Homework 2

Due: 09/12/2024 by 11:59 PM

For questions 1-4, please use ***dvdrental*** database. For questions 5 & 6, use ***northwind*** database. Write the queries down and answer the questions. The submission should contain both the query as well as the answer if required.

# Dvdrental Database

1. Write a SQL query to get the highest, lowest, total, and average replacement cost of all the films whose release year is ‘2006’ in the ***“film”*** table

## SELECT

MAX(replacement\_cost), MIN(replacement\_cost), SUM(replacement\_cost), AVG(replacement\_cost)

FROM film

WHERE release\_year = 2006;

*Explanation: I used aggregation functions (MAX, MIN, SUM, AVG) to calculate the highest, lowest, total, and average replacement cost of films from the film table that were released in the year 2006. The WHERE clause filters the films released in 2006.*

1. Write a SQL query to get the average rental rate and number of films that have a rating of ‘NC-17’ in the ***“film”*** table.

## SELECT

AVG(rental\_rate), COUNT(film\_id)

FROM film

WHERE rating = 'NC-17';

*Explanation: I used the AVG function to calculate the average rental rate and the COUNT function to count the number of films with the rating 'NC-17'. The WHERE clause filters the records for films with the 'NC-17' rating*

1. Write a SQL query to get the “title”, “length”, and “rating” of all the films from the

***“film”*** table with a “rental\_rate” greater than or equal to $2.99 and less than or equal to

$4.99, and the “rental\_duration” is 5. Display the data with the highest film length at the top.

## SELECT

title, length, rating FROM film

WHERE rental\_rate BETWEEN 2.99 AND 4.99 AND

rental\_duration = 5 ORDER BY length DESC;

***Explanation:*** *I filtered films with a rental rate between $2.99 and $4.99 and a rental duration of 5 using the BETWEEN operator. I then ordered the results by film length in descending order to display the longest films at the top.*

1. Write a SQL query to get the average replacement\_cost, and the number of films from the

***“film”*** table that have the “rating” of ‘PG-13’ and “rental\_rate” of $2.99 or more.

## SELECT

AVG(replacement\_cost), COUNT(\*)

FROM film

WHERE rating = 'PG-13' AND rental\_rate >= 2.99;

***Explanation:*** *I used the AVG function to calculate the average replacement cost and COUNT(\*) to count the number of films with a 'PG-13' rating and a rental rate greater than or equal to $2.99. The WHERE clause filters the relevant records.*

# Northwind Database

1. Write a SQL query to get “customer\_id”, “contact\_name” and “country” for the monthly lucky giveaways from the ***“customers”*** table. The customers having names starting with the letter P and countries having the second letter ‘e’ have been selected for this month’s giveaways.

## SELECT

customer\_id,

contact\_name, country

FROM customers

WHERE contact\_name ILIKE 'P%' AND country LIKE '\_e%';

***Explanation:*** *I filtered customers whose names start with 'P' using the ILIKE operator (for case-insensitive matching). I also filtered countries where the second letter is 'e' by using the LIKE '\_e%' pattern. This retrieves customer details for the monthly lucky giveaways.*

1. Write a query to find all products whose names contain the word “choco” in the

***“products”*** table.

## SELECT \*

FROM products

WHERE product\_name ILIKE '%choco%';

***Explanation:*** *I used the ILIKE operator with a wildcard (%) to search for products whose names contain the word "choco", ignoring case. The % symbols allow for a match to any characters before or after "choco" in the product name.*