

Courier Delivery Service

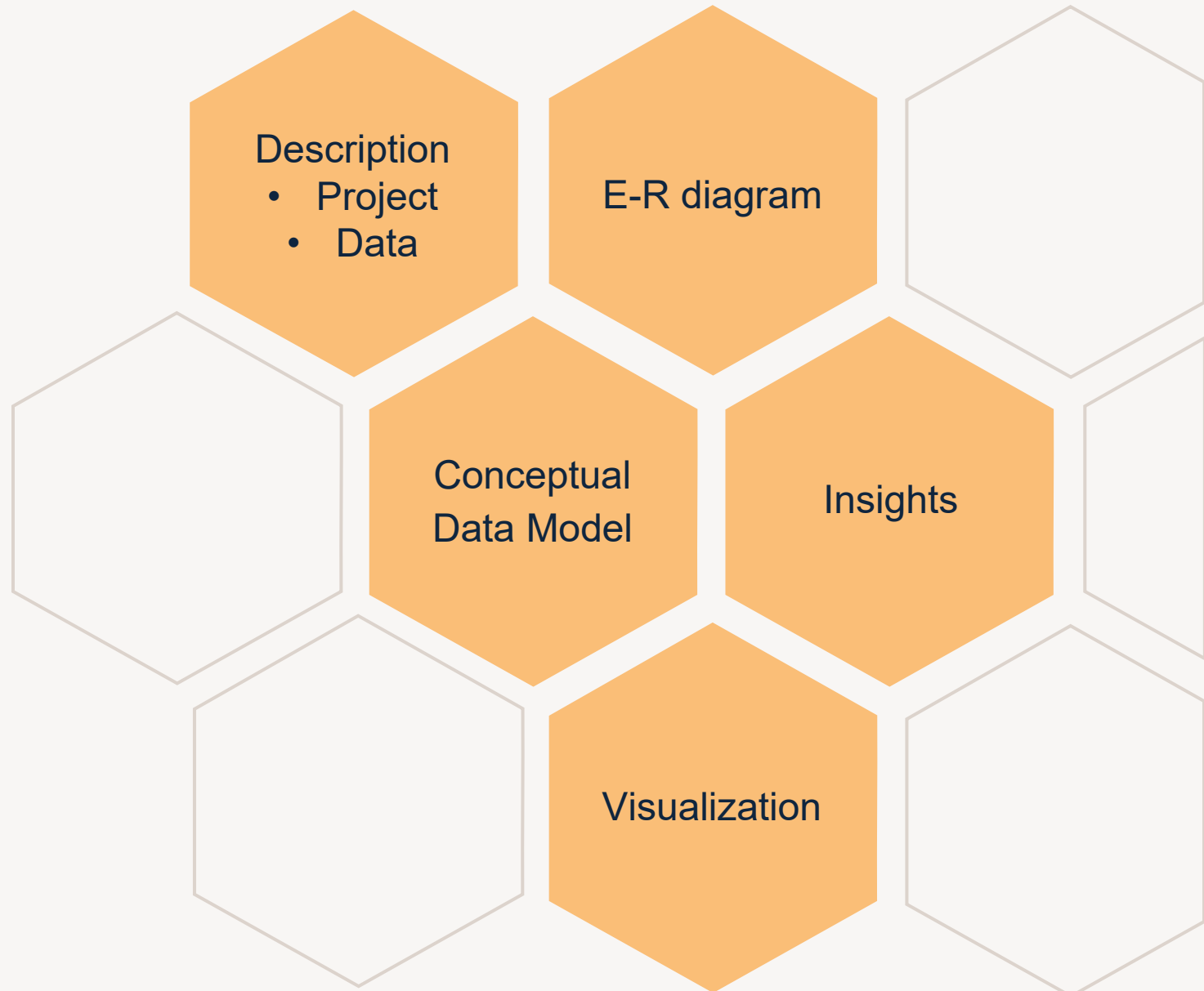
Optimizing Food Order Management and Delivery

