|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА − Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИИТ)**

**Кафедра прикладной математики (ПМ)**

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ**

по дисциплине «Методы анализа данных»

**Практическая работа № 2**

|  |  |  |  |
| --- | --- | --- | --- |
| Студент группы ИНБО-01-17 | *ИНБО-05-22, Новиков Денис Владиславович* | (подпись) | |
| Преподаватель | *Шкерин Алексей Леонидович* | (подпись) | |
| Отчет представлен | «11» сентября 2024 г. | |  | |

Москва 2024 г.

## Постановка задачи:

Импортировать и экспортировать таблицы из РСУБД в HDFS (Hadoop Distributed Filesystem), используя Sqoop.

## Ход работы:

### Исследование данных с помощью Sqoop для РСУБД (MariaDB)

В терминале авторизуемся в MariaDB и выбираем базу данных labs на Рисунке 1.

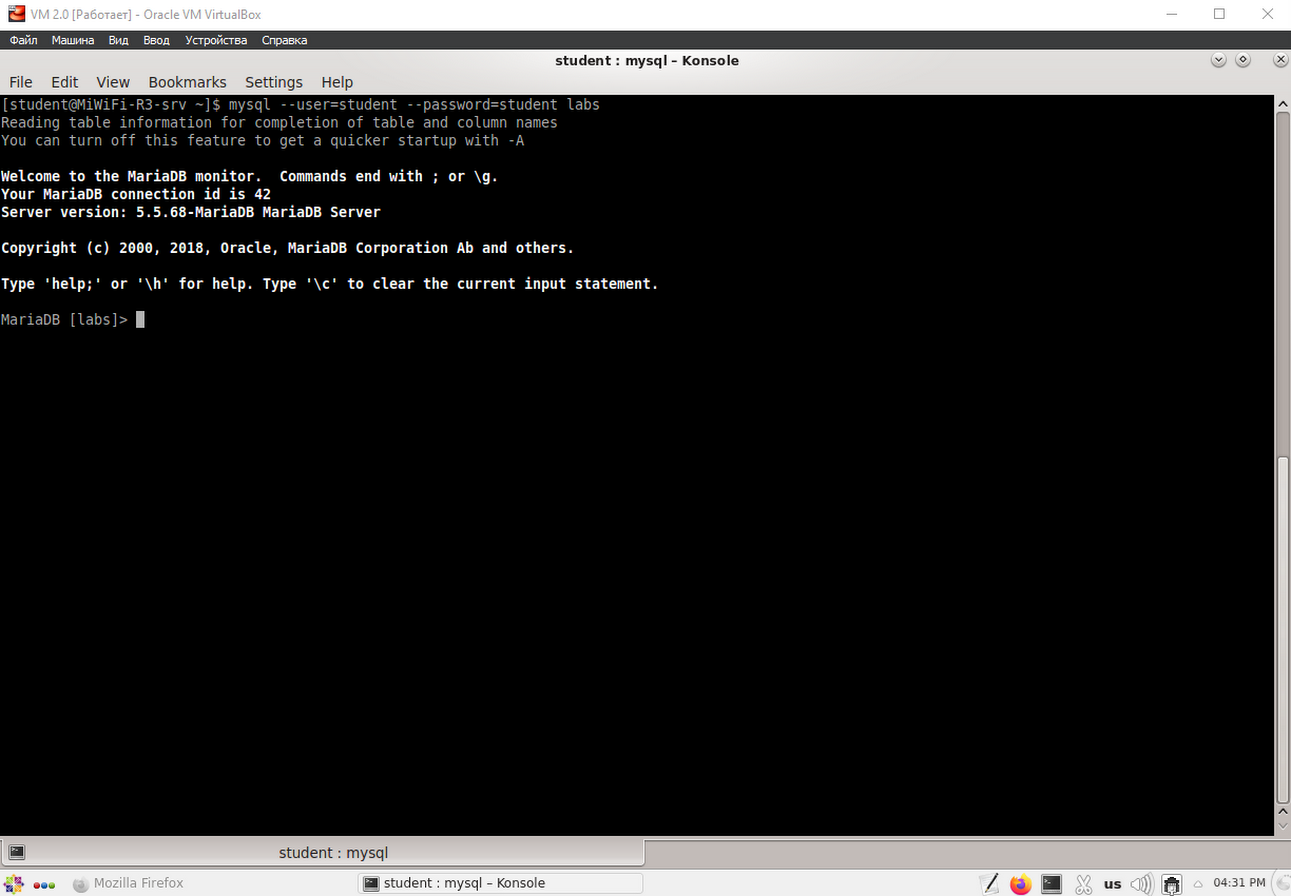


Рисунок 1 — Вход в MariaDB

Введем команду для проверки того, какие базы данных доступны на Рисунке 2.

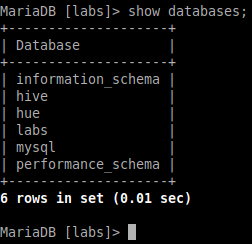


Рисунок 2 — Просмотр датабаз

Введем команду для просмотра таблиц в базе labs на Рисунке 3.

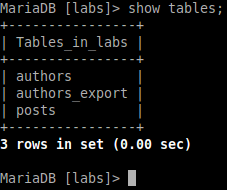


Рисунок 3 — Таблицы

На экране отображены таблицы authors и posts. Эти таблицы мы будем импортировать и экспортировать через команды Sqoop на Рисунке 4.

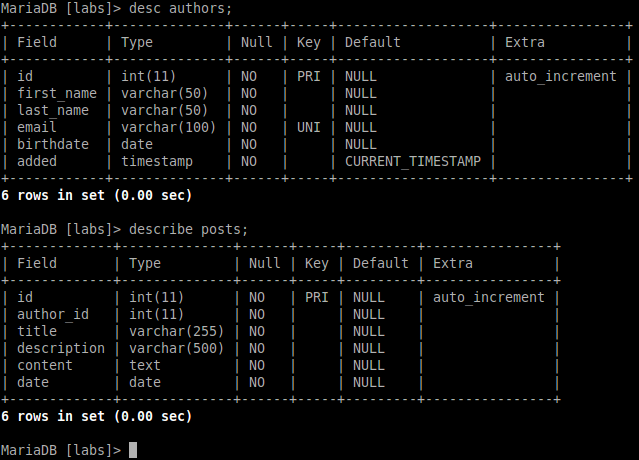


Рисунок 4 — Одинаковые команды desc и describe

Посмотрим на структуру таблиц authors и post, а также на несколько записей оттуда на Рисунке 5.

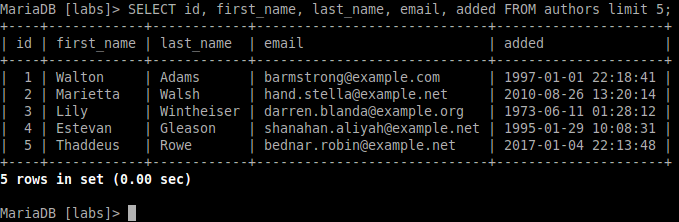


Рисунок 5 — Структура таблицы authors

Выходим из сеанса MariaDB на Рисунке 6.



Рисунок 6 — Выход из MariaDB

Для получения помощи по базовым командам sqoop запустим следующую команду на Рисунке 7.

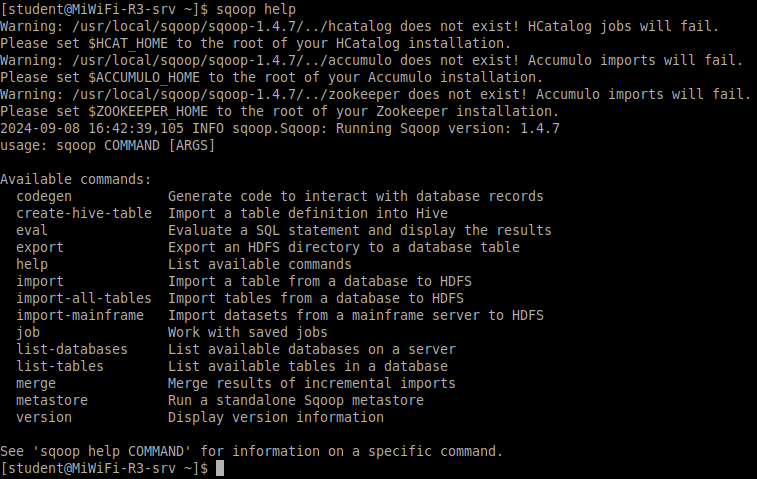


Рисунок 7 — Помощь sqoop

Для получения детализированной информации по каждой подкоманде, введем ее название после help на Рисунке 8.

