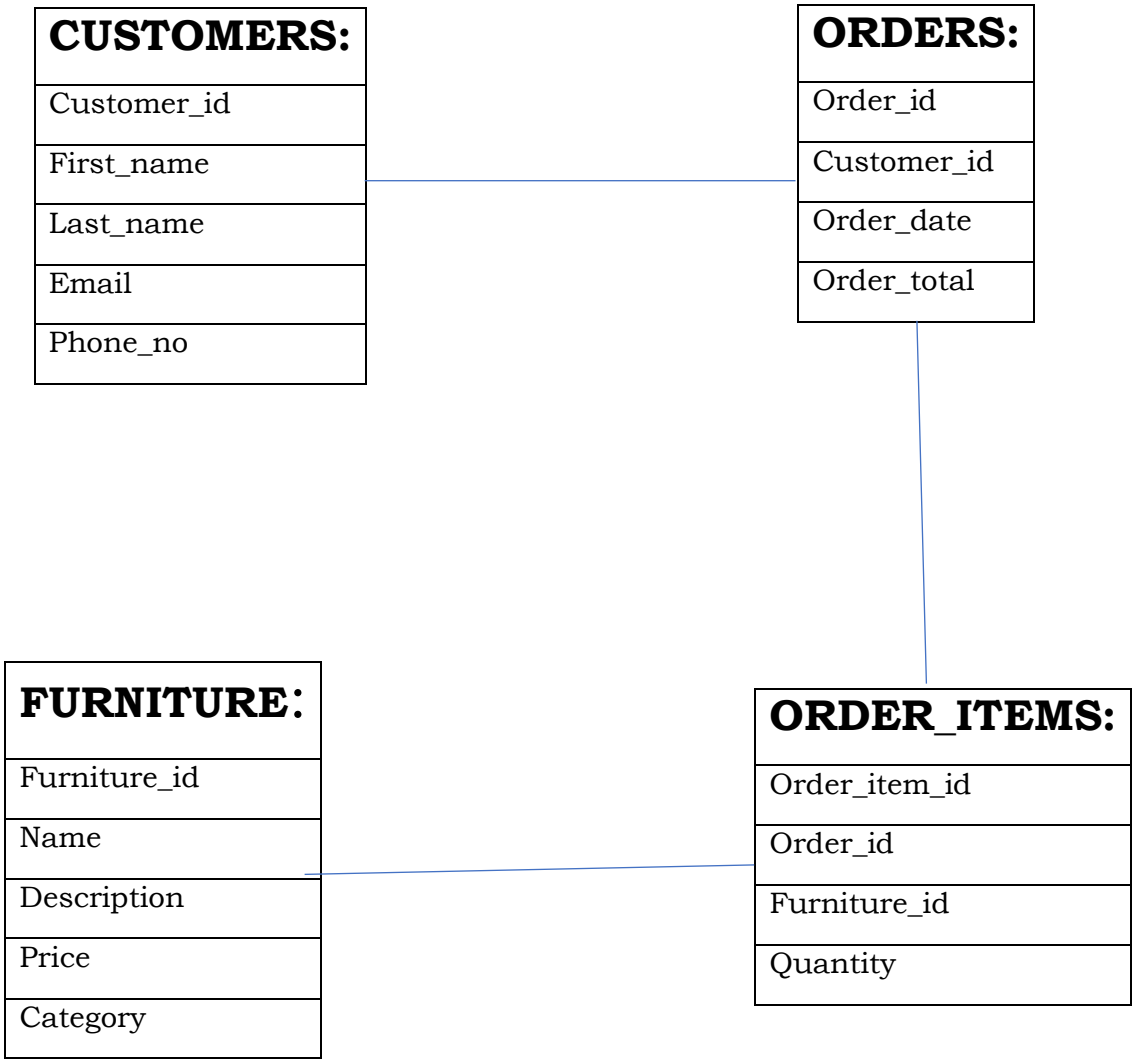
AJEENKYA
D Y PATIL UNIVERSITY
THE INNOVATION UNIVERSITY

School of Engineering

Name: Sarthak Shinde
Urn: 2022-B-17012005
Program: B. Tech ITDS
Semester: 4
Section: B
Subject: Data Analytics using SQL
Professor: Dipmala Kamdi.
Course Code: DS206E

ASSIGNMENT 1:

ER DIAGRAM:



