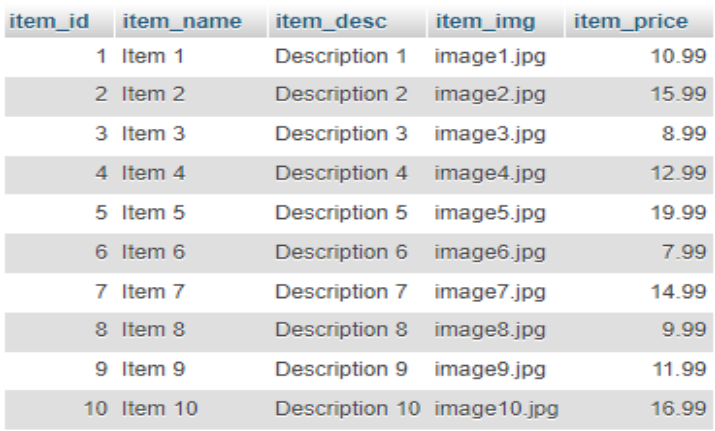
Week04 – SQL - QUESTIONS

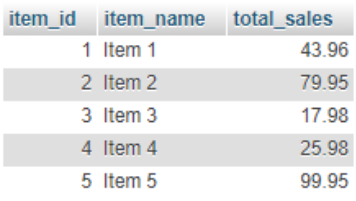
These questions and queries cover a wide range of scenarios commonly encountered in a MKTIME database, utilising joins, subqueries, and aggregate functions to extract meaningful output from the database.

1. List all products.

Code	Expected Answer	Actual Answer
SELECT * FROM items;	10 products	

item_id	item_name	item_desc	item_img	item_price
1	Item 1	Description 1	image1.jpg	10.99
2	Item 2	Description 2	image2.jpg	15.99
3	Item 3	Description 3	image3.jpg	8.99
4	Item 4	Description 4	image4.jpg	12.99
5	Item 5	Description 5	image5.jpg	19.99
6	Item 6	Description 6	image6.jpg	7.99
7	Item 7	Description 7	image7.jpg	14.99
8	Item 8	Description 8	image8.jpg	9.99
9	Item 9	Description 9	image9.jpg	11.99
10	Item 10	Description 10	image10.jpg	16.99

2. Find the total sales amount for each product.

Code	Expected Answer	Actual Answer
SELECT items.item_id, items.item_name, SUM(orders.total) AS total_sales FROM items JOIN orders ON items.item_id = orders.item_id GROUP BY items.item_id, items.item_name;	5 sales as only 5 products have been sold	

item_id	item_name	total_sales
1	Item 1	43.96
2	Item 2	79.95
3	Item 3	17.98
4	Item 4	25.98
5	Item 5	99.95

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

Code	Expected Answer	Actual Answer
SELECT DISTINCT users.user_id, users.firstname, users.lastname FROM users JOIN orders ON users.user_id = orders.user_id WHERE DATE(orders.order_date) = '2023-05-03';	1 purchase with payment_id 1003	user_id firstname lastname 2 Jane Smith

4. Find the top 5 costing items.

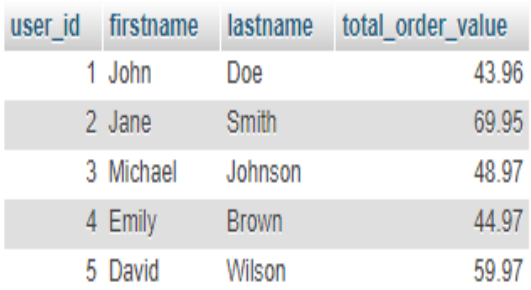
Code	Expected Answer	Actual Answer
SELECT * FROM items ORDER BY item_price DESC LIMIT 5;	Items 5 – 10 – 2 – 7 – 4	item_id item_name item_desc item_img item_price ▼ 1 5 Item 5 Description 5 image5.jpg 19.99 10 Item 10 Description 10 image10.jpg 16.99 2 Item 2 Description 2 image2.jpg 15.99 7 Item 7 Description 7 image7.jpg 14.99 4 Item 4 Description 4 image4.jpg 12.99

5. List all items and who purchased those items.

Code	Expected Answer	Actual Answer
SELECT items.item_id, items.item_name, users.user_id, users.firstname, users.lastname FROM items JOIN orders ON items.item_id = orders.item_id JOIN users ON orders.user_id = users.user_id;	Item 1 (users 1-2) Item 2 (users 2-4) Item 3 (users 1-3) Item 4 (users 1-4) Item 5 (users 3-5)	

item_id	item_name	user_id	firstname	lastname
1	Item 1	1	John	Doe
1	Item 1	2	Jane	Smith
2	Item 2	2	Jane	Smith
2	Item 2	4	Emily	Brown
3	Item 3	1	John	Doe
3	Item 3	3	Michael	Johnson
4	Item 4	1	John	Doe
4	Item 4	4	Emily	Brown
5	Item 5	3	Michael	Johnson
5	Item 5	5	David	Wilson