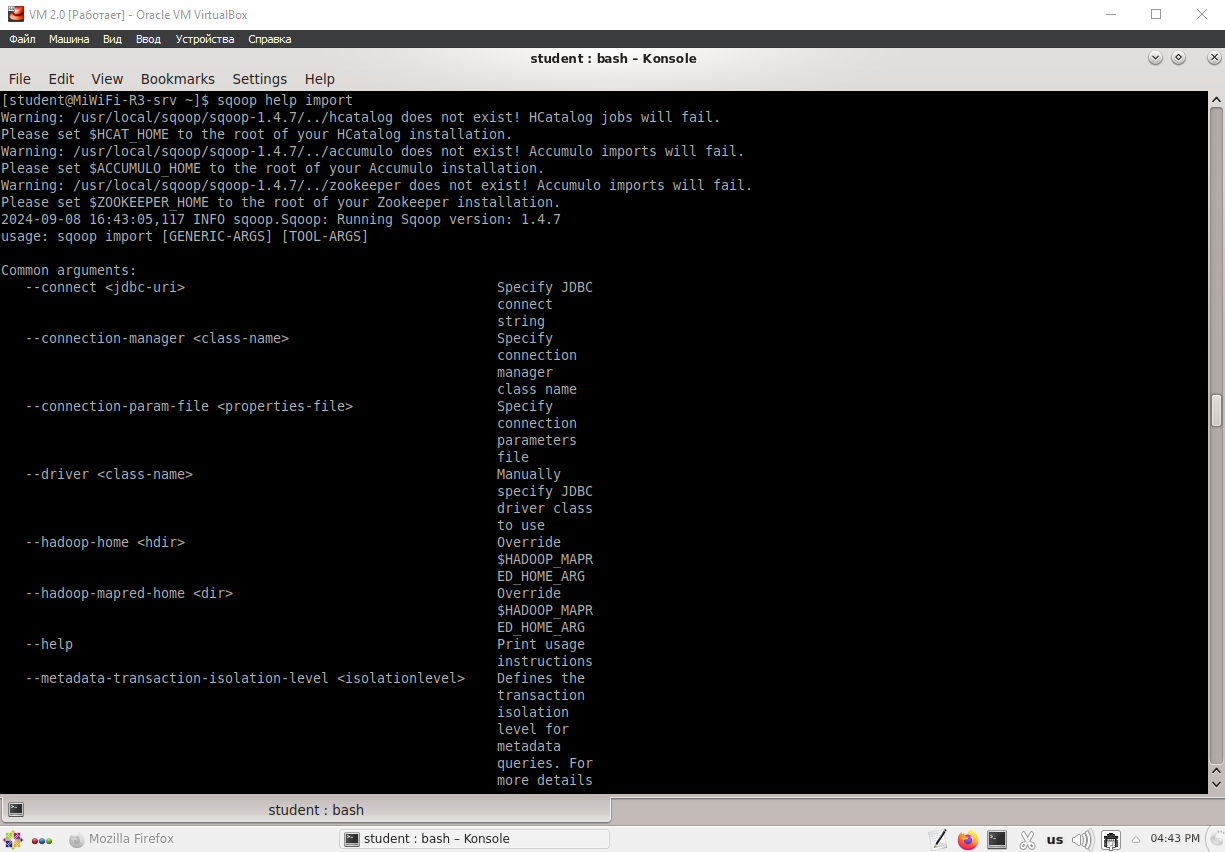


Рисунок 8 — Детализированная помощь sqoop

Просмотр списка баз данных в MariaDB и таблиц в базе данных labs осуществляется следующей командой на Рисунке 9.



Рисунок 9 — Выход из пользователя hadoop

Альтернативой использованию аргумента --password является использование ключа -P (заглавная буква) и ввод пароля при запросе. При этом вводимые символы будут невидимы на Рисунке 10.

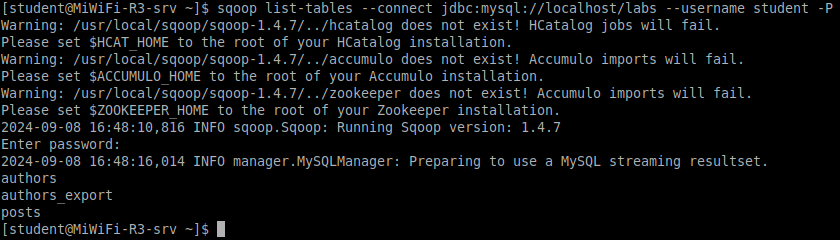


Рисунок 10 — Альтернативный вход

Импортируем все таблицы в базе labs используя команду import-all-tables на Рисунке 11.

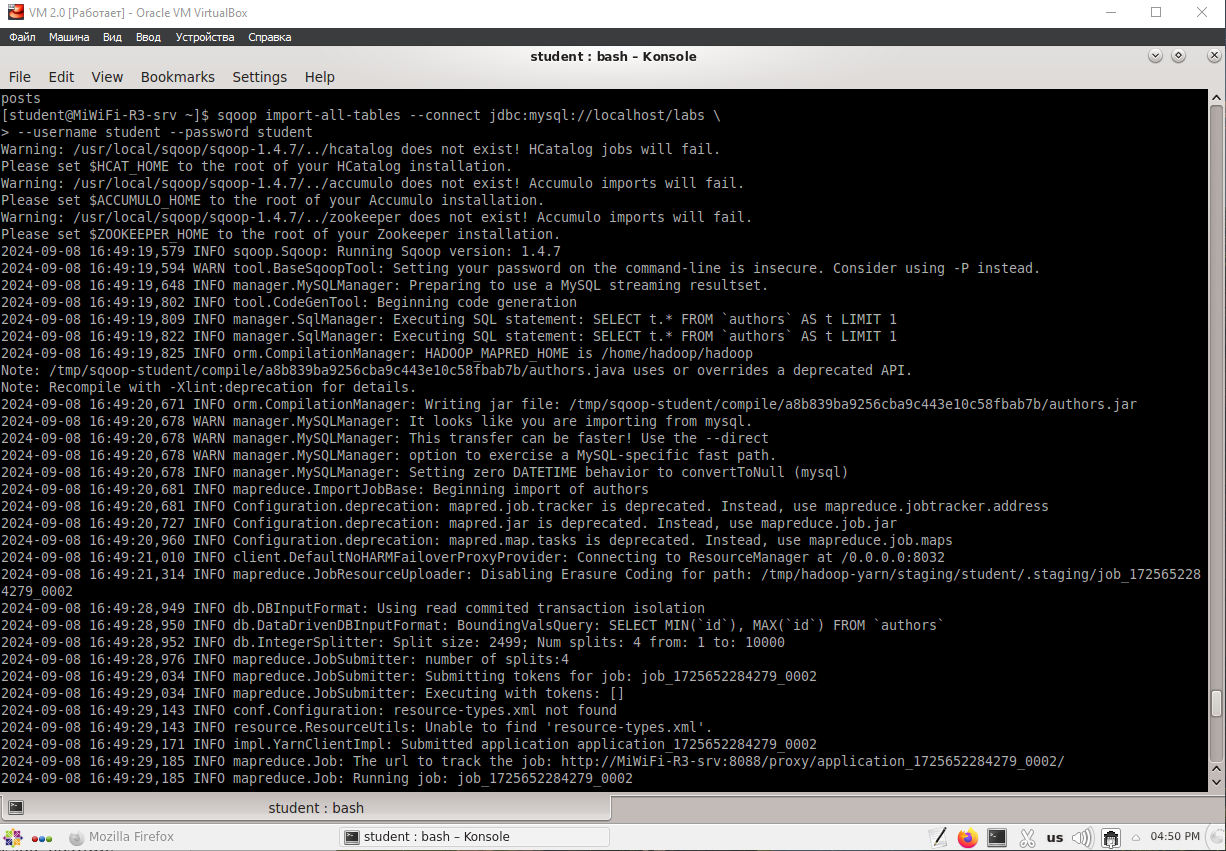


Рисунок 11 — Импорт

Выполним эту команду для получения таблицы posts из базы labs и сохранения последней в HDFS на Рисунке 12.

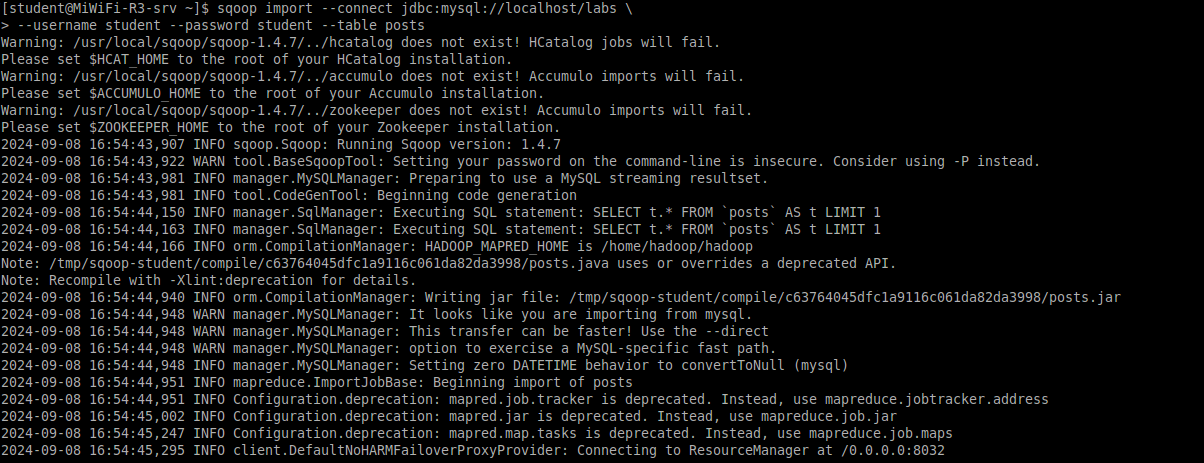


Рисунок 12 — Получения таблицы posts из базы labs

Проверим, как сохранились данные на Рисунке 13.

