# SQL활용

SQL 활용 단원 정리





# SQL 활용 단원 정리

이번 영상에서는 SQL 활용 단원 정리를 하겠습니다.



연산자는 비교, 논리, 특수, 산술, 집합 연산자로 나뉩니다.



구분	연산자	설명
비교 연산자	= / > / < / >= / <= / <>	같음 / 보다 큼 / 보다 작음 / 크거나 같음 / 작거나 같음 / <mark>같지 않음</mark>
	AND / OR	앞, 뒤 조건 모두 만족 / 하나라도 만족
논리 연산자	NOT	뒤에 오는 조건과 <mark>반대</mark>
	BETWEEN a AND b / NOT BETWEEN a AND b	a와 b의 값 사이 / a와 b의 값 사이가 <mark>아님</mark>
	IN (List) / NOT IN (List)	리스트(List) 값 / 리스트(List) 값이 <mark>아님</mark>
특수 연산자	LIKE '비교문자열'	비교문자열과 같음
국구 한인사	NOT LIKE '비교문자열'	비교문자열이 <mark>아님</mark>
	IS NULL	NULL과 같음
	IS NOT NULL	NULL이 아님
산술 연산자	+,-,*,/	덧셈, 뺄셈, 곱셉, 나눗셈
	UNION	2개 이상 테이블 중복된 행 제거 하여 집합(* 열 개수와 데이터 타입 일치)
집합 연산자	UNION ALL	2개 이상 테이블 중복된 행 제거 <mark>없이</mark> 집합(* 열 개수와 데이터 타입 일치)



- UNION: 2개 이상 테이블의 중복된 행들을 제거 하여 집합

- UNION ALL: 2개 이상 테이블의 중복된 행들을 제거 없이 집합

\* 열 개수와 데이터 타입 일치



### < UNION >

order_no	mem_no	order_date	product_code	sales_qty
1	1000970	2019-05-02	505	2
2	1000970	2019-05-03	75	3



order_no	mem_no	order_date	product_code	sales_qty
2	1000970	2019-05-03	75	3
3	1000021	2019-05-07	494	2



order_no	mem_no	order_date	product_code	sales_qty
1	1000970	2019-05-02	505	2
2	1000970	2019-05-03	75	3
3	1000021	2019-05-07	494	2

### < UNION ALL >

order_no	mem_no	order_date	product_code	sales_qty
1	1000970	2019-05-02	505	2
2	1000970	2019-05-03	75	3



order_no	mem_no	order_date	product_code	sales_qty
2	1000970	2019-05-03	75	3
3	1000021	2019-05-07	494	2



order_no	mem_no	order_date	product_code	sales_qty
1	1000970	2019-05-02	505	2
2	1000970	2019-05-03	75	3
2	1000970	2019-05-03	75	3
3	1000021	2019-05-07	494	2

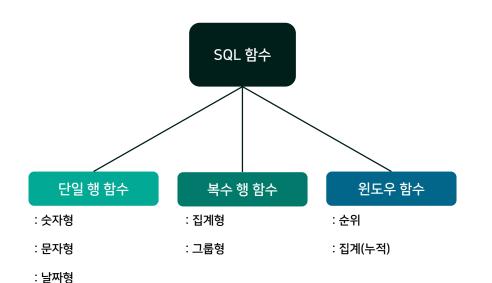


### 함수 - 단일, 복수, 윈도우 함수

- 단일 행 함수: 숫자형, 문자형, 날짜형, 형변환, 일반함수

- <mark>복수</mark> 행 함수: 집계, 그룹 함수

- 윈도우 함수: 순위, 집계(누적) 함수



구분	함수	설명
숫자형 함수	ABS(숫자)	절대값 반환
문자형 함수	LOWER(문자) / UPPER(문자)	소문자 / 대문자 반환
날짜형 함수	YEAR / MONTH / DAY(날짜)	연 / 월 / 일 반환
형변환 함수	DATE_FORMAT(날짜, 형식)	날짜형식으로 변환
일반 함수	IFNULL(A, B)	A가 NULL이면 B를 반환, 아니면 A 반환

구분	함수	설명
집계 함수	COUNT / SUM / AVG / MAX / MIN	행수 / 합계 / 평균 / 최대 / 최소
그룹함수	WITH ROLLUP	GROUP BY 열들을 오른쪽에서 왼쪽순으로 그룹

