INTRODUCTION

Column Functions

EMPNO	LASTNAME	SALARY	JOB	EDLEVEL
000120	O'CONNELL	29250.00	CLERK	14
000100	SPENSER	26150.00	MANAGER	14
000130	QUINTANA	23800.00	ANALYST	16
000280	SCHNEIDER	26250.00	OPERATOR	17
000250	SMITH	19180.00	CLERK	15
000060	STERN	32250.00	MANAGER	16

EMPLOYEE Table

```
SELECT SUM(SALARY) AS TOTAL, MIN(SALARY) AS
MINIMUM,
       MAX(SALARY) AS MAXIMUM,
       DECIMAL(AVG(SALARY),8,2) AS AVERAGE,
       COUNT(*) AS #EMP,
       COUNT(DISTINCT EDLEVEL) AS #LVL
   FROM EMPLOYEE
```

	TOTAL	MINIMUM	MAXIMUM	AVERAGE	#EMP	#LVL
Result	156880.00	19180.00	32250.00	26146.66	6	4

Summarizing Group Values

EMPNO	LASTNAME	SALARY	JOB	EDLEVEL
000120	O'CONNELL	29250.00	CLERK	14
000100	SPENSER	26150.00	MANAGER	14
000130	QUINTANA	23800.00	ANALYST	16
000280	SCHNEIDER	26250.00	OPERATOR	17
000250	SMITH	19180.00	CLERK	15
000060	STERN	32250.00	MANAGER	16

EMPLOYEE Table

```
SELECT EDLEVEL, SUM(SALARY) AS GROUP_TOTAL
    FROM EMPLOYEE
    GROUP BY EDLEVEL
    HAVING COUNT(*) > 1
```
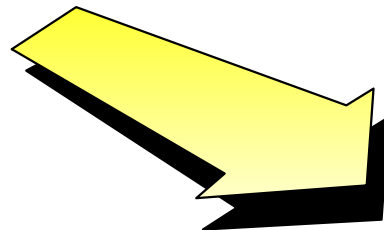
Result

EDLEVEL	GROUP_TOTAL
14	55400.00
16	56050.00

GROUP BY ROLLUP

EMPNO	LASTNAME	SALARY	JOB	EDLEVEL
000120	O'CONNELL	29250.00	CLERK	14
000100	SPENSER	26150.00	MANAGER	14
000130	QUINTANA	23800.00	ANALYST	16
000280	SCHNEIDER	26250.00	OPERATOR	17
000250	SMITH	19180.00	CLERK	15
000060	STERN	32250.00	MANAGER	16

EMPLOYEE Table

```
SELECT EDLEVEL, DECIMAL(AVG(SALARY),8,2)
   AS  AVG_SALARY
FROM EMPLOYEE
GROUP BY ROLLUP(EDLEVEL)
```

Result

EDLEVEL	AVG_SALARY
- (null)	**26146.66**
14	27700.00
15	19180.00
16	28025.00
17	26250.00

GROUP BY CUBE

EMPNO	LASTNAME	SALARY	JOB	EDLEVEL
000120	O'CONNELL	29250.00	CLERK	14
000100	SPENSER	26150.00	MANAGER	14
000130	QUINTANA	23800.00	ANALYST	16
000280	SCHNEIDER	26250.00	OPERATOR	17
000250	SMITH	19180.00	CLERK	15
000060	STERN	32250.00	MANAGER	16

EMPLOYEE Table

```
SELECT EDLEVEL,JOB,
   DECIMAL(AVG(SALARY),8,2) AS
   AVG_SALARY
FROM EMPLOYEE
WHERE EDLEVEL IN(14,15)
GROUP BY CUBE(JOB,EDLEVEL)
```
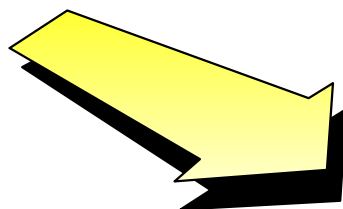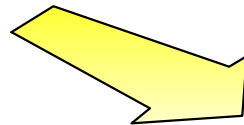
Result

EDLEVEL	JOB	AVG_SALARY
14	-	27700.00
15	-	19180.00
-	-	24860.00
-	CLERK	24215.00
-	MANAGER	26150.00
14	CLERK	29250.00
15	CLERK	19180.00
14	MANAGER	26150.00

