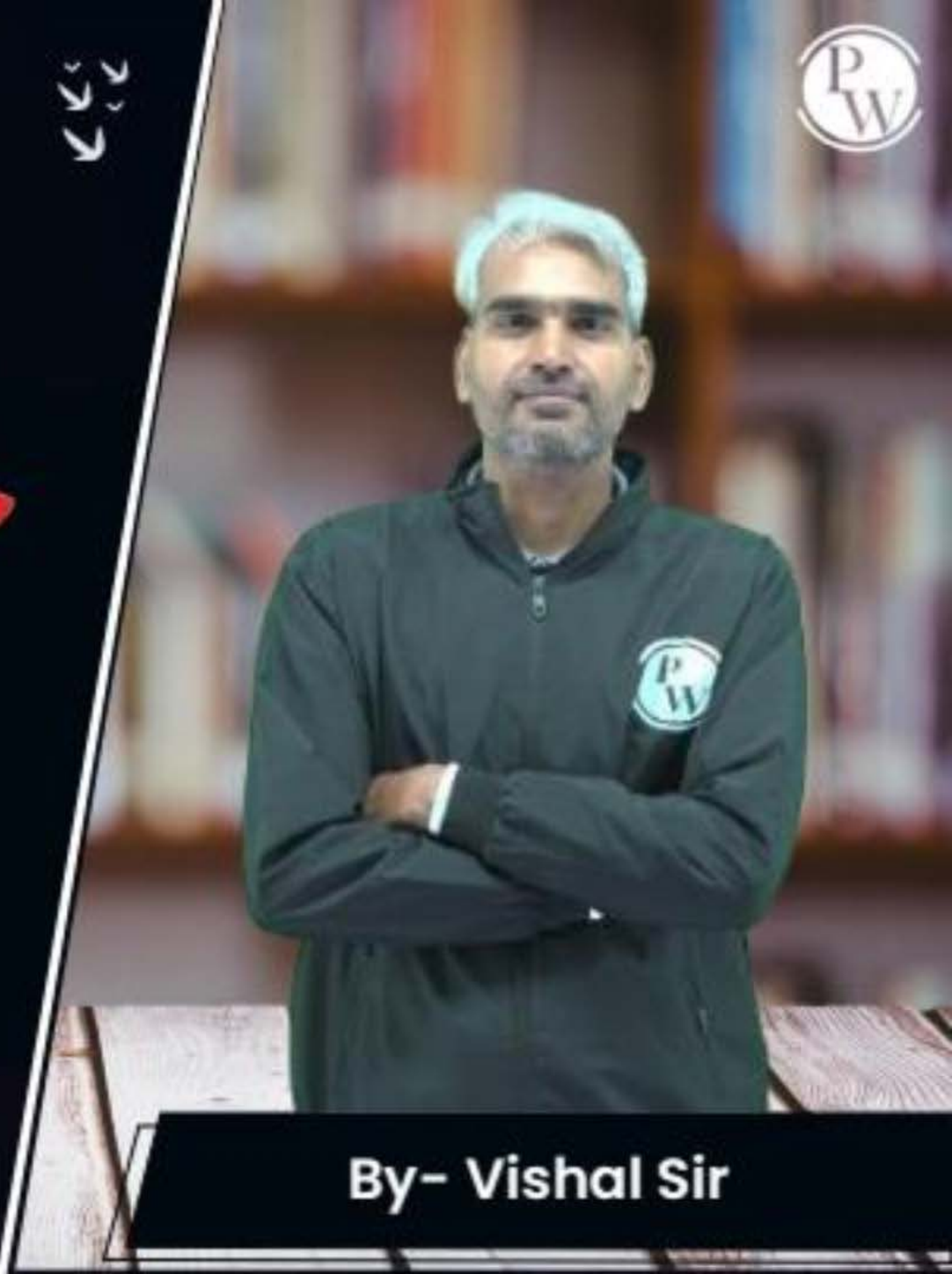
DS & AI

Database Management System

Super 1500+

Lecture No. 05



By- Vishal Sir

Recap of Previous Lecture

Topic Relational Algebra

Slide

Topics to be Covered

Topic Structure Query Language (SQL)

Slide

25 #Q. Consider the following relational schema

- actor (insta_id, name, language, age) Note: unique name of each actor.
- movie (movie_id, title, year, director_id) Note: title is unique for each movie.
- acts_in (insta_id, movie_id, character_name)
- director (director_id, name, language) Note: unique name of each director.

