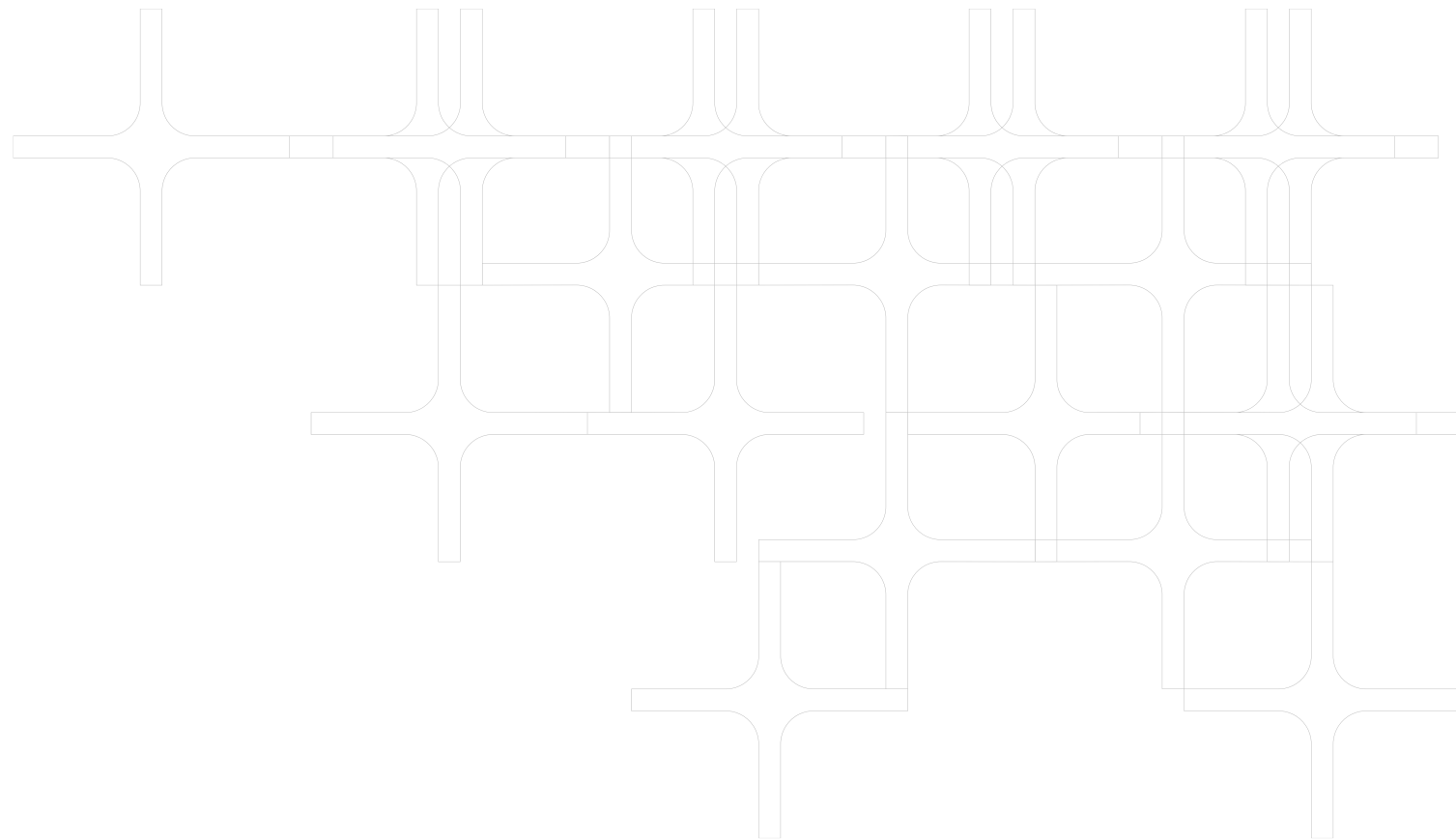


# SQL 문법

---

## 테이블 결합(JOIN)



# 테이블 결합(JOIN)

이번 영상에서는  
테이블 결합 개념을 학습합니다.