Grouping Function - Why?
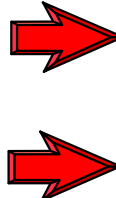
EMPNO	LASTNAME	SALARY	JOB	EDLEVEL
000120	O'CONNELL	29250.00	CLERK	14
000100	SPENSER	26150.00	MANAGER	14
000130	QUINTANA	23800.00	ANALYST	16
000280	SCHNEIDER	26250.00	OPERATOR	17
000250	SMITH	19180.00	CLERK	-
000060	STERN	32250.00	MANAGER	16

EMPLOYEE Table

```
SELECT EDLEVEL, DECIMAL(AVG(SALARY),8,2)
  AS  AVG_SALARY
FROM EMPLOYEE
GROUP BY ROLLUP(EDLEVEL)
```

Result

EDLEVEL	AVG_SALARY
-	**26146.66**
14	27700.00
-	19180.00
16	28025.00
17	26250.00

Grouping Function - Example

EMPNO	LASTNAME	SALARY	JOB	EDLEVEL
000120	O'CONNELL	29250.00	CLERK	14
000100	SPENSER	26150.00	MANAGER	14
000130	QUINTANA	23800.00	ANALYST	16
000280	SCHNEIDER	26250.00	OPERATOR	17
000250	SMITH	19180.00	CLERK	-
000060	STERN	32250.00	MANAGER	16

EMPLOYEE Table

SELECT EDLEVEL, **GROUPING(EDLEVEL)**,
DECIMAL(AVG(SALARY),8,2)
   AS AVG_SALARY
FROM EMPLOYEE
GROUP BY ROLLUP(EDLEVEL)

Result

EDLEVEL		AVG_SALARY
-	1	**26146.66**
14	0	27700.00
-	0	19180.00
16	0	28025.00
17	0	26250.00

GROUP BY Grouping Sets

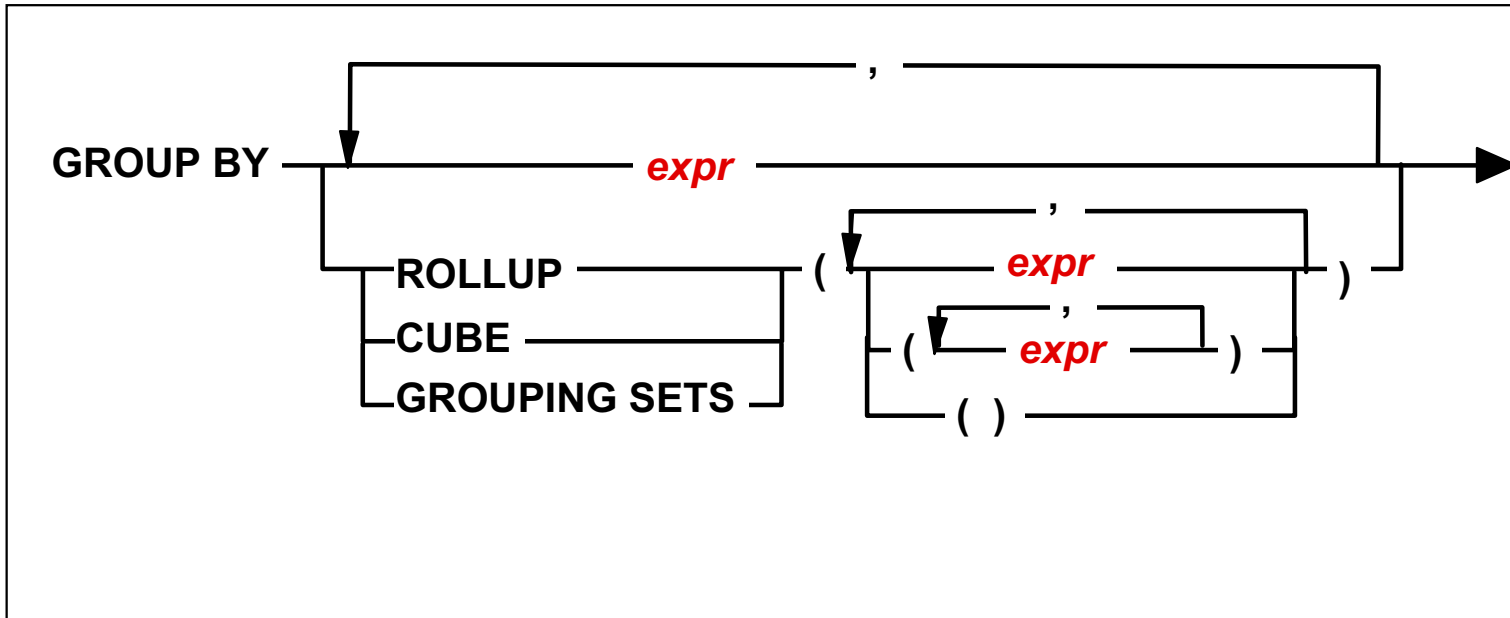| EMPNO	LASTNAME	SALARY	JOB	BIRTHDATE
000120	O'CONNELL	29250.00	CLERK	1942-10-18
000100	SPENSER	26150.00	MANAGER	1956-12-18
000130	QUINTANA	23800.00	ANALYST	1925-09-15
000280	SCHNEIDER	26250.00	OPERATOR	1936-03-28
000250	SMITH	19180.00	CLERK	1939-11-12
000060	STERN	32250.00	MANAGER	1945-07-07

EMPLOYEE Table

```
SELECT JOB, DAY(BIRTHDATE),
  DECIMAL(AVG(SALARY),8,2) AS
  AVG_SALARY
FROM EMPLOYEE
WHERE YEAR(BIRTHDATE) > 1938
GROUP BY
GROUPING SETS (DAY(BIRTHDATE), JOB)
```
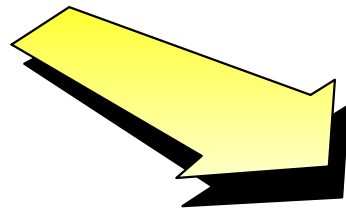
Result

JOB	DAY	AVG_SALARY
CLERK	-	24215.00
MANAGER	-	29200.00
-	7	32250.00
-	12	19180.00
-	18	27700.00

Super Groups - Syntax

GROUP BY

,

expr

ROLLUP
CUBE
GROUPING SETS

(expr ,)

(expr ,)

( )

OLAP Function for Ranking

EMPNO	LASTNAME	SALARY	JOB	BIRTHDATE
000120	O'CONNELL	29250.00	CLERK	1942-10-18
000100	SPENSER	26150.00	MANAGER	1956-12-18
000130	QUINTANA	23800.00	ANALYST	1925-09-15
000280	SCHNEIDER	26250.00	OPERATOR	1936-03-28
000250	SMITH	19180.00	CLERK	1939-11-12
000060	STERN	32250.00	MANAGER	1945-07-07

EMPLOYEE Table

```
SELECT EMPNO, LASTNAME, SALARY,
RANK () OVER (ORDER BY SALARY DESC)
AS SALARY_RANK
FROM EMPLOYEE
ORDER BY EMPNO;
```
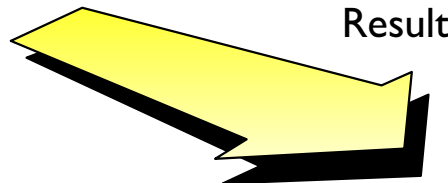
Result

EMPNO	LASTNAME	SALARY	SALARY_RANK
000060	STERN	32250,00	1
000100	SPENSER	26150,00	4
000120	O'CONNELL	29250,00	2
000130	QUINTANA	23800,00	5
000250	SMITH	19180,00	6
000280	SCHNEIDER	26250,00	3

Checkpoint

1. List three operating systems that can run DB2.

2. T/F. There is one DB2 catalog for an entire DB2 for OS/390 subsystem, which can contain many databases, and a separate DB2 catalog for each database in DB2 UDB for UNIX, Windows and OS/2.

3. Assume that you need to list the minimum, the maximum, and the average salary in your company. Which functions do you use?

4. Assume you have to prepare a report that lists each union classification on a line by itself, along with the total salary paid out to all employees in that union classification. Which clause will you need in your SQL statement to ensure that there is a separate line on the report for each union classification?

Summary

Now that you have completed this unit, you should be able to:

- Identify the purpose of the clauses in the SELECT statement

- Describe the key differences among the IBM DB2 platforms

- Describe and use the *super groups* feature (ROLLUP and CUBE)

- Describe and use the OLAP function for ranking