Database Schema:-

1. Customers table:

Customer_id, First_name, Last_name, Email, Phone_number.

customer_id	First_name	Last_name	Email	Phone_no
1	John	Doe	john.doe@example.com	123-456-7890
2	Jane	Smith	jane.smith@example.com	987-654-3210
3	Alice	Johnson	alice.johnson@example.com	555-123-4567
4	Bob	Brown	bob.brown@example.com	777-555-9999
5	Emily	Davis	emily.davis@example.com	111-222-3333
6	Michael	Wilson	michael.wilson@example.com	444-888-2222
7	Sarah	Martinez	sarah.martinez@example.com	666-999-1111
8	David	Garcia	david.garcia@example.com	222-777-8888
9	Jessica	Lee	jessica.lee@example.com	333-444-5555
10	Ryan	Anderson	ryan.anderson@example.com	999-333-7777
11	Laura	Taylor	laura.taylor@example.com	888-222-5555
12	Christopher	Hernandez	christopher.hernandez@example.com	666-444-2222
13	Melissa	Lopez	melissa.lopez@example.com	777-666-3333
14	Matthew	Gonzalez	matthew.gonzalez@example.com	555-999-4444
15	Jennifer	Perez	jennifer.perez@example.com	111-777-8888

2. Orders table:

Order_id, Customer_id, Order_date, Order_total

Order_id	Customer_id	Order_date	Order_total
1	1	2024-04-01	100.50
2	2	2024-04-02	75.25
3	3	2024-04-03	200.00
4	4	2024-04-04	150.75
5	5	2024-04-05	300.00
6	6	2024-04-06	50.00
7	7	2024-04-07	125.80
8	8	2024-04-08	180.25
9	9	2024-04-09	95.50
10	10	2024-04-10	220.75
11	11	2024-04-11	75.00
12	12	2024-04-12	180.20
13	13	2024-04-13	140.00
14	14	2024-04-14	260.75
15	15	2024-04-15	320.50

3. Products table:

Furniture_id | Name | Description | Price | Category

Furniture_id	Name	Description	Price	Category
1	Sofa	Comfortable 3-seater sofa	15000.00	Living Room
2	Dining Table	Wooden dining table with 4 chairs	25000.00	Dining Room
3	Bed Frame	Queen-sized bed frame with headboard	20000.00	Bedroom
4	Coffee Table	Glass-top coffee table with metal legs	6000.00	Living Room
5	Bookshelf	Tall wooden bookshelf with adjustable shelves	10000.00	Study
6	TV Stand	Modern TV stand with storage compartments	12000.00	Living Room
7	Dresser	White dresser with drawers	15000.00	Bedroom
8	Desk	Simple writing desk with one drawer	8000.00	Study
9	Armchair	Plush armchair with fabric upholstery	12000.00	Living Room
10	Nightstand	Bedside table with two drawers	5000.00	Bedroom
11	Office Chair	Ergonomic office chair with adjustable height	10000.00	Office
12	Bar Stool	Set of 2 bar stools with cushioned seats	7000.00	Kitchen
13	Wardrobe	Large wardrobe with sliding doors	30000.00	Bedroom
14	Side Table	Wooden side table with shelf	4000.00	Living Room
15	Accent Chair	Patterned accent chair with armrests	12000.00	Living Room

4. **Order items :**

Order_item_id | Order_id | Furniture_id | Quantity

Order_item_id	Order_id	Furniture_id	Quantity
1	1	1	3
2	2	2	2
3	3	3	4
4	4	4	1
5	5	5	5
6	6	6	3
7	7	7	4
8	8	8	2
9	9	9	5
10	10	10	1
11	11	11	4
12	12	12	2
13	13	13	3
14	14	14	5
15	15	15	1

Assignment 2

Questions:-

Let's consider only one table for generating 10 questions: -

1.Find the most expensive item in each category.

