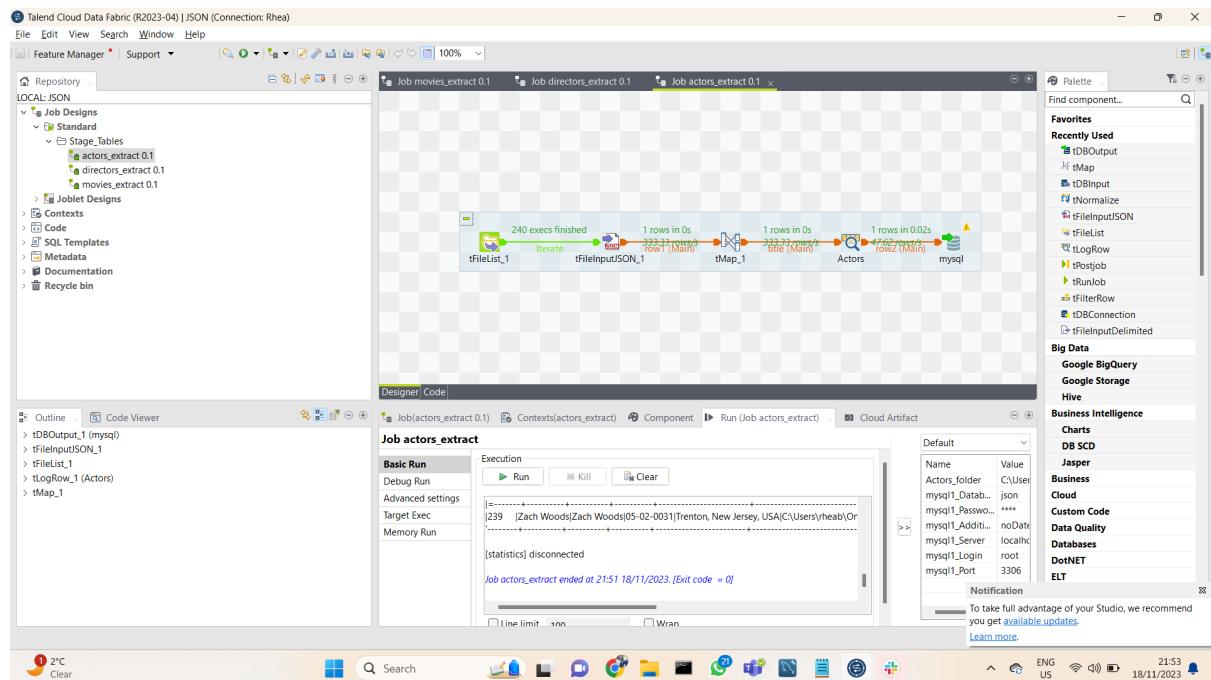
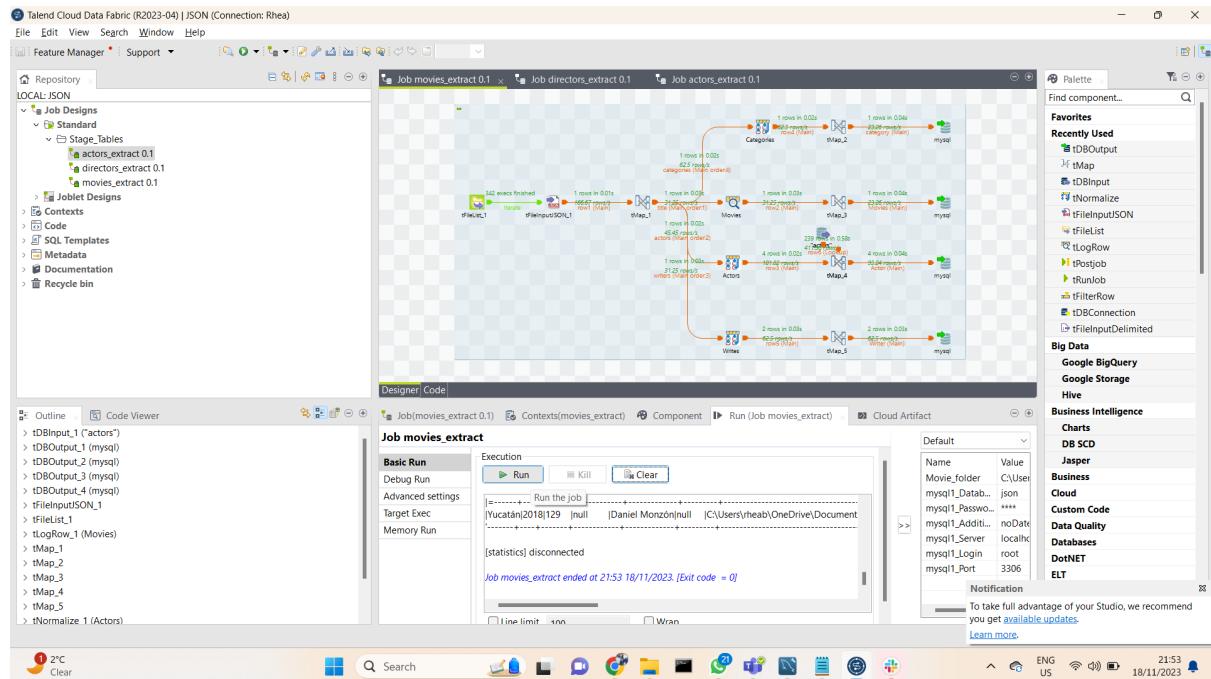