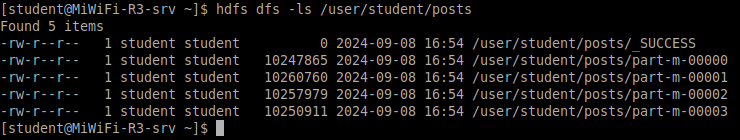


Рисунок 13 — результат импорта

Создаем целевую директорию в HDFS для импортирования данных в нее на Рисунке 14.



Рисунок 14 — Создаем целевую директорию в HDFS

Импортируем таблицу authors и сохраняем ее в директорию HDFS, которую мы создали выше, используя ‘,’ для разделения полей. Аргумент --field-terminated-by ‘,’ используется для задания запятой в качестве разделителя полей в HDFS файле на Рисунке 15.

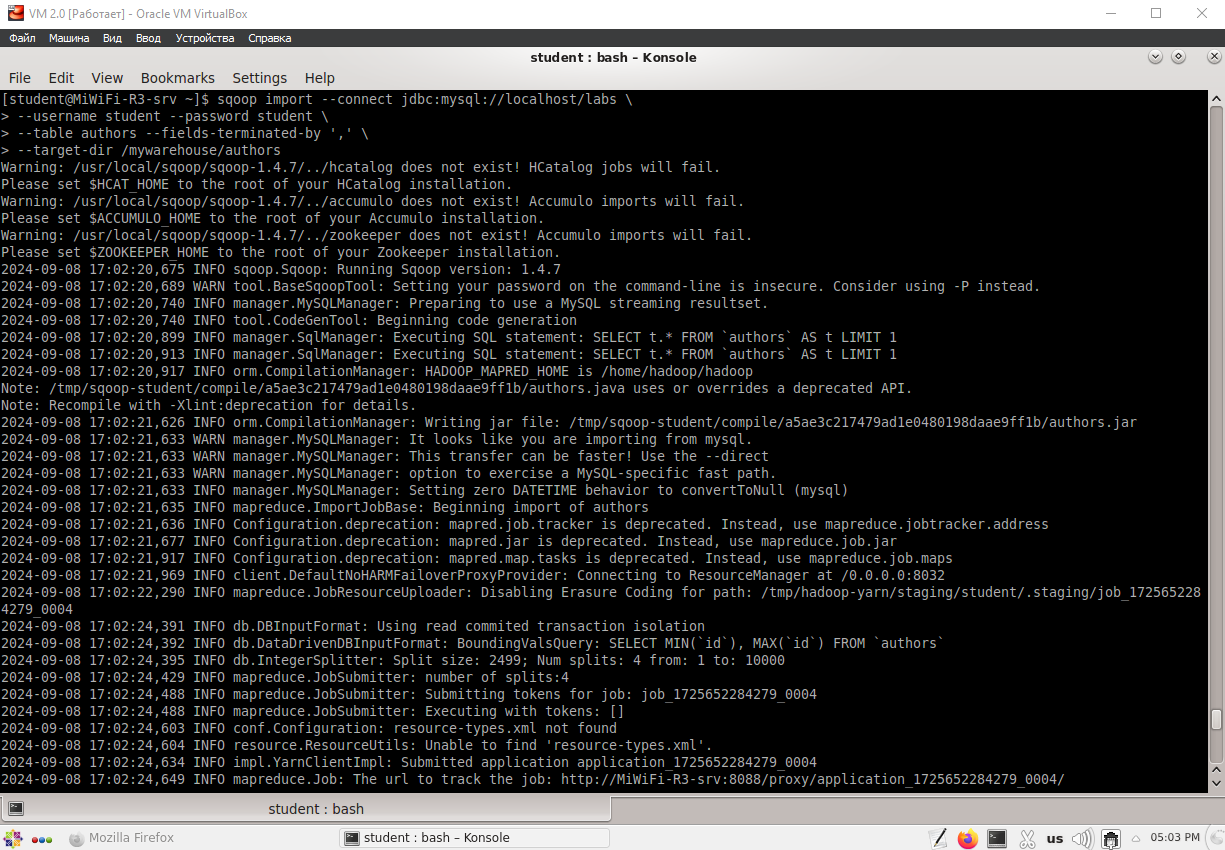


Рисунок 15 — Импортируем таблицу authors

Для проверки воспользуемся командой hdfs по целевой директории на Рисунке 16.

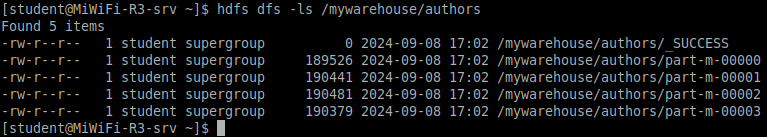


Рисунок 16 — Проверка целевой директории

Посмотрим содержание authors на Рисунке 17.

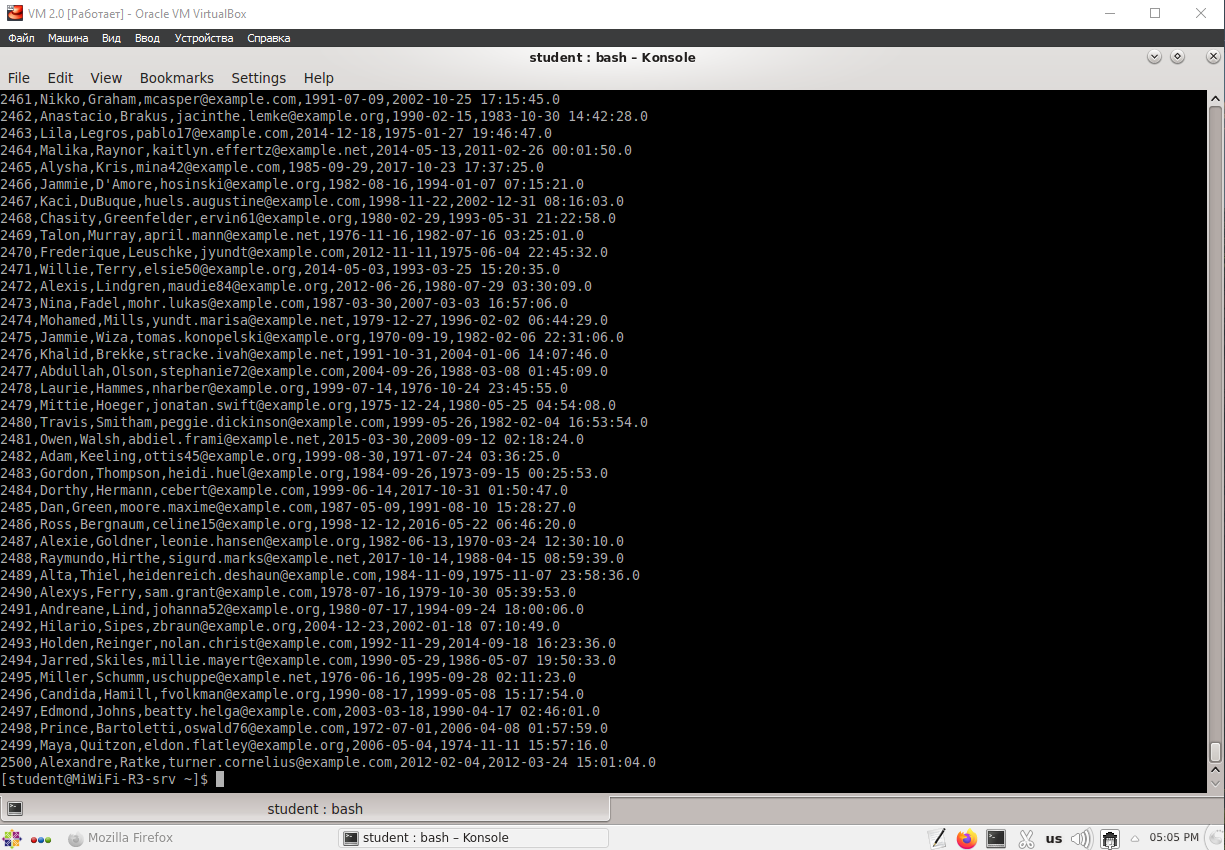


Рисунок 17 — Содержание authors

Импортирование определенных столбцов осуществляется указанием аргумента --columns для папки authors в домашней директории HDFS. Импортируемые столбцы: first\_name, last\_name, email на Рисунке 18.

