

## ✓ Milestone 6 | FastKitchen Customer Analysis

**INTRODUCTION:** FastKitchen is a fictitious restaurant and the dataset you'll be working with is constructed. While this dataset might not represent real data, it emulates characteristics of real data. When you're interviewing for a job, you might be asked to look at this kind of data to show off your skills in a context related to the company and the job position!

In this Milestone, you'll step into the shoes of a data contractor who is helping a new fast-food restaurant understand their customer base. You will need to make use of one type of outer join to help the restaurant manager combine information about their customers. These customers include registered customers who have accounts on the restaurant's website, and guest customers who do not register for accounts.

**HOW IT WORKS:** Follow the prompts in the questions below to investigate your data. Post your answers in the provided boxes: the **yellow boxes** for the queries you write, and **blue boxes** for text-based answers. When you're done, export your document as a pdf file and submit it on the Milestone page – see instructions for creating a PDF at the end of the Milestone. Please don't ever remove (paste your query below 📌) or (write your **answer** below 📌). These help your Evaluator!

### – Data Set **Description**

The data in this Milestone (fastkitchen.\*) depicts orders made at a fictional takeout-only fast food restaurant in the Midwestern United States. The restaurant has an online site where customers can put in orders for carryout or delivery; customers can also make orders offline at the restaurant's storefront. You will be working with two tables in this Milestone: orders and users.

Each row in the `orders` table is a single order that was placed at the restaurant. This table has seven columns:

- `order_id` - unique order id, primary key

- **timestamp** - when the order was made
- **user\_id** - user\_id for registered accounts, blank if guest customer
- **order\_type** - whether the order was made onsite, online carryout, or online delivery
- **subtotal** - base amount for the order
- **tip** - amount of tip, if any, left by the customer
- **total** - subtotal + tip

Customers have the option of creating a user account, which can be used both in person and online. The `users` table has five columns:

- **user\_id** - unique user\_id value, primary key
  - **reg\_timestamp** - when the user registered their account
  - **city** - user city
  - **state** - two-letter code for state
  - **zip** - zip code
- 

## – Task 1: Explore information about orders.

To start off, let's warm up with some questions on the individual tables, before we ask questions that require joining the two tables together. Let's look at the `orders` table first.

**A.** What is the average total amount (including tips) spent per order?

(paste your query below 🖱️)

```
SELECT AVG(total) AS avg_total_per_order
FROM fastkitchen.orders
```

(write your **answer** below 🖱️)

```
22.2170348258706468
```

- B. Compare the average subtotals, tips, and totals spent by each order type (onsite, carryout, delivery).

(paste your query below 🖱)

```
SELECT
  order_type,
  AVG(subtotal) AS avg_subtotal,
  AVG(tip) AS avg_tip,
  AVG(total) AS avg_total
FROM fastkitchen.orders
GROUP BY order_type
```

Are there any major differences between order types?

(write your **answer** below 🖱)

The average subtotal and total amounts are very similar across all three order types, with only slight variation. However, carryout orders tend to have slightly higher tips on average than onsite orders and even slightly more than delivery.

- C. Write a query to count the number of orders made by **registered users**.

(paste your query below 🖱)

```
SELECT COUNT(*) AS n_registered_orders
FROM fastkitchen.orders
WHERE user_id IS NOT NULL
```

- D. Now, write a query to return the number of orders made by **non-registered customers**. Remember, non-registered customers don't have a user id.

(paste your query below 🖱)

```
SELECT COUNT(*) AS n_non_registered_orders
FROM fastkitchen.orders
WHERE user_id IS NULL
```

Compared alongside the returned count from the query in C, which group is larger?

(write your **answer** below 🖱)

The number of guest orders is greater than the number of orders placed by registered users. (2088 > 1932)

- E. While guest customers tend to place more orders, there might be underlying reasons behind this trend. Your goal is to understand user behavior and suggest a practical solution to help improve customer registration and retention.