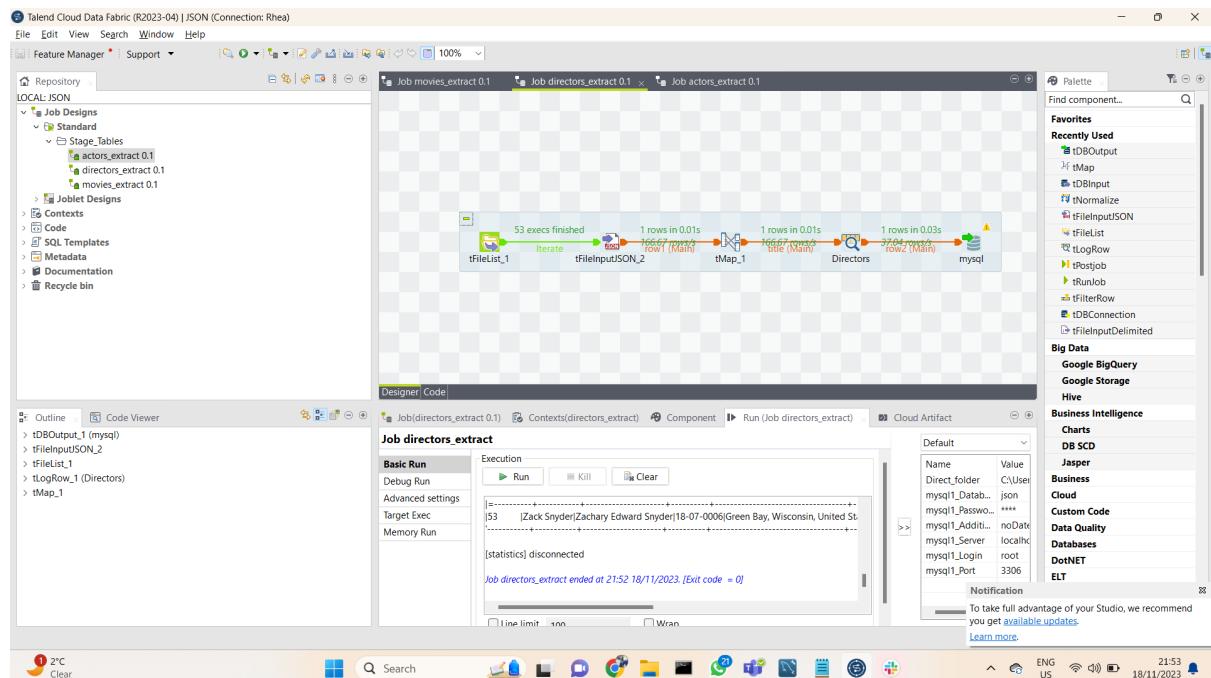


