STORED PROCEDURE

```
-- Stored Procedure
CREATE PROCEDURE P1
AS
SELECT * FROM DEPARTMENT
GO;

exec P1;
```

TCL commands

```
-- TCL
begin transaction;
DELETE FROM DEPARTMENT WHERE DNO=106;
commit;

begin transaction;
INSERT INTO [dbo].[DEPARTMENT] VALUES(106,'FINANCE');
rollback;

exec P1;
```

STRING FUNCTIONS

```
-- STRING FUNCTIONS
SELECT CONCAT([ENAME],' ',[CITY]) FROM [dbo].[EMPLOYEES]
```

117 %

Results | Messages

	(No column name)
1	KARTIK BANGALORE
2	SOMYA BANGALORE
3	RAJNIKANT CHENNAI
4	PRABHAKARAN CHENNAI

```
SELECT ENAME,LEN(ENAME) 'LENGTH'
FROM EMPLOYEES;
```

117 %

Results | Messages

	ENAME	LENGTH
1	KARTIK	6
2	SOMYA	5
3	RAJNIKANT	9
4	PRABHAKARAN	11

```
SELECT ENAME,LOWER(ENAME) 'LENGTH'
FROM EMPLOYEES;
```

117 %

Results | Messages

	ENAME	LENGTH
1	KARTIK	kartik
2	SOMYA	somya
3	RAJNIKANT	rajnikant
4	PRABHAKARAN	prabhakaran

```
SELECT ENAME,CHARINDEX('I',ENAME) FROM EMPLOYEES;
SELECT ENAME,PATINDEX('%KAR%',ENAME) FROM EMPLOYEES;
```

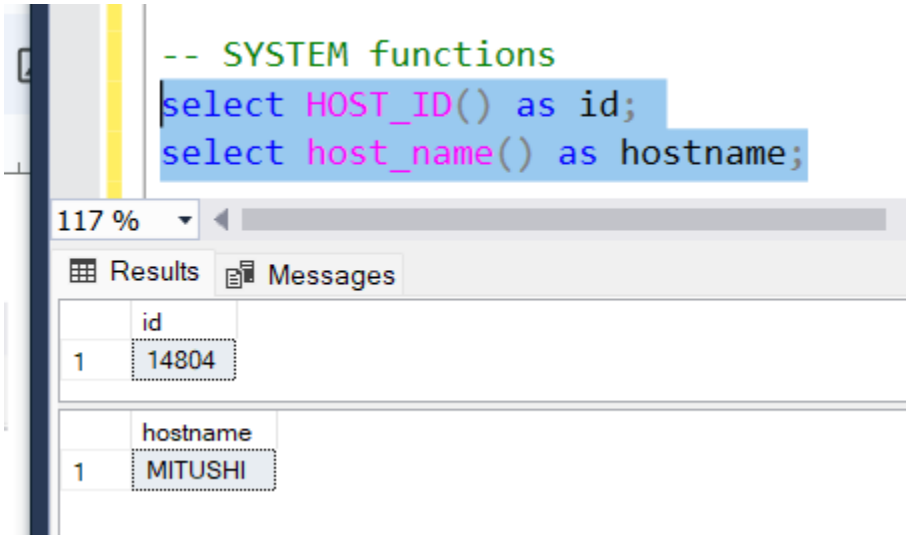
117 %

Results | Messages

	ENAME	(No column name)
1	KARTIK	5
2	SOMYA	0
3	RAJNIKANT	5
4	PRABHAKARAN	0

	ENAME	(No column name)
1	KARTIK	1
2	SOMYA	0
3	RAJNIKANT	0
4	PRABHAKARAN	7

SYSTEM FUNCTIONS



```
-- SYSTEM functions
select HOST_ID() as id;
select host_name() as hostname;
```

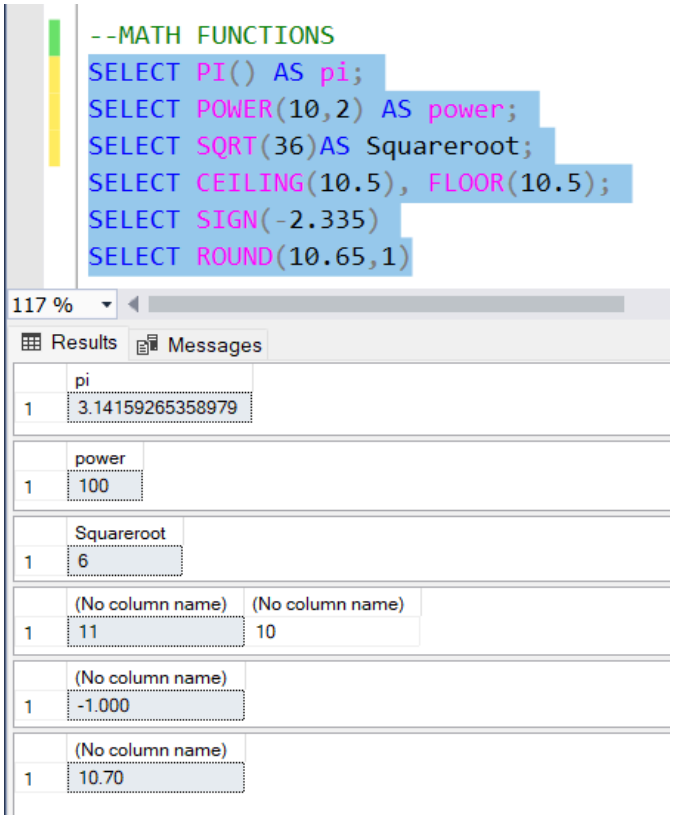
117 %

Results | Messages

	id
1	14804

	hostname
1	MITUSHI

MATH FUNCTIONS



```
--MATH FUNCTIONS
SELECT PI() AS pi;
SELECT POWER(10,2) AS power;
SELECT SQRT(36)AS Squareroot;
SELECT CEILING(10.5), FLOOR(10.5);
SELECT SIGN(-2.335)
SELECT ROUND(10.65,1)
```

117 %

Results | Messages

	pi
1	3.14159265358979

	power
1	100

	Squareroot
1	6

	(No column name)	(No column name)
1	11	10

	(No column name)
1	-1.000

	(No column name)
1	10.70

DATE FUNCTIONS

```
-- DATE FUNCTIONS
SELECT GETDATE();
SELECT SYSDATETIME();
SELECT MONTH(GETDATE());
```

117 %

Results | Messages

	(No column name)
1	2024-01-23 15:12:42.150

	(No column name)
1	2024-01-23 15:12:42.1514002

	(No column name)
1	1

```
SELECT MONTH([dob]) FROM student_info;
SELECT YEAR([dob]) FROM student_info;
SELECT DAY([dob]) FROM student_info;
```

117 %

Results | Messages

	(No column name)
1	8
2	9
3	10
4	9
5	11
6	1
7	8

	(No column name)
1	2001
2	2001
3	1994
4	2001
5	1991
6	2000
7	2000

	(No column name)
1	16
2	7
3	1
4	7
5	23
6	19
7	22

```
SELECT DATENAME(MONTH,[dob]) FROM student_info;
SELECT DATEPART(MONTH,[dob]) FROM student_info;
SELECT DATEDIFF(MONTH,[dob],GETDATE()) FROM student_info;
```

117 %

Results | Messages

	(No column name)
1	August
2	September
3	October
4	September
5	November
6	January
7	August

	(No column name)
1	8
2	9
3	10
4	9
5	11
6	1
7	8

	(No column name)
1	269
2	268
3	351
4	268
5	386
6	288
7	281

UNION

```
180 •   select * from employee
181     union
182     select *from emp;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id	name	age	email	salary	department
101	aparna	20	aparna@gmail.com	30000	sales
102	stella	20	stella@gmail.com	40000	HR
103	bob	45	bob@gmail.com	30000	IT
104	aparna	20	aparna@gmail.com	30000	sales
105	stella	20	stella@gmail.com	40000	HR
106	kishore	35	kishore@gmail.com	35000	IT
107	lucky	28	lucky@gmail.com	60000	salesforce
108	nimmi	30	nimmi@gmail.com	30000	IT
109	krish	30	krish@gmail.com	45000	sales
201	bob	45	bob@gmail.com	30000	IT
202	heshi	45	heshi@gmail.com	30000	HR
203	minnu	45	minnu@gmail.com	30000	HR

UNION ALL

```
184 •  select * from employee
185    union all
186    select * from emp;
```

Result Grid | Filter Rows: | Export: | Wrap Ce

	id	name	age	email	salary	department
▶	101	aparna	20	aparna@gmail.com	30000	sales
	102	stella	20	stella@gmail.com	40000	HR
	103	bob	45	bob@gmail.com	30000	IT
	104	aparna	20	aparna@gmail.com	30000	sales
	105	stella	20	stella@gmail.com	40000	HR
	106	kishore	35	kishore@gmail.com	35000	IT
	107	lucky	28	lucky@gmail.com	60000	salesforce
	108	nimmi	30	nimmi@gmail.com	30000	IT
	109	krish	30	krish@gmail.com	45000	sales
	107	lucky	28	lucky@gmail.com	60000	salesforce
	108	nimmi	30	nimmi@gmail.com	30000	IT
	201	bob	45	bob@gmail.com	30000	IT
	202	heshi	45	heshi@gmail.com	30000	HR
	203	minnu	45	minnu@gmail.com	30000	HR

INTERSECT

