Case study

Topic : FILM database of last 12 years;

This case study includes 5 tables which are connected internally using foreign key.

1. Films
2. Film\_crew
3. Film\_company
4. Reviews
5. Awards

In which you will get to know the details of the Movies their respective crew include cast, Writer, Director and their genre, ratings also which film company has produced those movies. This database also include awards table which help with the information regrading what movies has won the awards in which year.

Let’s take a look at the ER diagram to understand visual representation of the Table's structure and the relationships between logically related tables.

Graphical user interface, application

Description automatically generated

As you can see in the ER diagram Film tables include all the information related to the movies in which Film\_id field has the primary key constraints. Let’s describe the film table to get the structure of table which include name of the column, data-type of column and constraint information.

Syntax : Desc films;

Text

Description automatically generated

To retrieve the records from the film table, we can use select statement..

Syntax : Select \* from films;

Graphical user interface

Description automatically generated with low confidence

As you can see using the select statement we got all the information of movie example:

URI movie has unique film\_id “1” which is released in year 2019 in INDIA has duration of 120 minutes collecting gross over 352 million $.

Note: Duration field is in minutes and budget & gross is in million dollars($).

Next table:

Film\_company has the information of production\_company their respective Pid which is the unique id given to each production company and Pid column have the primary key constraint

Let’s describe the film\_company table to get information of schema

Syntax : Desc film\_company;

A picture containing text

Description automatically generated

To retrieve the records from the film table, we can use select statement..

Syntax : Select \* from film\_company;

A picture containing text

Description automatically generated

From the above table we can see that production company T-series has its own unique Pid “P107” founded in year 1983 based in INDIA and Chairman name is Bhushan Kumar Dua.

Next table:

film\_ crew has information of crew associate with respective movie.

Syntax : Desc film\_crew;

Text

Description automatically generated with medium confidence

As you can see the Film\_id column has foreign key constraint reference from Films table and Pid field also a have foreign key constraint reference from film\_company entity.

To retrieve the records from the film\_crew table, we can use select statement..

Syntax : Select \* from film\_crew;

A screenshot of a computer

Description automatically generated with medium confidence

From the above table we found information related to film\_crew like cast, Writer, Director and which film\_company production connected with that Movie.

Next table:

Reviews entity has detailed information of Movie like what certification does this movie have, what genre of movie is this and there respective IMDB score and rotten tomatoes

Type of Genre includes: Action, Adventure, Drama, Anime, Thriller, Sports, Sci-fi, War etc.

and Certification includes Pg-13, G, R, 18+, Not rated etc..

Syntax : Desc Review;

Text

Description automatically generated

As you can see in this table Film\_id has the foreign key constraint reference from Films(film\_id);

To retrieve the records from the film\_crew table, we can use select statement..

Syntax : Select \* from review;

Graphical user interface

Description automatically generated with low confidence

Last table:

Awards table is give the details of which movies are nominated and if that movie has won the award or not also helps with the type of award and how many awards that movie win in what year……

Syntax : Desc awards;

Diagram

Description automatically generated with low confidence

As you can see in this table Film\_id has the foreign key constraint reference from Films(film\_id);

To retrieve the records from the Awards table, we can use select statement..

Syntax : Select \* from awards;

Text

Description automatically generated

This awards table help us with the last 12 year awards won by the films..

Sub-Queries we might need to filter out data from above database :

1. From our film database if we want to know which movies are hit movies yearly having net profit more than 500 million dollars.

Ans= Syntax :

select title,release\_year,(gross-budget) as net\_profit from films where (gross-budget) in (select (Gross-budget) from films where (gross-budget) > 500) order by release\_year;

` A screenshot of a computer

Description automatically generated with low confidence

Similarly, If we wanted to know which are flop movies yearly having net profit less than

300 million dollars…

select title,release\_year,(gross-budget) as net\_profit from films where (gross-budget) in (select (Gross-budget) from films where (gross-budget) < 300) order by release\_year;

A picture containing text

Description automatically generated

1. From our film database if we want to know which films are favourites by the people’s choice having IMDB rating more than 8 and rotten tomatoes >75 ?

Ans=

Syntax :

select film\_id,title from films where film\_id in (select film\_id from reviews where (imdb\_score,rotten\_tomatoes) in (select imdb\_score,rotten\_tomatoes from reviews where imdb\_score>8 and rotten\_tomatoes>=75));

Text

Description automatically generated

Using Joins to joining the two or more tables from above database

To get desired results:

1. From our film database if we wanted to know which actor has won the award

“Best actor award” from which movie, we can simply use joins…

Ans = syntax :

select films.title,film\_crew.actor from films INNER JOIN film\_crew ON films.film\_id=film\_crew.film\_id where actor in (select actor from film\_crew where film\_id in (select film\_id from awards where type like '%actor%'));

Diagram, timeline

Description automatically generated with medium confidence

1. In our film database there are films which won awards and other films which have not if we want to get list of the all films along with their awards including their IMDB score and rotten tomatoes, we can simply use join…

Ans = syntax :

select films.title,awards.awards,awards.type,reviews.Imdb\_score,reviews.Rotten\_tomatoes from films LEFT JOIN awards on films.film\_id=awards.film\_id LEFT JOIN reviews on films.film\_id=reviews.film\_id;

Text

Description automatically generated

From the above table we can conclude that awards with the “0” value are films which are nominated but does not win on other hand NULL value represents they aren’t opted for awards.

Creating views Film\_view to get combined view of two or more tables.

If we just want to know the Movie title and their cast instead of using joins again and again we can simply create view to get desired result…

Syntax:

create view film\_view as select films.film\_id,films.title,film\_crew.actor as cast,film\_crew.actress as female\_cast from films,film\_crew where films.film\_id=film\_crew.film\_id;

To see the view we created, we can use select statement,

select \* from film\_view;

Graphical user interface, text

Description automatically generated

THE END..