DADABI Assignment JSON





MySQL Workbench

unconnected Mysql@localhost:3306

File Edit View Query Database Server Tools Scripting Help

Navigator: Schemas

SCHEMAS

- food_inspection
- json
 - Tables
 - actors
 - categories
 - directors
 - movie
 - movie_actor
 - movie_writer
 - movie_categories
 - Views
 - Stored Procedures

Administration Schemas

Table: movie

Columns:

SK_title	name	year	runtime	release_date	director	storyline	FileName
1	Dr. No	1962	110	0011-02-13 00:00:00	Terence Young	A resourceful British government agent seeks a...	C:\Users\rheab\OneDrive\Documents
14	To Kill a Mockingbird	1962	129	0031-04-15 00:00:00	Robert Mulligan	Atticus Finch, a lawyer in the Depression-era So...	C:\Users\rheab\OneDrive\Documents
36	Whatever Happened to Baby Jane?	1962	134	0037-02-12 00:00:00	Robert Aldrich	A former child star torments her paraplegic siste...	C:\Users\rheab\OneDrive\Documents
46	From Russia with Love	1963	115	0016-02-14 00:00:00	Terence Young	James Bond willingly falls into an assassination p...	C:\Users\rheab\OneDrive\Documents

Result Grid Filter Rows: Edit Export/Import Wrap Cell Content: Result Grid

movie 4 x Output: Action Output

#	Time	Action	Message	Duration / Fetch
132	20:00:56	DROP TABLE `json`.`movie_categories`	0 row(s) affected	0.031 sec
133	20:01:26	SELECT * FROM `json.movie_categories` LIMIT 0, 1000	341 row(s) returned	0.000 sec / 0.000 sec
134	20:03:04	DROP TABLE `json`.`movie_categories`	0 row(s) affected	0.016 sec
135	20:07:02	SELECT * FROM `json.categories` LIMIT 0, 1000	996 row(s) returned	0.000 sec / 0.000 sec
136	20:07:23	DROP TABLE `json`.`movie_categories`	0 row(s) affected	0.031 sec
137	21:41:30	SELECT * FROM `json.movie` LIMIT 0, 1000	341 row(s) returned	0.063 sec / 0.015 sec

Object Info Session

Temps to drop Monday 21:41 18/11/2023

MySQL Workbench

unconnected x Mysql@localhost:3306 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Category wise* List year SQL File 3* SQL File 4* movie_actor x categories directors movie movie_writer movie_categories

Limit to 1000 rows

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Table: movie_actor

Columns:

- SK_Actors int PK
- Actors varchar(500)
- Title varchar(500)
- Birthname varchar(1000)
- Birthdate datetime
- Birthplace varchar(1000)

Object Info Session

Result Grid | Filter Rows: | Edit | Export/Import: | Wrap Cell Content: | Fetch Rows: | Result Grid | Context Help | Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
133	20:01:26	SELECT * FROM json.movie_categories LIMIT 0, 1000	341 row(s) returned	0.000 sec / 0.000 sec
134	20:03:04	DROP TABLE `json`.`movie_categories`	0 row(s) affected	0.016 sec
135	20:07:02	SELECT * FROM json.categories LIMIT 0, 1000	996 row(s) returned	0.000 sec / 0.000 sec
136	20:07:23	DROP TABLE `json`.`movie_categories`	0 row(s) affected	0.031 sec
137	21:41:30	SELECT * FROM json.movie LIMIT 0, 1000	341 row(s) returned	0.063 sec / 0.015 sec
138	21:42:48	SELECT * FROM json.movie_actor LIMIT 0, 1000	1000 row(s) returned	0.015 sec / 0.000 sec

2°C Clear 21:42 18/11/2023

MySQL Workbench

unconnected x Mysql@localhost:3306 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Category wise* List year SQL File 3* SQL File 4* movie_actor categories directors movie movie_writer x movie_categories

Limit to 1000 rows

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Table: movie_actor

Columns:

- SK_Actors int PK
- Actors varchar(500)
- Title varchar(500)
- Birthname varchar(1000)
- Birthdate datetime
- Birthplace varchar(1000)

Object Info Session

Result Grid | Filter Rows: | Edit | Export/Import: | Wrap Cell Content: | Fetch Rows: | Result Grid | Form Editor | Field Types | Query Stats | Context Help | Snippets

