

# Flipmart Data Analysis Using My Sql

1. Find order name and corresponding category name

```
1 SELECT * FROM flipmart.order_details t1
2 JOIN flipmart.orders t2
3 ON t1.order_id = t2.order_id
4 JOIN flipmart.users t3
5 ON t3.user_id = t2.user_id
6 ;
```

order_id	amount	profit	quantity	category_id	order_id	user_id	order_date	user_id	name	state	city
B-25601	1275	-1148	7	1	B-25601	1	01-04-2018	1	Bharat	Gujarat	Ahmedabad
B-25614	494	54	4	1	B-25614	14	13-04-2018	14	Vandana	Himachal Pradesh	Simla
B-25618	362	127	1	1	B-25618	18	18-04-2018	18	Manju	Andhra Pradesh	Hyderabad
B-25638	182	-11	3	1	B-25638	38	26-04-2018	38	Parth	Maharashtra	Pune
B-25643	1061	-36	8	1	B-25643	43	29-04-2018	43	Kirti	Jammu and Kashmir	Kashmir
B-25651	200	-60	4	1	B-25651	51	07-05-2018	51	Anurag	Madhya Pradesh	Indore
B-25664	83	-48	1	1	B-25664	64	20-05-2018	64	Pratyusmita	Bihar	Patna
B-25667	516	69	4	1	B-25667	67	23-05-2018	67	Anjali	Haryana	Chandigarh
B-25670	656	-36	2	1	B-25670	70	25-05-2018	70	Charika	Goa	Goa
B-25718	115	1	1	1	B-25718	118	12-07-2018	118	Anjali	Maharashtra	Mumbai

## QUESTIONS

1. Find order\_id, name and city by joining users and orders.
2. Find order\_id, product category by joining order\_details and category

The screenshot shows a database IDE with a SQL query editor at the top and a result grid at the bottom. The query is a JOIN statement between flipmart.orders and flipmart.users. The result grid displays 16 rows of data with columns name, city, and order\_id.

```

1 • SELECT name , city , order_id
2 FROM flipmart.orders
3 JOIN flipmart.users
4 ON users.user_id = orders.user_id

```

name	city	order_id
Bharat	Ahmedabad	B-26011
Bharat	Ahmedabad	B-26074
Pearl	Pune	B-25602
Pearl	Pune	B-26012
Pearl	Pune	B-26075
Jahan	Bhopal	B-25603
Jahan	Bhopal	B-26013
Jahan	Bhopal	B-26076
Divsha	Jaipur	B-25604
Divsha	Jaipur	B-26014
Divsha	Jaipur	B-26077
Kasheen	Kolkata	B-25605
Kasheen	Kolkata	B-26015
Kasheen	Kolkata	B-26078
Hazel	Bangalore	B-25606
Hazel	Bangalore	B-26016

## QUESTIONS

1. Find all the orders placed in pune
2. Find all orders under Chairs category
3. |

ServerToolsScriptingHelp

order\_details (1)flipmart.category (1)category - Tableorder\_details (1)flipmart.order\_details (1)order\_details - Tableorders - Tableusers - Tableusersorder\_detailsusersorders

Limit to 1000 rows

```
1 SELECT *
2 FROM flipmart.orders
3 JOIN flipmart.users
4 ON users.user_id = orders.user_id
5 WHERE users.city = 'Pune';
```

Result Grid

Filter Rows:

Export:

