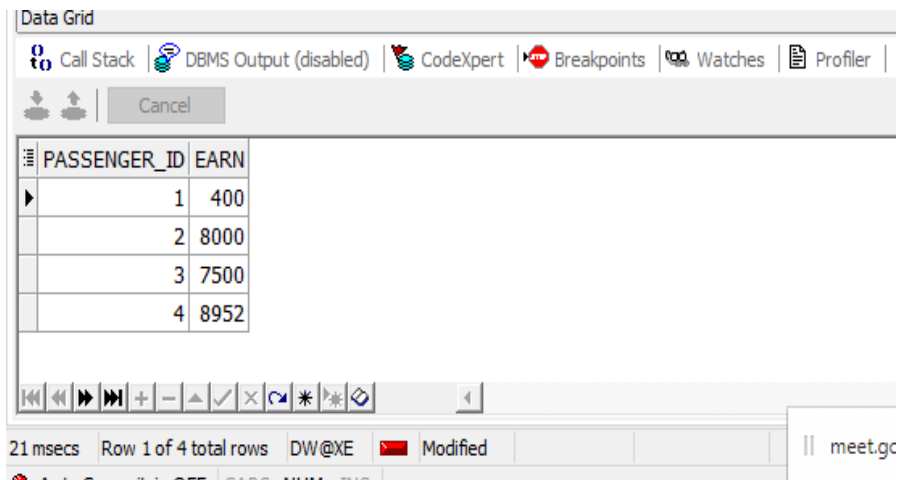


SQL QUERIES

-- [1] how much miles they earn

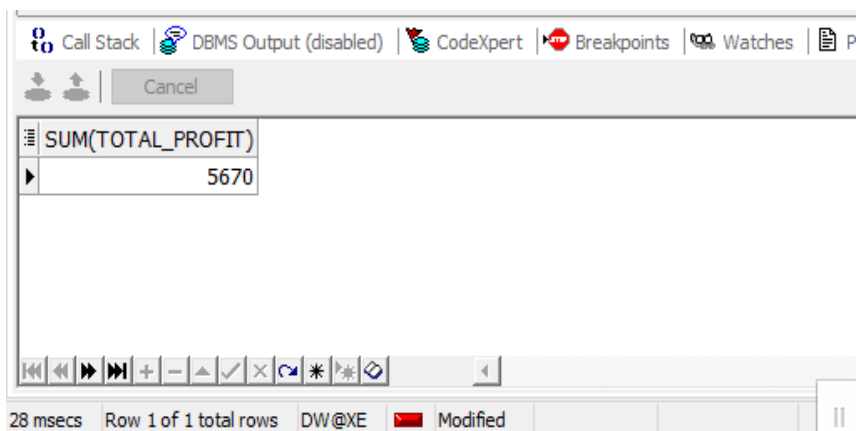
```
SELECT PASSENGER_ID, sum(no_miles) AS EARN
FROM FLIGHT_ACTIVITY_fact
GROUP BY PASSENGER_ID
order by passenger_id;
```



PASSENGER_ID	EARN
1	400
2	8000
3	7500
4	8952

--[2] company's profit

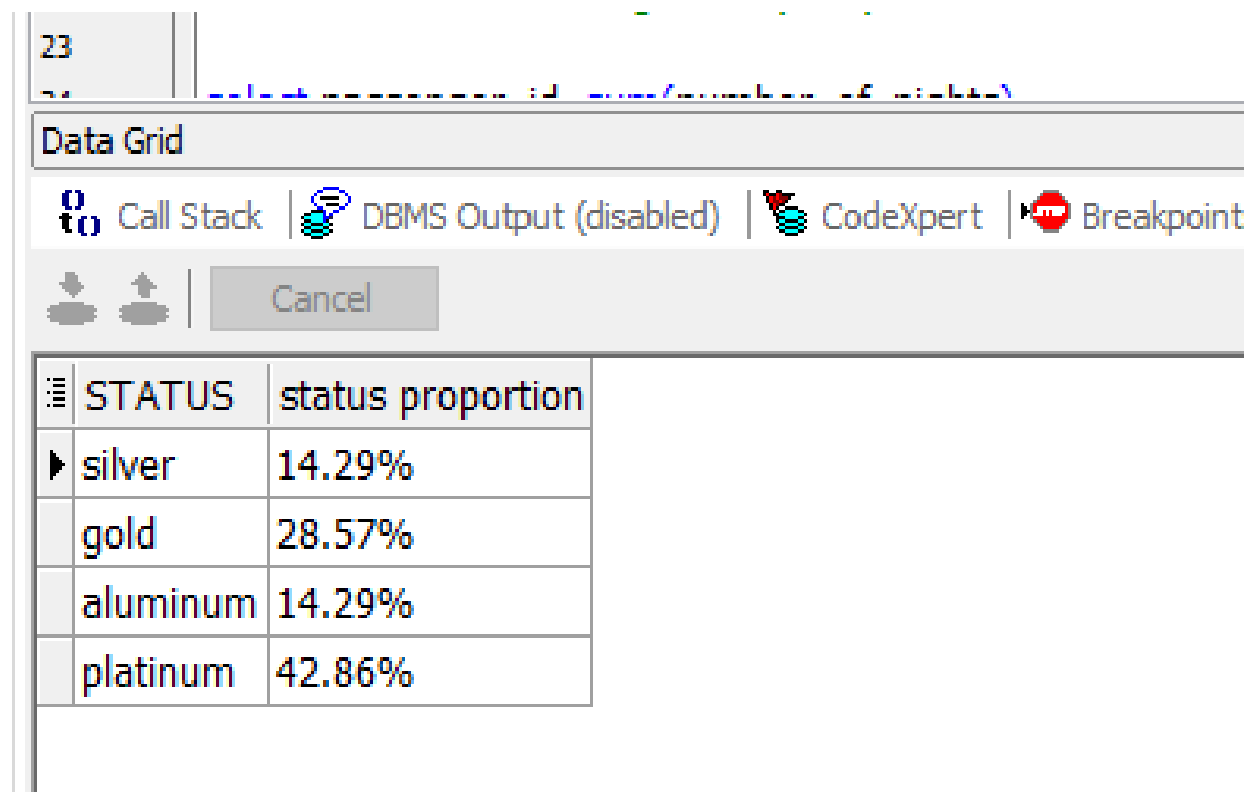
```
select sum(total_profit)
from flight_activity_fact;
```



SUM(TOTAL_PROFIT)
5670

--[3] what proportion of these frequent flyers have gold, platinum, aluminum, or titanium status.

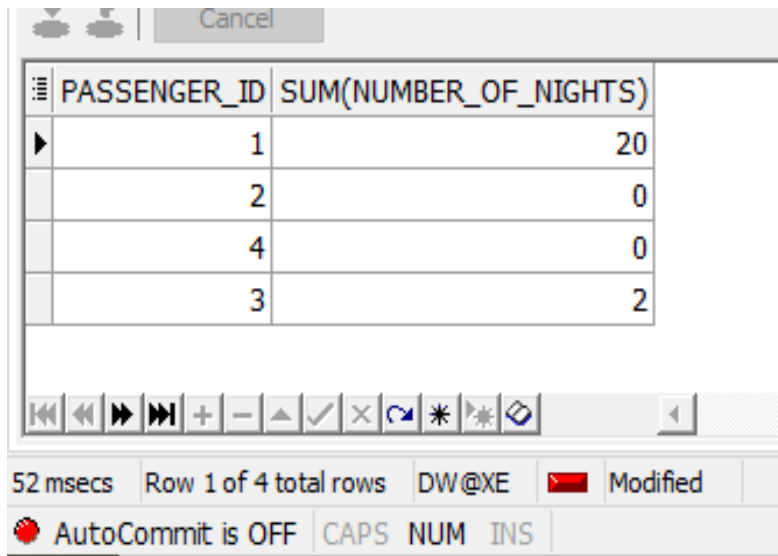
```
SELECT STATUS,round (COUNT(STATUS)/ (SELECT COUNT(*) FROM passenger_dim)
*100,2)|| '%' as "status proportion"
FROM passenger_dim
GROUP BY STATUS
```



STATUS	status proportion
silver	14.29%
gold	28.57%
aluminum	14.29%
platinum	42.86%

--[4] total number of nights they stay in hotels

```
select passenger_id, sum(number_of_nights)
from overnight_stay_fact
group by passenger_id;
```

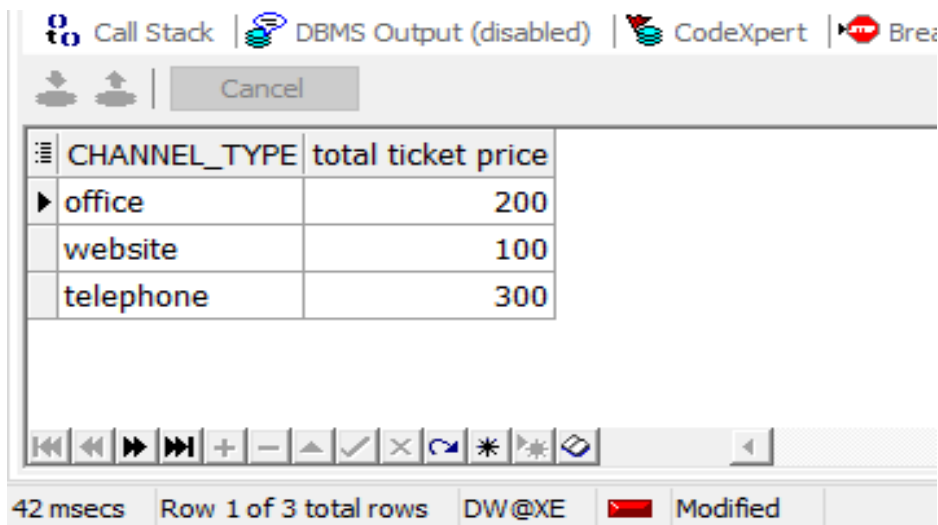


A screenshot of a SQL query result window. The window has a 'Cancel' button at the top. Below it is a table with two columns: 'PASSENGER_ID' and 'SUM(NUMBER_OF_NIGHTS)'. The table contains four rows of data. Below the table is a toolbar with various navigation and editing icons. At the bottom, there is a status bar showing '52 msec', 'Row 1 of 4 total rows', 'DW@XE', a red flag icon, and 'Modified'. Below the status bar, there is a row of icons and the text 'AutoCommit is OFF', 'CAPS', 'NUM', and 'INS'.

PASSENGER_ID	SUM(NUMBER_OF_NIGHTS)
1	20
2	0
4	0
3	2

--[5] total ticket price per channel

```
SELECT channel_type, SUM(ticket_price) AS "total ticket price"
FROM channel_dim, RESERVATION_fact
WHERE channel_dim.channel_id=RESERVATION_fact.channel_id
GROUP BY channel_type
ORDER BY 'total ticket price' DESC
```











A screenshot of a SQL query result window. The window has a 'Cancel' button at the top. Below it is a table with two columns: 'CHANNEL_TYPE' and 'total ticket price'. The table contains three rows of data. Below the table is a toolbar with various navigation and editing icons. At the bottom, there is a status bar showing '42 msec', 'Row 1 of 3 total rows', 'DW@XE', a red flag icon, and 'Modified'.

CHANNEL_TYPE	total ticket price
office	200
website	100
telephone	300

--[6] number and type of interactions per flight

```
SELECT cc.FLIGHT_id, INTERACTION_TYPE, count(*) as NUM_OF_INTERACTIONS
FROM customer_care_fact cc,
     date_dim d,
     flight_dim F,
     passenger_dim p
WHERE cc.DATE_id = d.DATE_id
      AND cc.Passenger_id = P.Passenger_id
      AND cc.FLIGHT_id = F.FLIGHT_id
      group by cc.FLIGHT_id, INTERACTION_TYPE;
```

Data Grid			
 Call Stack  DBMS Output (disabled)  CodeXpert  Breakpoints  Watches  Profiler			
  <input type="button" value="Cancel"/>			
	FLIGHT_ID	INTERACTION_TYPE	NUM_OF_INTERACTIONS
▶	2	within	1
	1	after	2