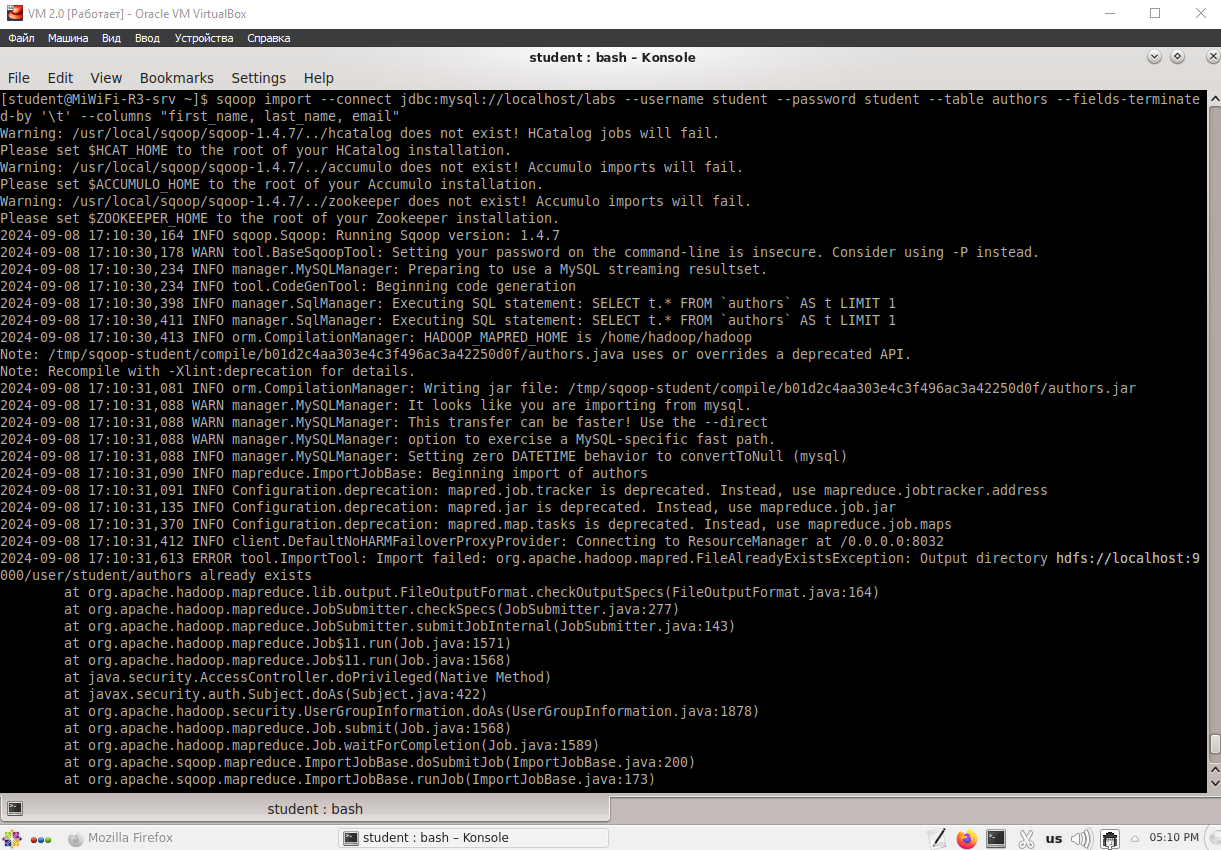


Рисунок 18 — Импорт столбцов first\_name, last\_name, email

Импортируем только подходящие строки с помощью --where. Для примера импортируем строки из таблицы authors, где first\_name указано, как ‘Dorthy’ на Рисунке 19.

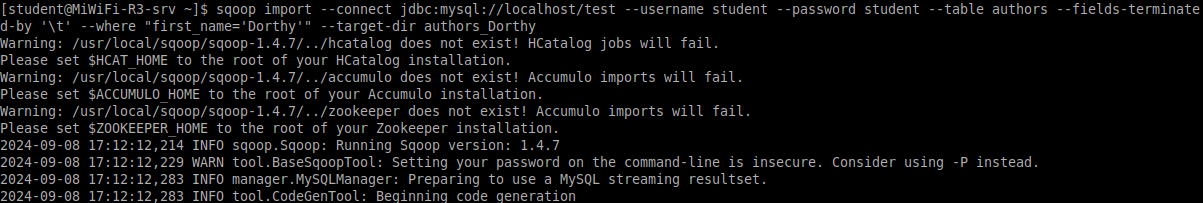


Рисунок 19 — импортируем строки из таблицы authors, где first\_name указано, как ‘Dorthy’

Импортируем таблицу, используя другой формат файла, вместо простого текстового. Импортируем таблицу authors в файл формата Parquet на Рисунке 20.

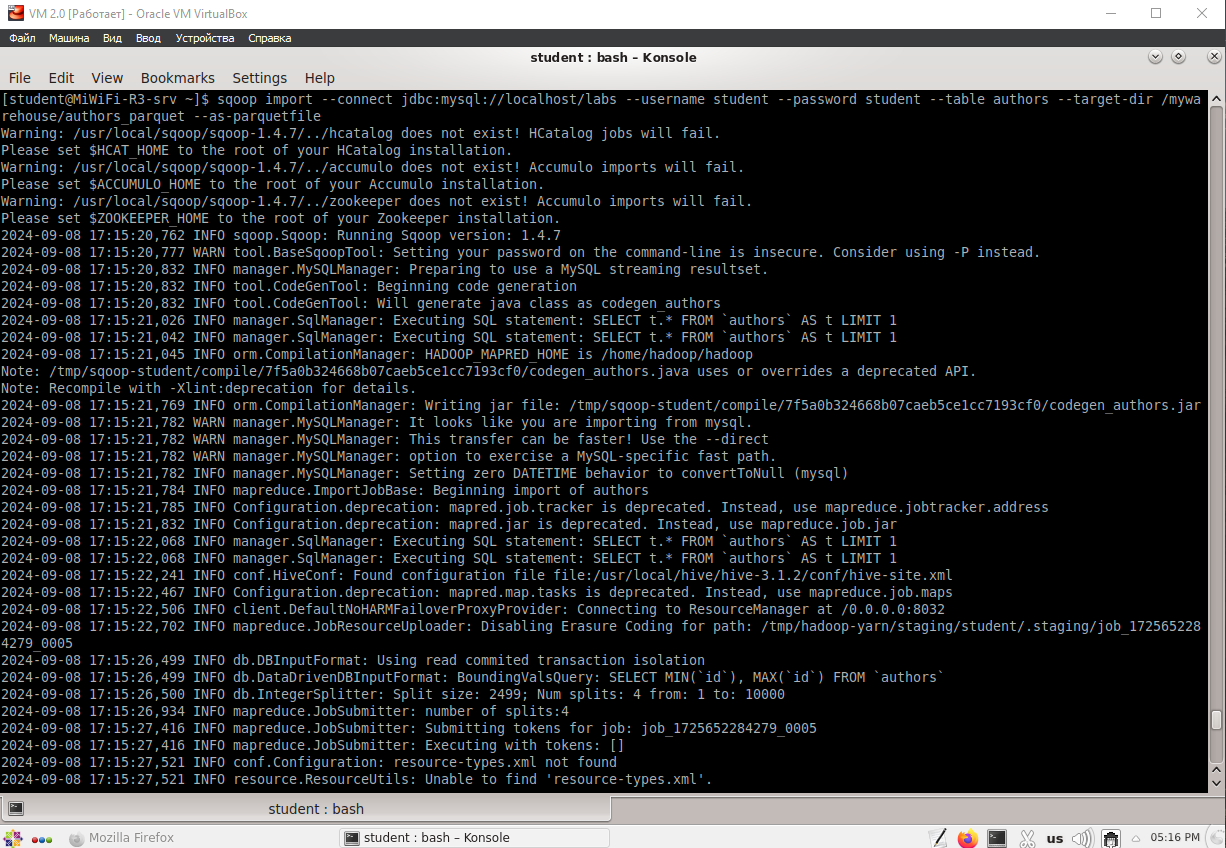


Рисунок 20 — Импортируем таблицу authors в файл формата Parquet

Смотрим результат при помощи отображения содержимого целевой директории HDFS на Рисунке 21.

