|  |
| --- |
| 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 №6**

по дисциплине «Методы анализа данных»

|  |  |  |
| --- | --- | --- |
| Студент группы: | ИМБО-02-21 Новичков С.Д. |  |
| Ассистент кафедры ПМ: | Горячев А.А. |  |
| Отчет представлен | «\_\_\_»\_\_\_\_\_\_\_\_2023г. |  | |

Москва 2023 г.

**Доступ к данным с помощью HBase**

В этой лабораторной работе выполняются базовые операции с созданием, удалением и изменением таблицы в оболочке HBase. Мы будем использовать оболочку для размещения и получения данных в HBase.

Запустим оболочку HBase. Выполним команду справки и просмотрим основную информацию об использовании оболочки HBase.

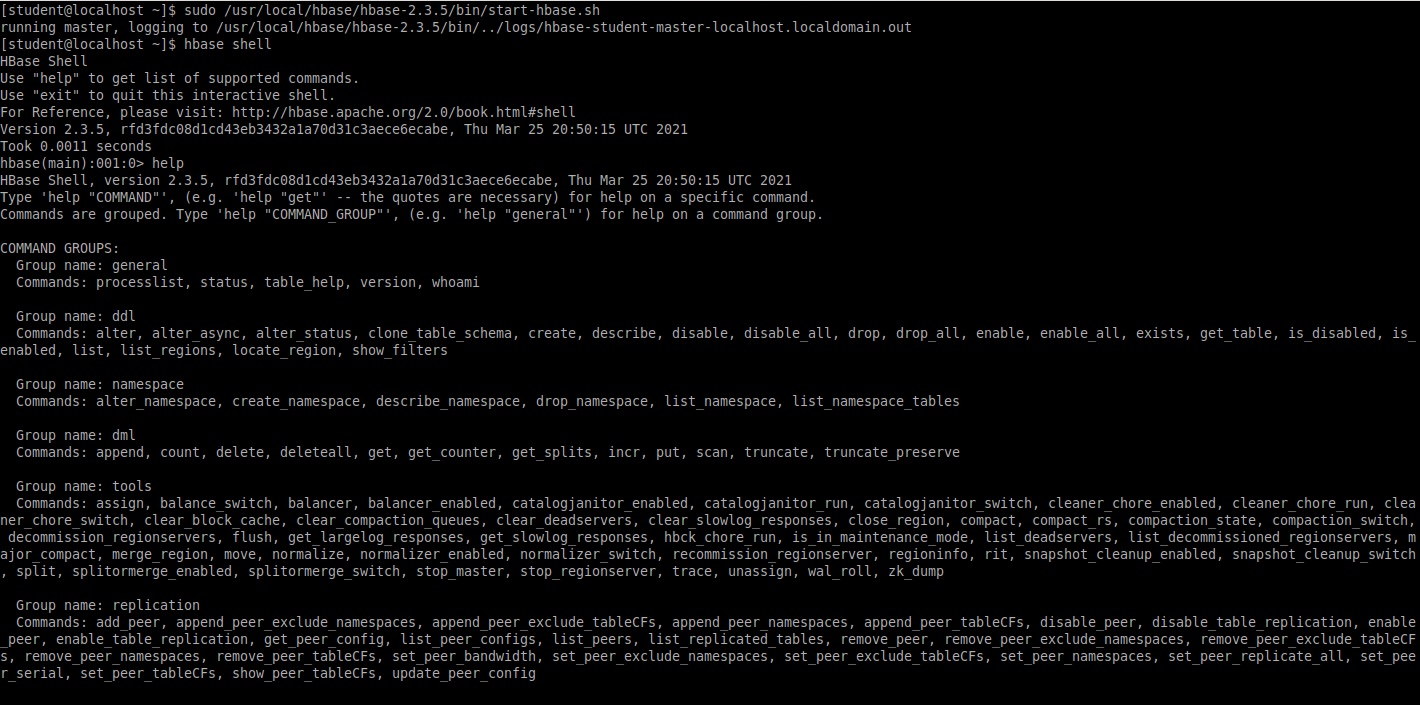


Рисунок 1 — Запуск HBase и просмотр справки

Отобразим версию и состояние для базового использования

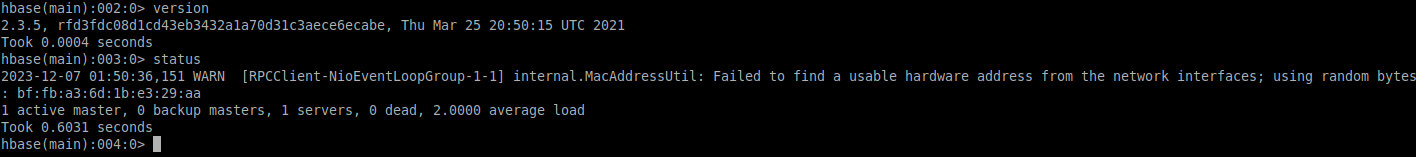


Рисунок 2 — Версия и статус

Мы видим версию Hbase (2.3.5) и автономное выполнение (1 активный мастер, 1 сервер).

