```
-- USE zomato
-- SELECT COUNT(*) FROM order_details

-- replicated sample function from pandas
-- SELECT * FROM users ORDER BY rand() LIMIT 5

-- To find the NULL values
-- SELECT * FROM orders WHERE restaurant_rating IS NULL

-- To replace NULL values with 0
-- UPDATE orders SET restaurant_rating = 0
-- WHERE restaurant_rating IS NULL

-- Q5
-- SELECT t2.name,COUNT(*) AS '#orders' FROM orders t1
-- JOIN users t2
-- ON t1.user_id = t2.user_id
-- GROUP BY t2.user_id

-- Q6
-- SELECT r_name,COUNT(*) AS 'menu_items' FROM restaurants t1
-- JOIN menu t2
-- ON t1.r_id = t2.r_id
-- GROUP BY t2.r_id

-- Q7
SELECT r_name,COUNT(*) AS 'num_votes',ROUND(AVG(restaurant_rating),2) AS 'rating'
FROM orders t1
JOIN restaurants t2
ON t1.r_id = t2.r_id
WHERE restaurant_rating IS NOT NULL
GROUP BY t1.r_id;

-- Q8
SELECT f_name,COUNT(*) FROM menu t1
JOIN food t2
ON t1.f_id = t2.f_id
GROUP BY t1.f_id
ORDER BY COUNT(*) DESC LIMIT 1;

-- Q9 -> May
-- SELECT MONTHNAME(DATE(date)),date FROM orders
SELECT r_name,SUM(amount) AS 'revenue' FROM orders t1
JOIN restaurants t2
```

```
ON t1.r_id = t2.r_id
WHERE MONTHNAME(DATE(date)) = 'July'
GROUP BY t1.r_id
ORDER BY revenue DESC LIMIT 1;

-- month by month revenue for a particular restautant = kfc
SELECT MONTHNAME(DATE(date)),SUM(amount) AS 'revenue' FROM orders t1
JOIN restaurants t2
ON t1.r_id = t2.r_id
WHERE r_name = 'box8'
GROUP BY MONTHNAME(DATE(date))
ORDER BY MONTH(DATE(date));

-- Q10
SELECT r_name,SUM(amount) AS 'revenue' FROM orders t1
JOIN restaurants t2
ON t1.r_id = t2.r_id
GROUP BY t1.r_id
HAVING revenue > 1500;

-- Q11
SELECT user_id,name FROM users
EXCEPT
SELECT t1.user_id,name FROM orders t1;

-- Q12
SELECT t1.order_id,f_name,date FROM orders t1
JOIN order_details t2
ON t1.order_id = t2.order_id
JOIN food t3
ON t2.f_id = t3.f_id
WHERE user_id = 5 AND date BETWEEN '2022-05-15' AND '2022-07-15';

-- Q13
SELECT t1.user_id,t3.f_id,COUNT(*) FROM users t1
JOIN orders t2
ON t1.user_id = t2.user_id
JOIN order_details t3
ON t2.order_id = t3.order_id
GROUP BY t1.user_id,t3.f_id
ORDER BY COUNT(*) DESC;

-- Q14
SELECT r_name,SUM(price)/COUNT(*) AS 'Avg_price' FROM menu t1
```

```
JOIN restaurants t2
ON t1.r_id = t2.r_id
GROUP BY t1.r_id
ORDER BY Avg_price ASC LIMIT 1;

-- Q15
SELECT partner_name,COUNT(*) * 100  + AVG(delivery_rating)*1000 AS 'salary'
FROM orders t1
JOIN delivery_partner t2
ON t1.partner_id = t2.partner_id
GROUP BY t1.partner_id
ORDER BY salary DESC;

-- Q17
-- SELECT CORR(delivery_time,delivery_rating) AS 'corr'
-- FROM orders;

-- Q19
SELECT r_name FROM menu t1
JOIN food t2
ON t1.f_id = t2.f_id
JOIN restaurants t3
ON t1.r_id = t3.r_id
GROUP BY t1.r_id
HAVING MIN(type) = 'Veg' AND MAX(type) = 'Veg';

-- Q 20
SELECT name,MIN(amount),MAX(amount),AVG(amount) FROM orders t1
JOIN users t2
ON t1.user_id = t2.user_id
GROUP BY t1.user_id
```