Action Output

#	Time	Action	Message	Duration / Fetch
134	20:03:04	DROP TABLE `json`.`movie_categories`	0 row(s) affected	0.016 sec
135	20:07:02	SELECT * FROM json.categories LIMIT 0, 1000	996 row(s) returned	0.000 sec / 0.000 sec
136	20:07:23	DROP TABLE `json`.`movie_categories`	0 row(s) affected	0.031 sec
137	21:41:30	SELECT * FROM json.movie LIMIT 0, 1000	341 row(s) returned	0.063 sec / 0.015 sec
138	21:42:48	SELECT * FROM json.movie_actor LIMIT 0, 1000	1000 row(s) returned	0.015 sec / 0.000 sec
139	21:43:28	SELECT * FROM json.movie_writer LIMIT 0, 1000	827 row(s) returned	0.016 sec / 0.000 sec

2°C Clear 21:43 18/11/2023

MySQL Workbench

unconnected x Mysql@localhost:3306 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Category wise List year SQL File 3* Category wise movie_actor categories directors movie movie_writer movie_categories sdmovies actors x

SCHEMAS: food_inspection json Tables: actors categories directors movie movie_actor movie_writer sdmovies

1 • SELECT * FROM json.actors;

Result Grid | Filter Rows: | Edit | Export/Import: | Wrap Cell Contents: |

SK_Actor	name	birthdate	birthplace	filename
1	Aamir Khan	Mohammed Aamir Hussain Khan	0019-07-17 00:00:00	Mumbai, Maharashtra, India
2	Adam Sandler	Adam Richard Sandler	0015-01-18 00:00:00	New York, United States
3	Adrienne Palicki	Adrienne Lee Palicki	0011-10-04 00:00:00	Toledo, Ohio, USA
4	Alicia Tyler	Alicia N. Tyler	0094-04-20 00:00:00	San Francisco, California, United States
5	Asintha Rai Bachchan	Asintha Rai	0007-03-27 00:00:00	Manesar, Haryana, India
6	Akshay Kumar	Rajiv Hari Om Bharia	0015-01-19 00:00:00	Amitsar, Punjab, India
7	Alexandra Daddario	Alexandra Anna Daddario	0021-08-07 00:00:00	Manhattan, New York City, New York, USA
8	Anika Heard	Amber Laura Heard	0023-09-07 00:00:00	Austin, Texas, US
9	Amitabh Bachchan	Amitabh Harivansh Rai Bachchan	0017-01-23 00:00:00	Allahabad, India
10	Amy Adams	Amy Lou Adams	0026-02-25 00:00:00	Vicenza, Italy
11	Anatole Taubman	Anatole Taubman	0029-04-22 00:00:00	Zurich, Switzerland
12	Angeline Jolie	Angeline Jolie Voight	0009-10-27 00:00:00	Los Angeles, California, United States
13	Anne Hathaway	Anne Jacqueline Hathaway	0018-04-05 00:00:00	Brooklyn, New York, USA

Action Output:

#	Time	Action	Message	Duration / Fetch
149	22:04:28	WITH CTE_movie AS (SELECT T.name, w.SK_Writers, w.Writers, a.actors, c.CategoriesCSV FROM movie...	0.016 sec
150	22:04:41	WITH CTE_movie AS (SELECT T.name, w.SK_Writers, w.Writers, a.actors, c.CategoriesCSV FROM movie...	0.125 sec / 0.000 sec
151	22:05:26	WITH CTE_movie AS (SELECT T.name, w.SK_Writers, w.Writers, a.actors, c.CategoriesCSV F...	0.078 sec / 0.000 sec
152	22:05:58	WITH CTE_movie AS (SELECT T.name, w.SK_Writers, w.Writers, a.actors, c.CategoriesCSV F...	0.078 sec / 0.000 sec
153	22:13:33	SELECT * FROM json.sdmovies LIMIT 0, 1000	339 row(s) returned	0.000 sec / 0.000 sec
154	22:21:47	SELECT * FROM json.actors LIMIT 0, 1000	239 row(s) returned	0.000 sec / 0.000 sec

Output:

Object Info Session

Query Completed

1°C Clear

ENG US 22:21 18/11/2023

MySQL Workbench

unconnected x Mysql@localhost:3306 x

File Edit View Query Database Server Tools Scripting Help

Navigator: Category wise List year SQL File 4* movie_actor categories x directors movie movie_writer movie_categories

SCHEMAS: food_inspection json Tables: actors categories directors movie movie_actor movie_writer sdmovies

1 • SELECT * FROM json.categories;

Execute the selected portion of the script or everything, if there is no selection