Далее используем команду create для создания новой таблицы. Необходимо указать имя таблицы и имя семейства столбцов.

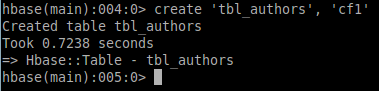


Рисунок 3 — Создание таблицы

Посмотрим список созданных таблиц с помощью команды list:

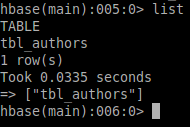


Рисунок 4 — Список таблиц

Выполним команду describe для просмотра подробных сведений, включая настройки по умолчанию.

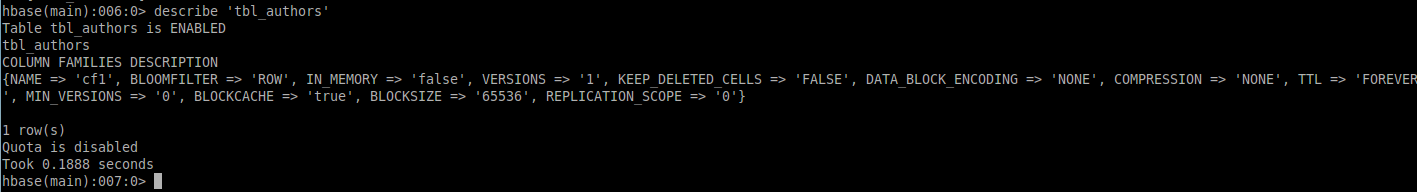


Рисунок 5 — Свойства таблицы

Попробуем удалить только что созданную таблицу.

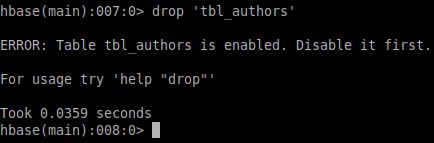


Рисунок 6 — Ошибка удаления таблицы

Возникла ошибка, т. к. таблица tbl\_authors включена. Сначала отключим её, чтобы затем удалить её.

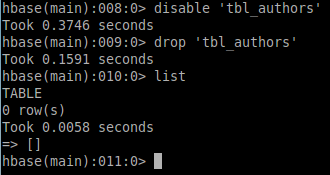


Рисунок 7 — Отключение и удаление таблицы

Создадим тестовую таблицу и используем команду put. Здесь мы вставляем три значения, по одному за раз.

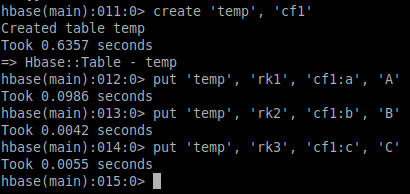


Рисунок 8 — Создание и заполнение таблицы

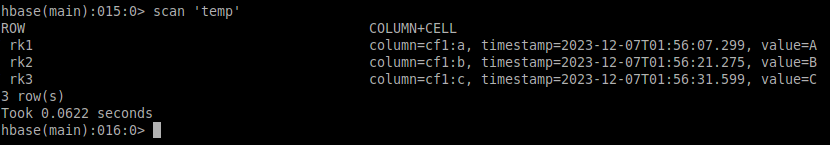


Рисунок 9 — Содержимое таблицы

Проведем подсчет количества строк во временной таблице HBase, созданной в предыдущем шаге с помощью команды count.

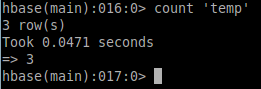


Рисунок 10 — Количество строк в таблице

Добавим еще значений. Ниже приведен результат сканирования данных в таблице temp. Общий список – 6, но фактическое количество строк – 5 из-за того, что в cf1 есть две строчки с ROWKEY rk=1.

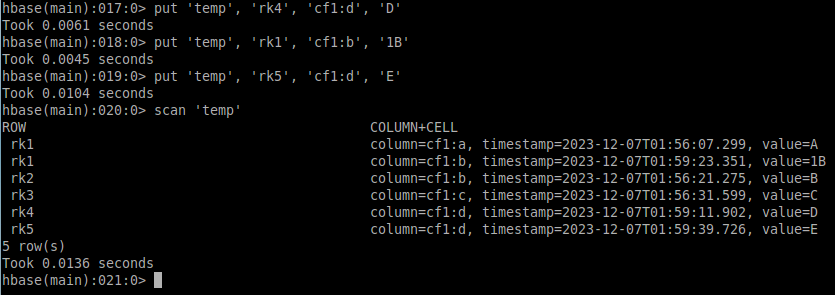


Рисунок 11 — Содержимое таблицы

Изменим атрибут version cf1 в таблице temp на 3.

