Week 6 Submission Tasks

Submission Process

Download **DAD_task_submission_template.docx** from the LMS.

Paste the required screen captures from the tasks below into this file.

Submit the (.docx) file into the appropriate weekly task on **the LMS**.

Setup

- Download the file named **movie_rating_colour.sql** from the LMS.
- Edit the script.
- Perform a find and replace. Change all occurrences of movieXXXX to movie9999 ratingXXXX to rating9999 colourtypeXXXX to colourtype9999 where 9999 is the last 4 digits of your student ID.
- **Save the changes**.
- Go to your Azure Database Query Editor.
- Using the Query Editor 'Open Query' tab, open and execute the script that you modified earlier named movie_rating_colour.sql.

Task 1.

Write a single SQL statement that:

- lists all columns of movies that have one of these movie no values: 70, 114, 1858, 286217.
- Use OR operators in your solution.
- The list must be in Ascending movie no sequence.

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position in **DAD_submission_template.docx**

Task 2.

Write a single SQL statement that:

- lists all columns of movies that have one of these movie no values: 137, 10764, 65754, 211672.
- Do NOT use the OR operators in your solution. Use the IN operator.
- The list must be in Ascending movie no sequence.

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position in **DAD_submission_template.docx**

Task 3.

Write a single SQL statement that:

- lists all columns of movies that have a title that begins with the word 'the'.
- Make the query work for any combination of upper or lowercase characters.
- The list must be in Ascending Title sequence.

Screen Capture the SQL text box plus the first 6 rows of the result set

Paste the screen captures in the appropriate position.

Task 4.

Write a single SQL statement that lists all columns of movies that match all the following criteria:

- Rating is "MA"
- Length of the film is less than or equal to 100 minutes
- The list must be in Ascending movie no sequence.

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position.

Task 5.

Write a single SQL statement that:

- lists all columns of movies that match either of the following criteria:
- Rating 'PG' and the tmdb score is greater than 7.6
- Rating is 'M' and the tmdb score is greater than 7.8
- The list must be in Ascending movie no sequence.

CHECK YOUR RESULTS and ensure that all movies in the result meet the criteria above.

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position.

Task 6.

Same as task 5, except:

- ensure that movies listed have a runtime greater than 110 minutes.

CHECK YOUR RESULTS and ensure that all movies in the result meet the criteria above.

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position.

Task 7.

Write a **single** SQL statement that:

- lists all columns of movies that match **all** of the following criteria:
- Release Year is one of the following: 2015 or 2014
- Run Time > 130 The list must be in Ascending movie no sequence.

CHECK YOUR RESULTS and ensure that all movies in the result meet the criteria above.

Screen Capture the SQL text box plus the all rows of the result set.

Paste the screen captures in the appropriate position.

Task 8.

Write a **single** SQL statement that does the following:

For each row in the MOVIE table:

- display the title,
- release year,
- rating code,
- the matching short description from the RATING table,
- The list must be in Ascending movie no sequence.

This will require you to Join two tables

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position

Task 9.

Write a **single** SQL statement that does the following:

For each row in the MOVIE table:

- display the title,
- release year,
- rating code
- matching long description from the RATING table.
- Only do this if the release year is 2015 or greater.
- The list must be in Ascending movie no sequence.

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position.

Task 10.

Write a single SQL statement that does the following:

For each row in the MOVIE table display:

- the title,
- release year,
- rating code ,
- matching long description from the RATING table
- the colour name from the COLOURTYPE table.
- Only display rows that have runtime of less than 90 minutes
- The list must be in Ascending movie no sequence.

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position.

Task 11.

Write a single SQL statement which displays the following columns:

- Display the Movie No,
- Title,
- Rating
- Short Description,
- tmdb score

for movies that meet **any** of these criteria:

- Rating code of MA plus a runtime between 100-102 (inclusive)
- Rating code of G plus a runtime 90 minutes or less
- Rating code of PG plus a runtime between 120-125 (inclusive)
- Rating code of M plus a runtime 180 minutes or more

AND ALSO:

- has a tmdb_score between 6.5 to 7.5 (inclusive)

Finally:

- The list must be in Ascending movie no sequence.

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position.

Task 12.

Write a single SQL statement which displays the following columns

- Display the Movie No,
- Title, Run time,
- tmdb score,
- Rating Long Description

for movies that meet **any** of these criteria:

- The Rating Long Description contains the word 'recommended' (any case)
- The tmdb id contains the text '44'
- The title of the movie contains the letter 't' (any case)

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position.

Task 13.

Write a single SQL statement that:

- displays all the tmdb scores in the movie table.
- The list must not repeat any value.
- The list must be in Ascending tmdb score sequence.

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position.

Task 14.

Write a single SQL statement that:

- displays all columns from movies in the movie table.
- Only display movies where the title does not contain any of these letters: A E I (any case)
- The list must be in Ascending title sequence.

Screen Capture the SQL text box plus the all rows of the result set

Paste the screen captures in the appropriate position.

Task 15 below

Task 15.

a) Write and Execute a query to:

Update the tmdb_votes value to 0 for all rows.

b) Write an execute a query to:

Update the tmdb_votes to 1 where rows meet **either** of the following criteria:

- The run time is 180 or greater and the rating code is M
- The run time is 160 or greater and the rating code is MA

Screen Capture the SQL text box plus the output message after running the update.
Paste the screen captures in the appropriate position.

c) Write a single SQL statement that displays:

- the movie no,
- title,
- release year,
- run time
- rating code
- for all movies that have tmdb-votes equal to the value 1
- The list must be in Ascending title sequence.

Screen Capture the SQL text box plus the all rows of the result set
Paste the screen captures in the appropriate position.