

SELECT 실행 순서

-문법 순서-

SELECT	- 1
FROM	- 2
WHERE	- 3
GROUP BY	- 4
HAVING	- 5
ORDER BY	- 6

-실행 순서-

FROM	- 1
WHERE	- 2
GROUP BY	- 3
HAVING	- 4
SELECT	- 5
ORDER BY	- 6

<테이블 내 지일 위에 위치한
인스턴스부터 차례대로 읽으며
조건이 맞는 인스턴스로 찾는다.>

<여러 조건들을 처리한 후 남은
인스턴스에서 어떤 열을 출력할지
선택한다.>

해당 데이터가 있는 곳을 찾아가서
조건에 맞는 데이터만 가져와서
원하는 데이터로 가공
가공한 데이터에서 조건에 맞는 것만
뽑아내서
정렬

(FROM)
(WHERE)
(GROUP BY)
(HAVING)
(SELECT)
(ORDER BY)

실행순서는 문법, 권한 검사 순서이기도 하고,
Alias 등록 순서이기도 하다.

별칭(Alias)

FROM 절에서 테이블에 Alias를 사용했다면
SELECT, ORDER BY 절에서 사용할 수 있고
SELECT 절에서 컬럼에 Alias를 사용했다면
ORDER BY 절에서 사용할 수 있다.

(FROM Table1 AS T1)
(SELECT T1.Col1, ORDER BY T1.Col1)
(SELECT T1.Col1 AS a)
(ORDER BY AS a)

ORDER BY절에 T1.a가 안되는 것으로 보아
a는 T1.col1을 대신하는 것 같다.

【출처】 [SELECT 실행 순서](#) | 작성자 [연항지](#)

FROM

	FIRST_NAME	SALARY	JOB_ID
1	Steven	24000	AD_PRES
2	Neena	17000	AD_VP
3	Lex	17000	AD_VP
4	Alexander	9000	IT_PROG
5	Bruce	6000	IT_PROG
6	David	4000	IT_PROG
7	Valli	4000	IT_PROG
8	Diana	4200	IT_PROG
9	Nancy	12000	FI_MGR
10	Daniel	9000	FI_ACCOUNT
11	John	8200	FI_ACCOUNT
12	Isabel	7700	FI_ACCOUNT
13	Jose Manuel	7800	FI_ACCOUNT
14	Luis	6900	FI_ACCOUNT
15	Den	12000	PU_MGR
16	Alexander	3100	PU_CLERK
17	Shelli	2900	PU_CLERK
18	Sigal	2000	PU_CLERK
19	Guy	2000	PU_CLERK

WHERE

	FIRST_NAME	SALARY	JOB_ID
1	Steven	24000	AD_PRES
2	Neena	17000	AD_VP
3	Lex	17000	AD_VP
4	Alexander	9000	IT_PROG
5	Bruce	6000	IT_PROG
6	David	4000	IT_PROG
7	Valli	4000	IT_PROG
8	Diana	4200	IT_PROG
9	Nancy	12000	FI_MGR
10	Daniel	9000	FI_ACCOUNT
11	John	8200	FI_ACCOUNT
12	Isabel	7700	FI_ACCOUNT
13	Jose Manuel	7800	FI_ACCOUNT
14	Luis	6900	FI_ACCOUNT
15	Den	12000	PU_MGR
16	Alexander	3100	PU_CLERK
17	Shelli	2900	PU_CLERK
18	Sigal	2000	PU_CLERK
19	Guy	2000	PU_CLERK

GROUP BY

	FIRST_NAME	SALARY	JOB_ID
1	Steven	24000	AD_PRES
2	Neena	17000	AD_VP
3	Lex	17000	AD_VP
4	Alexander	9000	IT_PROG
5	Bruce	6000	IT_PROG
6	David	4000	IT_PROG
7	Valli	4000	IT_PROG
8	Diana	4200	IT_PROG
9	Nancy	12000	FI_MGR
10	Daniel	9000	FI_ACCOUNT
11	John	8200	FI_ACCOUNT
12	Isabel	7700	FI_ACCOUNT
13	Jose Manuel	7800	FI_ACCOUNT
14	Luis	6900	FI_ACCOUNT
15	Den	12000	PU_MGR
16	Alexander	3100	PU_CLERK
17	Shelli	2900	PU_CLERK
18	Sigal	2000	PU_CLERK
19	Guy	2000	PU_CLERK

HAVING

	FIRST_NAME	SALARY	JOB_ID
1	Steven	24000	AD_PRES
2	Neena	17000	AD_VP
3	Lex	17000	AD_VP
4	Alexander	9000	IT_PROG
5	Bruce	6000	IT_PROG
6	David	4000	IT_PROG
7	Valli	4000	IT_PROG
8	Diana	4200	IT_PROG
9	Nancy	12000	FI_MGR
10	Daniel	9000	FI_ACCOUNT
11	John	8200	FI_ACCOUNT
12	Isabel	7700	FI_ACCOUNT
13	Jose Manuel	7800	FI_ACCOUNT
14	Luis	6900	FI_ACCOUNT
15	Den	12000	PU_MGR
16	Alexander	3100	PU_CLERK
17	Shelli	2900	PU_CLERK
18	Sigal	2000	PU_CLERK
19	Guy	2000	PU_CLERK

SELECT

	JOB_ID	SAL_AVG
1	SA_MGR	13750
2	AD_VP	17000

ORDER BY

	JOB_ID	SAL_AVG
1	AD_VP	17000
2	SA_MGR	13750

여러가지 쿼리된 데이터를
중에서 한 행씩 가져와
'SELECT' 값을 실어준다.

이처럼, 과다한 분할이 발생