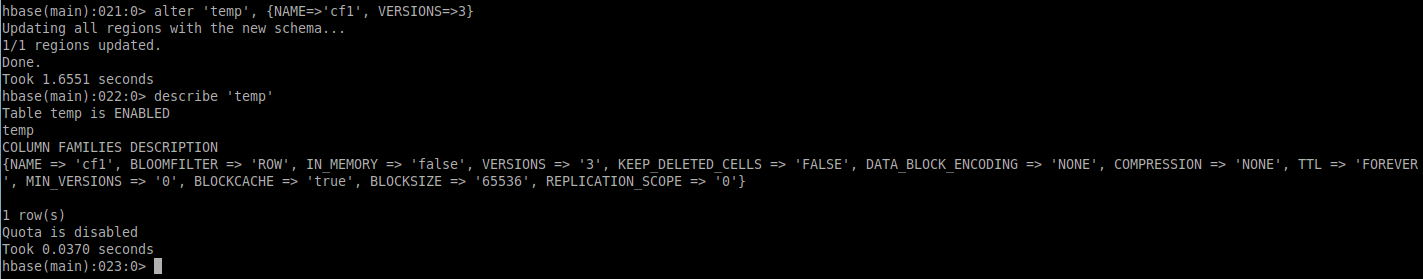


Рисунок 12 — Изменение атрибута version

Изменим временную таблицу, добавив семейство столбцов cf2 и cf3 с помощью команды ALTER.

Выполним команду desc, чтобы проверить, добавлены ли cf2 и cf3.

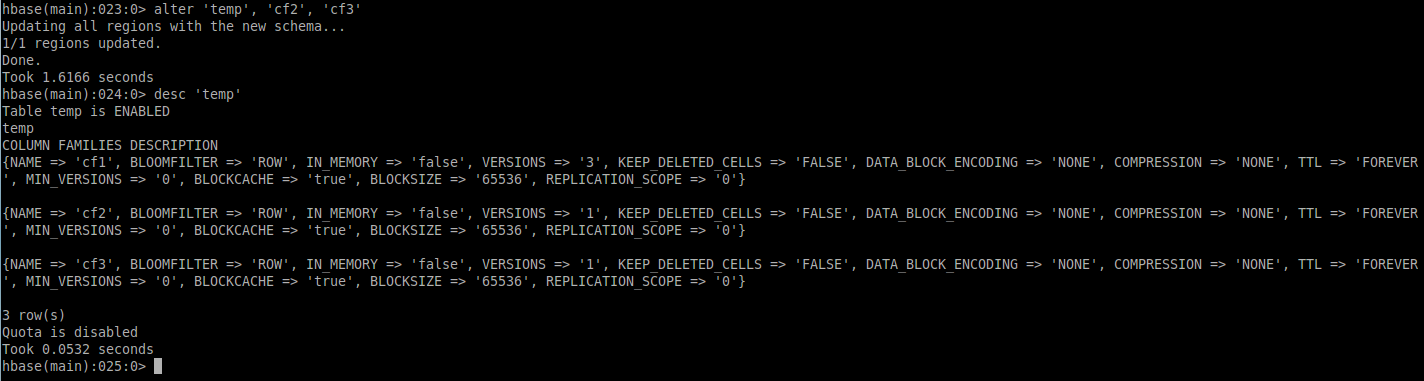


Рисунок 13 — Добавление семейства столбцов

Удалим с помощью delete только что добавленный cf3 и проверим результат.

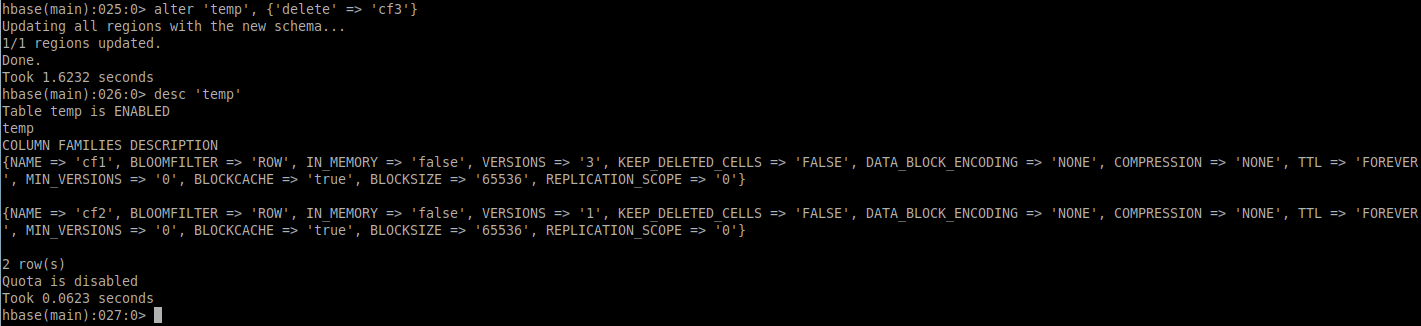


Рисунок 14 — Удаление cf3

Удалим временную таблицу. Выведем все таблицы, чтобы убедиться, что таблица была успешно удалена.