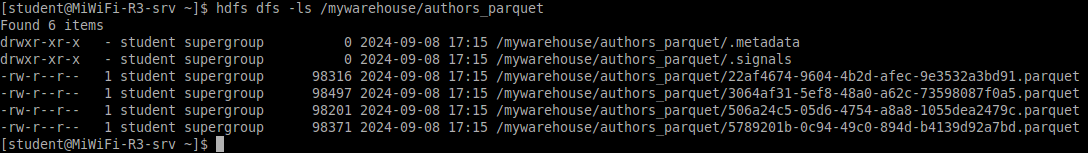


Рисунок 21 — Результат в целевой директории HDFS

Каждому файлу Parquet дано уникальное имя, например, 5e44cda6-728c-4912-864a-94d0659930f3.parquet. Данные в этом формате нельзя просмотреть напрямую, так как формат двоичный. Для просмотра записей в таких файлах используйте команду parquet-tools show. Сначала нужно получить файл parquet на локальный узел, а затем запустить команду parqet-tools на Рисунке 22.

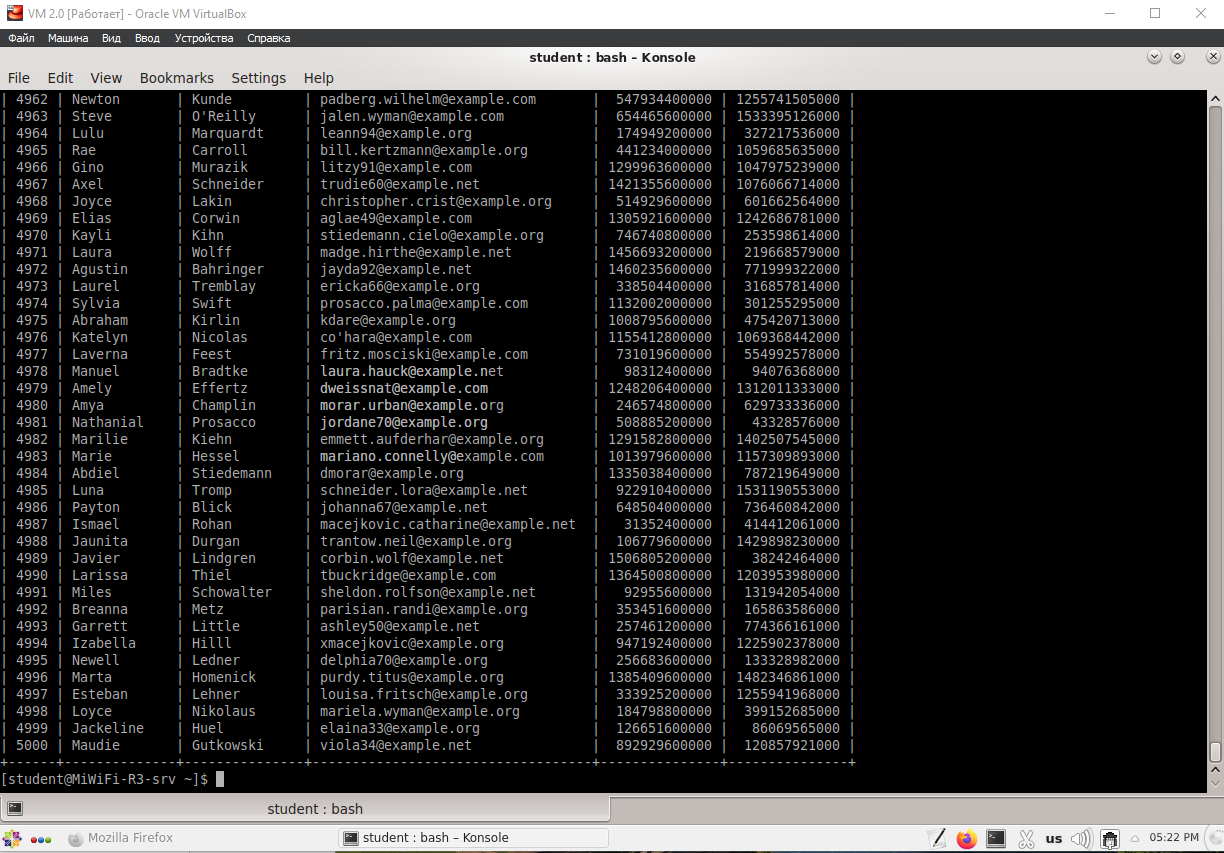


Рисунок 22 — Результат выполнения команды parqet-tools

Импортируем таблицу authors, используя сжатие (аргумент --compress или -z) на Рисунке 23.

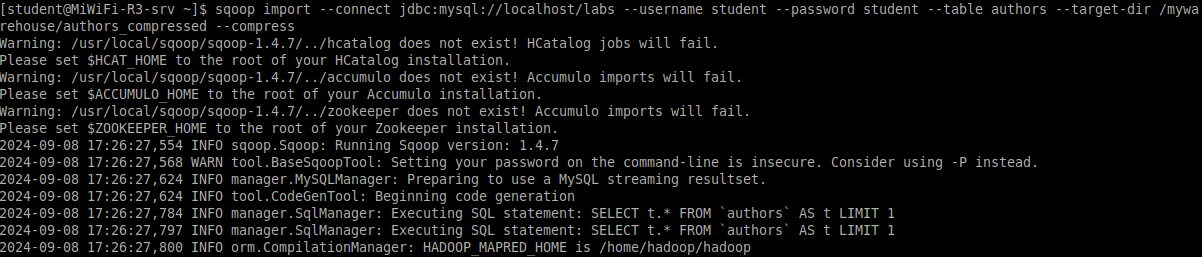


Рисунок 23 — Импорт со сжатием authors

Результат успешного импорта со сжатием показан на Рисунке 24.

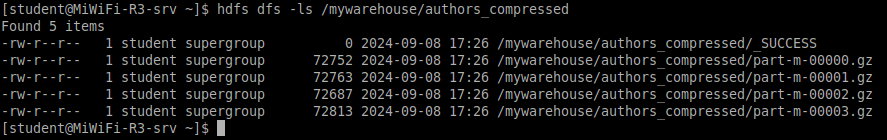


Рисунок 24 — Успешный импорт со сжатием

Импортируем строки, где first\_name указано, как ‘Dorthy’ (мы уже делали это в шаге 15) и сохраняем результат в папку dorthy в домашней директории HDFS на Рисунке 25.

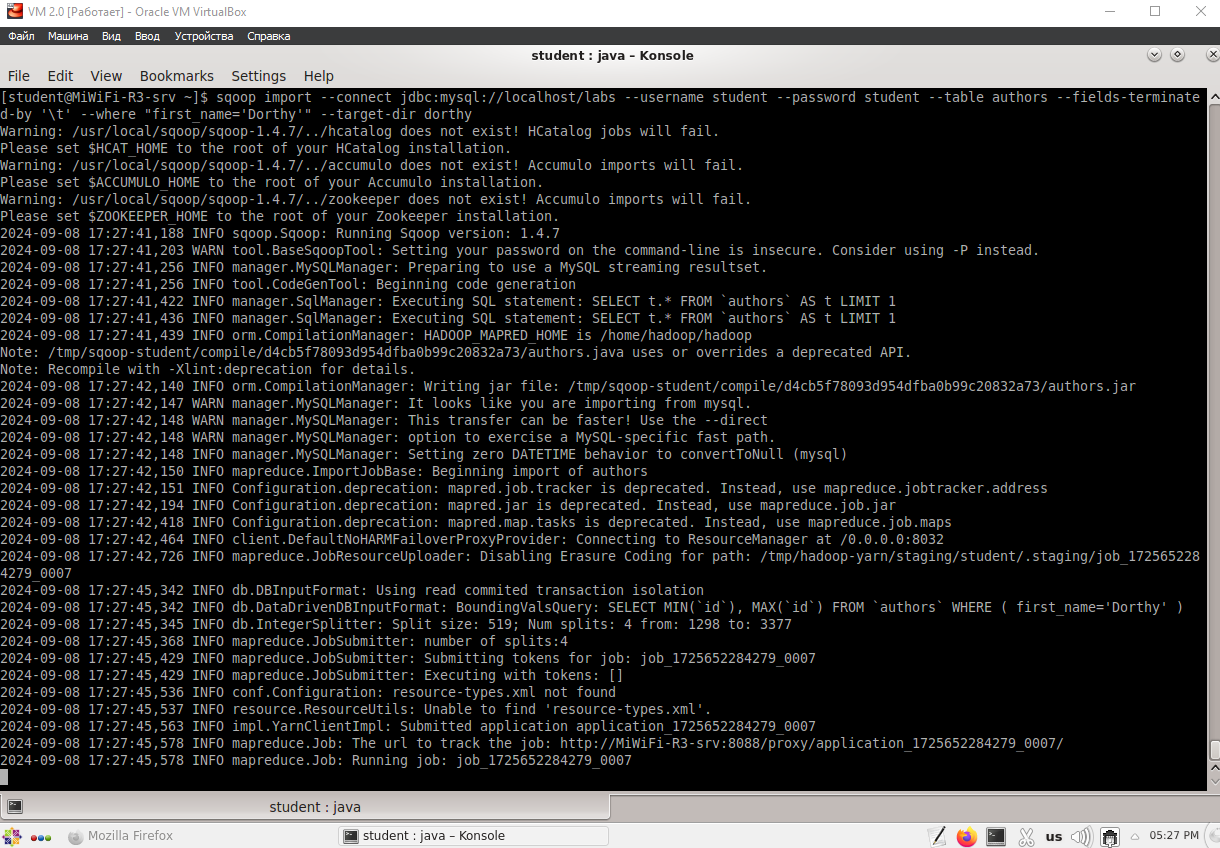


Рисунок 25 — Аналогичный импорт со сжатием

Экспортируем сохраненную папку dorthy в качестве таблицы в базу данных labs на Рисунке 26.

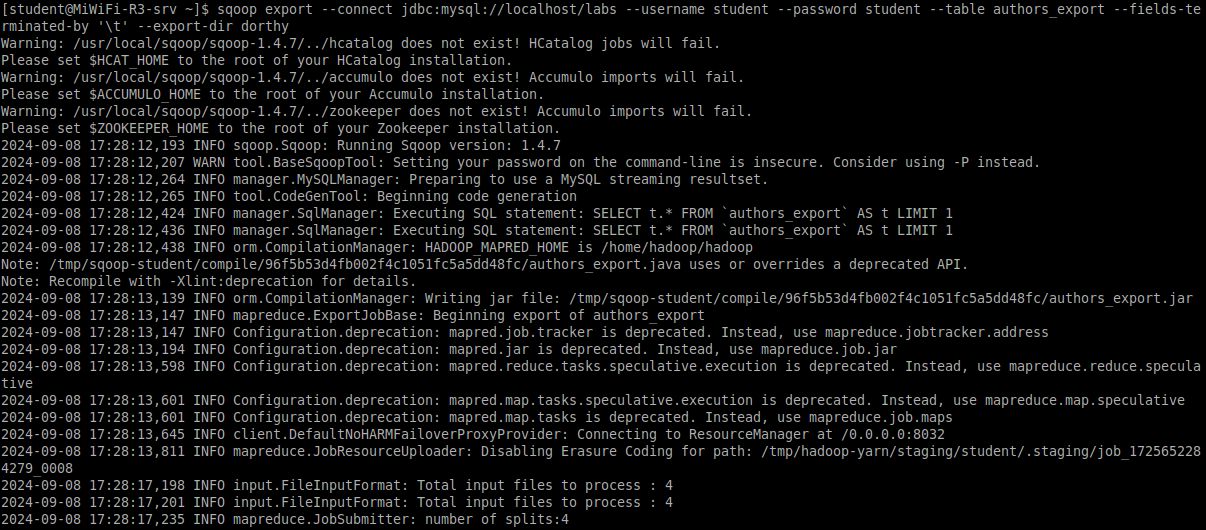


Рисунок 26 — Экспорт

Посмотрим содержимое экспортированных записей в MariaDB на Рисунке 27.

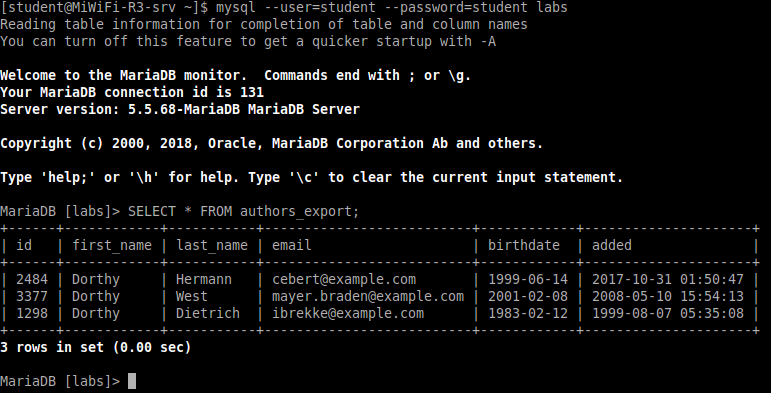


Рисунок 27 — Содержимое экспорта

Создаем целевую директорию в HDFS для импортирования данных таблицы из базы labs (/tmp/mylabs) на Рисунке 28.



Рисунок 28 — Создаем новую целевую директорию

Из таблицы posts импортируем первичный ключ, заголовки постов, дату публикации в директорию HDFS /tmp/mylabs/posts\_info. Сохраним файл в текстовом формате с табуляциями в качестве разделителей на Рисунке 29.

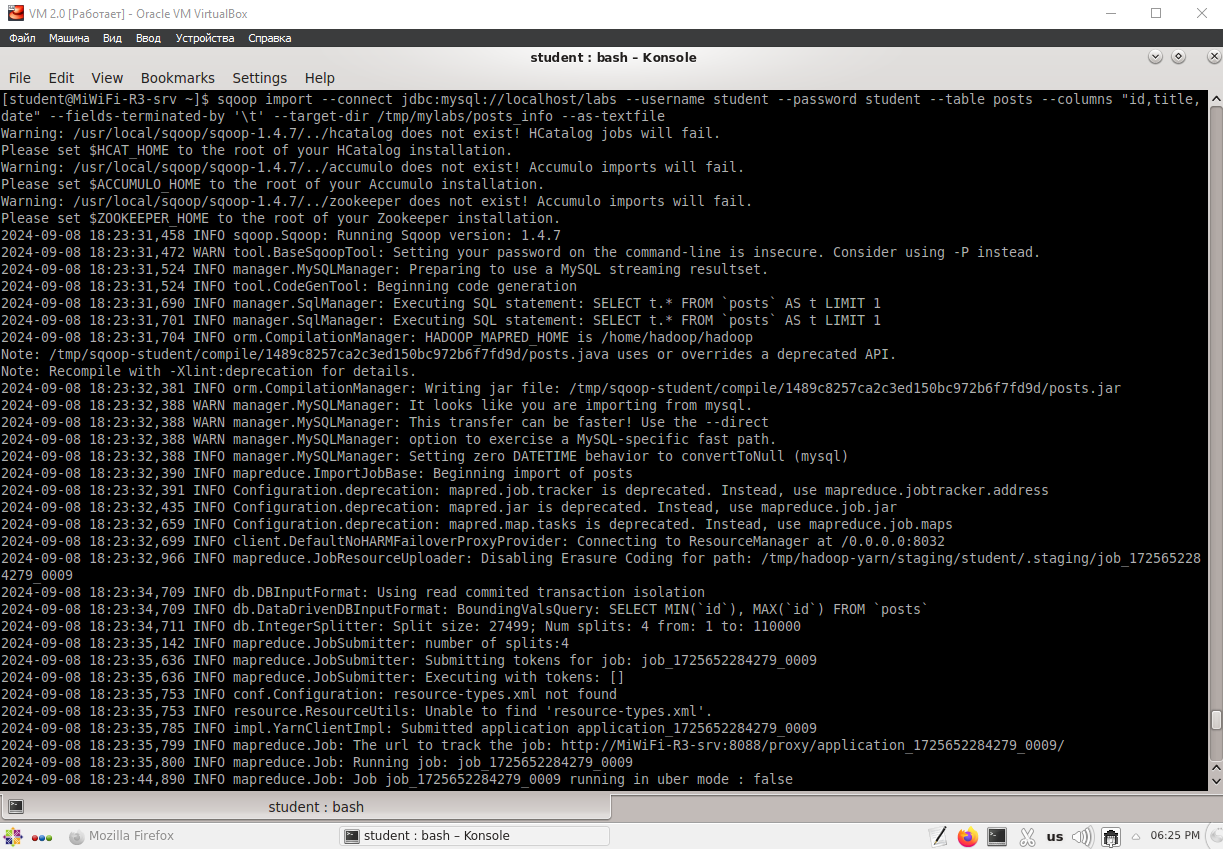


Рисунок 29 — Импорт

Теперь сохраним то же самое, только в формате parquet со сжатием snappy в папке /tmp/mylabs/posts\_compressed на Рисунке 30.

