

## SQL QUERIES WITH THE BUSINESS REQUIREMENTS

- ① Get Different Types of Professions for Any Given Personnel

```
SELECT nconst, PRIMARY_NAME, PRIMARY_PROFESSION, KNOWN_FOR_TITLES  
  
FROM PERSONNEL_DIM  
WHERE nconst = 'nm8492168';
```

- ② Find Personnel with More Than One Profession

```
SELECT nconst, PRIMARY_NAME, COUNT(DISTINCT PRIMARY_PROFESSION) AS  
num_professions  
FROM PERSONNEL_DIM  
GROUP BY nconst, PRIMARY_NAME  
HAVING COUNT(DISTINCT PRIMARY_PROFESSION) > 1  
ORDER BY num_professions DESC;
```

- ③ Get List of Genres for a Given Title & Identify Movies by Genre

```
SELECT t.primaryTitle, g.genre_name  
FROM Fact_Movie_Episode f  
JOIN Bridge_Title_Genre_Language bg ON f.tconst = bg.movie_id  
JOIN Dim_Genre g ON bg.entity_id = g.genre_id  
JOIN Dim_Title t ON f.tconst = t.tconst  
WHERE t.primaryTitle = '<movie_name>'; -- Replace with actual title
```

- ④ Find All Movies Released in a Given Year

```
SELECT primary_Title, START_YEAR, TITLE_TYPE  
FROM DIM_TITLE;  
WHERE startYear = 2022 -- Replace with required year  
ORDER BY primaryTitle;
```

- ⑤ Track Movie Length to Generate Metrics on Runtime by Year

```
SELECT startYear, AVG(runtimeMinutes) AS avg_runtime,  
       MIN(runtimeMinutes) AS min_runtime,  
       MAX(runtimeMinutes) AS max_runtime  
FROM Dim_Title  
WHERE runtimeMinutes IS NOT NULL  
GROUP BY startYear  
ORDER BY startYear;
```

⑥ List All Adult & Non-Adult Movies

```
SELECT primary_Title, START_YEAR, IS_ADULT
FROM Dim_Title
ORDER BY IS_ADULT DESC, START_YEAR;
```

⑦ Get All Languages a Title is Associated With

```
SELECT t.primaryTitle, l.language_name
FROM Fact_Movie_Episode f
JOIN Bridge_Title_Genre_Language bl ON f.tconst = bl.movie_id
JOIN Dim_Language l ON bl.entity_id = l.language_id
JOIN Dim_Title t ON f.tconst = t.tconst
WHERE t.primaryTitle = '<movie_name>'; -- Replace with actual movie name
```

⑧ Get List of Regions Where a Movie Was Released

```
SELECT t.primary_Title, l.region
FROM Fact_Movie_Episode f
JOIN REGION_DIM R ON f.location_id = R.REGION_id
JOIN Dim_Title t ON f.tconst = t.tconst
WHERE t.primary_Title = '<movie_name>'; -- Replace with actual movie name
```

⑨ Get List of All Directors & Writers for a Given Movie

```
SELECT t.primary_Title, p.primary_Name, B.CATEGORY
FROM Fact_Movie_Episode f
JOIN Bridge_Title_Personnel b ON f.tconst = b.movie_id
JOIN PERSONNEL_DIM p ON b.person_id = p.nconst
JOIN Dim_Title t ON f.tconst = t.tconst
WHERE b.category IN ('Director', 'Writer')
AND t.primaryTitle = '<movie_name>'; -- Replace with actual movie name
```

⑩ Get the Number of Episodes Associated With a TV Series

```
SELECT parent_Tconst AS series_id, COUNT(*) AS num_episodes
FROM Fact_Movie
WHERE PARENT_TCONST IS NOT NULL
GROUP BY PARENT_TCONST
ORDER BY num_episodes DESC;
```

⑪ Find All Different Types of Crew, Cast, Directors, Writers for a Movie

```
SELECT t.primary_Title, p.primary_Name, b.category, b.job
FROM Fact_Movie f
JOIN Bridge_Title_Personnel b ON f.tconst = b.movie_id
JOIN PERSONNEL_DIM p ON b.person_id = p.nconst
JOIN Dim_Title t ON f.tconst = t.tconst
WHERE t.primary_Title = '<movie_name>'; -- Replace with actual movie name
```

12 Get List of Jobs Involved in a Given Title

```
SELECT DISTINCT job
FROM Bridge_Title_Personnel
WHERE movie_id = '<tconst>'; -- Replace with actual movie ID
SELECT * FROM FACT_MOVIE;
```

13 Get List of Characters in a Given Title

```
SELECT t.primary_Title, p.primary_Name, b.characters
FROM Fact_Movie f
JOIN Bridge_Title_Personnel b ON f.tconst = b.movie_id
JOIN PERSONNEL_DIM p ON b.person_id = p.nconst
JOIN Dim_Title t ON f.tconst = t.tconst
WHERE t.primaryTitle = '<movie_name>' -- Replace with actual movie name
AND b.characters IS NOT NULL;
```

14 Find Top-Rated Movies by Year and Genre

```
SELECT t.start_Year, g.genre_name, t.primary_Title, f.averageRating, f.numVotes
FROM Fact_Movie_Episode f
JOIN Bridge_Title_Genre_Language bg ON f.tconst = bg.movie_id
JOIN Dim_Genre g ON bg.entity_id = g.genre_id
JOIN Dim_Title t ON f.tconst = t.tconst
WHERE f.averageRating IS NOT NULL
ORDER BY f.averageRating DESC, f.numVotes DESC
LIMIT 10;
```