2. Find the total price of all items in each category.

3. Find the average price of items in each category.

4. Find the number of items in each category.

5. Find the total number of items and the average price across all categories.

6. Retrieve the details (first name, last name, email) of customers whose phone numbers start with '555'.

7. Find the total number of customers in the database

8. List the customers whose email addresses end with '@gmail.com'.

9. Retrieve the customers sorted alphabetically by their last names

10. Calculate the average length of customer email addresses

Let's consider only two tables for generating 10 questions:

1. Retrieve the names and email addresses of customers who have placed orders
2. List the furniture items ordered along with their quantities and total order amounts
3. Find the total number of orders placed by each customer
4. Retrieve the details of customers who have not placed any orders
5. Calculate the total sales amount for each category of furniture
6. Find the top 5 customers with the highest total order amounts
7. Retrieve the furniture items that have not been ordered yet
8. Find the average order total for each customer

9. List the customers who have ordered furniture items in the "Living Room" category.
10. Retrieve the details of customers who have ordered a specific furniture item.

Let's consider only three tables for generating 10 questions: -

Questions :-

1.Find the total revenue generated from all orders.

2.Find the total number of items ordered.

3. Find the customer who placed the highest total order.

4. Find the most popular category of furniture based on the total quantity ordered.

5. Find the total revenue generated from orders placed in April 2024.

6. Find the average order total.

7. Find the top 3 highest-priced furniture items.

8.Find the customer who placed the earliest order.

9. Find the total quantity ordered for each furniture item.

10. Find the average price of furniture items ordered.

Assignment 3

Question 1: Find the most expensive item in each category.

Input:

Output:

```
mysql> SELECT Category, MAX(Price) AS Max_Price
    -> FROM Furniture
    -> GROUP BY Category;
+-------------+-----------+
| Category    | Max_Price |
+-------------+-----------+
| Living Room |  15000.00 |
| Dining Room |  25000.00 |
| Bedroom     |  30000.00 |
| Study       |  10000.00 |
| Office      |  10000.00 |
| Kitchen     |   7000.00 |
+-------------+-----------+
```

Question 2 : Find the total price of all items in each category:

input:

output:

```
mysql> SELECT Category, SUM(Price) AS Total_Price
    -> FROM Furniture
    -> GROUP BY Category;
+-------------+-------------+
| Category    | Total_Price |
+-------------+-------------+
| Living Room |    61000.00 |
| Dining Room |    25000.00 |
| Bedroom     |    70000.00 |
| Study       |    18000.00 |
| Office      |    10000.00 |
| Kitchen     |     7000.00 |
+-------------+-------------+
```

Question 3 : Find the average price of items in each category.

Input:

output:

```
mysql> SELECT Category, AVG(Price) AS Avg_Price
    -> FROM Furniture
    -> GROUP BY Category;
+-------------+--------------+
| Category    | Avg_Price    |
+-------------+--------------+
| Living Room | 10166.666667 |
| Dining Room | 25000.000000 |
| Bedroom     | 17500.000000 |
| Study       |  9000.000000 |
| Office      | 10000.000000 |
| Kitchen     |  7000.000000 |
+-------------+--------------+
```

Question 4. Find the number of items in each category.

Input:

Output:

```
mysql> SELECT Category, COUNT(*) AS Item_Count
    -> FROM Furniture
    -> GROUP BY Category;
+-------------+------------+
| Category    | Item_Count |
+-------------+------------+
| Living Room |          6 |
| Dining Room |          1 |
| Bedroom     |          4 |
| Study       |          2 |
| Office      |          1 |
| Kitchen     |          1 |
+-------------+------------+
```

Question 5. Find the total number of items and the average price across all categories.

input:

Output:

```
mysql> SELECT
    ->     COUNT(*) AS Total_Items,
    ->     AVG(Price) AS Avg_Price
    -> FROM Furniture;
+--------------+--------------+
| Total_Items | Avg_Price    |
+--------------+--------------+
|           15 | 12733.333333 |
+--------------+--------------+
```

Question 6. Retrieve the details (first name, last name, email) of customers whose phone numbers start with '555'.

input:

Output:

```
mysql> SELECT First_name, Last_name, Email
    -> FROM customers
    -> WHERE Phone_no LIKE '555%';
+-----------+----------+-----------------------------+
| First_name | Last_name | Email                      |
+-----------+----------+-----------------------------+
| Alice      | Johnson   | alice.johnson@example.com    |
| Matthew    | Gonzalez  | matthew.gonzalez@example.com |
+-----------+----------+-----------------------------+
```

Question 7. . Find the total number of customers in the database.

Input:

Output:

```
mysql> SELECT COUNT(*) AS Total_Customers
    -> FROM customers;
+-----------------+
| Total_Customers |
+-----------------+
|              15 |
+-----------------+
```

Question 8. Question: List the customers whose email addresses end with '@gmail.com'.