구분	함수	설명
	ROW_NUMBER	고유한 순위 반환
순위 함수	RANK	동일한 값이면 동일한 순위 반환
	DENSE_RANK	동일한 값이면 동일한 순위 반환(+ 하나의 등수로)
집계 함수 ( <mark>누적</mark> )	COUNT / SUM / AVG / MAX / MIN	<mark>누적</mark> 행수 / 합계 / 평균 / 최대 / 최소



: 형 변환

: 일반

- <mark>단일</mark> 행 함수: 모든 행에 대하여 <mark>각각</mark> 함수가 적용되어 반환
- 복수 행 함수: 여러 행들이 하나의 결과값으로 반환

### <단일 행 함수 >

		Customer				
mem_no	gender	birthday	addr	join_date	연도반환	YEAR(join_date)
1000001	man	1942-07-27	Seoul	2019-04-30	<b>→</b>	2019
1000002	man	1983-08-21	Seoul	2019-04-30	<b></b>	2019
1000004	man	1955-01-24	Busan	2019-05-05	<b>→</b>	2019
1002638	women	1988-10-06	Jeju	2020-09-22	<b>→</b>	2020
1002639	women	1990-01-04	Gangwon	2020-09-22	<b></b>	2020
1002640	women	1991-01-30	Incheon	2020-09-22	<b></b>	2020
1002641	man	1993-01-06	Jeju	2020-09-22	<b></b>	2020
92642	w .	1977-09	Jeju	9-23		2020
		##.	1			

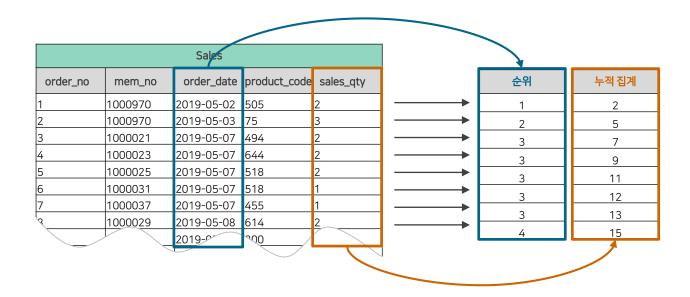
### < 복수 행 함수 >

		Sales			
order_no	mem_no	order_date	product_code	sales_qty	
1	1000970	2019-05-02	505	2	
2	1000970	2019-05-03	75	3	ı
3	1000021	2019-05-07	494	2	ı
4	1000023	2019-05-07	644	2	
5	1000025	2019-05-07	518	2	
6	1000031	2019-05-07	518	1	
7	1000037	2019-05-07	455	1	
8	1000029	2019-05-08	614	2	
	_	2019-0-	700		



- 윈도우 함수: 행과 행간의 관계를 정의하여 결과 값을 반환

### < 윈도우 행 함수 >



- 윈도우 함수 기본 문법(ORDER BY)

함수

OVER (ORDER BY 열 ASC or DESC)

- 윈도우 함수 기본 문법(ORDER BY) + PARTITION BY(그룹화)

함수

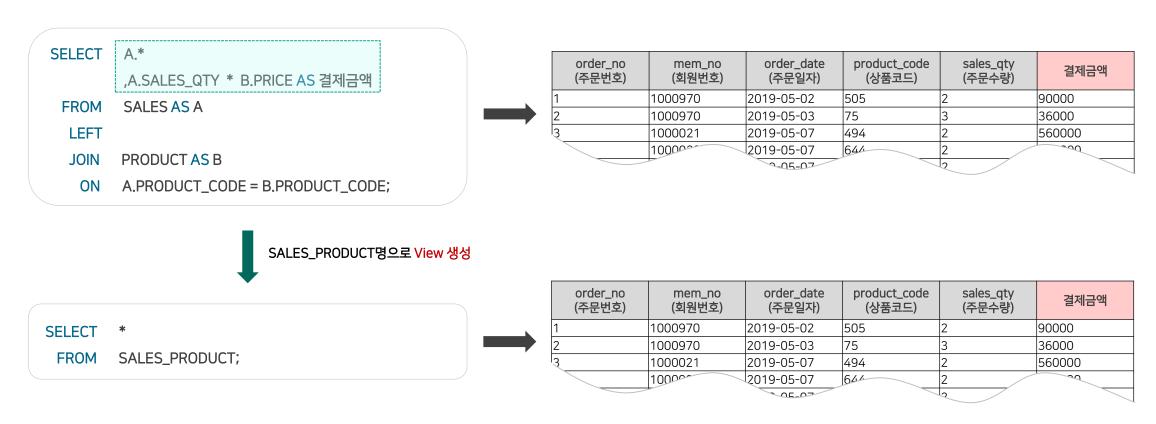
OVER (PARTITION BY 9 OVER BY 9 ASC or DESC)



### View 및 Procedure - View

View는 하나 이상의 테이블들을 활용하여, 사용자가 정의한 가상 테이블입니다.

