```
USE window_fun;
CREATE TABLE [dbo].[Orders]
(
	order_id INT,
	order_date DATE,
	customer_name VARCHAR(250),
	city VARCHAR(100),
	order_amount MONEY
)

INSERT INTO [dbo].[Orders]
SELECT '1001','04/01/2017','David Smith','GuildFord',10000
UNION ALL
SELECT '1002','04/02/2017','David Jones','Arlington',20000
UNION ALL
SELECT '1003','04/03/2017','John Smith','Shalford',5000
UNION ALL
SELECT '1004','04/04/2017','Michael Smith','GuildFord',15000
UNION ALL
SELECT '1005','04/05/2017','David Williams','Shalford',7000
UNION ALL
SELECT '1006','04/06/2017','Paum Smith','GuildFord',25000
UNION ALL
SELECT '1007','04/10/2017','Andrew Smith','Arlington',15000
UNION ALL
SELECT '1008','04/11/2017','David Brown','Arlington',2000
UNION ALL
SELECT '1009','04/20/2017','Robert Smith','Shalford',1000
UNION ALL
SELECT '1010','04/25/2017','Peter Smith','GuildFord',500;

select * from emp.dbo.employees;

/*Aggregate Window Functions
SUM(), MAX(), MIN(), AVG(). COUNT()
Ranking Window Functions
RANK(), DENSE_RANK(), ROW_NUMBER(), NTILE()
Value Window Functions
LAG(), LEAD(), FIRST_VALUE(), LAST_VALUE()*/


/*SUM*/
select city,sum(order_amount) as total from Orders group by city;
```

```
select order_id,
	order_date,
	customer_name,
	city,
	order_amount,
	sum(order_amount) over(partition by city) as City_wish_total from
window_fun.dbo.Orders;

/*AVERAGE*/
SELECT order_id,
		order_date,
		customer_name,
		city,
		order_amount,
		AVG(order_amount) OVER(PARTITION BY city, MONTH(order_date)) as
average_order_amount
FROM [dbo].[Orders]

/*MINIMUM*/
SELECT order_id,
		order_date,
		customer_name,
		city,
		order_amount,
		MIN(order_amount) over (partition by city) as min_order_citywise
FROM [dbo].[Orders];

/*MAXIMUM*/
SELECT order_id,
		order_date,
		customer_name,
		city,
		order_amount,
		MAX(order_amount) over (partition by city) as min_order_citywise
FROM [dbo].[Orders];

/*COUNT*/
SELECT city,COUNT(*) as count_ FROM [dbo].[Orders] GROUP BY city;

SELECT order_id,
		order_date,
		customer_name,
```

```
		city,
		order_amount,
		COUNT(order_amount) over (partition by city) as min_order_citywise
FROM [dbo].[Orders];

/*RANKING*/

/*using order by*/
SELECT order_id,
		order_date,
		customer_name,
		city,
		order_amount,
		RANK() over(order by order_amount DESC) as RANK_
FROM [dbo].[Orders];

/*using patition by*/

SELECT order_id,
		order_date,
		customer_name,
		city,
		order_amount,
		rank() over(partition by city order by order_date DESC) as RANK_
FROM [dbo].[Orders];



SELECT order_id,
		order_date,
		customer_name,
		city,
		order_amount,
		rank() over(partition by city,MONTH(order_date) order by order_date DESC) as
RANK_
FROM [dbo].[Orders];


SELECT order_id,
		order_date,
		customer_name,
		city,
		order_amount,
```

```
		rank() over(partition by city,MONTH(order_date) order by order_amount) as
RANK_
FROM [dbo].[Orders];

SELECT order_id,
		order_date,
		customer_name,
		city,
		order_amount,
		rank() over(partition by city order by MONTH(order_date)) as RANK_
FROM [dbo].[Orders];


/*DENSE RANK*/


SELECT *,
		DENSE_RANK() over(order by order_amount DESC) as RANK_
FROM [dbo].[Orders];



SELECT order_id,
		order_date,
		customer_name,
		city,
		order_amount,
		DENSE_RANK() over(partition by city order by MONTH(order_date)) as RANK_
FROM [dbo].[Orders];


/*ROW NUMBER*/

SELECT order_id,order_date,customer_name,city, order_amount,
ROW_NUMBER() OVER(ORDER BY order_id) [row_number]
FROM [dbo].[Orders];

SELECT order_id,order_date,customer_name,city, order_amount,
ROW_NUMBER() OVER(ORDER BY city DESC) [row_number]
FROM [dbo].[Orders];


SELECT order_id,order_date,customer_name,city, order_amount,
ROW_NUMBER() OVER(PARTITION BY city ORDER BY order_amount DESC) [row_number]
```

```
FROM [dbo].[Orders]


/*NTILE*/
SELECT order_id,order_date,customer_name,city, order_amount,
NTILE(4) OVER(ORDER BY order_amount) [row_number]