Result Grid | Filter Rows: | Edit | Export/Import: | Wrap Cell Contents: |

SK_Categories	name	CategoriesCSV
11	Dr. No	action
12	Dr. No	adventure
13	Dr. No	thriller
34	To Kill a Mockingbird	crime
35	To Kill a Mockingbird	drama
43	Whatever Happened To Baby Jane?	drama
44	Whatever Happened To Baby Jane?	horror
45	Whatever Happened To Baby Jane?	thriller
54	From Russia with Love	action
55	From Russia with Love	adventure
56	From Russia with Love	thriller
66	Goldfinger	action
67	Goldfinger	adventure
68	Goldfinger	thriller
78	The Sound of Music	biography

Action Output:

#	Time	Action	Message	Duration / Fetch
135	20:07:02	SELECT * FROM json.categories LIMIT 0, 1000	996 row(s) returned	0.000 sec / 0.000 sec
136	20:07:23	DROP TABLE json.movie_categories	0 row(s) affected	0.031 sec
137	21:41:30	SELECT * FROM json.movie LIMIT 0, 1000	341 row(s) returned	0.063 sec / 0.015 sec
138	21:42:48	SELECT * FROM json.movie_actor LIMIT 0, 1000	1000 row(s) returned	0.015 sec / 0.000 sec
139	21:43:28	SELECT * FROM json.movie_writer LIMIT 0, 1000	827 row(s) returned	0.016 sec / 0.000 sec
140	21:43:48	SELECT * FROM json.categories LIMIT 0, 1000	996 row(s) returned	0.000 sec / 0.000 sec

Output:

Object Info Session

2°C Clear

ENG US 21:43 18/11/2023

The screenshot shows the MySQL Workbench interface. The left sidebar displays the database schema with tables like `actors`, `categories`, `directors`, and `movie`. A query window is open with the following SQL code:

```
1 • SELECT * FROM json.directors;
```

The results grid shows 14 rows of director data. The history pane at the bottom lists the execution of various statements, including the creation of a table named 'movie_categories'.

SK_director	name	birthname	birthdate	birthplace	FileName
1	Akira Kurosawa	Akira Kurosawa	0028-05-22 00:00:00	Tokyo, Japan	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
2	Alfred Hitchcock	Alfred Joseph Hitchcock	0018-10-12 00:00:00	Leytonstone, Essex, England	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
3	Ang Lee	Ang Lee	0029-02-04 00:00:00	Pingtung, Taiwan	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
4	Anurag Kashyap	Anurag Singh Kashyap	0016-01-24 00:00:00	Gorakhpur, Uttar Pradesh, India	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
5	Billy Wilder	Samuel Wilder	0027-08-19 00:00:00	Sucha, Galicia, Austria-Hungary	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
6	Brad Bird	Brad Bird	0030-01-09 00:00:00	Kalispell, Montana, USA	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
7	Chris Columbus	Chris Joseph Columbus	0016-01-10 00:00:00	Spangler, Pennsylvania, USA	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
8	Christopher Nolan	Christopher Edward Nolan	0035-11-21 00:00:00	Westminster, London, England	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
9	Clint Eastwood	Clinton Eastwood Jr.	0036-08-11 00:00:00	San Francisco, California, U.S.	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
10	Damien Chazelle	Imtiaz Ali	0024-06-07 00:00:00	Providence, Rhode Island, United States	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
11	David Fincher	David Andrew Leo Fincher	0033-12-14 00:00:00	Denver, Colorado, United States	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
12	David Lynch	David Keith Lynch	0025-04-29 00:00:00	Missoula, Montana, USA	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
13	Dibakar Banerjee	Dibakar Banerjee	0026-10-21 00:00:00	New Delhi, India	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2
14	Francis Ford Coppola	Francis Ford Coppola	0012-07-19 00:00:00	Detroit, Michigan, United States	C:\Users\Head\OneDrive\Documents\databas.json\OneDrive_2

```
WITH CTE_movie AS (
    SELECT t.name,
    w.SK_Writers,
    w.Writers,
    a.actors,
    c.CategorieCSV
    FROM movie t
    LEFT JOIN json.movie_actor a
    ON t.name = a.title
    LEFT JOIN movie_writer w
    ON t.name = w.title
    LEFT JOIN categories c
    ON t.name = c.name
)
```

```
SELECT name,
CategorieCSV,
COUNT(DISTINCT SK_Writers) AS no_of_writer,
```

```
COUNT(DISTINCT actors) AS no_of_actor
```

```
FROM CTE_movie
GROUP BY name,
CategorieCSV
HAVING no_of_writer > 4 AND no_of_actor > 3;
```