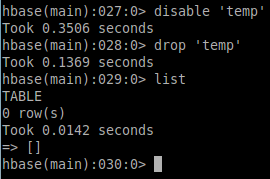


Рисунок 15 — Удаление таблицы

Выйдем из оболочки HBase с помощью команды quit и создадим файл с командами HBase.



Рисунок 16 — Выход из HBase и создание скрипта

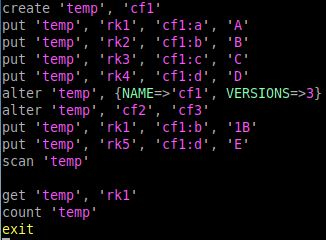


Рисунок 17 — Содержимое скрипта

Теперь этот файл можно выполнить из оболочки HBase.

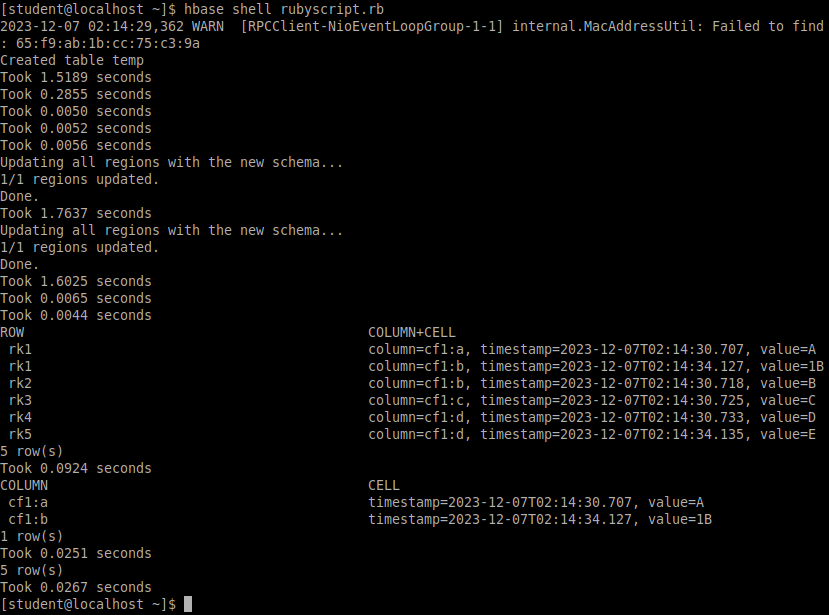


Рисунок 18 — Выполнение скрипта

**Доступ к данным с помощью команд DML**

В этой лабораторной работе мы будем использовать команды для вставки, извлечения, сканирования и удаления строк.

Запустим оболочку HBase. Введем данные в таблицу temp, созданной в предыдущей главе с помощью скрипта, и отобразим результат.

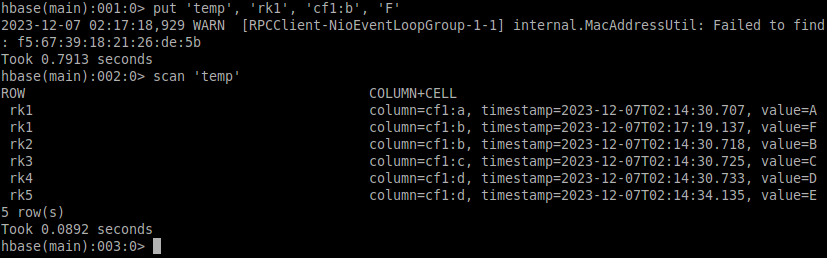


Рисунок 19 — Запуск оболочки и внесение данных в таблицу temp

Получим строки с ключом rk1 из временной таблицы с помощью get

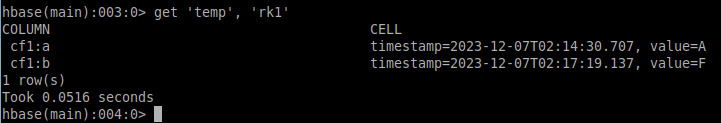


Рисунок 20 — Получение значений в строке

Получим два значения предыдущих версий столбца 'b' в строке rk1.

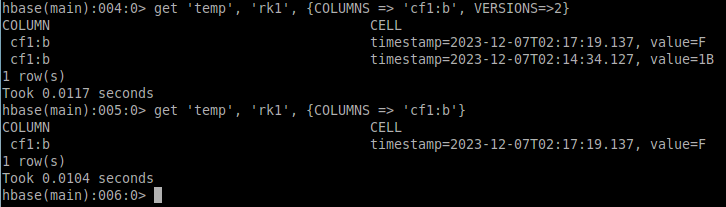


Рисунок 21 — Получение предыдущих версий ячейки

Просмотрим всю таблицу, но отобразим только столбцы 'a', 'b', 'c' с помощью явного их указания.

