скрипта**



**Рисунок 3.** Входной файл tweets.csv



**Рисунок 4.** Входной файл users.csv



**Рисунок 5.** Результат выполнения скрипта

**Вывод**: в ходе выполнения лабораторной работы были сформированы практические навыки реализации pig-скриптов для обработки больших данных.