```
188 •  select * from employee
189    intersect
190 ⊠  select * from emp;
```

Result Grid | Filter Rows: | Export: | Wr

	id	name	age	email	salary	department
▶	107	lucky	28	lucky@gmail.com	60000	salesforce
	108	nimmi	30	nimmi@gmail.com	30000	IT

EXCEPT

```
196 •   select name,salary,age from employee
197 ⊗   except
198     select name,salary,age from emp;
```

Result Grid | Filter Rows: | Export:

	name	salary	age
▶	aparna	30000	20
	stella	40000	20
	kishore	35000	35
	krish	45000	30

RANK

```
-- RANK
select first_name,email,
RANK() over (order by first_name) as row_no
from Students
```

117 %

Results | Messages

	first_name	email	row_no
1	Ava	ava.wilson@example.com	1
2	David	david.johnson@example.com	2
3	Emily	emily.williams@example.com	3
4	Ethan	ethan.moore@example.com	4
5	Jane	jane.smith@example.com	5
6	John	john.doe@example.com	6
7	John	johndoe@example.com	6
8	Matthew	matthew.miller@example.com	8
9	Michael	michael.jones@example.com	9
10	Olivia	olivia.davis@example.com	10
11	Riya	riyasharma@example.com	11

PARTITION BY

```
select first_name,email,
RANK() over (PARTITION BY first_name order by email) as row_no
from Students
```

117 %

Results | Messages

	first_name	email	row_no
1	Ava	ava.wilson@example.com	1
2	David	david.johnson@example.com	1
3	Emily	emily.williams@example.com	1
4	Ethan	ethan.moore@example.com	1
5	Jane	jane.smith@example.com	1
6	John	john.doe@example.com	1
7	John	johndoe@example.com	2
8	Matthew	matthew.miller@example.com	1
9	Michael	michael.jones@example.com	1
10	Olivia	olivia.davis@example.com	1
11	Riya	riyasharma@example.com	1

DENSE RANK

```
-- DENSE RANK
select first_name,email,
DENSE_RANK() over (order by first_name ) as row_no
from Students
```

117 %

Results | Messages

	first_name	email	row_no
1	Ava	ava.wilson@example.com	1
2	David	david.johnson@example.com	2
3	Emily	emily.williams@example.com	3
4	Ethan	ethan.moore@example.com	4
5	Jane	jane.smith@example.com	5
6	John	john.doe@example.com	6
7	John	johndoe@example.com	6
8	Matthew	matthew.miller@example.com	7
9	Michael	michael.jones@example.com	8
10	Olivia	olivia.davis@example.com	9
11	Riya	riyasharma@example.com	10

```sql
select first_name,email,
DENSE_RANK() over (PARTITION BY first_name order by email) as row_no
from Students
```

117 %

Results | Messages

	first_name	email	row_no
1	Ava	ava.wilson@example.com	1
2	David	david.johnson@example.com	1
3	Emily	emily.williams@example.com	1
4	Ethan	ethan.moore@example.com	1
5	Jane	jane.smith@example.com	1
6	John	john.doe@example.com	1
7	John	johndoe@example.com	2
8	Matthew	matthew.miller@example.com	1
9	Michael	michael.jones@example.com	1
10	Olivia	olivia.davis@example.com	1
11	Riya	riyasharma@example.com	1

Cleaning and Transformation

```sql
-- capitalizing name
update  Students
set first_name=UPPER(first_name);
```

117 %

Results | Messages

	student_id	first_name	last_name	date_of_birth	email	phone_number
1	1	RIYA	Sharma	1999-05-20	riyasharma@example.com	123-456-5555
2	2	JANE	Smith	1998-08-21	jane.smith@example.com	987-654-3210
3	3	DAVID	Johnson	1997-03-10	david.johnson@example.com	555-123-4567
4	4	EMILY	Williams	1996-11-28	emily.williams@example.com	789-456-1230
5	5	MICHAEL	Jones	1999-04-05	michael.jones@example.com	333-555-7777
6	7	MATTHEW	Miller	1998-12-02	matthew.miller@example.com	444-111-0000
7	8	OLIVIA	Davis	1996-06-20	olivia.davis@example.com	666-999-3333
8	9	ETHAN	Moore	1999-02-14	ethan.moore@example.com	111-222-4444
9	10	AVA	Wilson	1997-07-08	ava.wilson@example.com	999-777-6666
10	11	JOHN	Doe	1995-08-15	john.doe@example.com	123-456-7890

