1. **Describe your solution. What tradeoffs did you make when designing your solution, and why did you make those decisions?**

   To import data from the PunkAPI I used the async HTTPX module to send a GET request to the API endpoint and receive the data in JSON format. Due to pagination limits, 5 requests were sent to receive all 325 beer recipes. Then, I created a database schema with two tables: “beers” and “hops”. The “beers” table contains information about each beer, and the “hops” table contains information about the type and amount of hops used in each beer. I then looped over each beer and inserted its data into the “beers” table, and for each beer, I looped over the hops and inserted their data into the “hops” table. I used ORM provided by SQLAlchemy to interact with database.

   To show the average (mean) fermentation temperature for each type of hops, I created a SQL query that joins the beers and hops tables, groups by the hop name, and calculates the average fermentation temperature.
   To show the average (mean) fermentation temperature for the primary hops I created a SQL query that joins the beers and hops, groups by the beer id, and calculates the primary hops with the highest amount in the recipe.

   Tradeoffs:

   I decided to extract only relevant data to the required queries from JSON response due to its nested complexity. I checked for the missing data in the extracted data. Beer ID 169 was missing fermentation temperature and had no hops. Beer ID 72 was missing fermentation temperature. I decided to drop both beers from the results. Another solution would be probably use the average fermentation temperature from all beers to fill Beer ID 72 missing value.

2. **Take a look at the views you created, and make any other queries you think might be useful. From this data (not from your knowledge about brewing beer, if you have any), do you think there is a correlation between hop types chosen for a recipe and fermentation temperature? You don't need to demonstrate a correlation formally (or lack thereof). Just tell us what you see.**

   I made 2 additional queries “get_ten_most_used_hops” and “get_beers_by_hop” to show the top 10 most used hops in the recipes and show the beers that use a particular hop.
   From the data, I couldn’t notice any correlation between hop types and fermentation temperature. Some hop types, such as "Ahtanum" and "Styrian Goldings", have a wide range of fermentation temperatures, while others, such as "Bramling Cross" and "Chinook", have a narrower range.

3. **We use the data from our data warehouse to make important product and business decisions, and some reports are visible to important stakeholders (investors), so it's important that the data is always as accurate and as up-to-date as possible. What kinds of**

problems could arise with the data import as you have implemented it? How would you mitigate these issues and monitor and/or alert when there are problems?

One problem that could arise with the data import is if the PunkAPI changes its schema or returns data in a different format. To mitigate this, we could implement error handling to detect any changes in the API response and modify the import process accordingly. Another problem could be if the data in the PunkAPI is inaccurate or incomplete. To mitigate this, we could implement data validation and cleaning routines to ensure that only valid and complete data is inserted into the database.

4. **Suppose that now we would like to know how the beer recipes are updated over time, so we would like to store a history of the data. How would you approach this? (No need to implement anything, but describe your solution).**

   To store a history of the beer recipe data, we could create a new table “beer_recipe_history” with additional timestamp column and relationship to “beers” and “hops”. Whenever a beer recipe is updated, a new row can be inserted into the “beer_recipe_history” table with the updated information and a timestamp. We could then query this table to retrieve the history of beer recipe changes over time. We could also implement versioning with data migration tool like Alembic.