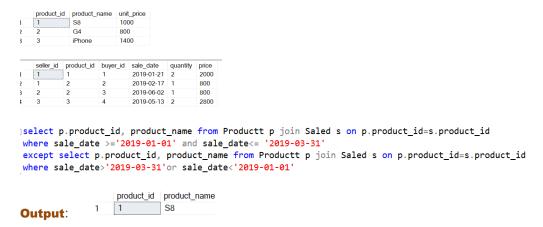
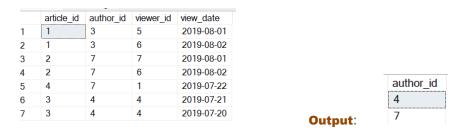
Q-1. Write an SQL query that reports the products that were only sold in the first quarter of 2019. That is, between 2019-01-01 and 2019-03-31 inclusive. TABLE- Productt and Saled.



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.



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

|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.

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

ad_id	user_id	action			
1	1	Clicked			
2	2	Clicked			
3	3	Viewed			
5	5	Ignored			
1	7	Ignored			
2	7	Viewed		ad_id	(No column name)
3	5	Clicked		1	66.67
1	4	Viewed		2	33.33
2	11	Viewed		3	50.00
1	2	Clicked	Output:	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 views from AdActions group by ad_id)as c
```

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

1 2 3 4	oyee_id team_id
2 3 4	
3	
4	
	O

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:

		couriny_iu	weather_state	Day
		2	15	2019-11-0
		2	12	2019-10-2
accombact id	accomplete manage	2	12	2019-10-2
country_id	country_name	3	-2	2019-11-1
2	LICA	3	0	2019-11-1
2	USA	3	3	2019-11-1
		5	16	2019-11-0
3	Australia	5	18	2019-11-0
_	_	5	21	2019-11-2
/	Peru	7	25	2019-11-2
_		7	22	2019-12-0
5	China	7	20	2019-12-0
		8	25	2019-11-0
8	Morocco	8	27	2019-11-1
		8	31	2019-11-2
9	Spain	9	7	2019-10-2
•	Opulli	9	3	2019-12-2

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

### Output:

Iselect 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

country id weather state Day

#### 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

	product_id	Average_price
	1	6.96
Output:	2	16.96

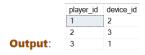
select p.product\_id,cast(sum(price\*units)/cast(sum(units)as decimal(10,2))as decimal(10,2)) Average\_price
from Prices p join UnitsSold u on p.product\_id=u.product\_id
where u.purchase\_date>=p.start\_date\_and\_u.purchase\_date<=p.end\_date\_group\_by\_p.product\_id</pre>

# $\ensuremath{\text{Q-8}}$ . 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		player id	logindate
1	2	2016-05-02	6		1	2016-03-01
2	3	2017-06-25	1		2	2017-06-25
3	1	2016-03-02	0		2	
3	4	2018-07-03	5	Output:	3	2016-03-02

select player id, min(event date) as logindate from Activity group by player id

|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



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	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

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

product_name	units
Leetcode Kit	100
Leetcode Solutions	130

Output: