*-- calculates all revenue (earned and unearned) from people taking flights back home*

SELECT SUM (F.BASE\_FARE\_REVENUE) AS "going back home revneue"

FROM fact\_reservations f

JOIN dim\_passenger dp

ON F.PASSENGER\_KEY = DP.passenger\_KEY

JOIN dim\_airport da

ON F.DESTINATION\_AIRPORT\_KEY = DA.AIRPORT\_KEY

JOIN dim\_city\_pair\_route dc

ON F.CITY\_PAIR\_ROUTE\_KEY = DC.CITY\_PAIR\_ROUTE\_KEY

WHERE DP.country = DA.country AND DC.DOM\_INTL\_INDICATOR = 'International';

A screenshot of a computer

Description automatically generated

*-- Fully Booked Flights 30 days before taking off, the airplane model for the flight, the airport it's heading to, and the revenue it generated*

SELECT FFR.FLIGHT\_NUMBER,

DA.MODEL,

DAP.AIRPORT\_NAME,

FFR.UNEARNED\_REVENUE

FROM fact\_flight\_revenue ffr

JOIN dim\_flight\_milestone dfm

ON FFR.FLIGHT\_MILESTONE\_KEY = DFM.FLIGHT\_MILESTONE\_KEY

JOIN dim\_aircraft da

ON FFR.AIRCRAFT\_KEY = DA.AIRCRAFT\_KEY

JOIN dim\_airport dap

ON FFR.DESTINATION\_AIRPORT\_KEY = DAP.AIRPORT\_KEY

WHERE DFM.FULLY\_BOOKED\_INDICATOR = 'Fully Booked'

AND DFM.DAYS\_BEFORE\_FLIGHT = 30;

A screenshot of a computer

Description automatically generated

*-- displays frequent flyer tier and status, base fare info, class of service flown info, and the measures related to one flight segment*

*-- filtered by home airport*

SELECT pp.TIER,

pp.FREQUENT\_FLYER\_STATUS,

fb.FARE\_TYPE,

fb.DESCRIPTION,

fb.DISCOUNT\_PERCENTAGE,

cas.CLASS\_PURCHASED,

cas.CLASS\_FLOWN,

cas.CLASS\_CHANGE\_INDICATOR,

fasa.BASE\_FARE\_REVENUE,

fasa.UPGRADE\_FEES,

fasa.SEGMENT\_MILES\_FLOWN,

fasa.SEGMENT\_MILES\_EARNED

FROM FACT\_FLIGHT\_ACTIVITY\_SEGMENT fasa

JOIN DIM\_PASSENGER\_PROFILE pp

ON fasa.PASSENGER\_PROFILE\_KEY = pp.PROFILE\_KEY

JOIN DIM\_FARE\_BASIS fb

ON fasa.FARE\_BASIS\_KEY = fb.FARE\_BASIS\_KEY

JOIN DIM\_CLASS\_OF\_SERVICE cas

ON fasa.CLASS\_OF\_SERVICE\_FLOWN\_KEY = cas.CLASS\_OF\_SERVICE\_KEY

WHERE PP.HOME\_AIRPORT = 'Texas';

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

*-- calculates the average duration of overnight stays for passengers grouped by their frequent flyer status, and*

*-- displays what percentage of all flyers customers in this tier account for*

SELECT dp.frequent\_flyer\_status,

AVG (faf.Overnight\_Stay\_Duration) AS avg\_overnight\_stay\_duration,

ROUND ( (COUNT (1) / SUM (COUNT (1)) OVER ()) \* 100, 2)

AS tier\_perc\_of\_total\_flyers

FROM Fact\_flight\_activity\_trip faf

JOIN

dim\_passenger\_profile dp

ON faf.Passenger\_Profile\_Key = dp.profile\_key

GROUP BY dp.frequent\_flyer\_status;

A screenshot of a computer

Description automatically generated

*--what proportion of frequent flyers that upgrade have gold, platinum or titanium status.*

SELECT ROUND (

SUM (

CASE

WHEN DCP.FREQUENT\_FLYER\_STATUS IN

('Gold', 'Platinum', 'Titanium')

THEN

1

ELSE

0

END)

/ SUM (1),

2)

AS perc\_of\_gold\_plat\_titanum

FROM fact\_passenger\_tier fpt

JOIN

dim\_tier\_change\_profile dcp

ON DCP.TIER\_CHANGE\_KEY = FPT.TIER\_CHANGE\_KEY

WHERE DCP.IS\_UPGRADE = 'Y';

A close-up of a computer screen

Description automatically generated

*-- how they earn and redeem their frequent flyer points*

SELECT DPT.CATEGORY,

DPT.SUB\_CATEGORY,

DPT.COMPANY,

SUM (

CASE

WHEN dpt.transaction\_type = 'earn' THEN points\_total

ELSE 0

END)

points\_earned\_from,

SUM (

CASE

WHEN dpt.transaction\_type = 'redeem' THEN points\_total

ELSE 0

END)

points\_spent\_on

FROM fact\_points\_transactions fpt

JOIN

dim\_points\_transaction\_profile dpt

ON FPT.POINTS\_TRANSACTION\_PROFILE\_KEY = DPT.POINTS\_ACTIVITY\_KEY

GROUP BY DPT.CATEGORY, DPT.SUB\_CATEGORY, DPT.COMPANY;

A screenshot of a computer

Description automatically generated

*--Tracking passenger complaints for each month in 2019*

SELECT dd.month,

SUM (CASE WHEN DIT.INTERACTIONTYPE = 'Complaint' THEN 1 ELSE 0 END)

AS no\_of\_complaints

FROM fact\_customer\_interaction fci

JOIN dim\_date dd

ON FCI.DATE\_KEY = dd.date\_key

JOIN DIM\_INTERACTION\_TYPE dit

ON DIT.INTERACTIONTYPE\_KEY = FCI.INTERACTIONTYPE\_KEY

WHERE dd.year = 2019

GROUP BY dd.month, EXTRACT (MONTH FROM DD.FULL\_DATE)

ORDER BY EXTRACT (MONTH FROM DD.FULL\_DATE)

A screenshot of a computer

Description automatically generated