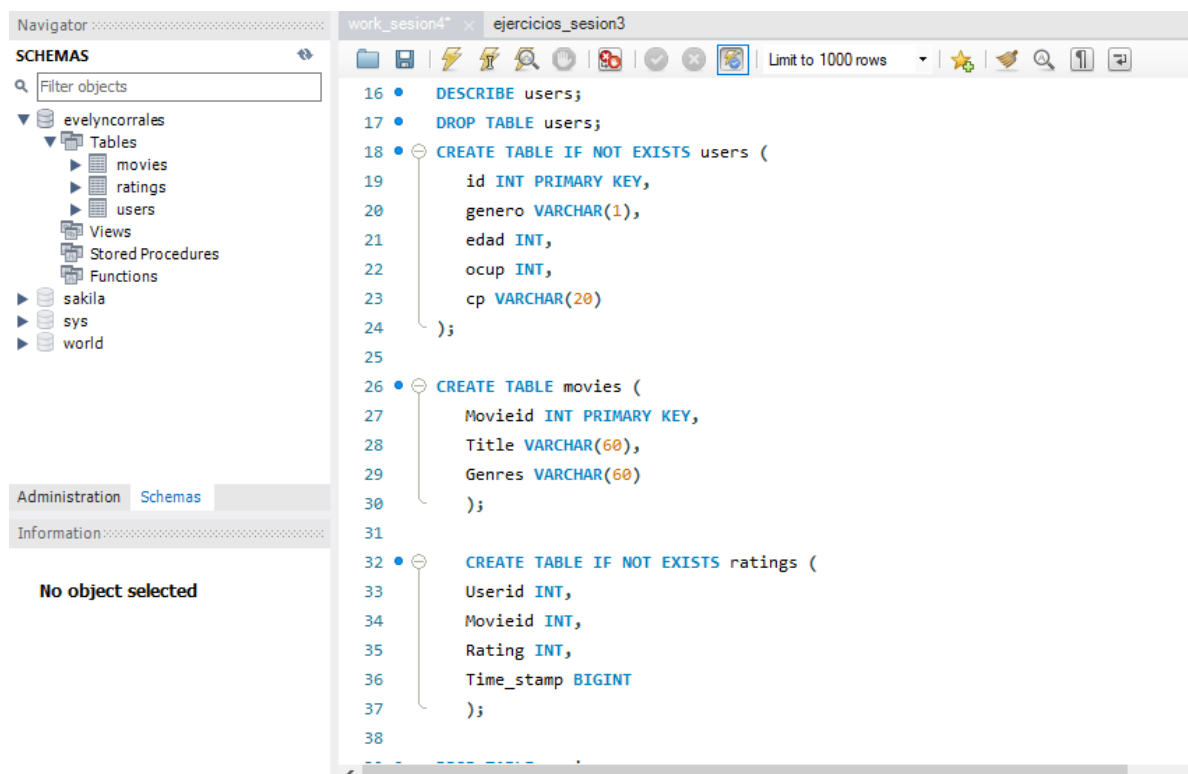


## Sesión 4

### Retos MYSQL\_1

1. Definir los campos y tipos de datos para la tabla `movies` haciendo uso de los archivos `movies.dat` y `README`.
2. Crear la tabla `movies` (recuerda usar el mismo nombre del archivo sin la extensión para vincular nombres de tablas con archivos).
3. Definir los campos y tipos de datos para la tabla `ratings` haciendo uso de los archivos `ratings.dat` y `README`.
4. Crear la tabla `ratings` (recuerda usar el mismo nombre del archivo sin la extensión para vincular nombres de tablas con archivos)



## Reto 2

1. Usando como base el archivo `movies.dat`, limpiarlo e importar los datos en la tabla `movies` creada en el Reto 1.
2. Usando como base el archivo `ratings.dat`, limpiarlo e importar los datos en la tabla `ratings` creada en el Reto 2.
3. Finalmente, añade un registro en cada tabla usando `INSERT INTO`.

The screenshot shows a database management tool interface. On the left, the 'Navigator' pane displays the 'evelyncorrales' schema with tables 'movies', 'ratings', and 'users'. The main editor shows a SQL script in 'work\_session4' with the following content:

```
37 );
38
39 DROP TABLE movies;
40 DROP TABLE ratings;
41 SELECT * FROM users LIMIT 10;
42 SELECT count(Movieid) FROM movies;
43 SELECT count(Movieid) FROM ratings;
44
45
```

Below the script, the 'Result Grid' shows a single row with the column 'count(Movieid)' and the value '70'. The 'Administration' tab is selected, and the 'Schemas' section shows 'No object selected'.

The 'Output' pane at the bottom shows the 'Action Output' with the following log:

| # | Time     | Action  |
|---|----------|---|
| 1 | 20:04:03 | SELECT count(Movieid) FROM movies LIMIT 0, 1000 |
| 2 | 20:04:07 | USE EvelynCorrales                              |
| 3 | 20:04:12 | SELECT count(Movieid) FROM movies LIMIT 0, 1000 |

File Edit View Query Database Server Tools Scripting Help

Navigator work\_session4 x

**SCHEMAS**

Filter objects

- evelyncorales
  - Tables
    - movies
    - ratings
    - users
  - Views
  - Stored Procedures
  - Functions
- sakila
- sys
- world

Administration Schemas

Information

No object selected

35 Rating INT,  
36 Time\_stamp BIGINT  
37 );  
38  
39 • DROP TABLE movies;  
40 • DROP TABLE ratings;  
41 • SELECT \* FROM users LIMIT 10;  
42 • SELECT count(Movieid) FROM movies;  
43 • SELECT count(Movieid) FROM ratings;

Result Grid Filter Rows: Export: Wrap Cell

|   | count(Movieid) |
|---|----------------|
| ▶ | 2895           |

Result 2 x

Output

Action Output

| #   | Time     | Action   |
|-----|----------|--|
| ✗ 1 | 20:04:03 | SELECT count(Movieid) FROM movies LIMIT 0, 1000  |
| ✓ 2 | 20:04:07 | USE EvelynCorales                                |
| ✓ 3 | 20:04:12 | SELECT count(Movieid) FROM movies LIMIT 0, 1000  |
| ✓ 4 | 20:05:02 | SELECT count(Movieid) FROM ratings LIMIT 0, 1000 |

File Edit View Query Database Server Tools Scripting Help

Navigator work\_session4\* x

**SCHEMAS**

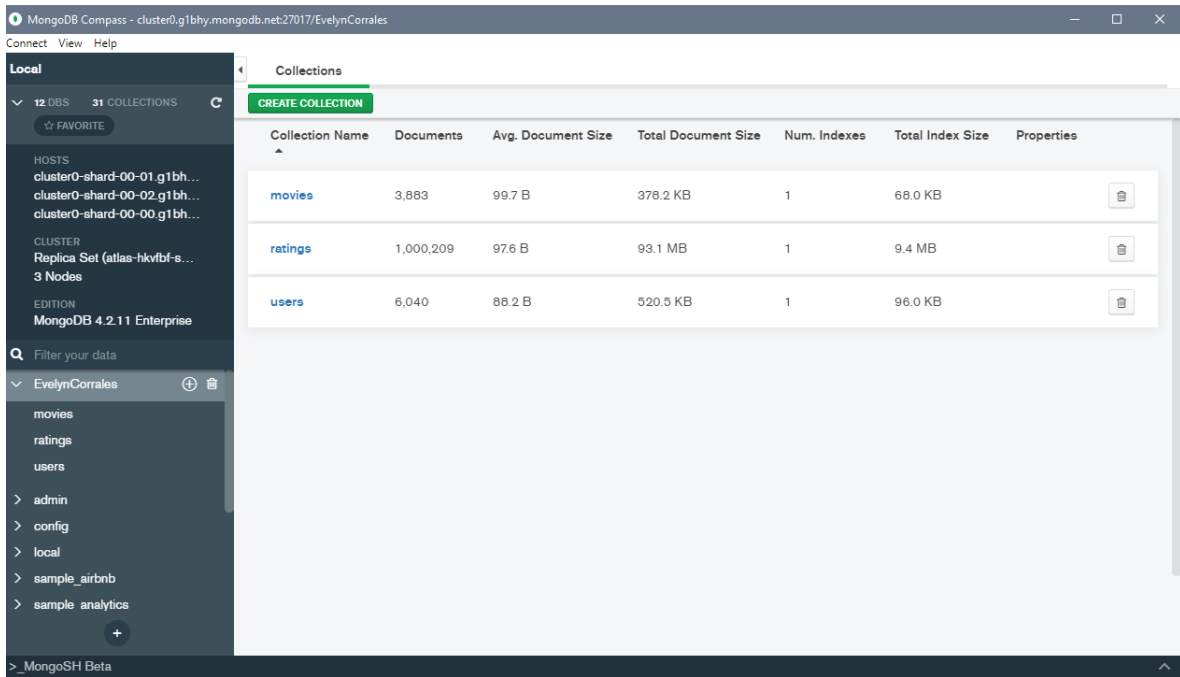
Filter objects

- evelyncorales
  - Tables
    - movies
    - ratings
    - users
  - Views
  - Stored Procedures
  - Functions
- sakila
- sys
- world

