In [1]:
```
pip install ipython-sql
```

```
Requirement already satisfied: ipython-sql in c:\users\helin\anaconda3\lib\site-packages
(0.4.1)
Requirement already satisfied: sqlalchemy>=0.6.7 in c:\users\helin\anaconda3\lib\site-pack
ages (from ipython-sql) (1.4.22)
Requirement already satisfied: ipython>=1.0 in c:\users\helin\anaconda3\lib\site-packages
(from ipython-sql) (7.29.0)
Requirement already satisfied: six in c:\users\helin\anaconda3\lib\site-packages (from ipy
thon-sql) (1.16.0)
Requirement already satisfied: sqlparse in c:\users\helin\anaconda3\lib\site-packages (fro
m ipython-sql) (0.4.3)
Requirement already satisfied: prettytable<1 in c:\users\helin\anaconda3\lib\site-packages
(from ipython-sql) (0.7.2)
Requirement already satisfied: ipython-genutils>=0.1.0 in c:\users\helin\anaconda3\lib\sit
e-packages (from ipython-sql) (0.2.0)
Requirement already satisfied: pygments in c:\users\helin\anaconda3\lib\site-packages (fro
m ipython>=1.0->ipython-sql) (2.10.0)
Requirement already satisfied: backcall in c:\users\helin\anaconda3\lib\site-packages (fro
m ipython>=1.0->ipython-sql) (0.2.0)
Requirement already satisfied: matplotlib-inline in c:\users\helin\anaconda3\lib\site-pack
ages (from ipython>=1.0->ipython-sql) (0.1.2)
Requirement already satisfied: setuptools>=18.5 in c:\users\helin\anaconda3\lib\site-packa
ges (from ipython>=1.0->ipython-sql) (58.0.4)
Requirement already satisfied: decorator in c:\users\helin\anaconda3\lib\site-packages (fr
om ipython>=1.0->ipython-sql) (5.1.0)
Requirement already satisfied: colorama in c:\users\helin\anaconda3\lib\site-packages (fro
m ipython>=1.0->ipython-sql) (0.4.4)
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0 in c:\users\he
lin\anaconda3\lib\site-packages (from ipython>=1.0->ipython-sql) (3.0.20)
Requirement already satisfied: traitlets>=4.2 in c:\users\helin\anaconda3\lib\site-package
s (from ipython>=1.0->ipython-sql) (5.1.0)
Requirement already satisfied: jedi>=0.16 in c:\users\helin\anaconda3\lib\site-packages (f
rom ipython>=1.0->ipython-sql) (0.18.0)
Requirement already satisfied: pickleshare in c:\users\helin\anaconda3\lib\site-packages
(from ipython>=1.0->ipython-sql) (0.7.5)
Requirement already satisfied: parso<0.9.0,>=0.8.0 in c:\users\helin\anaconda3\lib\site-pa
ckages (from jedi>=0.16->ipython>=1.0->ipython-sql) (0.8.2)
Requirement already satisfied: wcwidth in c:\users\helin\anaconda3\lib\site-packages (from
prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0->ipython>=1.0->ipython-sql) (0.2.5)
Requirement already satisfied: greenlet!=0.4.17 in c:\users\helin\anaconda3\lib\site-packa
ges (from sqlalchemy>=0.6.7->ipython-sql) (1.1.1)
Note: you may need to restart the kernel to use updated packages.
```

In [2]:
```
%load_ext sql
```

In [3]:
```
%sql mysql://root:Helin2134*$@localhost/casestudies
```

In [4]:
```
%%sql
select * from menu;
```

```
 * mysql://root:***@localhost/casestudies
3 rows affected.
```

Out[4]:

product_id	product_name	price
1	sushi	10
2	curry	15
3	ramen	12

In [5]:
```
%%sql
select * from sales;
```

```
 * mysql://root:***@localhost/casestudies
15 rows affected.
```

Out[5]:

customer_id	order_date	product_id
A	2021-01-01	1
A	2021-01-01	2
A	2021-01-07	2
A	2021-01-10	3
A	2021-01-11	3
A	2021-01-11	3
B	2021-01-01	2
B	2021-01-02	2
B	2021-01-04	1
B	2021-01-11	1
B	2021-01-16	3
B	2021-02-01	3
C	2021-01-01	3
C	2021-01-01	3
C	2021-01-07	3

In [6]:
```
%%sql
select * from members;
```

```
 * mysql://root:***@localhost/casestudies
2 rows affected.
```

Out[6]:

customer_id	join_date
B	2021-01-09 00:00:00
A	2021-01-07 00:00:00

In [7]:
```
%%sql
select customer_id,sum(price) from sales join menu on sales.product_id=menu.product_id gro
```

```
 * mysql://root:***@localhost/casestudies
3 rows affected.
```

