Week04 – SQL - QUESTIONS

These questions and queries cover a wide range of scenarios commonly encountered in a MKTIME database, utilising joins, subqueries, and aggregate functions to extract meaningful output from the database.

1. List all products.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT \* FROM products; | How many records you expect to display:  5 for all watches available |  |

2. Find the total sales amount for each product.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT p.product\_name, SUM(oc.quantity \* oc.price) AS total\_sales\_amount FROM order\_contents oc JOIN products p ON oc.product\_id = p.product\_id GROUP BY p.product\_name; | List the 5 products total sales |  |

3. List all users who made purchase on 3rd May 2023.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT u.user\_id, u.first\_name, u.last\_name, u.email  FROM users u  JOIN orders o ON u.user\_id = o.user\_id  WHERE DATE(o.order\_date) = '2023-05-03'; | Probably no sales on that specific day |  |

4. Find the top 5 costing items.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT product\_name, product\_price  FROM products  ORDER BY product\_price DESC  LIMIT 5; | 5 items order by descending |  |

5. List all items and who purchased those items.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT p.product\_name, u.first\_name, u.last\_name  FROM order\_contents oc  JOIN products p ON oc.product\_id = p.product\_id  JOIN orders o ON oc.order\_id = o.order\_id  JOIN users u ON o.user\_id = u.user\_id; | 20 sold articles |  |

6. Find the total order value for each user.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT u.user\_id, u.first\_name, u.last\_name, SUM(o.total\_amount) AS total\_order\_value  FROM users u  JOIN orders o ON u.user\_id = o.user\_id  GROUP BY u.user\_id, u.first\_name, u.last\_name; | Many user can purchase many items, but needs to show the 16 current costumers |  |

7. List all products with their corresponding orders.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT p.product\_name, o.order\_id  FROM order\_contents oc  JOIN products p ON oc.product\_id = p.product\_id  JOIN orders o ON oc.order\_id = o.order\_id; | 20 orders |  |

8. Find the customer who spent the most in total.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) u.user\_id, u.first\_name, u.last\_name, [SUM](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html%23function_sum)(o.total\_amount) AS total\_spent FROM users u JOIN orders o ON u.user\_id = o.user\_id GROUP BY u.user\_id, u.first\_name, u.last\_name ORDER BY total\_spent DESC LIMIT 1; | One costumer |  |

9. Find the top 3 categories with the highest total sales.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT c.category\_name, SUM(oc.quantity \* oc.price) AS total\_sales  FROM order\_contents oc  JOIN products p ON oc.product\_id = p.product\_id  JOIN product\_categories pc ON p.product\_id = pc.product\_id  JOIN categories c ON pc.category\_id = c.category\_id  GROUP BY c.category\_name  ORDER BY total\_sales DESC  LIMIT 3; | My MKT only have two categories so will expect them both |  |

10. List all orders made by a specific customer (e.g., John Doe).

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT u.first\_name, u.last\_name, o.order\_id, o.total\_amount, o.order\_date  FROM users u  JOIN orders o ON u.user\_id = o.user\_id  WHERE u.first\_name = 'John' AND u.last\_name = 'Doe'; | All orders under the name of John Doe, less than 20 entries. |  |

11. Find the number of orders placed by user\_id = 2.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT u.first\_name, u.last\_name, COUNT(o.order\_id) AS number\_of\_orders  FROM orders o  JOIN users u ON o.user\_id = u.user\_id  WHERE o.user\_id = 2; | User\_id = 2 = John Doe, therefore orders = 2 |  |

12. List all items with their respective quantities sold.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT p.product\_name, SUM(oc.quantity) AS total\_quantity\_sold  FROM order\_contents oc  JOIN products p ON oc.product\_id = p.product\_id  GROUP BY p.product\_name; | 5 items |  |

13. List all items with their respective quantities sold, including total sales.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT p.product\_name, SUM(oc.quantity) AS total\_quantity\_sold  FROM order\_contents oc  JOIN products p ON oc.product\_id = p.product\_id  GROUP BY p.product\_name; | 5 items (Quantities should be higher after Cypress test) |  |

14. Find the total sales made by each user.

|  |  |  |
| --- | --- | --- |
| Code | Expected Answer | Actual Answer |
| SELECT u.user\_id, u.first\_name, u.last\_name, SUM(oc.quantity \* oc.price) AS total\_sales  FROM users u  JOIN orders o ON u.user\_id = o.user\_id  JOIN order\_contents oc ON o.order\_id = oc.order\_id  GROUP BY u.user\_id, u.first\_name, u.last\_name; | All 16 users with the total sales for each one |  |