/\*solutions for case study 2. All challenges can be found here https://8weeksqlchallenge.com/case-study-3/

1. How many pizzas were ordered?

2. How many unique customer orders were made?

3. How many successful orders were delivered by each runner?

4. How many of each type of pizza was delivered?

5. How many Vegetarian and Meatlovers were ordered by each customer?

6. What was the maximum number of pizzas delivered in a single order?

7. For each customer, how many delivered pizzas had at least 1 change and how many had no changes?

8. How many pizzas were delivered that had both exclusions and extras?

9. What was the total volume of pizzas ordered for each hour of the day?

10. What was the volume of orders for each day of the week?

\*/

select

--1

max(

select count(customer\_orders.order\_id)

from customer\_orders

) as num\_of\_pizzas,

--10

day,

max(day\_count) as day\_count,

--9

hours,

max(hours\_count) as hours\_count,

--7

max(changes) as changes,

--7.2

max(no\_changes) as no\_changes,

--8

max(both\_changes) as both\_changes,

--2

max(

select count(distinct order\_id)

from customer\_orders

) as num\_of\_orders,

--3

max(

select count(order\_id) - ( select count(cancellation) from runner\_orders where cancellation like '%Restaurant%' or cancellation like '%Customer%')

from runner\_orders

) as successful\_delivered,

customer\_orders.customer\_id,

--4

max(num\_of\_pizzatype) as num\_of\_pizzatype,

--6 wie kann der maximalwert eines count-ergebnisses ausgwählt werden? ->order by & limit

max(

select max\_pizza\_per\_order

from(

select

count(order\_id) as max\_pizza\_per\_order

from customer\_orders

group by customer\_id

order by max\_pizza\_per\_order desc

)

limit 1

) as max\_pizza\_per\_order,

--5

count(customer\_orders.pizza\_id) as num\_of\_pizzatype\_per\_cus,

pizza\_name

from customer\_orders

join pizza\_names

on customer\_orders.pizza\_id = pizza\_names.pizza\_id

--4

join(

select

count(customer\_orders.pizza\_id) as num\_of\_pizzatype,

pizza\_id

from customer\_orders

group by pizza\_id

) as temp1

on customer\_orders.pizza\_id = temp1.pizza\_id

--7

left join(

select count(customer\_orders.customer\_id) as changes, customer\_orders.customer\_id

from customer\_orders

where (exclusions not like '%null%'

and LENGTH(TRIM(exclusions)) > 0)

or (extras not like '%null%'

and length(trim(extras)) > 0)

group by customer\_orders.customer\_id

) as temp2

on customer\_orders.customer\_id = temp2.customer\_id

-----------------------------------------------------------

--7.2 testen von leeren zellen mit length(trim(column)) > 0

-----------------------------------------------------------

left join(

select count(customer\_orders.customer\_id) as no\_changes, customer\_orders.customer\_id

from customer\_orders

where

(exclusions like '%null%'

and extras like '%null%')

or (LENGTH(TRIM(exclusions)) = 0

and LENGTH(TRIM(extras)) = 0

)

or (LENGTH(TRIM(exclusions)) = 0

and extras like '%null%' = 0

)

or (LENGTH(TRIM(extras)) = 0

and exclusions like '%null%'

)

group by customer\_orders.customer\_id

) as temp3

on customer\_orders.customer\_id = temp3.customer\_id

--8

left join(

select count(customer\_orders.customer\_id) as both\_changes,

customer\_orders.customer\_id

from customer\_orders

where

exclusions not like '%null%'

and extras not like '%null%'

and LENGTH(TRIM(exclusions)) >0

and LENGTH(TRIM(extras)) > 0

group by customer\_orders.customer\_id

) as temp4

on customer\_orders.customer\_id = temp4.customer\_id

--10

join(

select dayname(order\_time) as day,

count(dayname(order\_time)) as day\_count

from customer\_orders

group by day

)

--9 auswählen eines string abschnittes mit split\_part(coloumn, dilimeter, ' of part')

join(

select split\_part(hours, ':',1) as hours,

count(split\_part(hours, ':',1)) as hours\_count

from customer\_orders

join(

--

select split\_part(temp6.order\_time, ' ',2) as hours,

customer\_orders.order\_time

from customer\_orders

join(

select count(order\_time) as count,

order\_time

from customer\_orders

group by order\_time

)

as temp6

on customer\_orders.order\_time = temp6.order\_time

group by customer\_orders.order\_time, temp6.order\_time

--

) as temp5

on customer\_orders.order\_time = temp5.order\_time

group by hours

order by hours

) as hours

group by pizza\_name, customer\_orders.customer\_id, day, hours;