6. Find the total order value for each user.

Code	Expected Answer	Actual Answer
SELECT users.user_id, users.firstname, users.lastname, SUM(orders.total) AS total_order_value FROM users JOIN orders ON users.user_id = orders.user_id GROUP BY users.user_id, users.firstname, users.lastname;	User 1 - 3 items = 43.96 User 2 – 2 items = 69.95 User 3 – 2 items = 48.97 User 4 – 2 items = 44.97 User 5 – 1 item = 59.97	

user_id	firstname	lastname	total_order_value
1	John	Doe	43.96
2	Jane	Smith	69.95
3	Michael	Johnson	48.97
4	Emily	Brown	44.97
5	David	Wilson	59.97

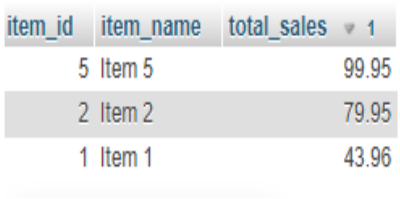
7. List all products with their corresponding orders.

Code	Expected Answer	Actual Answer
SELECT items.item_id, items.item_name, orders.order_date, orders.quantity, orders.total FROM items JOIN orders ON items.item_id = orders.item_id;	10 orders for 5 products (items 1 to 5)	item_id | item_name | order_date | quantity | total 1 | Item 1 | 2023-05-01 00:00:00 | 2 | 21.98 3 | Item 3 | 2023-05-02 00:00:00 | 1 | 8.99 4 | Item 4 | 2023-05-10 00:00:00 | 1 | 12.99 1 | Item 1 | 2023-05-06 00:00:00 | 2 | 21.98 2 | Item 2 | 2023-05-03 00:00:00 | 3 | 47.97 3 | Item 3 | 2023-05-07 00:00:00 | 1 | 8.99 5 | Item 5 | 2023-05-04 00:00:00 | 2 | 39.98 2 | Item 2 | 2023-05-09 00:00:00 | 2 | 31.98 4 | Item 4 | 2023-05-05 00:00:00 | 1 | 12.99 5 | Item 5 | 2023-05-08 00:00:00 | 3 | 59.97

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

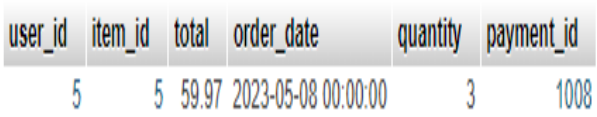
Code	Expected Answer	Actual Answer
SELECT users.user_id, users.firstname, users.lastname, SUM(orders.total) AS total_spent FROM users JOIN orders ON users.user_id = orders.user_id GROUP BY users.user_id, users.firstname, users.lastname ORDER BY total_spent DESC LIMIT 1;	User 2	user_id | firstname | lastname | total_spent 2 | Jane | Smith | 69.95

9. Find the top 3 (categories) items with the highest total sales.

Code	Expected Answer	Actual Answer
SELECT items.item_id, items.item_name, SUM(orders.total) AS total_sales FROM items LEFT JOIN orders ON items.item_id = orders.item_id GROUP BY items.item_id, items.item_name ORDER BY total_sales DESC LIMIT 3;	Item 5 Item 2 Item 1	

item_id	item_name	total_sales ▼ 1
5	Item 5	99.95
2	Item 2	79.95
1	Item 1	43.96

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

Code	Expected Answer	Actual Answer
SELECT orders.* FROM orders JOIN users ON orders.user_id = users.user_id WHERE users.firstname = 'David' AND users.lastname = 'Wilson';	1 order made by David Wilson	

user_id	item_id	total	order_date	quantity	payment_id
5	5	59.97	2023-05-08 00:00:00	3	1008

12. Find the number of orders placed by user_id = 2.

Code	Expected Answer	Actual Answer
SELECT COUNT(*) AS order_count FROM orders WHERE user_id = 2;	2 orders	order_count 2

13. List all items with their respective quantities sold.

Code	Expected Answer	Actual Answer
SELECT items.item_id, items.item_name, SUM(orders.quantity) AS total_quantity_sold FROM items JOIN orders ON items.item_id = orders.item_id GROUP BY items.item_id, items.item_name;	Item 1 = 4 Item 2 = 5 Item 3 = 2 Item 4 = 2 Item 5 = 5	item_id | item_name | total_quantity_sold 1 | Item 1 | 4 2 | Item 2 | 5 3 | Item 3 | 2 4 | Item 4 | 2 5 | Item 5 | 5

14. Find the total sales made by each user.

Code	Expected Answer	Actual Answer
SELECT users.user_id, users.firstname, users.lastname, SUM(orders.total) AS total_sales FROM users JOIN orders ON users.user_id = orders.user_id	User 1 = 43.96 User 2 = 69.95 User 3 = 48.97 User 4 = 44.97 User 5 = 59.97	user_id | firstname | lastname | total_sales 1 | John | Doe | 43.96 2 | Jane | Smith | 69.95 3 | Michael | Johnson | 48.97 4 | Emily | Brown | 44.97 5 | David | Wilson | 59.97

GROUP BY users.user_id, users.firstname, users.lastname;		