Parul University

Name: Prerak Doshi
Email: prerak12102004@gmail.com
Roll no: 25UG033422
Phone: 8849921118
Branch: Parul University
Department: CSE10_Batch 1
Batch: 2028
Degree: B.Tech - CSE

Scan to verify results

PIET_Oracle DBMS_Course

PIET_Oracle DBMS_Session 6_PAH_Updated

Attempt : 2
Total Mark : 40
Marks Obtained : 40

Section 1 : COD

1. Problem Statement

Jordan is a data analyst at a fitness tracking company. They need to analyze workout data and instructor information to provide insights. They have been asked to use the MIN and MAX functions to extract specific information from the WORKOUT_SESSIONS and INSTRUCTOR_UPDATES tables.

Table Details:

Sample Input Records:

Table: INSTRUCTORS

Table: WORKOUTS

Table: WORKOUT_SESSIONS

Table: INSTRUCTOR_UPDATES

Tasks to perform:

Write a query to find the earliest timestamp (MIN_SESSION_TIMESTAMP) from the SESSION_TIMESTAMP column in the WORKOUT_SESSIONS table. The query should use the alias MIN_SESSION_TIMESTAMP to label the result.Write a query to find the lexicographically largest (MAX_UPDATE_DESCRIPTION) description from the UPDATE_DESCRIPTION column in the INSTRUCTOR_UPDATES table. The query should use the alias MAX_UPDATE_DESCRIPTION to label the result. Write a query to find the earliest timestamp (MIN_SESSION_TIMESTAMP) of workout sessions conducted by the instructor with INSTRUCTOR_ID = 'I3'. The result should use the alias MIN_SESSION_TIMESTAMP to label the output.Write a query to find the most recent update timestamp (MAX_UPDATE_TIMESTAMP) for the instructor with INSTRUCTOR_ID = 'I3'. The result should use the alias MAX_UPDATE_TIMESTAMP to label the output.Write a query to find the lexicographically smallest update description (MIN_UPDATE_DESCRIPTION) from the UPDATE_DESCRIPTION column in the INSTRUCTOR_UPDATES table. The result should use the alias MIN_UPDATE_DESCRIPTION to label the output.

Note: The user must write only the queries to select the required data with the relevant aggregate functions.

Answer

oracle.sql

```
SELECT
   MIN(SESSION_TIMESTAMP) AS MIN_SESSION_TIMESTAMP
FROM
   WORKOUT_SESSIONS;


SELECT
   MAX(UPDATE_DESCRIPTION) AS MAX_UPDATE_DESCRIPTION
FROM
   INSTRUCTOR_UPDATES;


SELECT
   MIN(SESSION_TIMESTAMP) AS MIN_SESSION_TIMESTAMP
FROM
   WORKOUT_SESSIONS
WHERE
   INSTRUCTOR_ID = 'I3';


SELECT
   MAX(UPDATE_TIMESTAMP) AS MAX_UPDATE_TIMESTAMP
FROM
   INSTRUCTOR_UPDATES
WHERE
   INSTRUCTOR_ID = 'I3';


SELECT
   MIN(UPDATE_DESCRIPTION) AS MIN_UPDATE_DESCRIPTION
FROM
   INSTRUCTOR_UPDATES;
```

Status : Correct ***Marks : 10/10***

2. Problem Statement

George, a data analyst at a sports analytics company, needs to analyze player performance and game outcomes to provide insights into the best-

performing teams and players. They need to use the MIN and MAX aggregate functions to extract specific information from the PLAYERS, GAMES, and PERFORMANCE tables.

Table Details:

Sample Input Records:

Table Name: PLAYERS

Table Name: GAMES

Table Name: PERFORMANCE

Tasks to perform:

Write a query to find both the minimum (MIN_SCORE) and maximum (MAX_SCORE) scores from the SCORE column in the PERFORMANCE table. The result should use the aliases MIN_SCORE and MAX_SCORE to label the output.Write a query to find both the earliest (EARLIEST_GAME_DATE) and latest (LATEST_GAME_DATE) game dates from the GAME_DATE column in the GAMES table. The result should use the aliases EARLIEST_GAME_DATE and LATEST_GAME_DATE to label the output.Write a query to find the lexicographically smallest (MIN_GAME_NAME) and largest (MAX_GAME_NAME) game names from the GAME_NAME column in the GAMES table. The result should use the aliases MIN_GAME_NAME and MAX_GAME_NAME to label the output.Write a query to find both the minimum (MIN_PLAY_TIME) and maximum (MAX_PLAY_TIME) play times from the PLAY_TIME column in the PERFORMANCE table. The result should use the aliases MIN_PLAY_TIME and MAX_PLAY_TIME to label the output.Write a query to find the lexicographically smallest (MIN_PLAYER_NAME) and largest

(MAX_PLAYER_NAME) player names from the PLAYER_NAME column in the PLAYERS table. The result should use the aliases MIN_PLAYER_NAME and MAX_PLAYER_NAME to label the output.Note:

The user must write only the queries to select the required data with the relevant aggregate functions.