FROM [dbo].[Orders];

/*LAG*/

SELECT order_id,customer_name,city, order_amount,order_date,
 --in below line, 1 indicates check for previous row of the current row
 LAG(order_date,1) OVER(ORDER BY order_date) prev_order_date
FROM [dbo].[Orders]

SELECT order_id,customer_name,city, order_amount,order_date,
 LAG(order_date,2) OVER(ORDER BY order_date) prev_order_date
FROM [dbo].[Orders]

SELECT order_id,customer_name,city, order_amount,order_date,
 LAG(order_date,1) OVER(PARTITION BY city ORDER BY order_date) prev_order_date
FROM [dbo].[Orders]

/*LEAD*/

SELECT order_id,customer_name,city, order_amount,order_date,
 LEAD(order_date,1) OVER(ORDER BY order_date) prev_order_date
FROM [dbo].[Orders]

SELECT order_id,customer_name,city, order_amount,order_date,
 LEAD(order_date,2) OVER(ORDER BY order_date) prev_order_date
FROM [dbo].[Orders]

SELECT order_id,customer_name,city, order_amount,order_date,
 LEAD(order_date,1) OVER(PARTITION BY city ORDER BY order_date) prev_order_date
FROM [dbo].[Orders];

/*FIRST_VALUE LAST_VALUE*/

SELECT order_id,order_date,customer_name,city, order_amount,
FIRST_VALUE(order_date) OVER(PARTITION BY city ORDER BY order_date)
first_order_date,
LAST_VALUE(order_date) OVER(PARTITION BY city ORDER BY order_date) last_order_date
FROM [dbo].[Orders];
```

```
/*Regex*/
--customer_name that starting with a,e,i,o,u
select distinct customer_name from [dbo].[Orders] where lower(customer_name) like '[aeiou]%'

--customer_name that not starting with a,e,i,o,u
select distinct customer_name from [dbo].[Orders] where lower(customer_name) like '[^aeiou]%'

--customer_name that starting with da
select distinct customer_name from [dbo].[Orders] where lower(customer_name) like '[d][a]%'

--customer_name that starting character between a and d
select distinct customer_name from [dbo].[Orders] where lower(customer_name) like '[a-d]%'

--customer_name that first character between a and n, second character between b and n
select distinct customer_name from [dbo].[Orders] where lower(customer_name) like
'[a-n][b-n]%'

--customer_name that ending character between a and n
select distinct customer_name from [dbo].[Orders] where lower(customer_name) like '%[a-n]'

--customer_name that starting character d and ending character h
select distinct customer_name from [dbo].[Orders] where lower(customer_name) like '[d]%[h]'

--starting letters excluding A to h
select distinct customer_name from [dbo].[Orders] where lower(customer_name) like '[^a-h]%'

--The first character should be from a and s character and ending with th
select distinct customer_name from [dbo].[Orders] where lower(customer_name) like
'[a-s]%[t][h]'

--First character should be from a and s character and containing ae
select distinct customer_name from [dbo].[Orders] where lower(customer_name) like
'[a-s]%[a][e]%'

--second letter A, it returns both A and a
select customer_name from [dbo].[Orders] where customer_name like '_[A]%'

--second letter A, it returns only A
select customer_name from [dbo].[Orders] where customer_name COLLATE
Latin1_General_BIN  like '_[A]%'

--second letter a, it returns only a
```

```
select customer_name from [dbo].[Orders] where customer_name COLLATE
Latin1_General_BIN  like '_[a]%'

--starting with Da
select customer_name from [dbo].[Orders] where customer_name COLLATE
Latin1_General_BIN  like '[D][a]%'

--starting with da
select customer_name from [dbo].[Orders] where customer_name COLLATE
Latin1_General_BIN  like '[d][a]%'

CREATE TABLE TSQLREGEX(
    Email VARCHAR(1000)
 )

 Insert into TSQLREGEX values('raj@gmail.com')
 Insert into TSQLREGEX values('HSDFX@gmail.com')
 Insert into TSQLREGEX values('JHKHKO.PVS@gmail.com')
 Insert into TSQLREGEX values('ABC@@gmail.com')
 Insert into TSQLREGEX values('ABC.DFG.LKF#@gmail.com')
--email validation
 Select * from TSQLREGEX where email LIKE '%[A-Z0-9][@][A-Z0-9]%[.][A-Z0-9]%'
```