Vutla, Sushma Reddy





Agenda



Introduction

- Courier Delivery service database stores information on food orders by users and delivery of orders by couriers to users.
- **Key Objectives and Goals**
 - Enhance Customer Experience
 - Minimize Order Processing Time
 - Improve Order Accuracy
 - Enhance Inventory Management
 - Increase Operational Efficiency
- **Importance of Efficient Food Order Processing and Delivery**
 - Customer Satisfaction
 - Competitive Advantage
 - Brand Reputation
 - Operational Cost Savings
 - Increased Revenue
 - Adaptability to Market Trends

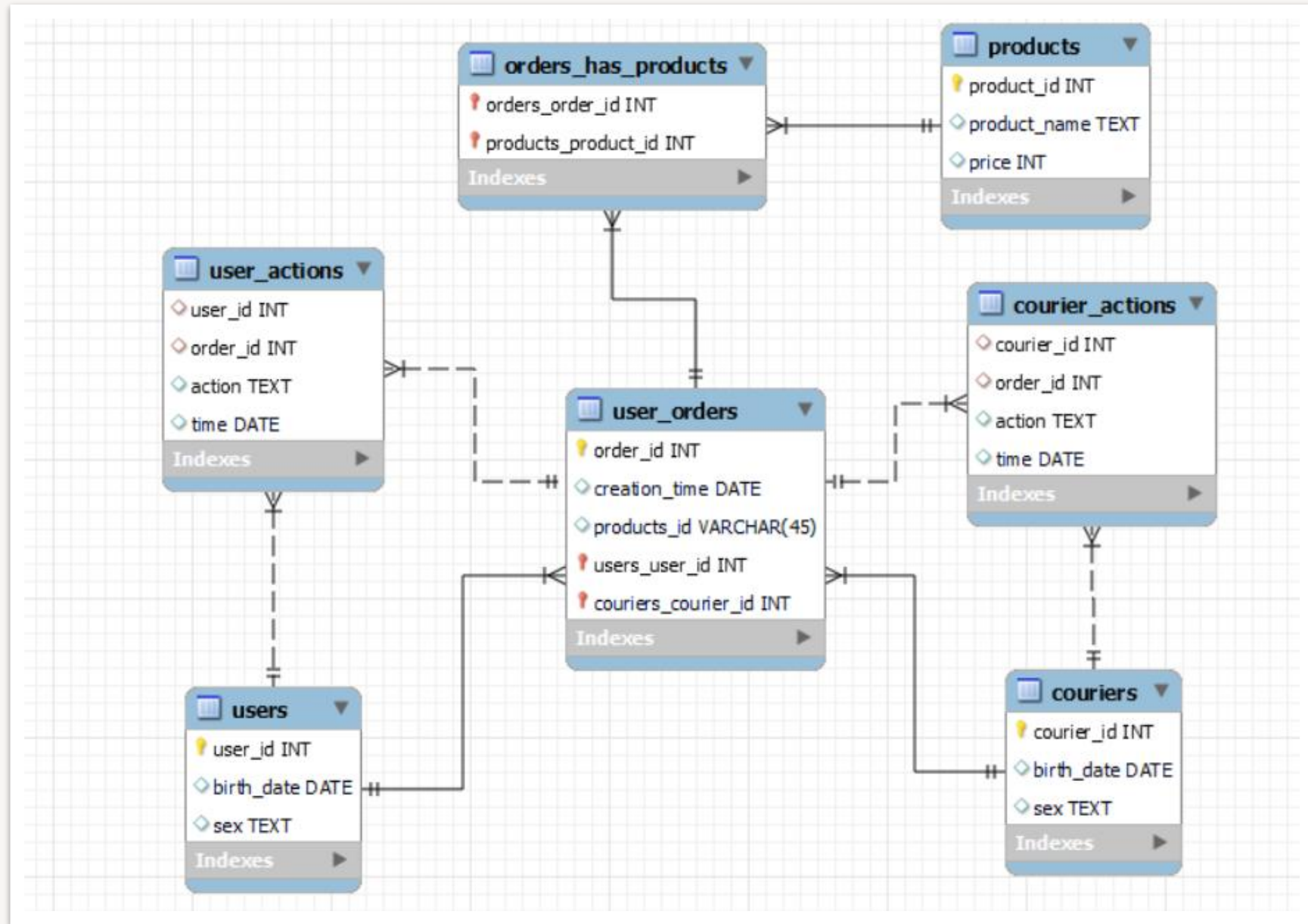
Description of Data

The courier service database is a comprehensive system storing crucial information for managing food orders, including details about couriers, users, and specific orders/products. It enables tracking the lifecycle of orders, actions by users and couriers, offering a complete view of the service's functionalities.