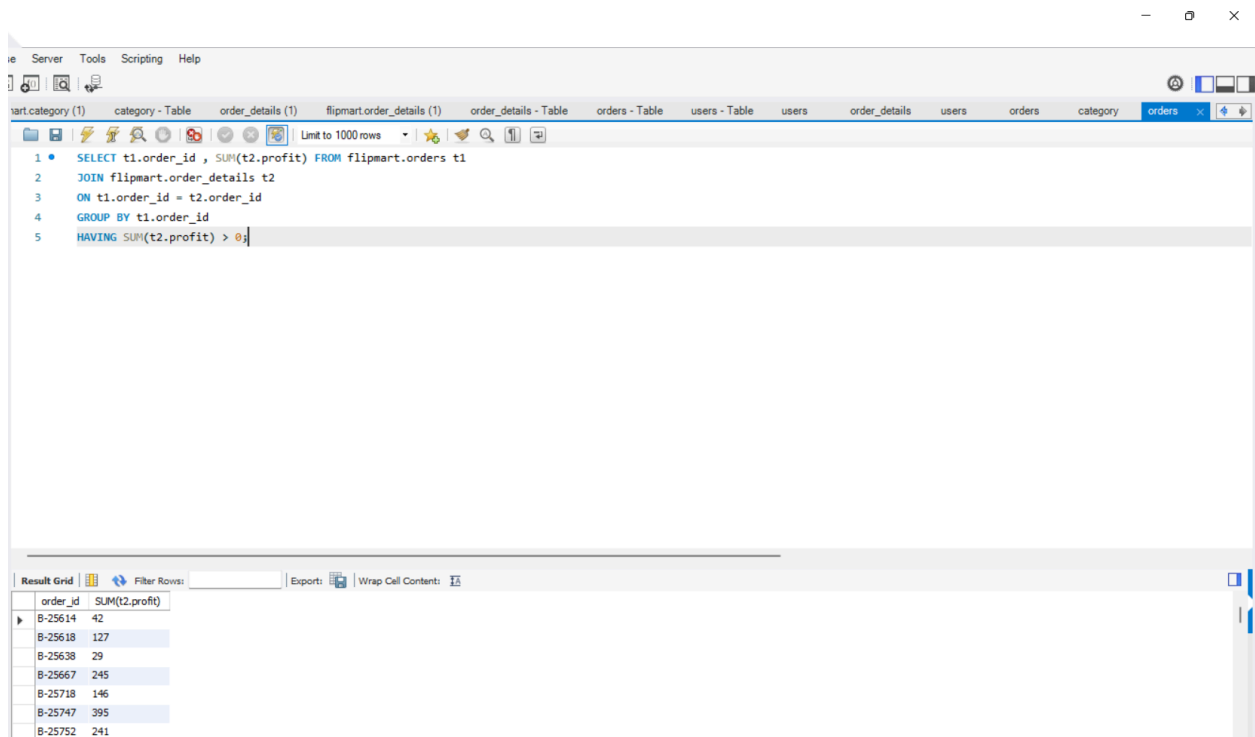
Wrap Cell Content:

order_id	user_id	order_date	user_id	name	state	city
B-26012	2	13-02-2019	2	Pearl	Maharashtra	Pune
B-26075	2	21-03-2019	2	Pearl	Maharashtra	Pune
B-25620	20	20-04-2018	20	Sarita	Maharashtra	Pune
B-26030	20	21-02-2019	20	Sarita	Maharashtra	Pune
B-26093	20	27-03-2019	20	Sarita	Maharashtra	Pune
B-25638	38	26-04-2018	38	Parth	Maharashtra	Pune
B-26048	38	04-03-2019	38	Parth	Maharashtra	Pune
B-25656	56	11-05-2018	56	Priyanka	Maharashtra	Pune
B-26064	56	16-03-2019	56	Priyanka	Maharashtra	Pune

# QUESTIONS

1. Find all profitable orders
2. Find the customer who has placed max number of orders
3. Which is the most profitable category
4. Which is the most profitable state
5. Find all categories with profit higher than 5000

1



The screenshot shows a database IDE with a SQL query editor and a results grid. The query is as follows:

```
1 SELECT t1.order_id , SUM(t2.profit) FROM flipmart.orders t1
2 JOIN flipmart.order_details t2
3 ON t1.order_id = t2.order_id
4 GROUP BY t1.order_id
5 HAVING SUM(t2.profit) > 0;
```

The results grid displays the following data:

order_id	SUM(t2.profit)
B-25614	42
B-25618	127
B-25638	29
B-25667	245
B-25718	146
B-25747	395
B-25752	241

2

Database client interface showing a SQL query and its results.

Query:

```
1 SELECT name , COUNT(*) FROM flipmart.orders t1
2 JOIN flipmart.users t2
3 ON t1.user_id = t2.user_id
4 GROUP BY t2.name
5 ORDER BY COUNT(*) DESC
6 LIMIT 1
7 ;
```

Result Grid:

name	COUNT(*)
Shreya	6

3

Database client interface showing a SQL query and its results.

Query:

```
1 SELECT t2.vertical , SUM(profit) FROM flipmart.order_details t1
2 JOIN flipmart.category t2
3 ON t1.category_id = t2.category_id
4 GROUP BY t2.vertical
5 ORDER BY SUM(profit) DESC LIMIT 1;
```

Result Grid:

vertical	SUM(profit)
Printers	5964

4

