SQL Queries

Google BigQuery used

Jobs Challenge

--Created the below tables by importing the provided CSVs in BigQuery

select \* from `Fotograf.Job statistics`;

select \* from `Fotograf.Job types`;

select \* from `Fotograf.Jobs`;

--Table created which joins jobs, job statistics and job types tables based on related columns between them

--jobs and job statistics join on Job\_ID

--jobs and job types on Job\_Type\_ID

create or replace table `Fotograf.Job-combined` as

select distinct j.Job\_Photographer\_ID, j.Job\_ID, j.Job\_Name, j.Job\_Guest\_Accesses\_Count\_Active, j.Job\_Type\_ID,

s.Job\_Statistic\_Count\_Login\_Unique, s.Job\_Statistic\_Count\_Order\_Paid, s.Job\_Statistic\_Total\_Revenue\_Paid, t.Job\_Type\_Name,

round((s.Job\_Statistic\_Count\_Login\_Unique/j.Job\_Guest\_Accesses\_Count\_Active),2) login\_rate,

round((s.Job\_Statistic\_Count\_Order\_Paid/j.Job\_Guest\_Accesses\_Count\_Active),2) order\_rate,

round((s.Job\_Statistic\_Total\_Revenue\_Paid/s.Job\_Statistic\_Count\_Order\_Paid),2) AOV,

round((s.Job\_Statistic\_Total\_Revenue\_Paid/j.Job\_Guest\_Accesses\_Count\_Active),2) revenue\_per\_head

from

`macro-magpie-365619.Fotograf.Jobs` j left join `macro-magpie-365619.Fotograf.Job statistics` s on j.Job\_ID=s.Job\_Statistic\_Job\_ID

left join `macro-magpie-365619.Fotograf.Job types` t on j.Job\_Type\_ID = t.Job\_Type\_ID order by 2 asc;

--Question 1. What job name had the lowest login rate?

select Job\_Name, login\_rate from `Fotograf.Job-combined` order by 2 asc limit 1;

--Question 2. What job name had the highest order rate?

select Job\_Name, order\_rate from `Fotograf.Job-combined` order by 2 desc limit 1;

--Question 3. What job type has the highest average AOV?

select Job\_Type\_Name, round(avg(AOV),2) avg\_AOV from `Fotograf.Job-combined` group by 1 order by 2 desc limit 1;

--Question 4. What job type has the highest average order rate?

select Job\_Type\_Name, round(avg(order\_rate),2) avg\_order\_rate from `Fotograf.Job-combined` group by 1 order by 2 desc limit 1;

--Job types and their revenues in desc order of revenues

select Job\_Type\_Name, round(sum(revenue\_per\_head),2) from `Fotograf.Job-combined` group by 1 order by 2 desc;

Photographer Onboarding Challenge

--Created the below tables by importing the provided CSVs in BigQuery

select \* from `Fotograf.invoice-settings`;

select \* from `Fotograf.job-statistics`;

select \* from `Fotograf.jobs-photographers`;

select \* from `Fotograf.photograhers`;

--Table created which joins photographers and invoice settings tables based on related columns between them

--they join on Photographer\_ID

create or replace table `Fotograf.photographer-combined` as

select Photographer\_ID, Photographer\_Country, Photographer\_Last\_Login, Photographer\_Created, Invoice\_Setting\_Photographer\_ID, Invoice\_Setting\_Disclaimer\_Agreed

from `Fotograf.photograhers` p join `Fotograf.invoice-settings` i on p.Photographer\_ID=i.Invoice\_Setting\_Photographer\_ID

order by 2,1 asc;

--Table created which joins job statistics and photographer jobs tables based on related columns between them

--they join on Photographer\_ID and Job\_ID

create or replace table `Fotograf.photographer job-combined` as

select Job\_Photographer\_ID, Job\_ID, Job\_Internal\_Number, Job\_Created\_at, Job\_Activated\_at, Job\_Statistic\_Total\_Revenue\_Paid

from `Fotograf.job-statistics` js right join `Fotograf.jobs-photographers` jp

on jp.Job\_Photographer\_ID=js.Job\_Statistic\_Photographer\_ID and jp.Job\_ID=js.Job\_Statistic\_Job\_ID

order by 1 asc;

--Table created which joins above created tables `Fotograf.photographer-combined` and `Fotograf.photographer job-combined` based on related columns between them

--they join on Photographer\_ID

--This table give photographer details like country, year they were created, year of last login, latest job details, total revenue needed for funnel

create or replace table `Fotograf.photographer\_data` as

select Photographer\_ID, Photographer\_Country, extract (year from Photographer\_Created) as year\_photographer\_created,

extract (year from Photographer\_Last\_Login) as year\_last\_login, Invoice\_Setting\_Disclaimer\_Agreed,

max(extract(year from Job\_Created\_at)) year\_job\_created, max(if(Job\_Internal\_Number="JOB00001","0","1")) Normal\_job,

max(extract(year from Job\_Activated\_at)) year\_job\_activated, sum(Job\_Statistic\_Total\_Revenue\_Paid) revenue\_paid

from `Fotograf.photographer-combined` pc left join `Fotograf.photographer job-combined` jc

on pc.Photographer\_ID=jc.Job\_Photographer\_ID

group by 1,2,3,4,5 order by 1 asc;

--Table for funnel

create or replace table `Fotograf.funnel` as

select 'total\_photograhers\_US&CA' as Stage, count(\*) At\_this\_Stage from `Fotograf.photographer\_data`

union all

select 'UScreatedthisyear' as Stage, count(\*) At\_this\_Stage from `Fotograf.photographer\_data` where Photographer\_Country = 'US' and year\_photographer\_created=2022

union all

select 'loggedin' as Stage, count(\*) At\_this\_Stage from `Fotograf.photographer\_data` where Photographer\_Country = 'US' and

year\_photographer\_created=2022 and year\_last\_login=2022

union all

select 'disclaimeragreed' as Stage, count(\*) At\_this\_Stage from `Fotograf.photographer\_data` where Photographer\_Country = 'US' and

year\_photographer\_created=2022 and year\_last\_login=2022 and Invoice\_Setting\_Disclaimer\_Agreed = 1

union all

select 'creatednormaljob' as Stage, count(\*) At\_this\_Stage from `Fotograf.photographer\_data` where Photographer\_Country = 'US' and

year\_photographer\_created=2022 and year\_last\_login=2022 and Invoice\_Setting\_Disclaimer\_Agreed = 1 and Normal\_job = '1'

union all

select 'activatednewjob' as Stage, count(\*) At\_this\_Stage from `Fotograf.photographer\_data` where Photographer\_Country = 'US' and year\_photographer\_created=2022 and

year\_last\_login=2022 and Invoice\_Setting\_Disclaimer\_Agreed = 1 and Normal\_job = '1' and year\_job\_activated = 2022

union all

select 'revenue500' as Stage, count(\*) At\_this\_Stage from `Fotograf.photographer\_data` where Photographer\_Country = 'US' and

year\_photographer\_created=2022 and year\_last\_login=2022 and Invoice\_Setting\_Disclaimer\_Agreed = 1 and Normal\_job = '1' and

year\_job\_activated = 2022 and revenue\_paid>=500

order by At\_this\_Stage desc;

Graphical user interface, text, application, email

Description automatically generated