관계는 1:1, 1:N, N:N 세가지 형태로,  
테이블 간의 연결이 가능하다는 것을 의미합니다.

회원 및 주문 테이블은 **회원번호** 기준으로 1:N 관계이다.

ex. 1명의 회원이 여러 번(N) 주문이 가능하다.

회원 테이블				
회원번호	이름	성별	나이	가입일자
1001	홍길동	남	28	2020-01-02
1002	이순신	남	43	2020-01-03
1003	장영실	남	22	2020-01-04
1004	유관수	여	33	2020-01-05

주문 테이블				
주문번호	구매일자	제품명	구매금액	회원번호
A10001	2020-02-01	티셔츠	80,000	1001
A10001	2020-02-02	긴바지	40,000	1001
A10002	2020-02-03	반바지	90,000	1002
A10002	2020-02-05	가디건	50,000	1002
A10003	2020-02-08	양말	10,000	1003

# 테이블 결합(JOIN) - 관계를 활용하여, 테이블을 결합하자.

테이블 결합(JOIN)은 두 테이블 관계를 활용하여, 테이블을 결합하는 명령어입니다.

Customer 및 Sales 테이블은 **mem\_no(회원번호)** 기준으로 1:N 관계이며,  
테이블 결합이 가능합니다.

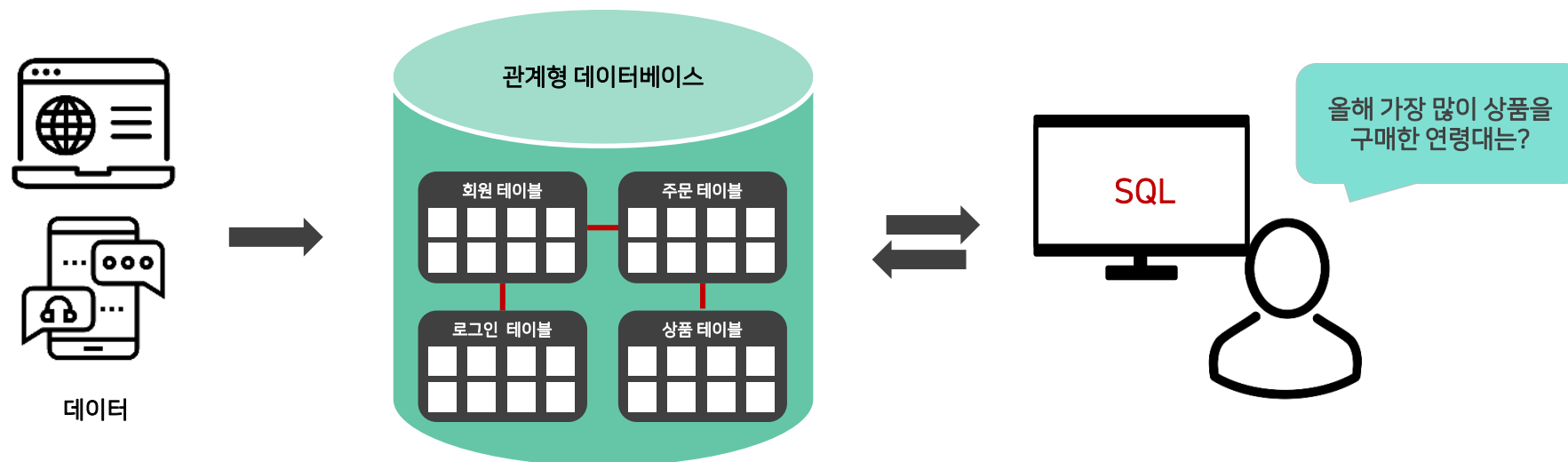
Customer				
mem_no	gender	birthday	addr	join_date
1000001	man	1942-07-27	Seoul	2019-04-30
1000002	man	1983-08-21	Seoul	2019-04-30
1000003	women	1945-12-09	Gyeonggi	2019-05-01
1000004	man	1955-01-24	Busan	2019-05-05
1000005	man	1955-03-14	Daejeon	2019-05-05
1000006	man	1955-03-14	Daejeon	2019-05-05