**Try this prompt:** Guest customers placed more orders than registered users. Why might guests order more often, and what could FastKitchen do to encourage guests to register accounts without reducing conversion?

Based on ChatGPT's response, what's one potential reason for high guest activity? What's one idea FastKitchen could try to convert guest users into registered users?

(write your **answer** below 🖱)

Convenience is one of the possible reasons why guest customers are more likely to order regularly as they might want to make the

order in a short period of time without creating an account or logging in.

Some of the benefits that FastKitchen can provide to make people want to register include loyalty points or free checkout or special benefits. Notably, the registration must be voluntary and fast lest it may work against its conversions.

## – Task 2: Explore information about registered users.

Next, we'll check out the `users` table.

A. Write a query that counts the number of users by city.

(paste your query below 🖱️)

```
SELECT
  city,
  COUNT(*) AS n_users_by_city
FROM fastkitchen.users
GROUP BY city
ORDER BY n_users_by_city DESC
```

Which city has the highest number of users, and how many users are there?

(write your **answer** below 🖱️)

Allen, 212 users. Followed by Nulle Pointe and then Maebe

B. Expand the query so that you group by zip code as well. Does this help explain what you found in part 2A?

(paste your query below 🖱️)

```
SELECT
  city,
  zip,
  COUNT(*) AS n_users
FROM fastkitchen.users
GROUP BY
  city,
  zip
ORDER BY n_users DESC
```

(write your **answer** below 🙋)

Grouping by zip code helps explain Allen's large user base. Most of Allen's users are concentrated across three nearby zip codes: 63218, 63216, and 63215. This suggests that the customer base in Allen is spread across multiple neighborhoods, each with a significant number of users. In contrast, cities like Nulle Pointe and Maebe seem to have just one dominant zip code. It also supports the fact that the largest user population come from the zipcode of Allen.

### – **Task 3:** How do orders compare between zip codes and cities?

Finally, we'll combine the `user` and `orders` tables into a single, joined table.

- A.** To start, simply write a query that returns all of the columns, joining the two tables on the `user_id` column. Make sure that you choose a join that keeps all of the orders, even when there isn't a matching registered user.



**Try this prompt:** I'm trying to join an orders table with a users table on user\_id in my SQL query, but not all orders have a matching user (some are guests). Which type of JOIN should I use if I want to keep all orders, even when there's no matching user, and why?

(paste your query below 📌)

```
SELECT
  o.*,
  u.*
FROM fastkitchen.orders AS o
LEFT JOIN fastkitchen.users as u
ON o.user_id = u.user_id
```

- B.** Add to the query from 3A to answer the following question: in which zip code is the user with the highest amount of money spent?

(paste your query below 📌)

(i figured i could also keep LEFT JOIN in this task, but will need to require a line to filter where zip code/ user id IS NOT NULL)

```
SELECT
  u.user_id,
  u.zip,
  SUM(o.total) as total_spent
FROM fastkitchen.orders AS o
INNER JOIN fastkitchen.users as u
ON o.user_id = u.user_id
GROUP BY u.user_id,
         u.zip
ORDER BY total_spent DESC
```

(write your **answer** below 📌)

63222 - 1103.11 of total spent

- C. Write a query that returns the average total amount spent per order by zip code.

(paste your query below 📌)

```
SELECT
  u.zip,
  AVG(o.total) AS avg_total_spent
FROM fastkitchen.orders AS o
LEFT JOIN fastkitchen.users as u
ON o.user_id = u.user_id
GROUP BY u.zip
ORDER BY avg_total_spent DESC
```

How many of the zip codes spend more on average than non-registered guest customers?

(write your **answer** below 📌)

3 zip codes, including 63216, 63215, 63222

## – Submission

Great work completing this Milestone! To submit your completed Milestone, you will need to download / export this document as a PDF and then upload it to the Milestone submission page. You can find the option to download as a PDF from the File menu in the upper-left corner of the Google Doc interface.