In this project, our group decided to choose a dataset off of www.kaggle.com, which is an open-source repository for datasets. The dataset that caught our eye was one that contained information regarding sales in shopping males. Information included in this dataset is as follows: invoice number, date of purchase, customer id, gender, age, category (clothes, technology, toys, etc…), quantity, selling price per unit, cost price per unit, payment method, shopping mall name, and region and state of shopping mall. The group decided to use SQL PostgreSQL 9.6, via db-fiddle/sqlfiddle, to query the dataset and find information relevant to questions raised. We had originally thought to use MonetDB, however, we had some complications with the installation of such software. Furthermore, the group decided to create an ER-diagram to show the relationships of entities in the dataset, this can be seen below.

A diagram of a diagram

Description automatically generated

Unfortunately, due to the size of the dataset, we had to decrease some of the entries in order to get db-fiddle/sqlfiddle to work correctly, as these services can only contain up to eight thousand entries at a time. The following image shows the general schema of our database, and DDL statements used to populate the database.

A screenshot of a computer code

Description automatically generated

Some questions the group had regarding the dataset, and queries performed to figure out the answers, are as follows:

* **What type of item sold the most?**

***Query performed:***

select category, sum(quantity) as total from transactions

group by category

order by total desc

limit 1;

***Result:***

A white rectangular object with a black border

Description automatically generated with medium confidence

The result seen above shows that throughout all malls in this database, the most sold category of item was clothing items. This is the most logical outcome, as shopping malls are more tailored for selling clothing items as opposed to equipment and sporting goods items.

* **What type of item sold the least?**

***Query Performed:***

select category, sum(quantity) as total from transactions

group by category

order by total asc

limit 1;

***Result:***

***A white rectangular object with a black border

Description automatically generated with medium confidence***

* **What shopping mall sold the most items?**

***Query Performed:***

select shopping\_mall as mall, sum(quantity) as total

from transactions natural join malls

group by shopping\_mall

order by total desc

limit 1;

***Result:***

***A screenshot of a computer

Description automatically generated***

* **Which gender buys the most of which category of item?**

***Query Performed:***

WITH Totals AS (

SELECT

category,

gender,

SUM(quantity) AS total,

RANK() OVER (PARTITION BY category ORDER BY SUM(quantity) DESC) AS r

FROM transactions

NATURAL JOIN customer

GROUP BY category, gender

)

SELECT

category,

gender,

total

FROM Totals

WHERE r = 1;

***Result:***

***A screenshot of a computer

Description automatically generated***

* **Which category of items sold the most at each shopping mall?**

***Query Performed:***

WITH Categories\_per\_mall AS (

SELECT

shopping\_mall,

category,

SUM(quantity) AS total,

ROW\_NUMBER() OVER (PARTITION BY shopping\_mall ORDER BY SUM(quantity) DESC) AS r

FROM transactions

NATURAL JOIN malls

GROUP BY shopping\_mall, category

)

SELECT

shopping\_mall,

category,

total

FROM Categories\_per\_mall

WHERE r = 1;

***Result:***

A screenshot of a white box

Description automatically generated

* **What shopping mall is more expensive on average, per category?**

***Query Performed:***

with mall\_rank as (

select

shopping\_mall as mall,

category,

avg(cast(selling\_price\_per\_unit as numeric)) as asp,

rank() over (partition by category order by avg(cast(selling\_price\_per\_unit as numeric)) desc) as r

from

transactions

natural join malls

group by

mall,

category

)

select

mall,

category,

round(asp, 2) as average\_price

from

mall\_rank

where

r = 1

order by

category;

***Results:***

A screenshot of a white table

Description automatically generated

**Conclusion**

The group was able to find solutions to all questions presented and were satisfied with our results. Some information found in this project was eye opening, for instance, the only item males bought more of than females were toys. This project forced us to expand our current knowledge of PostgreSQL 9.6 further when trying to find answers to our questions above.