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
3	1000021	2019-05-07	494	2
4	1000023	2019-05-07	644	2
5	1000023	2019-05-07	518	2

# 테이블 결합(JOIN) - 테이블을 결합하여, 분석하자.

테이블 결합(JOIN)

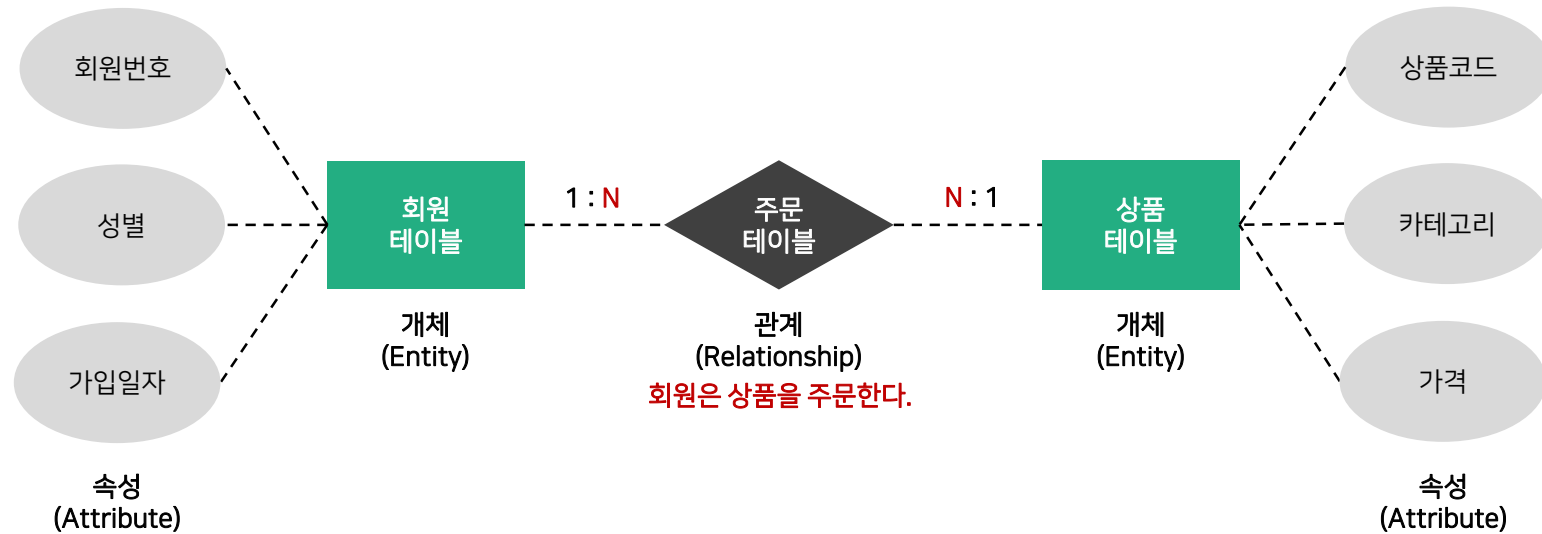
테이블 결합(JOIN)을 통해, 여러 테이블을 활용하여 분석이 가능합니다.



ERM(Entity-Relationship Modelling)은 개체-관계 모델링이며, 관계형 데이터베이스에 테이블을 모델링할 때 사용됩니다.