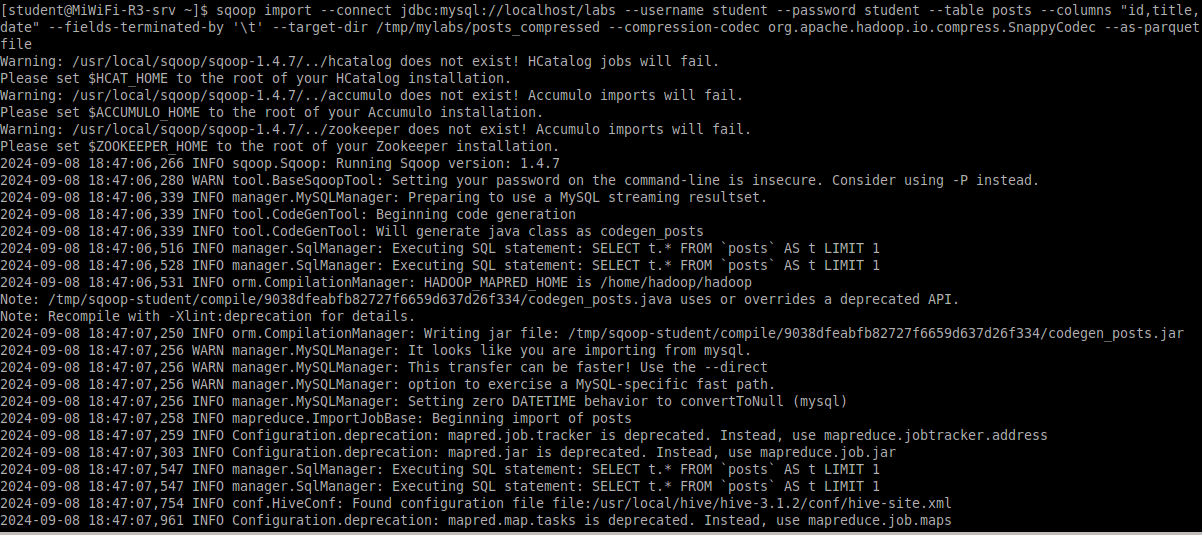


Рисунок 30 — Импорт со сжатием Snappy в формате parquet

Из окна терминала отобразим несколько записей из тех, что мы только что проимпортировали на Рисунке 31.

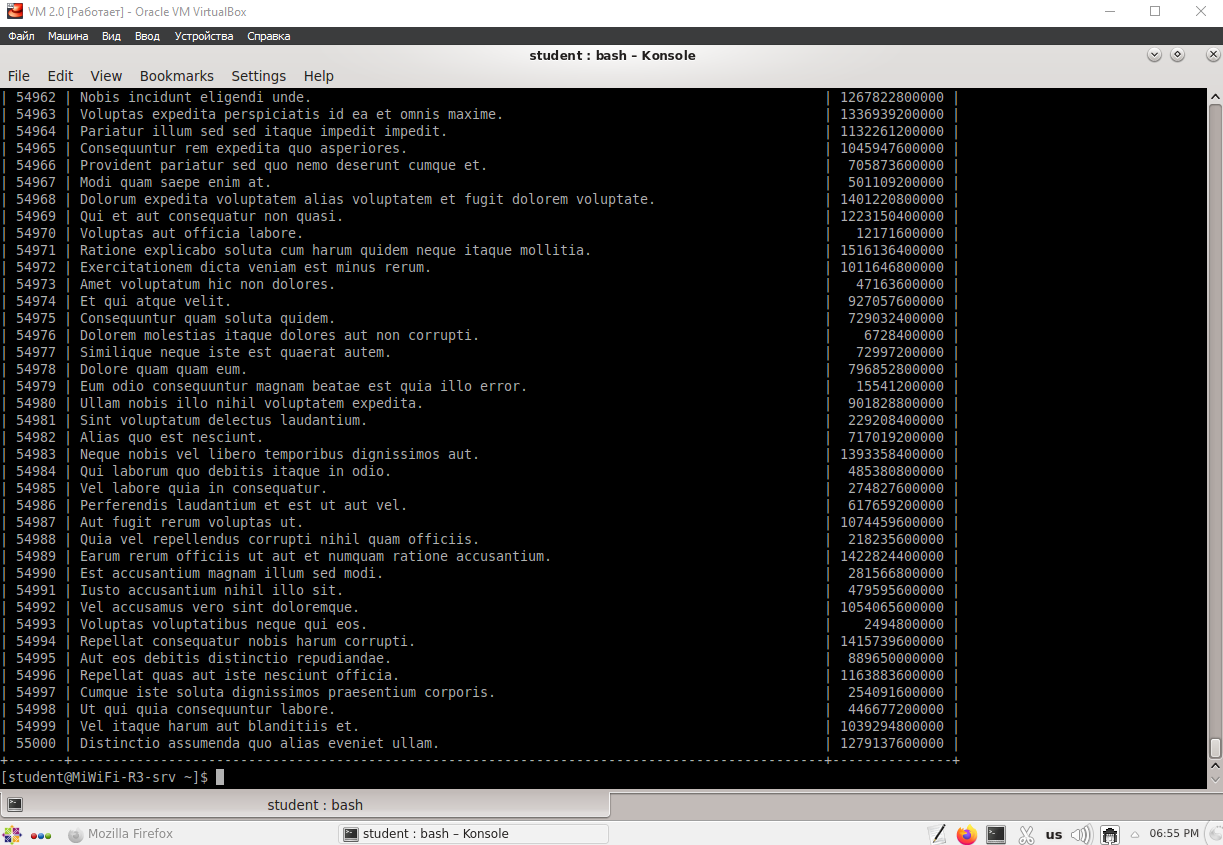


Рисунок 31 — Результат импорта

Очищаем ненужную папку на Рисунке 32.



Рисунок 32 — Очистка папки

Импортируем столбцы id, first\_name, last\_name, birthdate из таблицы authors в домашнюю директорию HDFS. Файл сохраним в текстовом формате с табуляциями в качестве разделителей на Рисунке 33.

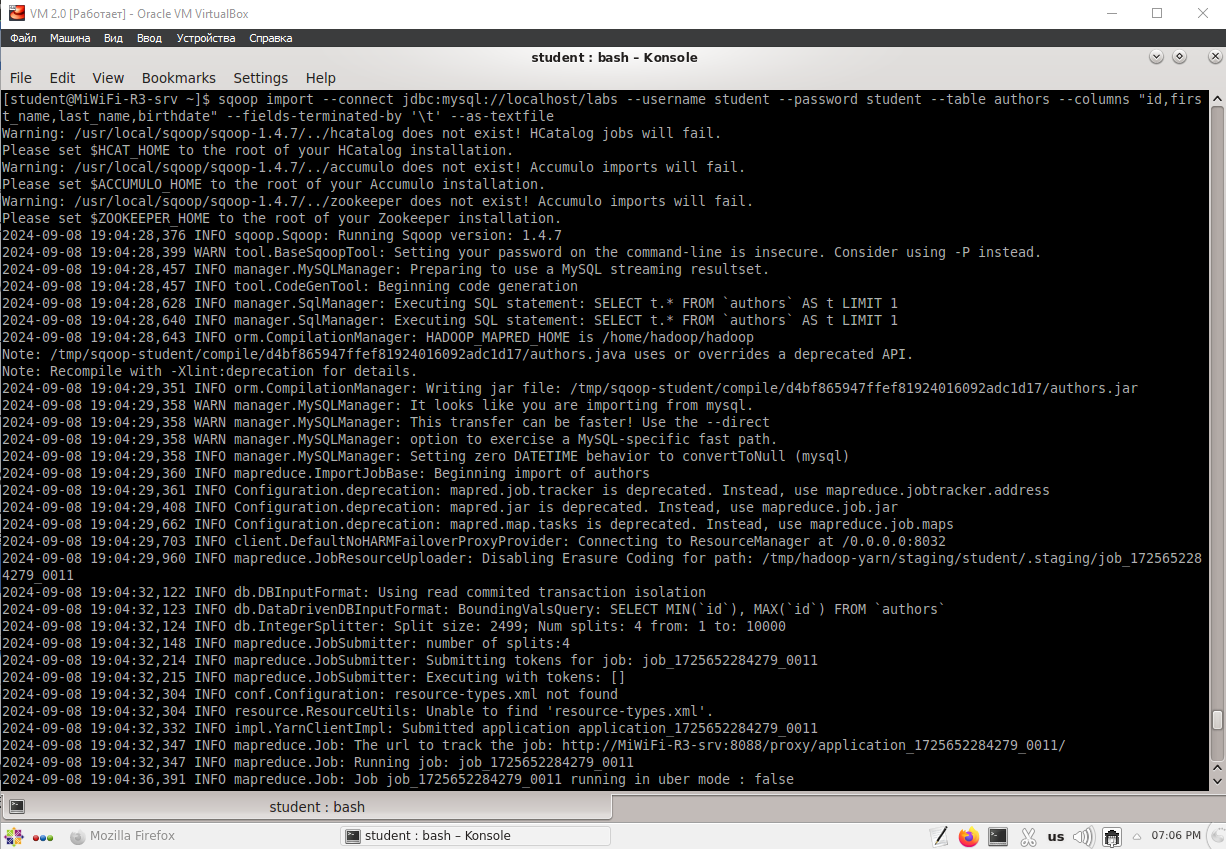


Рисунок 33 — Импорт

Результат импорта показан на Рисунке 34.