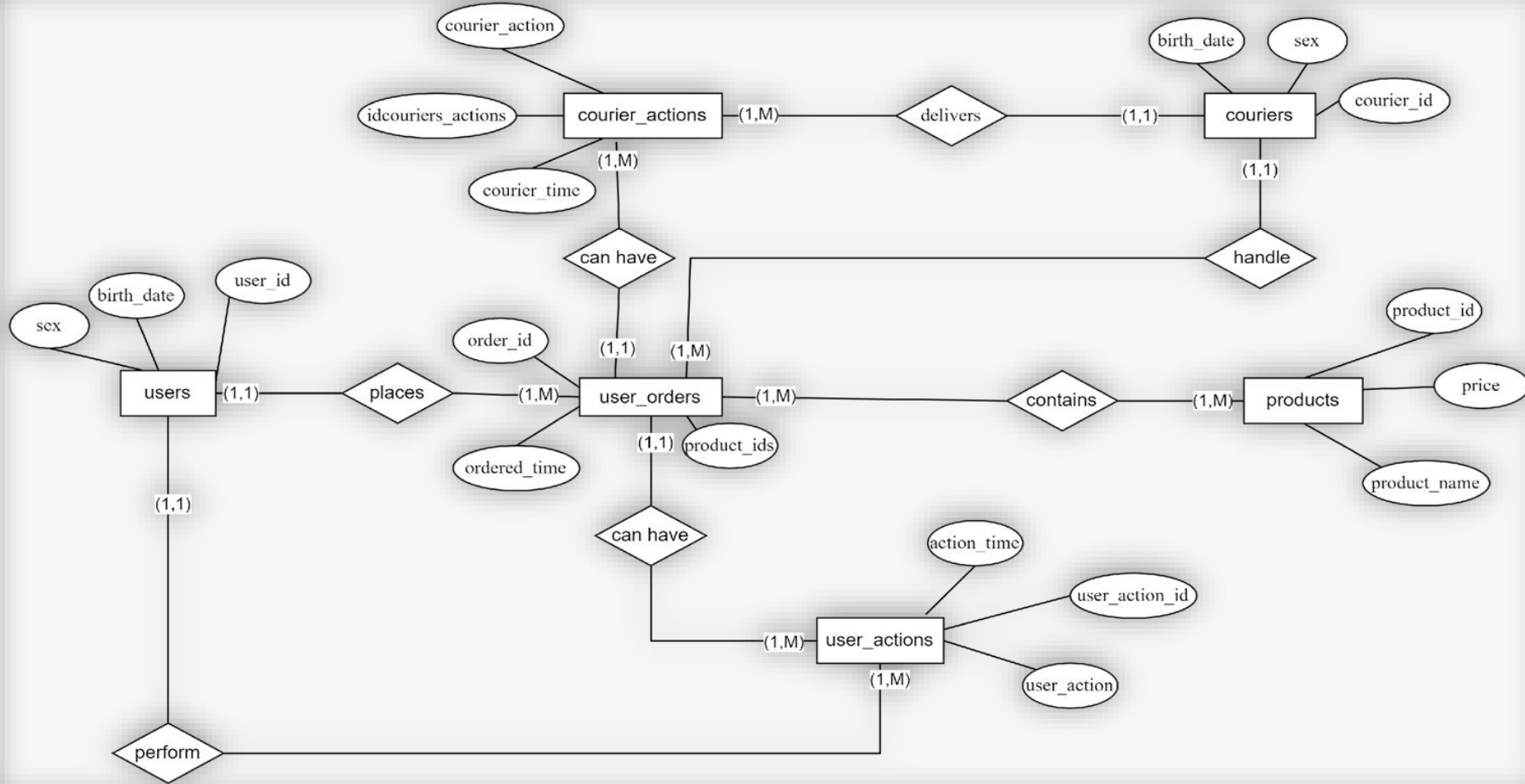
Tables & Contents:

1. courier_actions: Tracks courier actions (courier_id, order_id, action, time).
2. couriers: Contains courier details (Courier_id, Birth_date, Sex).
3. products: Stores product information (product_id, name, price).
4. user_actions: Records user actions (User_id, Order_id, Action, Time).
5. user_orders: Holds order details (Order_id, Creation_time, Product_id).
6. users: Stores user information (User_id, Birth_date, Sex).