\* 개체(Entity): 하나 이상의 속성(Attribute)으로 구성된 개체

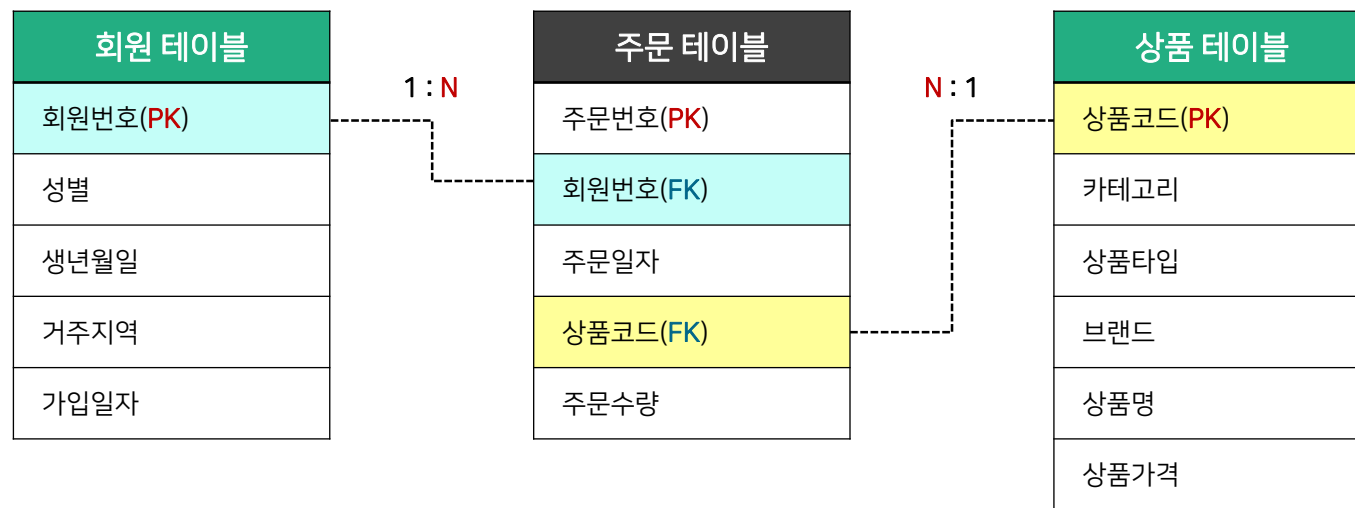
\* 관계(Relationship): 속성(Entity)들 간의 관계



# ERD(Entity-Relationship Diagram)

테이블 결합(JOIN)

ERD(Entity-Relationship Diagram)은 개체 간의 관계를 도표로 표현할 때 사용됩니다.



## FK(FOREIGN KEY)

: 다른 테이블에서 PK(PRIMARY KEY)

(Ex. 주문테이블의 회원번호는 회원테이블의 PK)

# 실습 테이블(Sales, Product) 가져오기

테이블 결합(JOIN)

실습 테이블(Sales, Product)은 주문 및 상품테이블입니다.

Sales				
order_no (주문번호)	mem_no (회원번호)	order_date (주문일자)	product_code (상품코드)	sales_qty (주문수량)
1	1000970	2019-05-02	505	2
2	1000970	2019-05-03	75	3
3	1000021	2019-05-07	494	2
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
	1000029	2019-05-08	614	2
		2019-05-09	30	