Input:

```
mysql> SELECT *
    -> FROM customers
    -> WHERE Email LIKE '%@gmail.com';
```

Output:

```
Empty set (0.00 sec)
```

Question 9. Retrieve the customers sorted alphabetically by their last names

Input:

```
mysql> SELECT *
    -> FROM customers
    -> ORDER BY Last_name;
```

Output:

customer_id	First_name	Last_name	Email	Phone_no
10	Ryan	Anderson	ryan.anderson@example.com	999-333-7777
4	Bob	Brown	bob.brown@example.com	777-555-9999
5	Emily	Davis	emily.davis@example.com	111-222-3333
1	John	Doe	john.doe@example.com	123-456-7890
8	David	Garcia	david.garcia@example.com	222-777-8888
14	Matthew	Gonzalez	matthew.gonzalez@example.com	555-999-4444
12	Christopher	Hernandez	christopher.hernandez@example.com	666-444-2222
3	Alice	Johnson	alice.johnson@example.com	555-123-4567
9	Jessica	Lee	jessica.lee@example.com	333-444-5555
13	Melissa	Lopez	melissa.lopez@example.com	777-666-3333
7	Sarah	Martinez	sarah.martinez@example.com	666-999-1111
15	Jennifer	Perez	jennifer.perez@example.com	111-777-8888
2	Jane	Smith	jane.smith@example.com	987-654-3210
11	Laura	Taylor	laura.taylor@example.com	888-222-5555
6	Michael	Wilson	michael.wilson@example.com	444-888-2222

Question 10 . Calculate the average length of customer email addresses **:**

Input:

```
mysql> SELECT AVG(LENGTH(Email)) AS Avg_Email_Length
    -> FROM customers;
```

Output:

Avg_Email_Length
24.7333

Assignment 4:

Question 1. Retrieve the names and email addresses of customers who have placed orders

Input:

```
mysql> SELECT c.First_name, c.Last_name, c.Email
    -> FROM customers c
    -> INNER JOIN orders o ON c.customer_id = o.Customer_id;
```

Output:

First_name	Last_name	Email
John	Doe	john.doe@example.com
Jane	Smith	jane.smith@example.com
Alice	Johnson	alice.johnson@example.com
Bob	Brown	bob.brown@example.com
Emily	Davis	emily.davis@example.com
Michael	Wilson	michael.wilson@example.com
Sarah	Martinez	sarah.martinez@example.com
David	Garcia	david.garcia@example.com
Jessica	Lee	jessica.lee@example.com
Ryan	Anderson	ryan.anderson@example.com
Laura	Taylor	laura.taylor@example.com
Christopher	Hernandez	christopher.hernandez@example.com
Melissa	Lopez	melissa.lopez@example.com
Matthew	Gonzalez	matthew.gonzalez@example.com
Jennifer	Perez	jennifer.perez@example.com

Question 2. List the furniture items ordered along with their quantities and total order amounts.

Input:

```
mysql> SELECT f.Name, oi.Quantity, (oi.Quantity * f.Price) AS Total_Amount
    -> FROM Furniture f
    -> INNER JOIN Order_items oi ON f.Furniture_id = oi.Furniture_id;
```

Output:

Name	Quantity	Total_Amount
Sofa	3	45000.00
Dining Table	2	50000.00
Bed Frame	4	80000.00
Coffee Table	1	6000.00
Bookshelf	5	50000.00
TV Stand	3	36000.00
Dresser	4	60000.00
Desk	2	16000.00
Armchair	5	60000.00
Nightstand	1	5000.00
Office Chair	4	40000.00
Bar Stool	2	14000.00
Wardrobe	3	90000.00
Side Table	5	20000.00
Accent Chair	1	12000.00

Question 3. Find the total number of orders placed by each customer.

Input:

```
mysql> SELECT c.First_name, c.Last_name, COUNT(o.Order_id) AS Total_Orde
    -> FROM customers c
    -> LEFT JOIN orders o ON c.customer_id = o.Customer_id
    -> GROUP BY c.customer_id;
```

Output:

First_name	Last_name	Total_Orders
John	Doe	1
Jane	Smith	1
Alice	Johnson	1
Bob	Brown	1
Emily	Davis	1
Michael	Wilson	1
Sarah	Martinez	1
David	Garcia	1
Jessica	Lee	1
Ryan	Anderson	1
Laura	Taylor	1
Christopher	Hernandez	1
Melissa	Lopez	1
Matthew	Gonzalez	1
Jennifer	Perez	1

Question 4. Retrieve the details of customers who have not placed any orders.