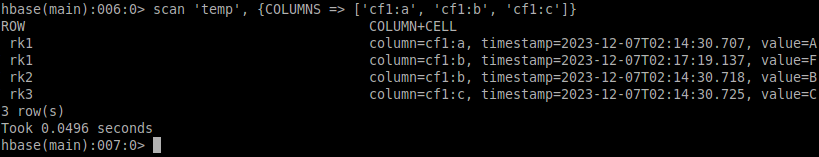


Рисунок 22 — Отображение конкретных столбцов

Удалим ячейки из столбца 'b' из таблицы temp с ключом строки = rk1 с помощью delete.

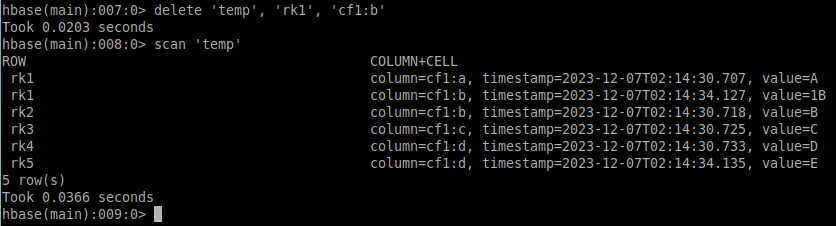


Рисунок 23 — Удаление ячейки

Удалим всю строку из временной таблицы в ключе строки rk1 с помощью deleteall.

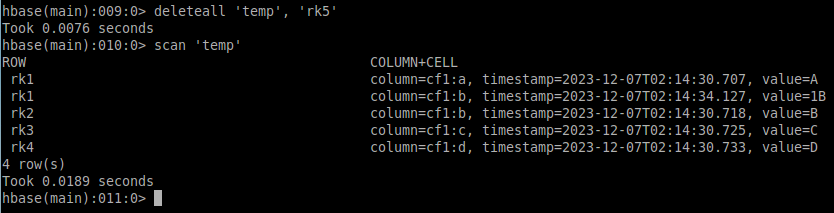


Рисунок 24 — Удаление строки

Выведем описание таблицы temp и проверим значение TTL столбцов из cf2. Значение TTL (Time-To-Live) по умолчанию — FOREVER, что означает, что срок действия версий ячейки никогда не истекает.

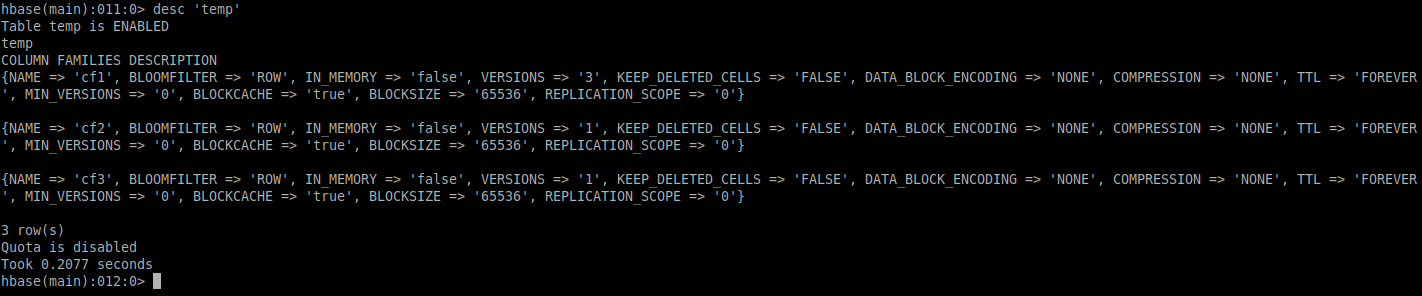


Рисунок 25 — Описание семейства столбцов cf2

Изменим значение с FOREVER на 30 секунд. Это означает, что версия будет удалена через 30 секунд после вставки данных в таблицу.

