**Assignment 2**

By Harsh Sharma

Part 3

**Part III. View**

**1.** Create a view named with **YearlyMovieRentals**) to facilitate this requirement. The view should list out the following columns: movie\_title, movie\_genre, rental\_fee (sourced from detail\_fee from the appropriate table), detail\_duedate, detail\_returndate, mem\_fname, and lname.

Solution:

CREATE VIEW YearlyMovieRentals AS

SELECT

MOVIE.MOVIE\_TITLE AS movie\_title,

MOVIE.MOVIE\_GENRE AS movie\_genre,

PRICE.PRICE\_RENTFEE AS rental\_fee,

DETAILRENTAL.DETAIL\_DUEDATE AS detail\_duedate,

DETAILRENTAL.DETAIL\_RETURNDATE AS detail\_returndate,

MEMBERSHIP.MEM\_FNAME AS mem\_fname,

MEMBERSHIP.MEM\_LNAME AS mem\_lname

FROM

MOVIE

JOIN

DETAILRENTAL ON MOVIE.MOVIE\_NUM = DETAILRENTAL.VID\_NUM

JOIN

RENTAL ON DETAILRENTAL.RENT\_NUM = RENTAL.RENT\_NUM

JOIN

MEMBERSHIP ON RENTAL.MEM\_NUM = MEMBERSHIP.MEM\_NUM

JOIN

PRICE ON MOVIE.PRICE\_CODE = PRICE.PRICE\_CODE

WHERE

EXTRACT(YEAR FROM RENTAL.RENT\_DATE) = EXTRACT(YEAR FROM SYSDATE);

Output:

A black background with white text

Description automatically generated

2. List all movies rented by members with the last name 'Knight' ('KNIGHT') this year from the

YearlyMovieRentals view.

Query:

SELECT \*

FROM YearlyMovieRentals

WHERE UPPER(mem\_lname) = 'KNIGHT';

Result:

A black background with white text

Description automatically generated

**3**. Which members have rented movies of the genre 'Action' ('ACTION') this year? Display the distinct first and last names of these members from the YearlyMovieRentals view.

Query:

SELECT DISTINCT mem\_fname, mem\_lname

FROM YearlyMovieRentals

WHERE UPPER(movie\_genre) = 'ACTION';

Result:

