Q-1 . Write an SQL query that reports the products that were only sold in the first quarter of 2019. That is,between 2019-01-01 and 2019-03-31 inclusive. TABLE- Productt and Saled.

	product_id	product_name	unit_price
1	1	S8	1000
2	2	G4	800
3	3	iPhone	1400

	seller_id	product_id	buyer_id	sale_date	quantity	price
1	1	1	1	2019-01-21	2	2000
2	1	2	2	2019-02-17	1	800
3	2	2	3	2019-06-02	1	800
4	3	3	4	2019-05-13	2	2800

```
select p.product_id, product_name from Productt p join Saled s on p.product_id=s.product_id
where sale_date >='2019-01-01' and sale_date<= '2019-03-31'
except select p.product_id, product_name from Productt p join Saled s on p.product_id=s.product_id
where sale_date>'2019-03-31'or sale_date<'2019-01-01'
```

Output:

	product_id	product_name
1	1	S8

Q-2 . Write an SQL query to find all the authors that viewed at least one of their own articles. Return the table - 'Viewss' sorted by id in ascending order.

	article_id	author_id	viewer_id	view_date
1	1	3	5	2019-08-01
2	1	3	6	2019-08-02
3	2	7	7	2019-08-01
4	2	7	6	2019-08-02
5	4	7	1	2019-07-22
6	3	4	4	2019-07-21
7	3	4	4	2019-07-20

Output:

author_id
4
7

```
select distinct author_id  from viewss where author_id=viewer_id
```

Q-3 . If the customer's preferred delivery date is same as order date, the the order is called immediately , otherwise scheduled. .Write an SQL query to find the percentage of immediate orders in the table- 'Delivery' rounded to 2 decimal places.

delivery_id	customer_id	order_date	customer_pref_delivery_date
1	1	2019-08-01	2019-08-02
2	5	2019-08-02	2019-08-02
3	1	2019-08-11	2019-08-11
4	3	2019-08-24	2019-08-26
5	4	2019-08-21	2019-08-22
6	2	2019-08-11	2019-08-13

Output:

(No column name)
33.33

```
select cast(round(count(case when order_date=customer_pref_delivery_date then 1 end) *100.00 /count(*),2) as decimal (10,2))
from Delivery
```

Q-4 . Write an SQL query to find CTR of each ad, round it to 2 decimal places.

$$CTR = \begin{cases} 0, & \text{if Ad total clicks + Ad total views = 0} \\ \frac{\text{Ad total clicks}}{\text{Ad total clicks + Ad total views}} \times 100, & \text{otherwise} \end{cases}$$

ad_id	user_id	action
1	1	Clicked
2	2	Clicked
3	3	Viewed
5	5	Ignored
1	7	Ignored
2	7	Viewed
3	5	Clicked
1	4	Viewed
2	11	Viewed
1	2	Clicked

Output:

ad_id	(No column name)
1	66.67
2	33.33
3	50.00
5	NULL

```
select ad_id, cast(100.00*c.clicks/nullif( c.clicks + c.viewss,0)as decimal(10,2)) from(
select ad_id,sum(case when action ='Clicked' then 1 else 0 end)  as clicks,sum(case when action ='Viewed' then 1  else 0 end)
as viewss from AdActions group by ad_id)as c
```

Q-5 . Write an SQL query to find team size of each employees from Employeee table.

employee_id	team_id
1	8
2	8
3	8
4	7
5	9
6	9

Output:

employee_id	Size
1	3
2	3
3	3
4	1
5	2
6	2

```
select employee_id, count(team_id) over(partition by team_id) as Size from employeee order by employee_id
```

Q-6

Write an SQL query to find the type of weather in each country for November 2019.The type of weather is:

- Cold if the average weather_state is less than or equal 15,
- Hot if the average weather_state is greater than or equal to 25, and
- Warm otherwise.

Table – Country and Weatherr are given below:

country_id	country_name
2	USA
3	Australia
7	Peru
5	China
8	Morocco
9	Spain

country_id	weather_state	Day
2	15	2019-11-01
2	12	2019-10-28
2	12	2019-10-27
3	-2	2019-11-10
3	0	2019-11-11
3	3	2019-11-12
5	16	2019-11-07
5	18	2019-11-09
5	21	2019-11-23
7	25	2019-11-28
7	22	2019-12-01
7	20	2019-12-02
8	25	2019-11-05
8	27	2019-11-15
8	31	2019-11-25
9	7	2019-10-23
9	3	2019-12-23

Output:

country_name	(No column name)
Australia	Cold
China	Warm
Morocco	Hot
Peru	Hot
USA	Cold

```
select country_name, case when avg(weather_state)<=15 then 'Cold' when avg(weather_state)>=25 then 'Hot' else 'Warm' end
from COUNTRY c join Weatherr w on c.country_id=w.country_id where month(DAY)=11 group by country_name
```

Q-7 Table- Prices & UnitSold

Write an SQL query to find the average selling price for each product. average_price should be rounded to 2 decimal places.

product_id	start_date	end_date	Price
1	2019-02-17	2019-02-28	5
1	2019-03-01	2019-03-22	20
2	2019-02-01	2019-02-20	15
2	2019-02-21	2019-03-31	30

product_id	purchase_date	units
1	2019-02-25	100
1	2019-03-01	15
2	2019-02-10	200
2	2019-03-22	30

Output:

product_id	Average_price
1	6.96
2	16.96

```
select p.product_id,cast(sum(price*units)/cast(sum(units)as decimal(10,2))as decimal(10,2)) Average_price
from Prices p join UnitsSold u on p.product_id=u.product_id
where u.purchase_date>=p.start_date and u.purchase_date<=p.end_date group by p.product_id
```

Q-8 . Write an SQL query to report the first login date for each player. Table - Activity

player_id	device_id	event_date	games_played
1	2	2016-03-01	5
1	2	2016-05-02	6
2	3	2017-06-25	1
3	1	2016-03-02	0
3	4	2018-07-03	5

Output:

player_id	logindate
1	2016-03-01
2	2017-06-25
3	2016-03-02

```
select player_id, min(event_date) as logindate from Activity group by player_id
```

Q-9 . Write an SQL query to report the device that is first logged in date for each player. Table - Activity

```
select player_id, device_id from
(select  player_id, device_id,DENSE_RANK() over(partition by player_id order by event_date) first_login
from Activity) dns where first_login=1
```

Output:

player_id	device_id
1	2
2	3
3	1

Q-10 Write an SQL query to get the names of products that have at least 100 units ordered in February 2020 and their amount.

Tables- Products & Orders

product_id	product_name	product_category
1	Leetcode Solutions	Book
2	Jewels of Stringology	Book
3	HP Laptop	Laptop
4	Lenovo Laptop	Laptop
5	Leetcode Kit	T-shirt

product_id	order_date	unit
1	2020-02-05	60
1	2020-02-10	70
2	2020-01-18	30
2	2020-02-11	80
3	2020-02-17	2
3	2020-02-24	3
4	2020-03-01	20
4	2020-03-04	30
4	2020-03-04	60
5	2020-02-25	50
5	2020-02-27	50
5	2020-03-01	50

Output:

product_name	units
Leetcode Kit	100
Leetcode Solutions	130

```
select  product_name, sum(o.unit) as units from products p join orders o on p.product_id=o.product_id
where month(order_date)>=2 and month(order_date)<3 group by p.product_name  having sum(o.unit) >=100
```