E-R Diagram



Conceptual Data Model



Insights

1. What are the top 3 most ordered products?

```
11  /* 1. Top 3 most ordered products: */
12  •  SELECT p.product_id,p.name,COUNT(uo.product_id) AS order_count
13      FROM user_orders uo
14      JOIN products p ON uo.product_id = p.product_id
15      GROUP BY p.product_id, p.name
16      ORDER BY order_count DESC
17      Limit 3;|
18
```

100% 9:17

Result Grid Filter Rows: Search Export: Fetch rows:

	product_id	name	order_count
	40	bread	4161
	77	chicken	4144
	1	sugar	4130

2. On which date were the highest number of orders placed?

```
19  /* 2. Date at which highest number of orders has been placed? */
20  •  SELECT COUNT(order_id) AS max_count, cast(ca.time AS date ) AS delivery_date
21      FROM courier_actions ca group by delivery_date order by max_count DESC;
22
23
```

100% 26:16

Result Grid Filter Rows: Search Export:

max_count	delivery_date
12296	2004-09-22

3. How is the distribution of order actions (placed, delivered, etc.) segmented by gender?

```
22
23  /* 3. What is the distribution of order actions (placed, delivered, etc.) by gender? */
24 • SELECT u.sex, ua.action, COUNT(ua.order_id) AS action_count
25 FROM users u
26 JOIN user_actions ua ON u.user_id = ua.user_id
27 GROUP BY u.sex, ua.action;
28
```

100% 75:21

Result Grid Filter Rows: Search Export:

	sex	action	action_count
	female	create_order	28146
	female	cancel_order	1434
	male	create_order	28362
	male	cancel_order	1404

4. What is the count of orders handled by each courier?

```
28
29  /* 4. What are the number of orders per courier? */
30 • SELECT courier_id, COUNT(*) AS num_orders_handled
31 FROM courier_actions
32 GROUP BY courier_id
33 ORDER BY num_orders_handled DESC;
34
```

100% 47:26

Result Grid Filter Rows: Search Export: Fetch rows:

	courier_id	num_orders_handled
	252	108
	291	108
	492	108
	708	107
	23	105
	276	103
	501	103
	329	101
	203	100
	66	100
	206	100
	179	99
	40	99

Result 12

• 5. How many orders were placed by individual users?

```
34
35  /* 5. what is the number of orders placed by an user ? */
36 • SELECT user_id, action, COUNT(action) AS action_count
37 FROM user_actions
38 GROUP BY user_id, action
39 ORDER BY action_count DESC;
40
```

100% 20:32

Result Grid Filter Rows: Search Export: Fetch rows:

	user_id	action	action_count
	3793	create_order	18
	3922	create_order	15
	362	create_order	15
	3333	create_order	14
	803	create_order	14
	3941	create_order	14
	12	create_order	13
	4046	create_order	13
	1170	create_order	13
	4015	create_order	13
	763	create_order	13
	1533	create_order	13
	2077	create_order	13

6. What is the average number of products per order?

```
41  /* 6. Average number of products per order: */
42 • SELECT AVG(num_products) AS avg_products_per_order
43 FROM (
44     SELECT order_id, COUNT(*) AS num_products
45     FROM user_orders
46     GROUP BY order_id
47 ) AS product_counts;
48
```

100% 26:38

Result Grid Filter Rows: Search Export:

avg_products_per_or...
3.3978

- 7. Which product generated the highest total revenue?

```
49  /* 7. Most popular product by total revenue generated: */
50  • SELECT p.name AS popular_product, SUM(p.price) AS total_revenue
51  FROM user_orders uo
52  JOIN Products p ON uo.product_id = p.product_id
53  GROUP BY p.name
54  ORDER BY total_revenue DESC
55  LIMIT 1;
```