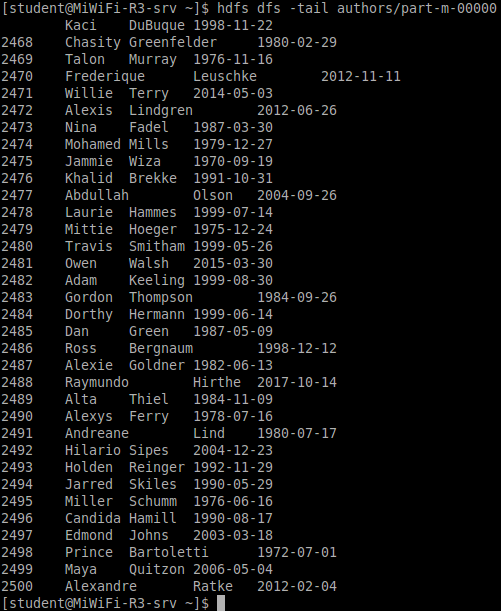


Рисунок 34 — Результат импорта

Импортируем и сохраним в директории /tmp/mylabs/posts\_NotN те посты, название которых не NULL. Импортирум только следующие столбцы: id, title, content. Сохраним файл в формате parquet, а для сжатия используем кодек snappy на Рисунке 35.

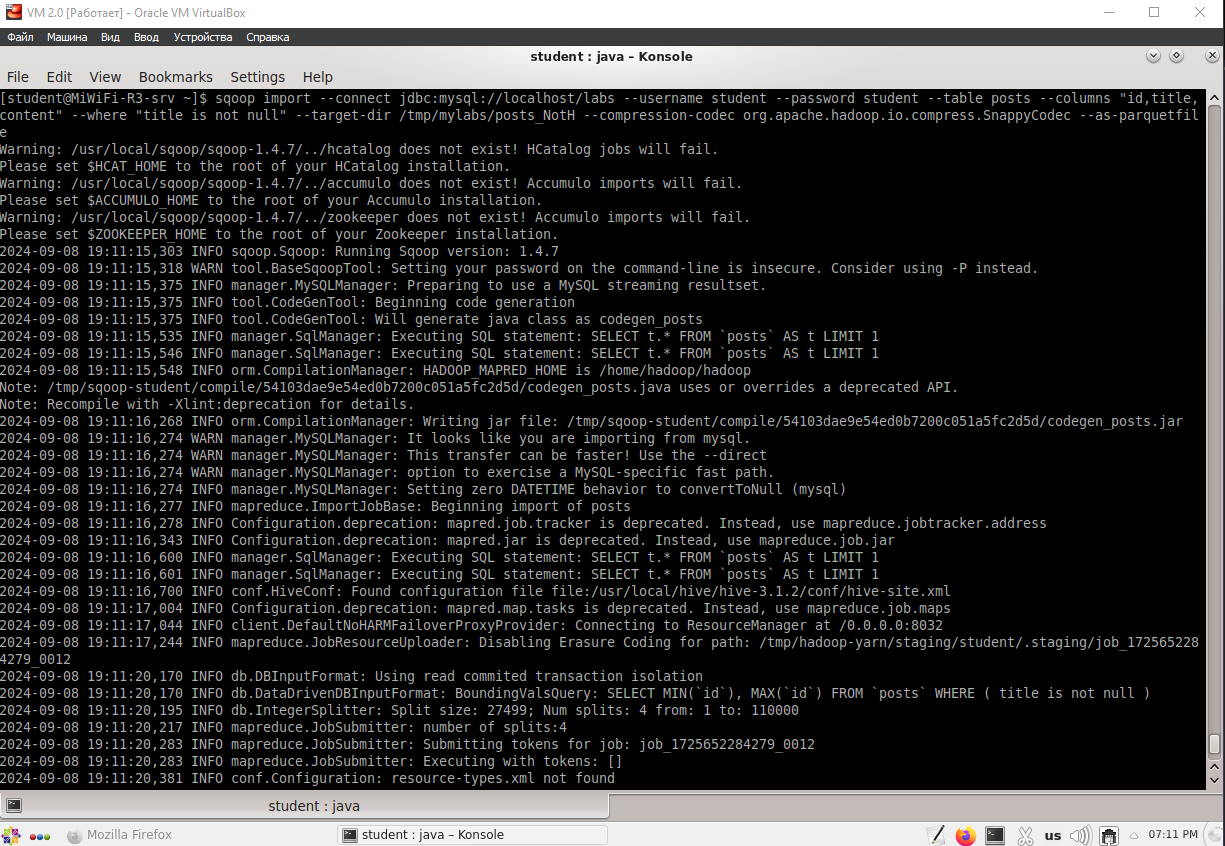


Рисунок 35 — Импорт

Результат импорта сжатых файлов показан на Рисунке 36.

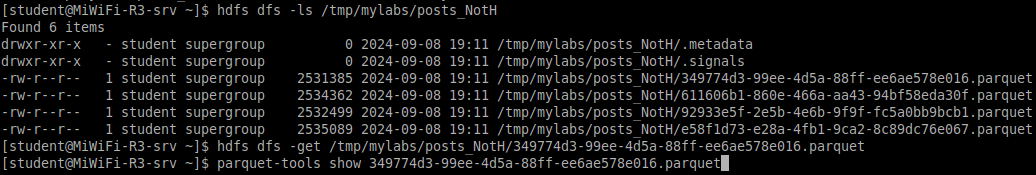


Рисунок 36 — Результат импорта сжатых файлов

Содержимое импортированного сжатого файла показано на Рисунке 37.

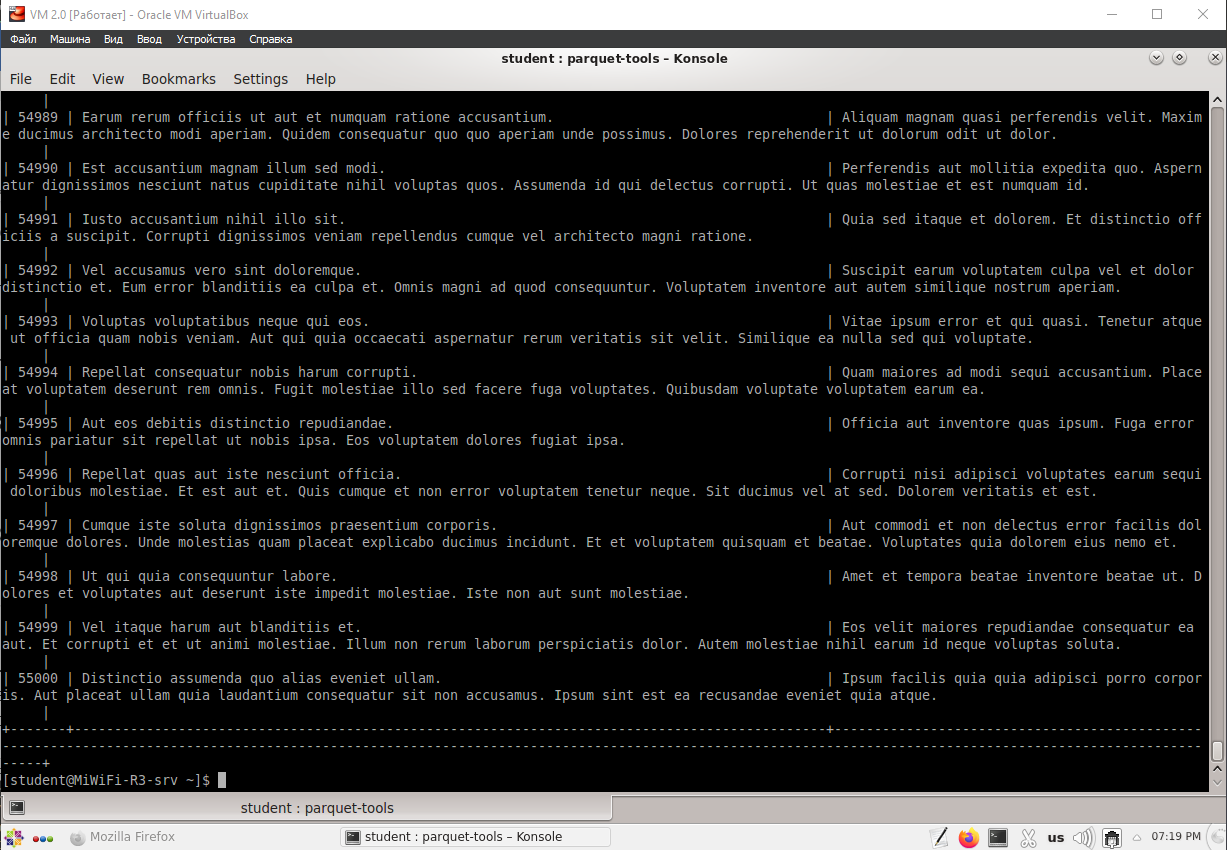


Рисунок 37 — Содержимое импортированного сжатого файла