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL code:

```
13     LEFT JOIN categories c
14     ON t.name = c.name
15   )
16
17   SELECT name,
18   CategorieCSV,
19   COUNT(DISTINCT SK_Writers) AS no_of_writer,
20   COUNT(DISTINCT actors) AS no_of_actor
21
22   FROM CTE_movie
23   GROUP BY name,
24   CategorieCSV
25   HAVING no_of_writer > 4 AND no_of_actor > 3;
```

The results grid displays the following data:

name	CategorieCSV	no_of_writer	no_of_actor
Ant-Man	action	6	4
Ant-Man	adventure	6	4
Ant-Man	comedy	6	4
Ant-Man and the Wasp	action	5	5
Ant-Man and the Wasp	adventure	5	5

The output pane shows the following log entries:

#	Time	Action	Message	Duration / Fetch
147	21:49:45	DROP TABLE `json`.`movie_writer`	0 row(s) affected	0.047 sec
148	22:03:12	SELECT name, CategorieCSV, COUNT(DISTINCT SK_Writers) AS writer_count, COUNT(DISTINCT actors) AS actor_count	Error Code: 1146. Table 'json.cte' doesn't exist	0.032 sec
149	22:04:28	WITH CTE_movie AS (SELECT t.name, w_SK_Writers, w_Writers, a.actors, c.CategorieCSV FROM movie...)	Error Code: 1054. Unknown column 'actor_count' in having clause	0.016 sec
150	22:04:41	WITH CTE_movie AS (SELECT t.name, w_SK_Writers, w_Writers, a.actors, c.CategorieCSV FROM movie...)	36 rows(s) returned	0.125 sec / 0.000 sec
151	22:05:26	WITH CTE_movie AS (SELECT t.name, w_SK_Writers, w_Writers, a.actors, c.CategorieCSV F...	36 rows(s) returned	0.078 sec / 0.000 sec
152	22:05:58	WITH CTE_movie AS (SELECT t.name, w_SK_Writers, w_Writers, a.actors, c.CategorieCSV F...	36 rows(s) returned	0.078 sec / 0.000 sec

**SELECT name, year
FROM movie
ORDER BY year;**

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Schemas:** food_inspection, json (selected), categories, directors, movie, movie_actor, movie_writer, new_schema_t1and, product_pricing, sys.
- Query Editor:** Contains the SQL query:


```
1 • SELECT name, year
2   FROM movie
3   ORDER BY year;
4
```
- Result Grid:** Displays the results of the query:

name	year
Dr. No	1962
To Kill a Mockingbird	1962
Whatever Happened To Baby Jane?	1962
From Russia with Love	1963
Goldfinger	1964
The Sound of Music	1965
Thunderball	1965
The Good, the Bad and the Ugly	1966
You Only Live Twice	1967
Diamonds Are Forever	1971
The Godfather	1972
La Patagonia rebelde	1974
The Godfather: Part II	1974
Monty Python and the Holy Grail	1975
- Output:** Shows the execution log with the following message:


```
Object Info Session
① 115 17:45:34 with cte as( SELECT t.name, w.SK_Writers, w.Writers, a.actors, c.CategorieCSV FROM movie t left join json.movie_writer w on t.id = w.movie_id left join json.actors a on w.Writers = a.id left join json.categories c on w.movie_id = c.movie_id ) select * from cte order by year
Message: Error Code: 1054. Unknown column 'actor_count' in 'having clause'
Duration / Fetch: 0.015 sec
```
- System Bar:** Shows the date (18/11/2023), time (17:48), and system status (ENG US).