100% 21:45

Result Grid Filter Rows: Search Export: Fetch rows:

popular_product	total_revenue
pork	1437750

- 8. What are the top most ordered products for male and female users, and how do their order counts differ?

```
57  /* 8. What are the top most ordered products for both male and female users, and how do they differ in terms of order count? */
58  • WITH RankedProducts AS (
59      SELECT
60          u.sex,
61          p.name,
62          COUNT(o.product_id) AS order_count,
63          ROW_NUMBER() OVER (PARTITION BY u.sex ORDER BY COUNT(o.product_id) DESC) AS product_rank
64      FROM users u
65      JOIN user_actions ua ON u.user_id = ua.user_id
66      JOIN user_orders o ON ua.order_id = o.order_id
67      JOIN products p ON o.product_id = p.product_id
68      WHERE ua.action = 'create_order'
69      GROUP BY u.sex, p.name
70  )
71  SELECT
72      sex, name AS most_ordered_product, order_count
73  FROM RankedProducts
74  WHERE product_rank = 1;
```

100% 16:53

Result Grid Filter Rows: Search Export:

sex	most_ordered_prod...	order_count
female	bread	2002
male	bananas	1975

9. What is the busiest day for orders (by count)?

76
77
78 •
79
80
81
82
83
84
85
86

```
/* 9. Busiest day for orders (by count): */  
SELECT creation_time AS order_date, COUNT(*) AS total_orders  
FROM user_Orders  
GROUP BY order_date  
ORDER BY total_orders DESC  
LIMIT 1;
```

100%
Result Grid
Filter Rows:
Export:
Fetch rows:

20:73

Search

Fetch rows:

order_date	total_orders
2/9/2022 20:40	80

10. How does the average delivery time vary across different couriers?

82
83
84 •
85
86
87
88
89
90

```
/*10. How does the average delivery time vary across different couriers? */  
SELECT courier_id,AVG(CAST(time AS SIGNED)) AS average_delivery_time  
FROM courier_actions  
WHERE action = 'deliver_order'  
GROUP BY courier_id  
ORDER BY average_delivery_time DESC;
```

100%
Result Grid
Filter Rows:
Export:
Fetch rows:

44:76

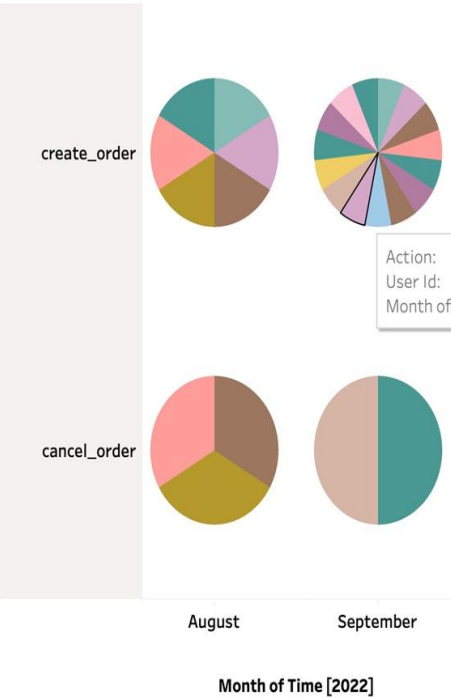
Search

Fetch rows:

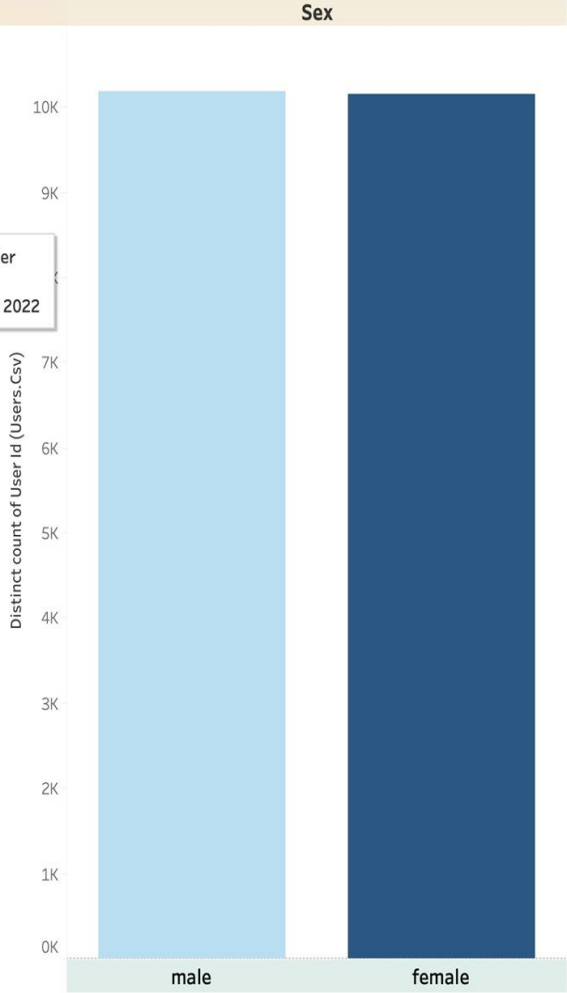
courier_id	average_delivery_ti...
544	23.5000
328	22.8276
214	22.4848
490	21.8333
572	21.6250
545	21.6250
4	21.6129
251	21.5882
736	21.4483
103	21.3023
179	21.2653
770	21.0385
642	20.9024
81	20.8919
13	20.8537