- JOIN 사용을 <u>최소화</u>하여, 편의성을 <u>최대화</u>합니다.
- View 테이블은 가상 테이블이기 때문에, 중복되는 열이 저장될 수 없습니다.





### View 및 Procedure - Procedure

Procedure는 매개변수를 활용해, 사용자가 정의한 작업을 저장합니다.

- IN: 매개 변수를 프로시저로 <mark>전달</mark>

- OUT: 프로시저 결과값 <mark>반환</mark>

- INOUT: 매개 변수를 프로시저로 <mark>전달</mark>, 프로시저 결과값 <mark>반환</mark>

### < Procedure 작업 저장 >

```
DELIMITER //
CREATE PROCEDURE [프로시저명] (IN 매개변수 VARCHAR(20))
BEGIN

SELECT *
FROM CUSTOMER
WHERE GENDER + 매개변수;
END //
DELIMITER;
```

# IN 매개변수 : 매개 변수를 프로시저로 전달 : 매개 변수를 프로시저로 전달 : 프로시저 결과값 반환 Procedure INOUT 매개변수 : 매개 변수를 프로시저로 전달 : 프로시저 결과값 반환 Procedure

### 데이터 마트는 분석에 필요한 데이터를 가공한 분석용 데이터입니다.

- 요약 변수: 수집된 데이터를 분석에 맞게 종합한 변수(기간별 구매 금액, 횟수, 수량 등)

- 파생 변수 : 사용자가 특정 조건 또는 함수로 의미를 부여한 변수(연령대, 선호 카테고리 등)

- 데이터 정합성: 데이터가 서로 모순 없이 일관되게 일치함을 나타낼 때 사용

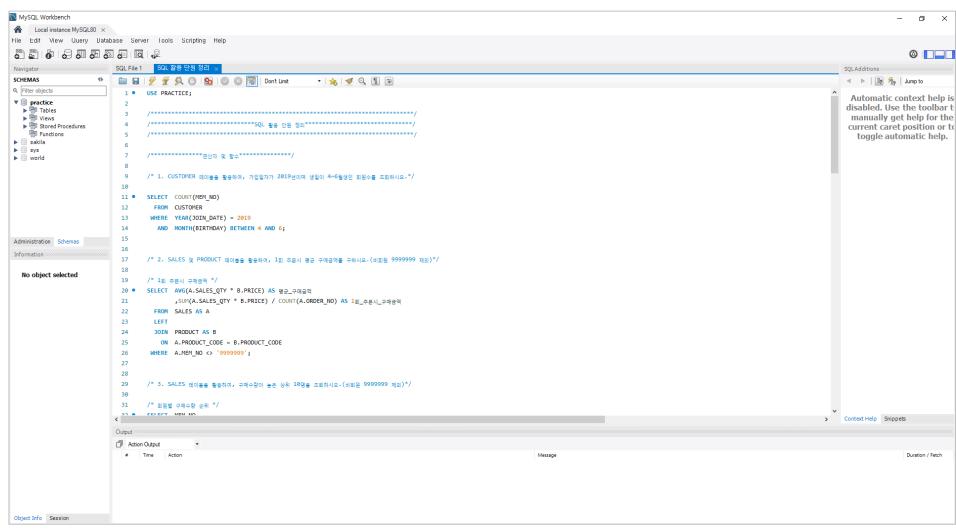


### < 데이터 정합성 >

- 1. 데이터 마트의 회원 수의 <mark>중복</mark>은 없는가?
- 데이터 마트의 요약 및 파생변수의 오류는 없는가?
- 3. 데이터 마트의 구매자 비중(%)의 오류는 없는가?



### SQL 활용 단원 정리



## End of Document

Thank you.