Input:

```
mysql> SELECT c.First_name, c.Last_name, COUNT(o.Order_id) AS Total_Orders
    -> FROM customers c
    -> LEFT JOIN orders o ON c.customer_id = o.Customer_id
    -> GROUP BY c.customer_id;
```

Output:

First_name	Last_name	Total_Orders
John	Doe	1
Jane	Smith	1
Alice	Johnson	1
Bob	Brown	1
Emily	Davis	1
Michael	Wilson	1
Sarah	Martinez	1
David	Garcia	1
Jessica	Lee	1
Ryan	Anderson	1
Laura	Taylor	1
Christopher	Hernandez	1
Melissa	Lopez	1
Matthew	Gonzalez	1
Jennifer	Perez	1

Question 5. Calculate the total sales amount for each category of furniture.

Input:

Output:

```
mysql> SELECT *
    -> FROM customers
    -> WHERE customer_id NOT IN (SELECT DISTINCT Customer_id FROM orders);
Empty set (0.01 sec)
```

Question 6. Find the top 5 customers with the highest total order amounts.

Input:

Output:

```
mysql> SELECT f.Category, SUM(oi.Quantity * f.Price) AS Total_Sales
    -> FROM Furniture f
    -> INNER JOIN Order_items oi ON f.Furniture_id = oi.Furniture_id
    -> GROUP BY f.Category;
+-------------+-------------+
| Category    | Total_Sales |
+-------------+-------------+
| Living Room |   179000.00 |
| Dining Room |    50000.00 |
| Bedroom     |   235000.00 |
| Study       |    66000.00 |
| Office      |    40000.00 |
| Kitchen     |    14000.00 |
+-------------+-------------+
```

Question 7. Retrieve the furniture items that have not been ordered yet.

Input:

Output:

```
mysql> SELECT c.First_name, c.Last_name, SUM(f.Price * oi.Quantity) AS Total_Order_Amount
    -> FROM customers c
    -> INNER JOIN orders o ON c.customer_id = o.Customer_id
    -> INNER JOIN Order_items oi ON o.Order_id = oi.Order_id
    -> INNER JOIN Furniture f ON oi.Furniture_id = f.Furniture_id
    -> GROUP BY c.customer_id
    -> ORDER BY Total_Order_Amount DESC
    -> LIMIT 5;
+------------+-----------+--------------------+
| First_name | Last_name | Total_Order_Amount |
+------------+-----------+--------------------+
| Melissa    | Lopez     |           90000.00 |
| Alice      | Johnson   |           80000.00 |
| Sarah      | Martinez  |           60000.00 |
| Jessica    | Lee       |           60000.00 |
| Emily      | Davis     |           50000.00 |
+------------+-----------+--------------------+
```

Question 8. Find the average order total for each customer.

Input:

```
mysql> SELECT *
    -> FROM Furniture
    -> WHERE Furniture_id NOT IN (SELECT DISTINCT Furniture_id FROM Order_items);
Empty set (0.00 sec)
```

Output:

Question 9. List the customers who have ordered furniture items in the "Living Room" category.

Input:

```
mysql> SELECT c.customer_id, AVG(o.Order_total) AS Avg_Order_Total
    -> FROM customers c
    -> INNER JOIN orders o ON c.customer_id = o.Customer_id
    -> GROUP BY c.customer_id;
```

Output:

customer_id	Avg_Order_Total
1	100.500000
2	75.250000
3	200.000000
4	150.750000
5	300.000000
6	50.000000
7	125.800000
8	180.250000
9	95.500000
10	220.750000
11	75.000000
12	180.200000
13	140.000000
14	260.750000
15	320.500000

Question 10. Retrieve the details of customers who have ordered a specific furniture item.