- Retrieve details of all movies that were released in 2010. The output schema should be the same as that of the movie table.

```
Select *
From movie
Where year : 2010
```

$\equiv \sigma_{\text{year : 2010}}(\text{movie})$

26 #Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve details of all actors that are not in their thirties(i.e., age<30 or age>39). The output schema should be the same as that of the actor table.

```
Select * From actor
    Where age < 30 OR age > 39
```

```
Select *
    From actor A
Where
A.insta_id NOT IN ( Select insta_id
                    From actor
                    Where age between 30 AND 39 )
```

27 #Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve the names of all directors.

```
Select  name
    from  director
```

29 #Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve the names of all "Telugu" language directors.

```
Select name
    From director
        Where ( language = 'Telugu' )
```

29 #Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve the name of each actor together with the titles of the movie he/she has performed in.

```
Select A.name, M.title
    From actor As A, movie As M, acts-in As In
    Where ( A.insta-id = In.insta-id AND In.movie-id = M.movie-id )
```

30 #Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve the names of all actors that have played the character of "Ravan".

```
Select A.name
    From actor As A, acts_in As In
    Where (A.insta_id = In.insta_id AND In.Character_name = 'Ravan')
```

30 #Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve the names of all actors that have played the character of "Ravan".

```
Select A.name
    From actor As A
    Where EXISTS ( Select *
                   From acts_in As In
                   Where (A.insta_id = In.insta_id AND In.Character_name
                          = 'Ravan' )
```

31 #Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve the names of all actors that have played the character of "Ravan", together with the year the corresponding movies were released.

```
Select A.name, M.year
    From actor As A, movie As M, acts_in As In
    Where (A.insta_id = In.insta_id AND In.Character_name = 'Ravan'
        AND M.movie_id = In.movie_id
```

32 #Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve all actors that acted in movie with title "Bahubali". The output schema should be the same as that of the actor table.

= Bahubali

```
Select *
    from actor A
        Where EXISTS ( Select *
                        from acts_in As In , movie As M
                        Where (A.insta_id = In.insta_id AND In.movie_id = M.movie_id
                        AND M.title = 'Bahubali' ) )
```

H.W.

33

#Q Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Find out the names of all actors that have performed in a movie directed by "Anurag Kashyap".

34

#Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve the titles of all movies in which Amitabh and Jaya have co-acted.

```
Select M1.title
    From actor A1, acts-in In1, movie M1
        Where ( A1.insta-id = In1.insta-id
                    AND
                In1.movie-id = M1.movie-id
                    AND
                A1.name = 'Amitabh' )
```

Intersect

```
Select M2.title
    From actor A2, acts-in In2, movie M2
        Where ( A2.insta-id = In2.insta-id
                    AND
                In2.movie-id = M2.movie-id
                    AND
                A2.name = 'Jaya' )
```

34

#Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve the titles of all movies in which Amitabh and Jaya have co-acted.

```
WITH Jaya(title-jaya , Mid-jaya ) AS (Select M.title, M.movie-id
                                      From actor A, acts-in In, movie M
                                      Where (A.insta_id = In.insta_id AND
                                             In.movie_id = M.movie_id AND
                                             A.name = 'Jaya'))

Select Jaya.title_jaya
From actor A1, act_in In1, Jaya
Where (A1.insta_id = In1.insta_id
                AND
       In1.movie_id = Jaya.Mid_jaya
                AND
       A1.name = 'Amitabh' )
```

H.W.

35 #Q. Consider the following relational schema

actor (insta_id, name, language, age) Note: unique name of each actor.

movie (movie_id, title, year, director_id) Note: title is unique for each movie.

acts_in (insta_id, movie_id, character_name)

director (director_id, name, language) Note: unique name of each director.

Retrieve names of all actors with their age more than the average age of all the actors who performed in movie "Bahubali"

2 mins Summary

Topic Structure Query Language (SQL)

Slide

THANK - YOU