**D205 Performance Assessment**

Shanay Murdock

[smurd32@wgu.edu](mailto:smurd32@wgu.edu)

D205 – Data Acquisition

Western Governors University

January 15, 2024

1. Research Question

The research question proposed for this performance assessment is *What age group requires the most amount of technical support service?* To determine this, the `customer\_id` and `age` columns were used from the customer table. `customer\_id` is of a text data type and `age` is of an integer data type. From the `service` table that was created, the `customer\_id` and `tech\_support` columns were used. The `customer\_id` was set up with a data type of VARCHAR(8) (although a data type of text could have also been used) and the `tech\_support` column was set up as a small integer type to restrict disk storage usage as the value would only be a `0` or a `1`. These columns were enough to join the tables together using the `customer\_id` column featured in both the `customer` and `service` tables while displaying the data needed to determine the ages of customers using the technical support service. The `customer` table doesn’t provide age groups as the research question intends; those were therefore created using a CASE statement in the SQL query.

1. ERD

The following SQL code was used to create the `service` table and add it to the entity relationship diagram to connect the tables.

-- Create services table

CREATE TABLE public.service (

id SERIAL PRIMARY KEY,

customer\_id VARCHAR(7) NOT NULL REFERENCES customer(customer\_id),

internet\_service VARCHAR(50),

phone SMALLINT NOT NULL,

multiple SMALLINT NOT NULL,

online\_security SMALLINT NOT NULL,

online\_backup SMALLINT NOT NULL,

device\_protection SMALLINT NOT NULL,

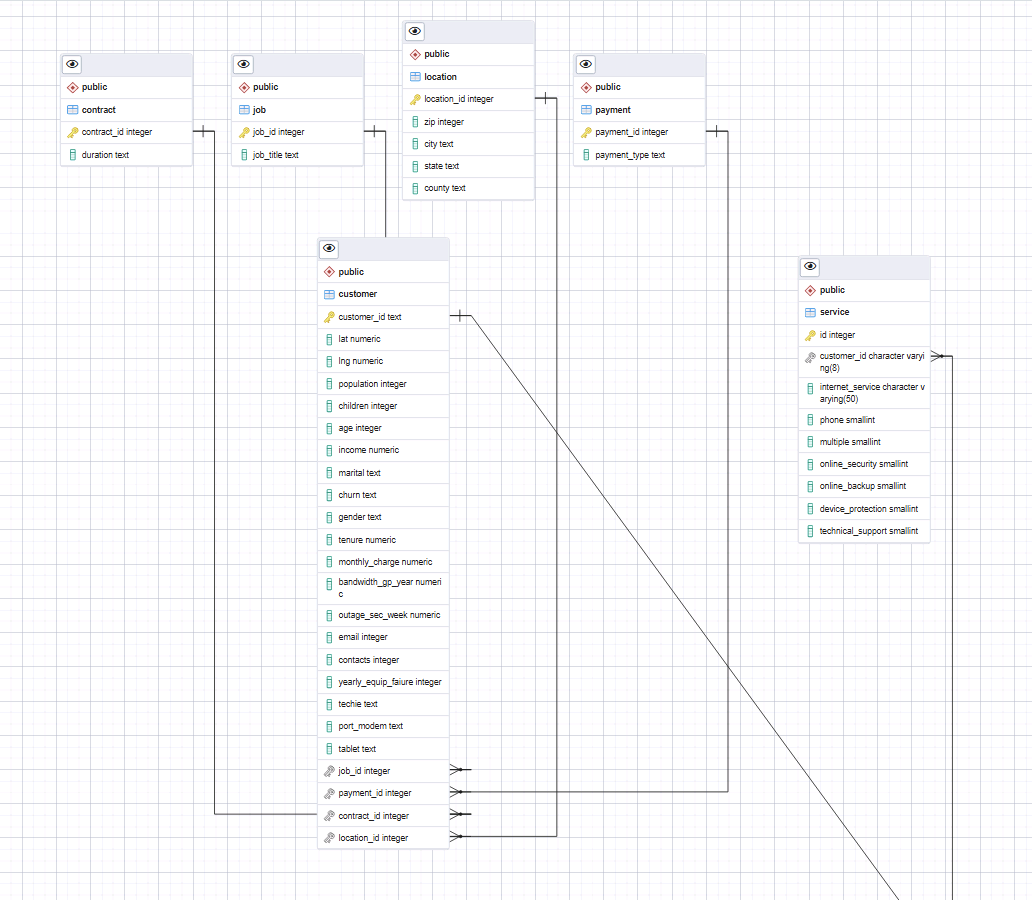
tech\_support SMALLINT NOT NULL

);

ALTER TABLE public.service

OWNER TO postgres;

The `Services\_cleaned.csv` file was updated so that the column headers used a traditional snake\_case styling convention. In addition, the `Yes` and `No` values used in the columns `phone`, `multiple`, `online\_security`, `online\_backup`, `device\_protection`, and `tech\_support` columns were converted to `1` for Yes values and `0` for No values so they would be easier to use aggregation functions. The `customer` table has a one-to-many relationship with the `service` table as a unique customer must appear in the `customer` table but the customer can be featured more than one time in the `service` table.



Small integer data types were used on the above columns to keep disk storage utilization minimal.

The following code was used to import the data from the `Services\_cleaned.csv` file that was cleaned for use in the table. This code imported the data to the `service` table as per the recommendation from Dr. Sewell’s “D205 Sunday Presentation” (Sewell, n.d.)

"\\copy public.service(customer\_id, internet\_service, phone, multiple, online\_security, online\_backup, device\_protection, technical\_support) FROM 'C:/LabFiles/SERVIC~1.CSV' DELIMITED','CSV HEADER QUOTE ]|'" ESCAPE "";""

1. Querying the data for the research question

The following code is used to answer the research question: *What age group requires the most amount of technical support service?*

SELECT SUM(technical\_support) AS total\_technical\_support,

CASE WHEN age < 25 THEN 'Under 25'

WHEN age BETWEEN 25 and 34 THEN '25-34'

WHEN age BETWEEN 35 AND 44 THEN '35-44'

WHEN age BETWEEN 45 AND 54 THEN '45-54'

WHEN age BETWEEN 55 AND 64 THEN '55-64'

WHEN age BETWEEN 65 AND 74 THEN '65-74'

ELSE 'Above 75'

END AS age\_range

FROM service

INNER JOIN customer

ON service.customer\_id = customer.customer\_id

GROUP BY age\_range

ORDER BY total\_technical\_support DESC;

Commonly used age groups in demographic studies were used to establish the age ranges used in the code. *Practical SQL: A beginner’s guide to storying telling with data* by Anthony Debarros provided a useful guide on setting up CASE statements featuring conditions (Debarros 2022). The output is provided in a separate CSV file, `query\_output.csv`.

1. Frequency of data update & query

This data would be helpful in determining departmental staffing needs. It would be recommended that this data be tracked, updated, and queried on a weekly basis or bi-weekly basis to assist with scheduling enough technicians to be able to provide technical support. In a secondary context, querying this data on a monthly or quarterly basis would be helpful if the company wishes to create any marketing campaigns around the technical support services available. Those could be used to target the largest existing group that requires technical support (‘Above 75’) or to target the group who is currently using technical support the least (‘Below 25’).

**References**

DeBarros, A. (2022). *Practical SQL a beginner’s guide to storytelling with data*. No Starch Press.

Sewell, W. (n.d.). *D205 SQL Sunday presentation*. Panopto. https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=96313931-aed9-4cdc-86fe-b02800e4f5df