This dataset included customer profile, offer detail and a list of every transaction a customer received. I used SQL to dive into the dataset to uncover some probing questions & insights. These included:

- Male/Female Split.

- Age Group.

- Status of each offer.

- Which offer was sent out most.

- Which offer was completed most.

I first wanted to look at the customer profile - gender split and age groups.

select gender, round(sum(num)/14825\*100,0) as gen\_pct from

(select gender, count(gender) as num

from profile

group by gender)

group by gender, num

order by gen\_pct desc;

Findings: 57% Male, 41% Female, 2% Other.

select age\_group, sum(total) as total

from(

select age, count(age) as total,

case

when age between 18 and 30 then '18-30'

when age between 31 and 40 then '31-40'

when age between 41 and 50 then '41-50'

when age between 51 and 60 then '51-60'

when age between 61 and 70 then '61-70'

when age between 71 and 80 then '71-80'

when age between 81 and 90 then '81-90'

when age between 91 and 105 then '90+'

else 'error'

end as age\_group

from(

select customer, gender, age, count(event) as total from

        (select profile.id as customer, gender, age, event

from profile

left join transcript

on profile.id = transcript.person)

    group by customer, gender, age

order by total desc)

group by age

order by age desc)

group by age\_group

order by total desc;

Findings: Highest number of customers aged between 51-60yo, with lowest in 81+yo.

Next, I wanted to find the status of each offer and show as a percentage of total:

select event, round(sum(total)/306534\*100,0) as total\_pct from(

select event, count(event) as total

from transcript

group by event)

group by event

order by event desc;

Findings: 45% of offers had been transacted with 11% completed.

It was now time to focus on the offers, so I wanted to see which offer had been sent out the most:

select portfolio.id, count(portfolio.id) as total\_offer

from portfolio

join transcript

on transcript.value like concat('%',portfolio.id,'%')

group by portfolio.id

order by total\_offer desc;

Findings: Offer 'fafdcd668e3743c1bb461111dcafc2a4' had been sent out the most times - 20241 which earned the customer 2 rewards if completed. The least sent was an informational offer '3f207df678b143eea3cee63160fa8bed' - 11761 times.

Lastly, it was time to find out which offer had completed the most times:

select transcript.value, transcript.event, count(transcript.event) as total

from transcript

join portfolio

on transcript.value like concat('%',portfolio.id,'%')

where transcript.event = 'offer completed'

group by transcript.value, transcript.event

order by total desc;

Findings: Offer 'fafdcd668e3743c1bb461111dcafc2a4' was completed the most times - 5317. Offer '4d5c57ea9a6940dd891ad53e9dbe8da0' was completed least - 3331 times, however had a high difficulty and minimum spend.

This concludes my Starbucks case study! It was a fun dataset to work with and posed plenty of challenges - How to group the customer ages into age groups!? How to link portfolio id column to a string in the transcript value column?

The dataset gave me a great chance to practice and use, Joins, Subqueries, Aggregation, Case statements and much more.