Out[7]:

customer_id	sum(price)
A	76
B	74
C	36

In [8]:
```
%%sql
select customer_id, count(distinct(order_date)) as visit_days from sales group by customer
```

```
 * mysql://root:***@localhost/casestudies
3 rows affected.
```

Out[8]:

customer_id	visit_days
A	4
B	6
C	2

In [9]:

```
%%sql
select product_name,order_date,customer_id, DENSE_RANK() OVER (PARTITION BY customer_id
ORDER BY order_date ) as rankk from sales join menu on sales.product_id=menu.product_id
```

```
 * mysql://root:***@localhost/casestudies
15 rows affected.
```

Out[9]:

product_name	order_date	customer_id	rankk
sushi	2021-01-01	A	1
curry	2021-01-01	A	1
curry	2021-01-07	A	2
ramen	2021-01-10	A	3
ramen	2021-01-11	A	4
ramen	2021-01-11	A	4
curry	2021-01-01	B	1
curry	2021-01-02	B	2
sushi	2021-01-04	B	3
sushi	2021-01-11	B	4
ramen	2021-01-16	B	5
ramen	2021-02-01	B	6
ramen	2021-01-01	C	1
ramen	2021-01-01	C	1
ramen	2021-01-07	C	2

RANK returns the rank of all in order but dense_rank returns each customer rank.

In [10]:

```
%%sql
select count(customer_id) as count, product_name from menu join sales on sales.product_id=
```

```
 * mysql://root:***@localhost/casestudies
1 rows affected.
```

Out[10]:

count	product_name
8	ramen

In [11]:

```
%%sql
SELECT customer_id, product_name
FROM sales
JOIN menu ON sales.product_id = menu.product_id
GROUP BY customer_id, product_name
HAVING COUNT(*) = (
```

```
  SELECT MAX(product_count)
  FROM (
    SELECT customer_id, product_id, COUNT(*) as product_count
    FROM sales
    GROUP BY customer_id, product_id
  ) AS counts
  WHERE counts.customer_id = customer_id
);
```

```
 * mysql://root:***@localhost/casestudies
2 rows affected.
```

Out[11]:

customer_id	product_name
A	ramen
C	ramen

In [53]:

```
%%sql
WITH cte_first_member AS (
SELECT members.customer_id AS mem,
menu.product_name AS product,
RANK() OVER (
PARTITION BY members.customer_id
ORDER BY sales.order_date
) AS raank
FROM members
JOIN sales ON sales.customer_id = members.customer_id
JOIN menu ON sales.product_id = menu.product_id
WHERE sales.order_date >= members.join_date
)
SELECT mem,
product
FROM cte_first_member
WHERE raank = 1;
```

```
 * mysql://root:***@localhost/casestudies
2 rows affected.
```

Out[53]:

mem	product
A	curry
B	sushi

In [55]:

```
%%sql
WITH cte_BEFORE_member_purchase AS (
SELECT members.customer_id AS mem,
menu.product_name AS product,
RANK() OVER (
PARTITION BY members.customer_id
ORDER BY sales.order_date desc
) AS raank
FROM members
JOIN sales ON sales.customer_id = members.customer_id
JOIN menu ON sales.product_id = menu.product_id
WHERE sales.order_date < members.join_date
)
SELECT mem,
product
FROM cte_BEFORE_member_purchase
WHERE raank = 1;
```

```
 * mysql://root:***@localhost/casestudies
3 rows affected.
```

Out[55]:

mem	product
A	sushi
A	curry
B	sushi

In [59]:

```
%%sql
WITH cte_BEFORE_member_countsum AS (
SELECT members.customer_id AS mem,
SUM(menu.price) AS summ,
COUNT(menu.product_id) AS countt
FROM members
JOIN sales ON sales.customer_id = members.customer_id
JOIN menu ON sales.product_id = menu.product_id
WHERE sales.order_date < members.join_date
GROUP BY mem
)
SELECT *
from cte_BEFORE_member_countsum
order by mem;
```

```
 * mysql://root:***@localhost/casestudies
2 rows affected.
```

Out[59]:

mem	summ	countt
A	25	2
B	40	3

In [61]:

```
%%sql
WITH cte_points AS(
SELECT *,
CASE WHEN product_id=1 THEN price*20
else price*10
end as cte_points
from menu
)
Select sales.customer_id, SUM(cte_points) as point
from sales
join cte_points
on sales.product_id=cte_points.product_id
group by customer_id
```

```
 * mysql://root:***@localhost/casestudies
3 rows affected.
```

Out[61]:

customer_id	point
A	860
B	940
C	360