SQL Test

The rows in the table of interest represent offers shown to a user during installations via Installer. When the same offer is shown to the same user on the same day, that offer is considered a duplicate. Each record specifies the position in which the offer was shown on that day and the rate of revenue it supplies the company.

Write a query to display for each day and offer, the amount of offers installed, the amount of duplicate offers installed and the sum of the revenue generated from duplicate installs. The result should look like this:

Day	Offer	Offers installed	Duplicate installs	Duplicate installs revenue
02/02/2017	avant	2000	140	45

Solution: Let's start with creating a table to practice on:

```
CREATE TABLE Offers(
        Dday date,
        Offer varchar(255),
        UserID int,
        Rate float,
        Position int,
        Aaction varchar(255));
```

```
INSERT INTO Offers (Dday,Offer,UserID,Rate,Position,Aaction)
VALUES('2/2/2017','avant',1,0.05,1,'Install'),
('2/2/2017','avant',1,0.05,1,'Install'),
('2/2/2017','avant',1,0.12,3,'Install'),
('2/2/2017','avant',2,0.15,1,'Quit'),
('2/2/2017','avis',2,0.03,1,'Install'),
('2/2/2017','avis',3,0.05,1,'Install'),
('2/2/2017','avis',4,0.12,2,'Install'),
('2/2/2017','avis',5,0.15,2,'Install'),
('2/2/2017','avis',5,0.15,1,'Install'),
('3/2/2017','avant',3,0.03,1,'Install'),
('3/2/2017','avant',4,0.05,1,'Install'),
('3/2/2017','avant',5,0.12,2,'Install'),
('3/2/2017','avant',6,0.15,3,'Install'),
('3/2/2017','avant',7,0.03,1,'Install'),
('3/2/2017','avant',8,0.05,2,'Install'),
('3/2/2017','avant',9,0.12,3,'Install'),
('3/2/2017','avant',10,0.15,1,'Quit'),
('3/2/2017','avant',11,0.15,2,'Quit'),
('3/2/2017','avant',12,0.2,1,'Quit');
```

Select * to have a look at what we got

Results Messages

Search to filter items...

DDAY	OFFER	USERID	RATE	POSITION
2017-02-02T00:00:00.0000000	avant	1	0.05	1
2017-02-02T00:00:00.0000000	avant	1	0.05	1
2017-02-02T00:00:00.0000000	avant	1	0.12	3

Now we can use a CTE (Common Table Expression) and the window function row_number() to create a table with an indication of duplication by my specified keys and order. From this table I can count all the offers and then sum the required information about the duplicate records.

```
with cte (Dday,Offer,Rate,rnk)
AS (
        select Dday,Offer,Rate,
        row_number( ) over (partition by Dday,Offer,UserID order by Position asc) as rnk
        from Offers
        where Aaction='Install'
)
select Dday,Offer,count(*) as Installs,sum(case when rnk>1 then 1 else 0 end) as duplicates,
sum(case when rnk>1 then Rate else 0 end) as revenue
from cte
group by Dday,Offer
```

Finally, we can have a look at the results:

Results Messages

Search to filter items...

DDAY	OFFER	INSTALLS	DUPLICATES	REVENUE
2017-02-02T00:00:00.0000000	avant	3	2	0.17
2017-03-02T00:00:00.0000000	avant	7	0	0
2017-02-02T00:00:00.0000000	avis	5	1	0.15

Exactly as we wanted.