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Schemas:** food_inspection, json (selected), categories, directors, movie, movie_actor, movie_writer, scdmovies (selected), movie_categories.
- Query Editor:** Contains the SQL query:


```
1 • SELECT * FROM json.scdmovies;
```
- Result Grid:** Displays the results of the query:

SK_title	name	year	runtime	release_date	director	storyline	Filename
1	Dr. No	1962	110	1962-01-01	Terence Young	A resourceful British government agent seizes a... C:\Users\heab\OneDrive\Documents\datab	
14	To Kill a Mockingbird	1962	129	1962-01-01	Robert Mulligan	Atticus Finch, a lawyer in the Depression-era Sa... C:\Users\heab\OneDrive\Documents\datab	
26	Whatever Happened To Baby Jane?	1962	134	1962-01-01	Robert Aldrich	A former child star torments her paraplegic sister... C:\Users\heab\OneDrive\Documents\datab	
46	From Russia with Love	1963	115	1963-01-01	Terence Young	James Bond willingly falls into an assassination p... C:\Users\heab\OneDrive\Documents\datab	
57	Goldfinger	1964	110	1964-01-01	Guy Hamilton	Investigating a gold magnate's smuggling, James... C:\Users\heab\OneDrive\Documents\datab	
69	The Sound of Music	1965	165	1965-01-01	Robert Wise	A woman leaves an Austrian convent to become a... C:\Users\heab\OneDrive\Documents\datab	
81	Thunderball	1965	130	1965-01-01	Terence Young	James Bond heads to the Bahamas to recover a... C:\Users\heab\OneDrive\Documents\datab	
93	The Good, the Bad and the Ugly	1966	161	1966-01-01	Sergio Leone	A bounty hunting com... C:\Users\heab\OneDrive\Documents\datab	
105	You Only Live Twice	1967	117	1967-01-01	Lewis Gilbert	Agent 007 and the Japanese secret service ninj... C:\Users\heab\OneDrive\Documents\datab	
116	Diamonds Are Forever	1971	120	1971-01-01	Guy Hamilton	A diamond smuggling investigation leads James... C:\Users\heab\OneDrive\Documents\datab	
126	The Godfather	1972	177	1972-01-01	Francis Ford C...	When the aging head of a famous crime family ... C:\Users\heab\OneDrive\Documents\datab	
158	La Patagonia rebelde	1974	103	1974-01-01	Hector Olivera	La Patagonia rebelde is an Argentinean film bas... C:\Users\heab\OneDrive\Documents\datab	
- Output:** Shows the execution log with the following messages:


```
Object Info Session
① 148 22:03:12. SELECT name, CategorieCSV, COUNT(DISTINCT SK_Writers) AS writer_count, COUNT(DISTINCT actors) AS actor_count
Message: Error Code: 1146. Table 'json.cte' doesn't exist
Duration / Fetch: 0.032 sec
② 149 22:04:28 WITH CTE_movie AS ( SELECT t.name, w.SK_Writers, w.Writers, a.actors, c.CategorieCSV FROM movie t left join json.movie_writer w on t.id = w.movie_id left join json.actors a on w.Writers = a.id left join json.categories c on w.movie_id = c.movie_id ) select * from cte
Message: Error Code: 1054. Unknown column 'actor_count' in 'having clause'
Duration / Fetch: 0.016 sec
③ 150 22:04:41 WITH CTE_movie AS ( SELECT t.name, w.SK_Writers, w.Writers, a.actors, c.CategorieCSV FROM movie t left join json.movie_writer w on t.id = w.movie_id left join json.actors a on w.Writers = a.id left join json.categories c on w.movie_id = c.movie_id ) select * from cte
Message: 36 row(s) returned
Duration / Fetch: 0.125 sec / 0.000 sec
④ 151 22:05:26 WITH CTE_movie AS ( SELECT t.name, w.SK_Writers, w.Writers, a.actors, c.CategorieCSV FROM movie t left join json.movie_writer w on t.id = w.movie_id left join json.actors a on w.Writers = a.id left join json.categories c on w.movie_id = c.movie_id ) select * from cte
Message: 36 row(s) returned
Duration / Fetch: 0.078 sec / 0.000 sec
⑤ 152 22:05:58 WITH CTE_movie AS ( SELECT t.name, w.SK_Writers, w.Writers, a.actors, c.CategorieCSV FROM movie t left join json.movie_writer w on t.id = w.movie_id left join json.actors a on w.Writers = a.id left join json.categories c on w.movie_id = c.movie_id ) select * from cte
Message: 36 row(s) returned
Duration / Fetch: 0.078 sec / 0.000 sec
⑥ 153 22:13:33 SELECT * FROM json.scdmovies LIMIT 0, 1000
Message: 339 row(s) returned
Duration / Fetch: 0.000 sec / 0.000 sec
```
- System Bar:** Shows the date (18/11/2023), time (22:13), and system status (ENG US).