37 );  
38  
39 • DROP TABLE movies;  
40 • DROP TABLE ratings;  
41 • SELECT \* FROM users LIMIT 10;  
42 • SELECT \* FROM movies;  
43 • SELECT \* FROM ratings;  
44 • SELECT count(Movieid) FROM movies;  
45 • SELECT count(Movieid) FROM ratings;  
46  
47 • INSERT INTO movies (Movieid,Title,Genres) VALUES (72,'Frozen 2','Fiction, Kids');  
48 • INSERT INTO ratings (UserId,Movieid,Rating,Time\_stamp) VALUES (11,72,3,979775689);

## Retos Mongo

1. Crear la colección movies
2. Importar datos a la colección movies desde el archivo movies.csv
3. Crear la colección ratings
4. Importar datos a la colección ratings desde el archivo ratings.csv



## Ejercicios sesión 4

MongoDB Compass - cluster0.g1bhy.mongodb.net:27017/EvelynCorrales.movies

Connect View Collection Help

Local

12 DBS 31 COLLECTIONS

☆ FAVORITE

HOSTS

- cluster0-shard-00-01.g1bh...
- cluster0-shard-00-02.g1bh...
- cluster0-shard-00-00.g1bh...

CLUSTER

Replica Set (atlas-hkvbf-s...

3 Nodes

EDITION

MongoDB 4.2.11 Enterprise

Filter your data

EvelynCorrales

- movies
- ratings
- users

EvelynCorrales.movies Documents

DOCUMENTS 3.9k TOTAL SIZE 378.4KB AVG. SIZE 100B INDEXES 1 TOTAL SIZE 76.0KB AVG. SIZE 76.0KB

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER {MovieId: {\$in: ["4000", "4001"]}}

ADD DATA VIEW

Displaying documents 0 - 2 of 2

REFRESH

```
{ "_id": "ObjectId('5fd6e4ed0c35d95b9f77c114')",  
  "MovieId": "4000",  
  "Title": "Avengers: Endgame (2019)",  
  "Genres": "Fantasy|Sci-Fi" }  
  
{ "_id": "ObjectId('5fd6e5630c35d95b9f77c115')",  
  "MovieId": "4001",  
  "Title": "Glass (2019)",  
  "Genres": "Drama|Fantasy" }
```

MongoDB Compass - cluster0.g1bhy.mongodb.net:27017/EvelynCorrales.movies

Connect View Collection Help

Local

12 DBS 31 COLLECTIONS

☆ FAVORITE

HOSTS

- cluster0-shard-00-01.g1bh...
- cluster0-shard-00-02.g1bh...
- cluster0-shard-00-00.g1bh...

CLUSTER

Replica Set (atlas-hkvbf-s...

3 Nodes

EDITION

MongoDB 4.2.11 Enterprise

Filter your data

users

- admin
- config
- local
- sample\_airbnb
- sample\_analytics
- sample\_geospatial
- sample\_mflix
- sample\_restaurants

EvelynCorrales.movies Documents

DOCUMENTS 3.9k TOTAL SIZE 378.4KB AVG. SIZE 100B INDEXES 1 TOTAL SIZE 76.0KB AVG. SIZE 76.0KB

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER {MovieId: {\$in: ["4000", "4001"]}}

ADD DATA VIEW

Displaying documents 0 - 2 of 2

REFRESH

```
{ "_id": "ObjectId('5fd6e4ed0c35d95b9f77c114')",  
  "MovieId": "4000",  
  "Title": "Avengers: Endgame (2019)",  
  "Genres": "Fantasy|Sci-Fi" }  
  
{ "_id": "ObjectId('5fd6e5630c35d95b9f77c115')",  
  "MovieId": "4001",  
  "Title": "Glass (2019)",  
  "Genres": "Drama|Fantasy",  
  "valoraciones": Array  
    0: Object  
      UserId: "1563"  
      MovieId: "4001"  
      Rating: "4"  
    1: Object  
      UserId: "434"  
      MovieId: "4001"  
      Rating: "5" }
```