 **🧪 Lab** | FastKitchen Customers

**INTRODUCTION:** In our previous lesson, you’ve learned additional ways of joining tables together, with three different types of outer join: the left join, right join, and the full outer join. While an inner join retains only information when there’s a match between the joined tables, an outer join will also output information that can only be found in one table.

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!

**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.

PROMPT: In this lab, 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 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.

**—** Data Set **Description**

The data in this lab (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 lab: fastkitchen\_orders and fastkitchen\_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 fastkitchen\_orders table first.

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

(paste your query below 👇)

|  |
| --- |
| select  avg(total)  from fastkitchen\_orders |

(write your **answer** below 👇)

|  |
| --- |
| 22.2170348258706 |

1. Compare the average subtotals, tips, and totals spent by each order type (onsite, carryout, delivery). Are there any major differences between order types?

(paste your query below 👇)

|  |
| --- |
| select  order\_type,  avg(subtotal),  avg(tip),  avg(total)  from fastkitchen\_orders  group by 1 |

(write your **answer** below 👇)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | carryout | 20.1586689419795 | 2.01127133105802 | 22.1699402730375 | | delivery | 20.6122655426765 | 1.86855637513172 | 22.4808219178082 | | onsite | 20.1781569246972 | 1.93611901000527 | 22.1142759347025 | |

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

(paste your query below 👇)

|  |
| --- |
| select  fu.user\_id,  count(order\_id)  from fastkitchen\_orders fo  join fastkitchen\_users fu  on fu.user\_id = fo.user\_id  group by 1 |

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

(paste your query below 👇)

|  |
| --- |
| select  fo.user\_id,  count(order\_id)  from fastkitchen\_orders fo  left join fastkitchen\_users fu  on fu.user\_id = fo.user\_id  where fo.user\_id is null  group by 1 |

(write your **answer** below 👇)

|  |
| --- |
| 2088 non-registered users is the largest group |

**— Task 2:** Explore information about registered users.

Next, we’ll check out the fastkitchen\_users table.

1. Write a query that counts the number of users by city. Which city has the highest number of users, and how many users are there?

(paste your query below 👇)

|  |
| --- |
| select  city,  count(user\_id) as num\_users  from fastkitchen\_users  group by 1  order by count(user\_id) desc |

(write your **answer** below 👇)

|  |
| --- |
| City: Allen count: 212 |

1. 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(user\_id) as num\_users  from fastkitchen\_users  group by 1,2  order by count(user\_id) desc |

(write your **answer** below 👇)

|  |
| --- |
| (Write your answer here) |

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

Finally, we’ll combine the fastkitchen\_users and fastkitchen\_orders tables into a single, joined table.

1. 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.

(paste your query below 👇)

|  |
| --- |
| select \*  from fastkitchen\_orders fo  left join fastkitchen\_users fu  on fo.user\_id = fu.user\_id |

1. 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 👇)

|  |
| --- |
| select  fo.user\_id,  zip,  sum(total) as total  from fastkitchen\_orders fo  left join fastkitchen\_users fu  on fo.user\_id = fu.user\_id  group by 1,2  order by sum(total) desc |

(write your **answer** below 👇)

|  |
| --- |
| 63222 |

1. Write a query that returns the average total amount spent per order by zip code. How many of the zip codes spend more on average than non-registered guest customers?

**HINT:** The null zip code represents non-registered guests!

(paste your query below 👇)

|  |
| --- |
| select  zip,  avg(total) as avg\_total  from fastkitchen\_orders fo  left join fastkitchen\_users fu  on fo.user\_id = fu.user\_id  group by 1  order by avg(total) desc |

(write your **answer** below 👇)

|  |
| --- |
| 3 |