```sql
 -- cleaning and transformation

 -- duplicate data
select first_name,count(first_name) as Actual_count from Students
group by first_name
having count(first_name)>1;

-- deleting duplicate data
with cte as
(
select first_name,email,ROW_NUMBER() over (partition by first_name order by email ) as row_no
from Students
)
delete from cte
where row_no>1;

-- REMOVING NULL VALUES
select * from Students
where first_name is null;

delete from Students
where first_name is null;

-- UPDATING NULL VALUES
select * from Students where student_id is null;

update Students set student_id=7
where student_id is null;
```

SUB TOTALS

```
-- SUB TOTALS
SELECT  SalesYear, SUM(SalesTotal) AS SalesTotal FROM SalesList
GROUP BY ROLLUP(SalesYear);
```

117 %

Results | Messages

	SalesYear	SalesTotal
1	2019	700.00
2	2020	350.00
3	NULL	1050.00

```
SELECT  SalesYear,SalesQuartes, SUM(SalesTotal) AS SalesTotal
FROM SalesList GROUP BY ROLLUP(SalesYear, SalesQuartes);
```

117 %

Results | Messages

	SalesYear	SalesQuartes	SalesTotal
1	2019	Q1	60.00
2	2019	Q2	30.00
3	2019	Q3	160.00
4	2019	Q4	450.00
5	2019	NULL	700.00
6	2020	Q1	220.00
7	2020	Q3	10.00
8	2020	Q4	120.00
9	2020	NULL	350.00
10	NULL	NULL	1050.00

GROUPING

```sql
SELECT SalesYear,SalesQuartes, SUM(SalesTotal) AS SalesTotal ,
GROUPING(SalesQuartes) AS SalesQuarterGrp,
GROUPING(SalesYear) AS SYearGrp
FROM SalesList
GROUP BY ROLLUP(SalesYear, SalesQuartes);
```

117 %

Results | Messages

	SalesYear	SalesQuartes	SalesTotal	SalesQuarterGrp	SYearGrp
1	2019	Q1	60.00	0	0
2	2019	Q2	30.00	0	0
3	2019	Q3	160.00	0	0
4	2019	Q4	450.00	0	0
5	2019	NULL	700.00	1	0
6	2020	Q1	220.00	0	0
7	2020	Q3	10.00	0	0
8	2020	Q4	120.00	0	0
9	2020	NULL	350.00	1	0
10	NULL	NULL	1050.00	1	1

RENAMING SUBTOTALS USING SWITCH CASE

```sql
SELECT
CASE
WHEN GROUPING(SalesQuartes)=1 AND GROUPING(SalesYear)=0 THEN 'SubTotal'
WHEN GROUPING(SalesQuartes)=1 AND GROUPING(SalesYear)=1 THEN 'Grand Total'
ELSE CAST(SalesYear AS VARCHAR(10))
END
AS SalesYear, SalesQuartes,SUM(SalesTotal) AS SalesTotal
FROM SalesList
GROUP BY ROLLUP(SalesYear,SalesQuartes);
```

117 %

Results | Messages

	SalesYear	SalesQuartes	SalesTotal
1	2019	Q1	60.00
2	2019	Q2	30.00
3	2019	Q3	160.00
4	2019	Q4	450.00
5	SubTotal	NULL	700.00
6	2020	Q1	220.00
7	2020	Q3	10.00
8	2020	Q4	120.00
9	SubTotal	NULL	350.00
10	Grand Total	NULL	1050.00

CTE AND ROW NUMBER

```sql
WITH CTE AS (
SELECT SalesMonth,SalesTotal ,
ROW_NUMBER() OVER(ORDER BY NEWID())
AS RowNumber FROM SalesList )

SELECT
    RowNumber ,SalesMonth,SUM(SalesTotal) AS SalesTotal
FROM CTE
GROUP BY ROLLUP(SalesMonth, RowNumber);
```

117 %

Results | Messages

	RowNumber	SalesMonth	SalesTotal
1	7	July	10.00
2	10	July	160.00
3	NULL	July	170.00
4	2	March	60.00
5	4	March	170.00
6	8	March	50.00
7	NULL	March	280.00
8	9	May	30.00
9	NULL	May	30.00
10	1	November	120.00
11	3	November	180.00
12	6	November	120.00
13	NULL	November	420.00
14	5	October	150.00
15	NULL	October	150.00
16	NULL	NULL	1050.00