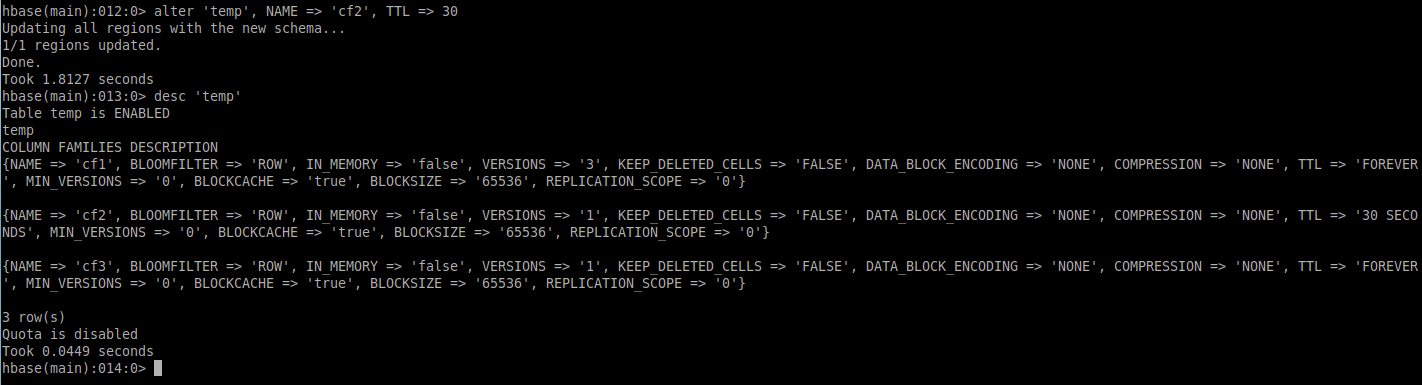


Рисунок 26 — Изменение атрибута TTL

Вставим новые данные и отобразим результаты.

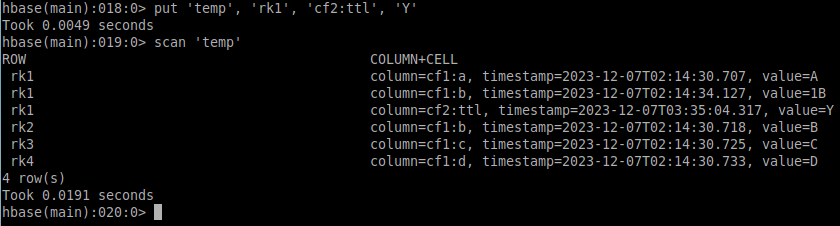


Рисунок 27 — Вставка новых данных в таблицу

Подождав не менее 30 секунд, выполним команду сканирования еще раз, чтобы узнать, истек ли срок действия вставленной строки и была ли она удалена.

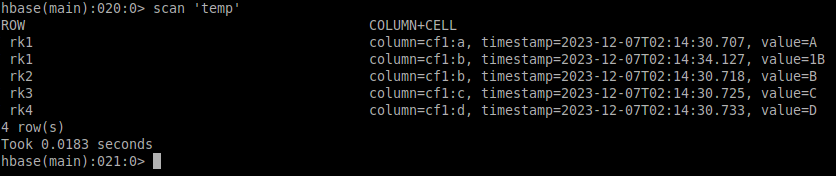


Рисунок 28 — Результат сканирования

Установим TTL на 10 секунд и установите MIN\_VERSIONS на 1 для cf2. Когда версия истекает с опцией MIN\_VERSION, она не будет удалена, если это единственная оставшаяся версия ячейки.

Проверим значения TTL и MIN\_VERSIONS в cf2.

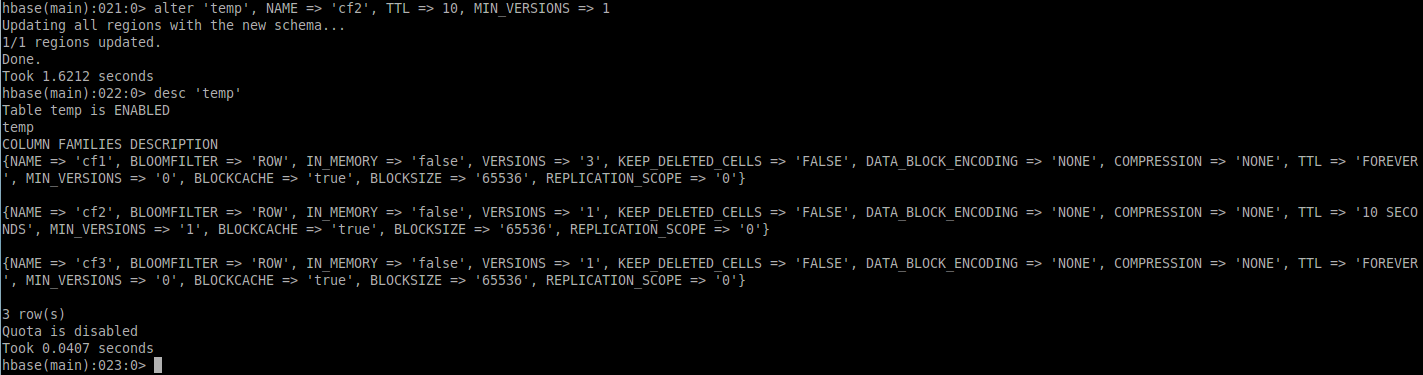


Рисунок 29 — Изменение атрибутов TTL и MIN\_VERSIONS в cf2

Вставим в таблицу новые данные и отобразим результат.