Input:

```
mysql> SELECT DISTINCT c.*
    -> FROM customers c
    -> INNER JOIN orders o ON c.customer_id = o.Customer_id
    -> INNER JOIN Order_items oi ON o.Order_id = oi.Order_id
    -> INNER JOIN Furniture f ON oi.Furniture_id = f.Furniture_id
    -> WHERE f.Category = 'Living Room';
```

Output:

customer_id	First_name	Last_name	Email	Phone_no
1	John	Doe	john.doe@example.com	123-456-7890
4	Bob	Brown	bob.brown@example.com	777-555-9999
6	Michael	Wilson	michael.wilson@example.com	444-888-2222
9	Jessica	Lee	jessica.lee@example.com	333-444-5555
14	Matthew	Gonzalez	matthew.gonzalez@example.com	555-999-4444
15	Jennifer	Perez	jennifer.perez@example.com	111-777-8888

Assignment 5 :

Question 1: Find the total revenue generated from all orders.

Input:

```
mysql> SELECT SUM(Order_total) AS Total_Revenue
    -> FROM orders;
```

Output:

```
+---------------+
| Total_Revenue |
+---------------+
|       2475.25 |
+---------------+
```

Question 2: Find the total number of items ordered.

Input:

```
mysql> SELECT AVG(f.Price) AS Avg_Furniture_Price
    -> FROM Order_items oi
    -> JOIN Furniture f ON oi.Furniture_id = f.Furniture_id;
```

Output:

```
+--------------------+
| Avg_Furniture_Price |
+--------------------+
|       12733.333333 |
+--------------------+
1 row in set (0.00 sec)
```

Question 3: Find the customer who placed the highest total order.

Input:

```
mysql> SELECT Customer_id, SUM(Order_total) AS Total_Order
    -> FROM orders
    -> GROUP BY Customer_id
    -> ORDER BY Total_Order DESC
    -> LIMIT 1;
```

Output:

```
+-------------+-------------+
| Customer_id | Total_Order |
+-------------+-------------+
|          15 |      320.50 |
+-------------+-------------+
```

Question 4: Find the most popular category of furniture based on the total quantity ordered.

Input:

```
mysql> SELECT f.Category, SUM(oi.Quantity) AS Total_Quantity
    -> FROM Order_items oi
    -> JOIN Furniture f ON oi.Furniture_id = f.Furniture_id
    -> GROUP BY f.Category
    -> ORDER BY Total_Quantity DESC
    -> LIMIT 1;
```

Output:

```
+-------------+----------------+
| Category    | Total_Quantity |
+-------------+----------------+
| Living Room |             18 |
+-------------+----------------+
```

Question 5: Find the total revenue generated from orders placed in April 2024.

Input:

```
mysql> SELECT SUM(Order_total) AS Total_Revenue_April
    -> FROM orders
    -> WHERE MONTH(Order_date) = 4 AND YEAR(Order_date) = 2024;
```

Output:

```
+--------------------+
| Total_Revenue_April |
+--------------------+
|             2475.25 |
+--------------------+
```

Question 7: Find the average order total.

Input:

```
mysql> SELECT AVG(Order_total) AS Avg_Order_Total
    -> FROM orders;
```

Output:

```
+-----------------+
| Avg_Order_Total |
+-----------------+
|      165.016667 |
+-----------------+
```

Question 8: Find the top 3 highest-priced furniture items.

Input:

```
mysql> SELECT Name, Price
    -> FROM Furniture
    -> ORDER BY Price DESC
    -> LIMIT 3;
```

Output:

Name	Price
Wardrobe	30000.00
Dining Table	25000.00
Bed Frame	20000.00

Question 9: Find the total quantity ordered for each furniture item.

Input:

```
mysql> SELECT f.Name, SUM(oi.Quantity) AS Total_Quantity_Ordered
    -> FROM Order_items oi
    -> JOIN Furniture f ON oi.Furniture_id = f.Furniture_id
    -> GROUP BY f.Name;
```

Output:

Name	Total_Quantity_Ordered
Sofa	3
Dining Table	2
Bed Frame	4
Coffee Table	1
Bookshelf	5
TV Stand	3
Dresser	4
Desk	2
Armchair	5
Nightstand	1
Office Chair	4
Bar Stool	2
Wardrobe	3
Side Table	5
Accent Chair	1

Question 10: Find the average price of furniture items ordered**.**

Input:

```
mysql> SELECT AVG(f.Price) AS Avg_Furniture_Price
    -> FROM Order_items oi
    -> JOIN Furniture f ON oi.Furniture_id = f.Furniture_id;
+---------------------+
| Avg_Furniture_Price |
+---------------------+
|        12733.333333 |
+---------------------+
1 row in set (0.00 sec)
```

Output: