Sixth hands-on exercise (answers)

Task 1: In the car retailer database, what is the name of the customer who generated the most profit for the company? ***Assume*** all the orders were accepted and shipped with no cancellation!

```
alter table orderdetails add column customerNumber int;

update orderdetails join orders on  orders.orderNumber =
orderdetails.orderNumber set orderdetails.customerNumber =
orders.customerNumber;

alter table orderdetails add column cost decimal(10,2);

UPDATE orderdetails
JOIN products ON products.productCode = orderdetails.productCode SET
orderdetails.cost = products.buyPrice;

select customerNumber, round(sum((priceEach-cost)*quantityOrdered),2) as
profit from orderdetails group by customerNumber
order by profit desc;
```

A simple code to solve the above problem by joining three tables.

```
select tb.customerNumber, c.customerName,c.contactLastName,
c.contactFirstName, sum((o.priceEach - p.buyPrice)*o.quantityOrdered) AS
profit

from orderdetails o
join orders tb on o.orderNumber = tb.orderNumber
join products p on o.productCode = p.productCode
JOIN customers c ON tb.customerNumber = c.customerNumber

group by tb.customerNumber ORDER BY profit desc
```

From the table customers, you can the name of customerNumber of 141 (Freyre, Diego)

Task 3: In TripAdvisor dataset, which location (locality) receives the largest amount of hotel reviews and which one receives the lowest amount of reviews?

- the calculation needs to be based on two tables [tripadvisor_review_sample_without_reviewtext.sql] and [tripadivsor_hotel_sample.sql]
- in [tripadvisor_review_sample_without_reviewtext] table, you can find the reviews for each hotel.
- in [tripadivsor_hotel_sample] table, you can find the locality for each hotel

```
select b.locality, count(a.id) as fre from
tripadvisor_review_sample_without_reviewtext a join tripadivsor_hotel_sample b
on a.hotel_id = b.hotel_id group by b.locality order by fre desc
```

Task 4: In TripAdvisor dataset, let's say that the review samples collected from location ('locality' of hotel) with less than 700 reviews are not worth to be included for further analysis, and we want to drop them. So please create a new table [name: new_data] that has a structure similar to "tripadvisor_review_sample_without_reviewtext" table, but only include the reviews from the locations ['locality'] that received over 700 reviews.

```
create table temp as (select b.locality, count(a.id) as fre from
tripadvisor_review_sample_without_reviewtext a join tripadivsor_hotel_sample
b on a.hotel_id = b.hotel_id group by b.locality having fre > 200);
```

```
create table new_data as (select * from
tripadvisor_review_sample_without_reviewtext where hotel_id in (select
hotel_id from tripadivsor_hotel_sample where locality in (select locality
from temp)));

drop table temp;
```

- What you would do, if it took a long time for your computer to run the query?!

Task 5: Based on the table "tripadvisor_review_sample_without_reviewtext", please create a new table [name: new_data2] including 1000 rows of reviews for hotels in "New York city" and 500 rows of reviews for hotels in "Chicago".

```
create table new_data2 (select * from
tripadvisor_review_sample_without_reviewtext  where hotel_id in (select
hotel_id from tripadivsor_hotel_sample where locality = 'New York city')
limit 1000)
union
(select * from tripadvisor_review_sample_without_reviewtext  where hotel_id
in (select hotel_id from tripadivsor_hotel_sample where locality = 'Chicago')
limit 500)
```

Task 6: In the "tripadvisor_data_for_handson_assignment_ONLY" dataset, please calculate the **number** of the reviews that hotels received, and only consider those hotels that received reviews having at least one review of an overall rating of 5, 3 and 1, respectively. If all the reviews given to a hotel XXX ONLY having an overall rating of either 4 or 5, it will be excluded, because it does not have any reviews offering an overall rating of 3 or 1. Please obtain following result in one command.

tripadvisor_data_for_handson_assignment_ONLY (4×20)

hotel_id	Num_of_5_star_rating	Num_of_3_star_rating	Num_of_1_star_rating
75,737	4	1	1
81,192	8	5	1
93,437	6	1	1
93,450	4	4	2
93,517	2	1	2
93,520	11	4	1

```
select tb1.*, tb2.Num_of_3_star_rating, tb3.Num_of_1_star_rating from (select
hotel_id, count(*) as Num_of_5_star_rating from
tripadvisor_data_for_handson_assignment_ONLY where overall_rating = 5 group
by hotel_id ) tb1
join
(select hotel_id,  count(*) as Num_of_3_star_rating from
tripadvisor_data_for_handson_assignment_ONLY where overall_rating = 3 group
by hotel_id ) tb2
on tb1.hotel_id = tb2.hotel_id
join
(select hotel_id, count(*) as  Num_of_1_star_rating from
tripadvisor_data_for_handson_assignment_ONLY where overall_rating = 1 group
by hotel_id ) tb3
on tb3.hotel_id = tb2.hotel_id
```

Task 7: Based on the task 6, think about how to do the task, if a hotel that did not receive any rating of 5, 3 or 1 will still be included in the result. By learning the ifnull (see. https://www.jquery-az.com/mysql-ifnull/) function by yourself, you will figure out the solution.

```
select tb0.hotel_id, ifnull(tb1.Num_of_5_star_rating,0),
ifnull(tb2.Num_of_3_star_rating,0), ifnull(tb3.Num_of_1_star_rating,0) from
(select hotel_id, count(*) as total_number from
tripadvisor_data_for_handson_assignment_ONLY group by hotel_id ) tb0

left join

(select hotel_id, count(*) as Num_of_5_star_rating from
tripadvisor_data_for_handson_assignment_ONLY where overall_rating = 5 group by
hotel_id ) tb1

on tb0.hotel_id = tb1.hotel_id

left join

(select hotel_id, count(*) as Num_of_3_star_rating from
tripadvisor_data_for_handson_assignment_ONLY where overall_rating = 3 group by
hotel_id ) tb2

on tb0.hotel_id = tb2.hotel_id

left join

(select hotel_id, count(*) as Num_of_1_star_rating from
tripadvisor_data_for_handson_assignment_ONLY where overall_rating = 1 group by
hotel_id ) tb3
```

```
on tb0.hotel_id = tb3.hotel_id
```

Task 8: In the "tripadvisor_data_for_handson_assignment_ONLY" dataset, please calculate the **proportions** of reviews giving an overall rating of 5 and 3 and 1 respectively, and only consider those hotels that received reviews having at least one review of an overall rating of 5, 3 and 1, respectively. Excluding those hotels with less than 10 reviews from the analysis. Please obtain following result via one command.

tripadvisor_data_for_handson_assignment_ONLY (5×17)

hotel_id	Num_of_total_review	ratio_of_5_star_rating	ratio_of_3_star_rating	ratio_of_1_star_rating
81,192	20	0.4000	0.2500	0.0500
93,437	15	0.4000	0.0667	0.0667
93,450	18	0.2222	0.2222	0.1111
93,517	11	0.1818	0.0909	0.1818
93,520	20	0.5500	0.2000	0.0500
93,545	13	0.0769	0.2308	0.0769

```
select tb1.hotel_id, tb4.Num_of_total_review,
tb1.Num_of_5_star_rating/tb4.Num_of_total_review as ratio_of_5_star_rating,
tb2.Num_of_3_star_rating/tb4.Num_of_total_review as ratio_of_3_star_rating,
tb3.Num_of_1_star_rating/tb4.Num_of_total_review as ratio_of_1_star_rating
from
(select hotel_id, count(*) as Num_of_5_star_rating from
tripadvisor_data_for_handson_assignment_ONLY where overall_rating = 5 group
by hotel_id ) tb1
join
(select hotel_id, count(*) as  Num_of_total_review from
tripadvisor_data_for_handson_assignment_ONLY  group by hotel_id ) tb4
on tb1.hotel_id = tb4.hotel_id
join
(select hotel_id,  count(*) as Num_of_3_star_rating from
tripadvisor_data_for_handson_assignment_ONLY where overall_rating = 3 group
by hotel_id ) tb2
on tb1.hotel_id = tb2.hotel_id
join
(select hotel_id, count(*) as  Num_of_1_star_rating from
tripadvisor_data_for_handson_assignment_ONLY where overall_rating = 1 group
by hotel_id ) tb3
on tb3.hotel_id = tb2.hotel_id where tb4.Num_of_total_review >= 10
```