Visualization Charts

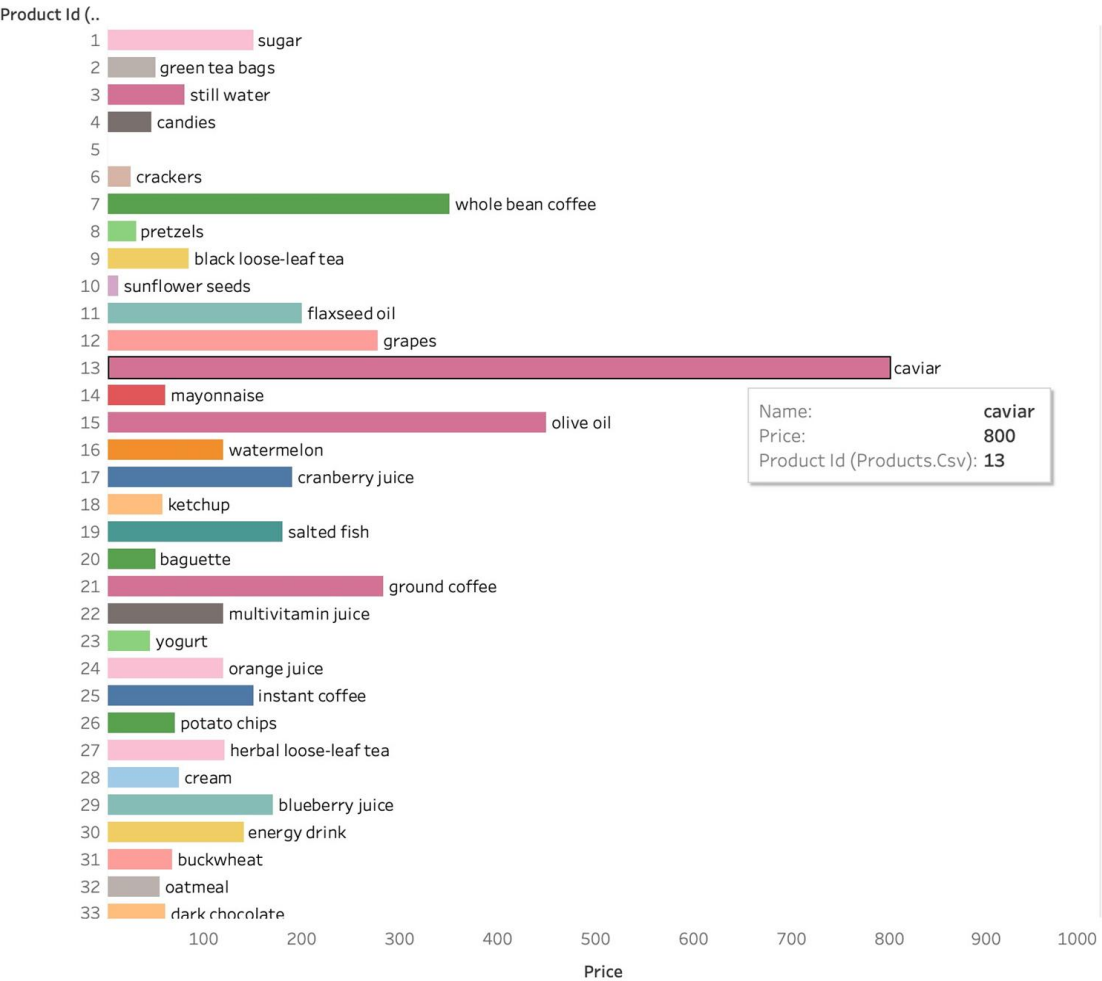
User Analysis



Segmented User Analysis

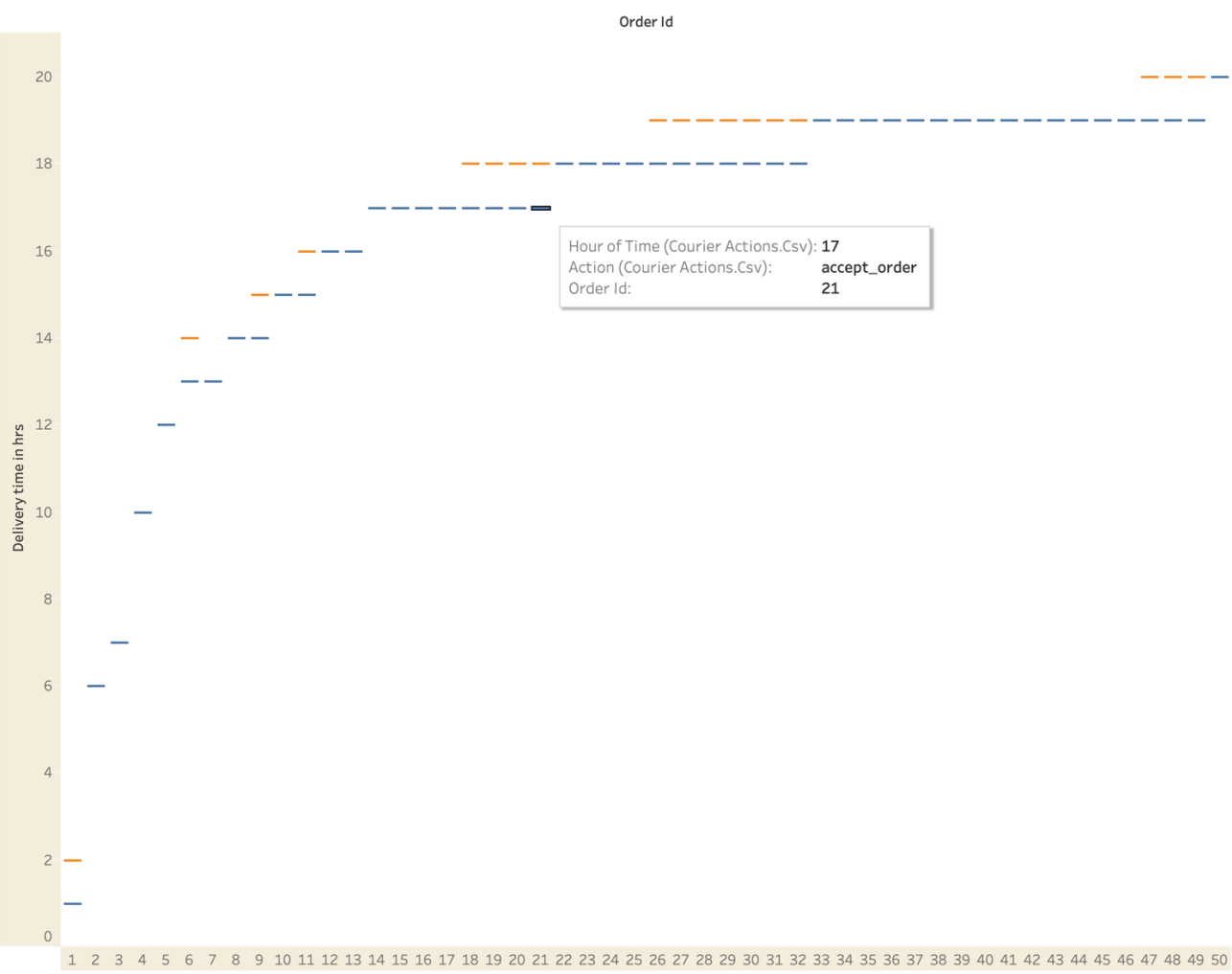


Product Pricing

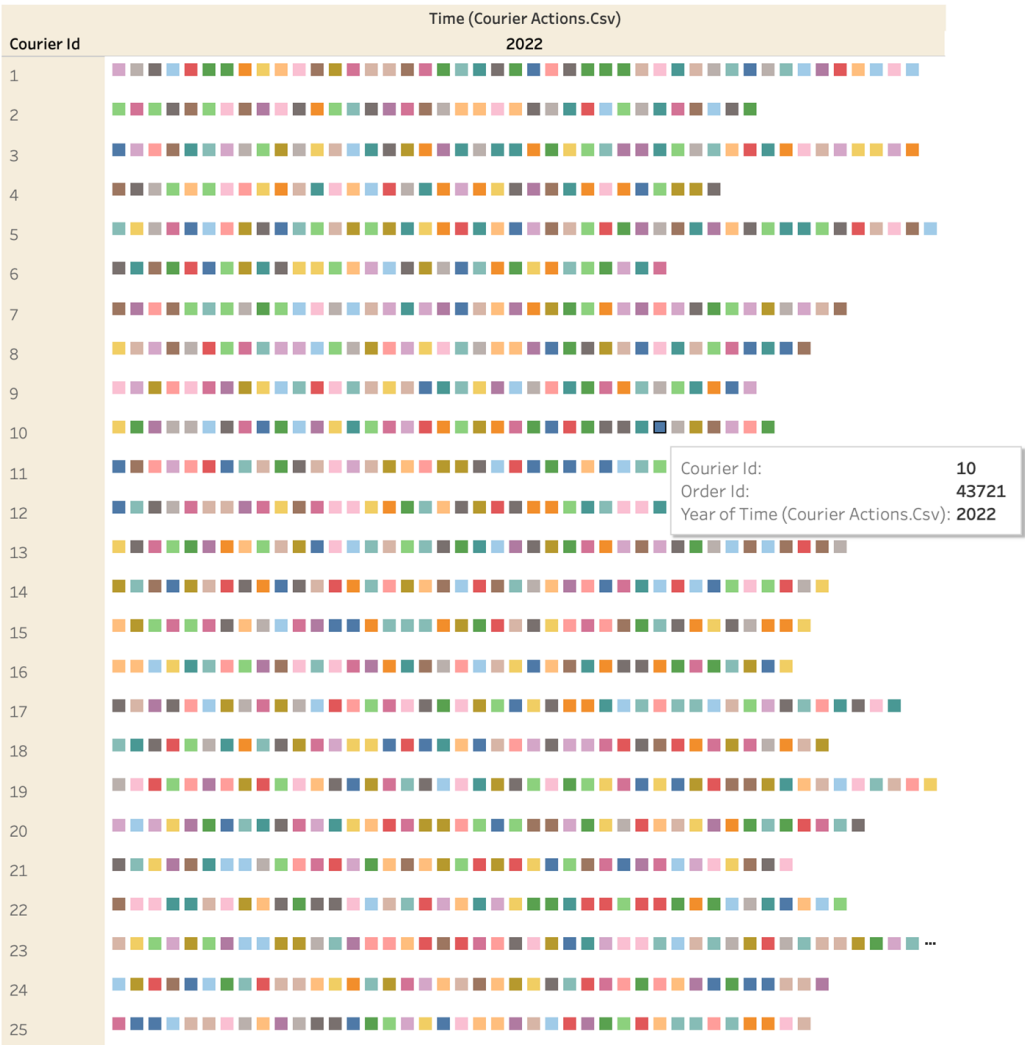


Visualization Charts

Courier Activity Monitor



Top 25 courier actions



Conclusion

- Uncovered insights for operational efficiency and customer-centric strategies.
- Identified patterns: top-selling products, gender-specific preferences, peak order periods, and courier efficiency.
- Actionable information for inventory management, targeted marketing, and service optimization.
- Recognized benefits of knowing most ordered products and peak order days for strategic stock management and resource allocation.
- Customer behavior insights, including preferences of high-ordering users and average products per order.
- Opportunities to enhance customer experiences by tailoring services to preferences and habits.
- Potential benefits: optimized workflow, better resource utilization, elevated service standard, and improved customer satisfaction.





Thank you