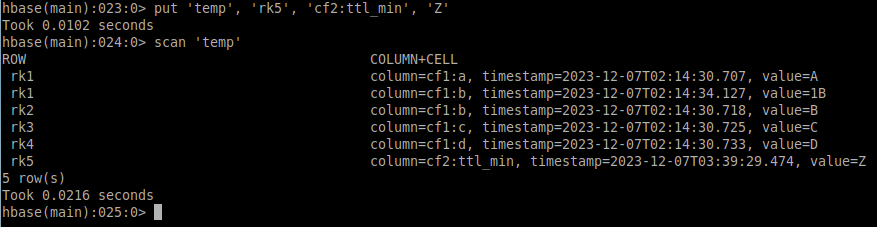


Рисунок 30 — Вставка новых данных в таблицу

Подождав не менее 10 секунд, выполним команду сканирования еще раз, чтобы узнать, истек ли срок действия вставленной строки и осталась ли она. Так как это единственная версия строки, она осталась, несмотря на TTL.

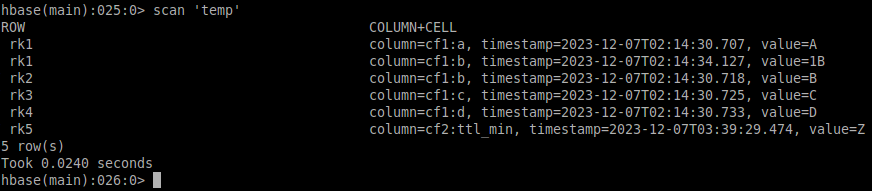


Рисунок 31 — Проверка через 10 секунд

**Самостоятельная работа с HBase**

Напишем команду для создания таблицы, отвечающей необходимым требованиям

Имя таблицы: **movie**. Содержит описание трех фильмов, каждому фильму соответствует столбец. Все столбцы находятся в семействе title. Название столбцов соответствуют названиям фильмов (выберите по желанию). В ячейках таблицы описание.

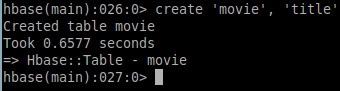


Рисунок 32 — Создание таблицы movie

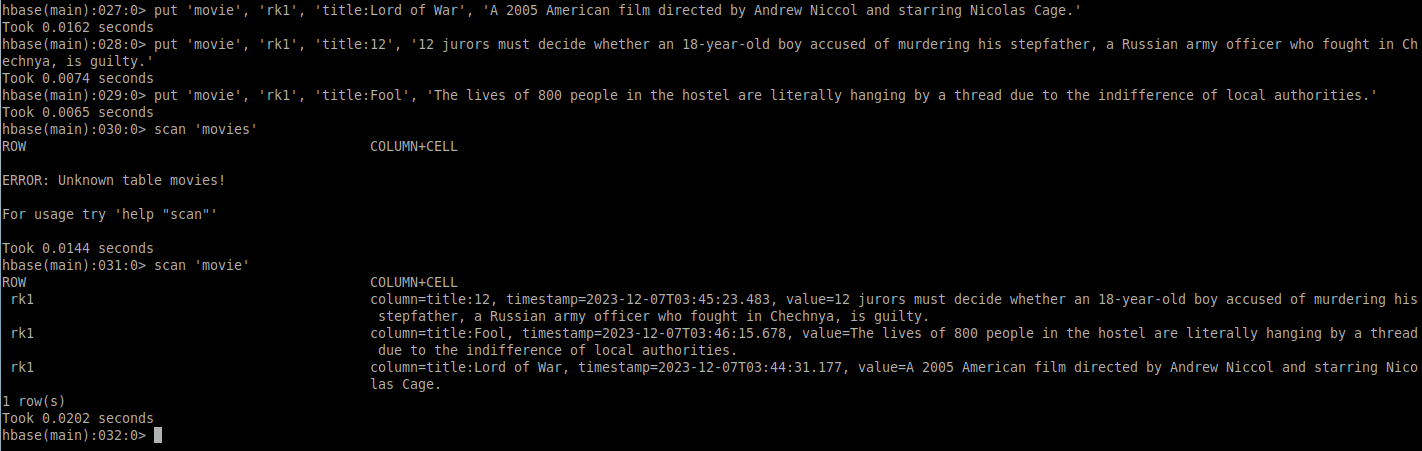


Рисунок 33 — Заполнение таблицы movie

Имя таблицы: **ranking**. Содержит информацию об оценках пользователей трех фильмов (опять же, фильмы - столбцы). В ячейках таблицы оценки. Используем команду alter, чтобы добавить семейство столбцов (column family) fl\_title для названий фильмов на другом языке.