Answer

oracle.sql

```
SELECT
   MIN(SCORE) AS MIN_SCORE,
   MAX(SCORE) AS MAX_SCORE
FROM
   PERFORMANCE;


SELECT
   MIN(GAME_DATE) AS EARLIEST_GAME_DATE,
   MAX(GAME_DATE) AS LATEST_GAME_DATE
FROM
   GAMES;

SELECT
   MIN(GAME_NAME) AS MIN_GAME_NAME,
   MAX(GAME_NAME) AS MAX_GAME_NAME
FROM
   GAMES;

SELECT
   MIN(PLAY_TIME) AS MIN_PLAY_TIME,
   MAX(PLAY_TIME) AS MAX_PLAY_TIME
FROM
   PERFORMANCE;

SELECT
   MIN(PLAYER_NAME) AS MIN_PLAYER_NAME,
   MAX(PLAYER_NAME) AS MAX_PLAYER_NAME
FROM
   PLAYERS;
```

Status : Correct ***Marks : 10/10***

3. Problem Statement

Alex is a data analyst at a retail chain and is tasked with analyzing sales performance and inventory management to provide insights into product profitability and sales trends. They need to use the aggregate functions SUM, and AVG to extract and analyze specific information from the `PRODUCTS`, `SALES`, and `INVENTORY` tables.

Sample Input Records:

Table Name: PRODUCTS

Table Name: PRODUCTS

Table Name: INVENTORY

Tasks to Perform:

Write a query to compute the average price (AVERAGE_PRICE) of products for each category in the PRODUCTS table. The result should use the alias AVERAGE_PRICE to label the computed average price.Write a query to compute the sum of the TOTAL_AMOUNT column across all records in the SALES table. The result should use the alias TOTAL_SALES_AMOUNT to label the computed total sales amount.Write a query to compute the sum of the STOCK column across all records in the INVENTORY table. The result should use the alias TOTAL_STOCK to label the computed total stock.Write a query to compute the average of the TOTAL_AMOUNT column across all records in the SALES table. The result should use the alias AVERAGE_TOTAL_SALES_AMOUNT to label the computed average amount.Write a query to compute the sum of the QUANTITY column across all records in the SALES table. The result should use the alias

TOTAL_QUANTITY_SOLD to label the computed total quantity.

Answer

oracle.sql

```
SELECT CATEGORY, AVG(PRICE) AS AVERAGE_PRICE FROM PRODUCTS
GROUP BY CATEGORY;

SELECT SUM(TOTAL_AMOUNT) AS TOTAL_SALES_AMOUNT FROM SALES;

SELECT SUM(STOCK) AS TOTAL_STOCK FROM INVENTORY;

SELECT AVG(TOTAL_AMOUNT) AS AVERAGE_TOTAL_SALES_AMOUNT FROM
SALES;

SELECT SUM(QUANTITY) AS TOTAL_QUANTITY_SOLD FROM SALES;
```

Status : Correct ***Marks : 10/10***

4. Problem Statement:

Taylor is a data analyst at a film streaming service. They need to analyze film ratings and release dates to provide insights into the latest trends. They have been asked to use the MIN and MAX aggregate functions to extract specific information from the MOVIES, REVIEWS, and AWARDS tables.

Table Details:

Sample input records:

Table Name: MOVIES

Table Name: REVIEWS

Table Name: AWARDS

Tasks to Perform:

Write a query to find the lowest and highest ratings from the RATING column in the REVIEWS table. The results should be labeled as MIN_RATING and MAX_RATING, respectively.Write a query to find the earliest and latest release dates from the RELEASE_DATE column in the MOVIES table. The results should be labeled as EARLIEST_RELEASE_DATE and LATEST_RELEASE_DATE, respectively.Write a query to find the earliest and latest dates from the AWARD_DATE column in the AWARDS table. The results should be labeled as EARLIEST_AWARD_DATE and LATEST_AWARD_DATE, respectively.Write a query to find the earliest and latest dates from the REVIEW_DATE column in the REVIEWS table. The results should be labeled as EARLIEST_REVIEW_DATE and LATEST_REVIEW_DATE, respectively.Write a query to find the lexicographically first and last movie names from the MOVIE_NAME column in the MOVIES table. The results should be labeled as EARLIEST_MOVIE_NAME and LATEST_MOVIE_NAME, respectively.

Note: The user must write only the queries to select the required data using the MIN and MAX aggregate functions.

Answer

oracle.sql

```
SELECT
  MIN(RATING) AS MIN_RATING,
  MAX(RATING) AS MAX_RATING
FROM
  REVIEWS;

SELECT
  MIN(RELEASE_DATE) AS EARLIEST_RELEASE_DATE,
  MAX(RELEASE_DATE) AS LATEST_RELEASE_DATE
FROM
  MOVIES;

SELECT
  MIN(AWARD_DATE) AS EARLIEST_AWARD_DATE,
  MAX(AWARD_DATE) AS LATEST_AWARD_DATE
FROM
```

```
    AWARDS;

SELECT
    MIN(REVIEW_DATE) AS EARLIEST_REVIEW_DATE,
    MAX(REVIEW_DATE) AS LATEST_REVIEW_DATE
FROM
    REVIEWS;

SELECT
    MIN(MOVIE_NAME) AS EARLIEST_MOVIE_NAME,
    MAX(MOVIE_NAME) AS LATEST_MOVIE_NAME
FROM
    MOVIES;
```

Status : Correct ***Marks : 10/10***