### 

Рисунок 34 — Создание таблицы ranking

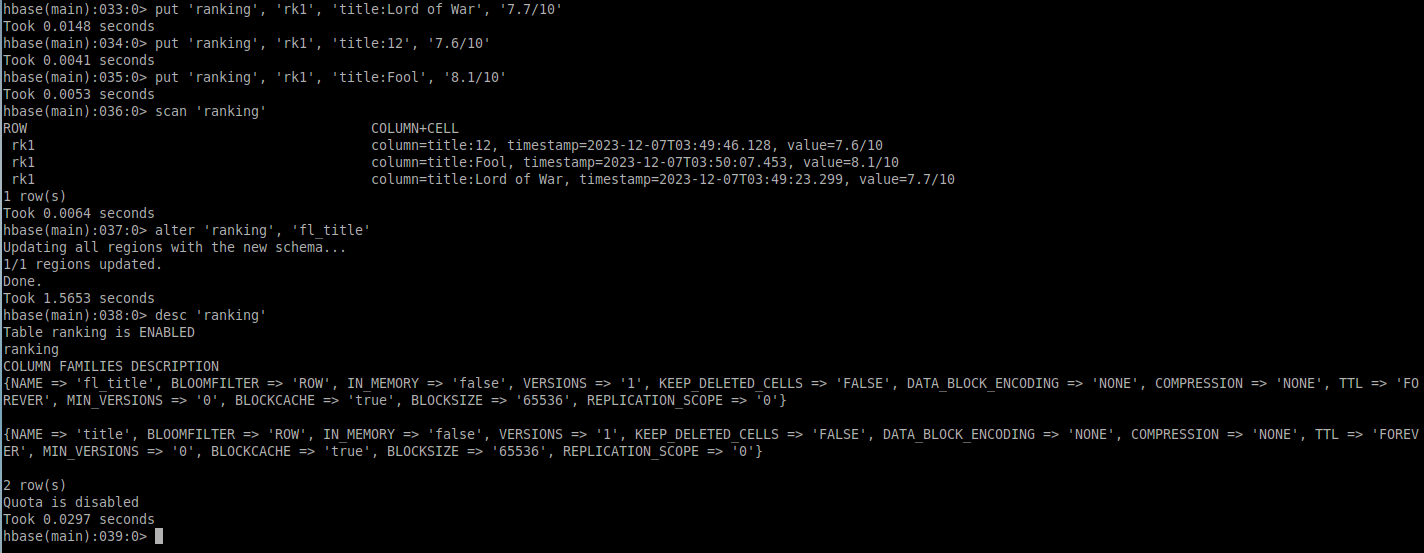


Рисунок 35 — Наполнение таблицы и дополнение семейства столбцов

Создадим таблицу user по следующим условиям:

* 1. Семейство столбцов: info.
  2. Создадим строку с Row key: rk100.
  3. Столбец age содержит значение 20.
  4. Столбец gender содержит значение F.
  5. Столбец zip содержит значение 18730.

Отобразим строку с ключом строки rk100 из пользовательской таблицы.

### 

Рисунок 36 — Создание и наполнение таблицы

Кроме того, вставим строку c rk100 в таблицу user со следующими атрибутами:

* 1. Row key: rk100, семейство столбцов: info, столбец age - значение 30.
  2. Row key: rk100, семейство столбцов: info столбец age - значение 40.

Отобразим строки с ключом строки rk100 из пользовательской таблицы.

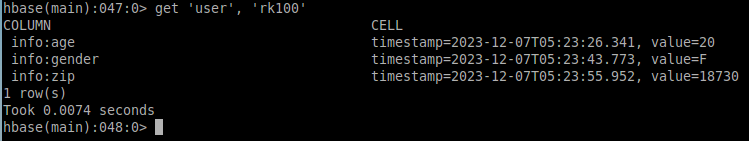


Рисунок 37 — Строка rk100

В пользовательской таблице получим все старые версии столбца age в строке с ключом rk100.

### 

Рисунок 38 — Версии ячейки age в строке rk100

Отобразим всю таблицу с помощью команды сканирования.

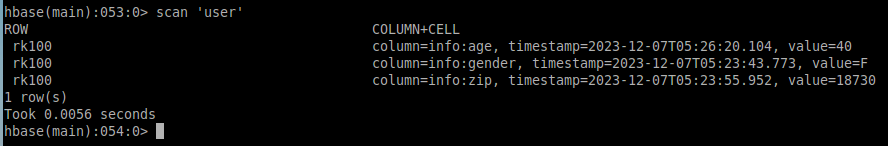


Рисунок 39 — Содержимое таблицы

В таблице пользователей удалим столбец info:age строки с ключом строки rk100.

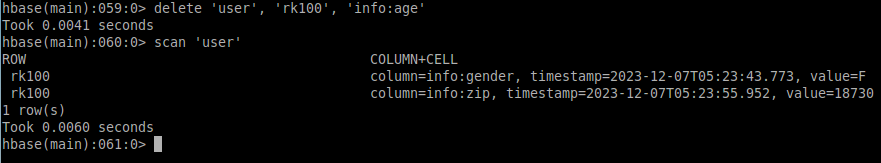


Рисунок 40 — Удаление ячейки

Удалим все строки с ключом строки rk100 из таблицы user.

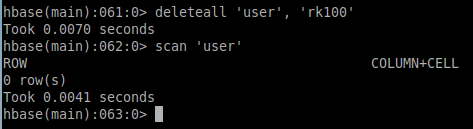


Рисунок 41 — Удаление rk100 и отображение содержимого