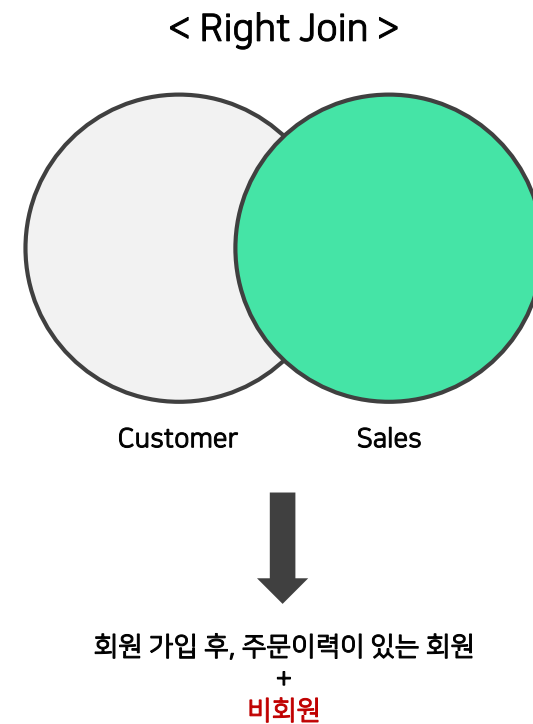
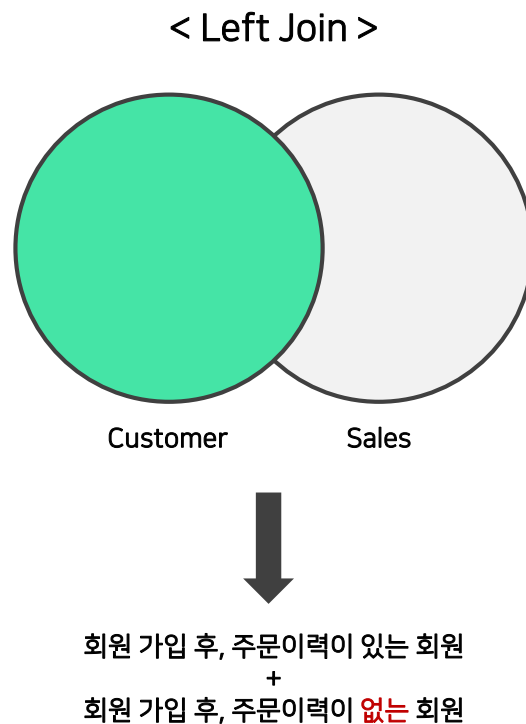
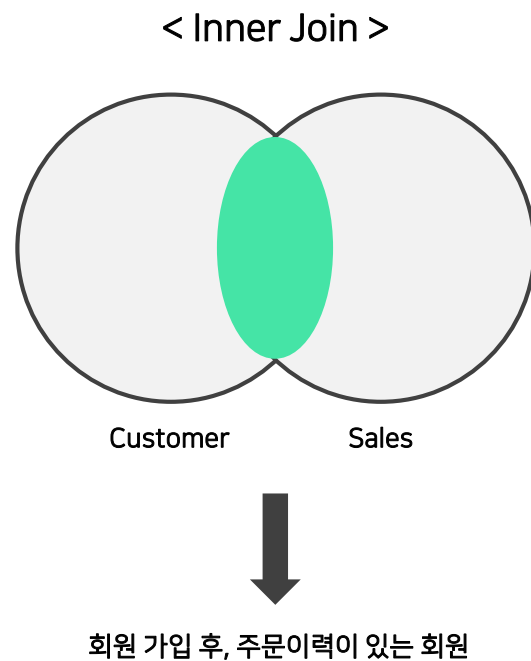


Product					
product_code (상품코드)	category (카테고리)	type (상품타입)	brand (브랜드)	product_name (상품명)	price (상품가격)
1	electronic	camera	sony	sony new alpha	880000
2	electronic	camera	canon	canon powershot	650000
3	electronic	camera	olympus	olympus tough	730000
4	electronic	television	tcl	tcl roku	930000
5	electronic	television	samsung	samsung class	1250000
6	electronic	television	lg	lg webos	1300000
7	electronic	headphone	samsung	samsung galaxy	90000
8	electronic	headphone	bose	bose	170000
	electronic	headphone	sony	sony professional	190000





- Inner Join: 두 테이블의 공통 값이 **매칭**되는 데이터만 결합
- left Join: 두 테이블의 공통 값이 매칭되는 데이터만 결합 + **왼쪽** 테이블의 매칭되는 않는 데이터는 NULL
- Right Join: 두 테이블의 공통 값이 매칭되는 데이터만 결합 + **오른쪽** 테이블의 매칭되는 않는 데이터는 NULL



테이블 결합(JOIN) 및 데이터 조회(SELECT)를 활용하여, 여러 테이블들을 결합하여 조회 및 분석합니다.

회원 및 주문테이블 **결합**

FROM 절 + JOIN

: 테이블 확인 및 **결합**

성별이 남성 조건으로 필터링하여

WHERE 절

: **FROM**절 테이블을 특정 조건으로 필터링

거주지역별로 **구매횟수** 집계

GROUP BY 절

: 열 별로 **그룹화**

**구매횟수** 100회 미만 조건으로 필터링

HAVING 절

: 그룹화된 **새로운 테이블**을 특정 조건으로 필터링

모든 열 조회

SELECT 절

: 열 **선택**

**구매횟수**가 낮은 순으로

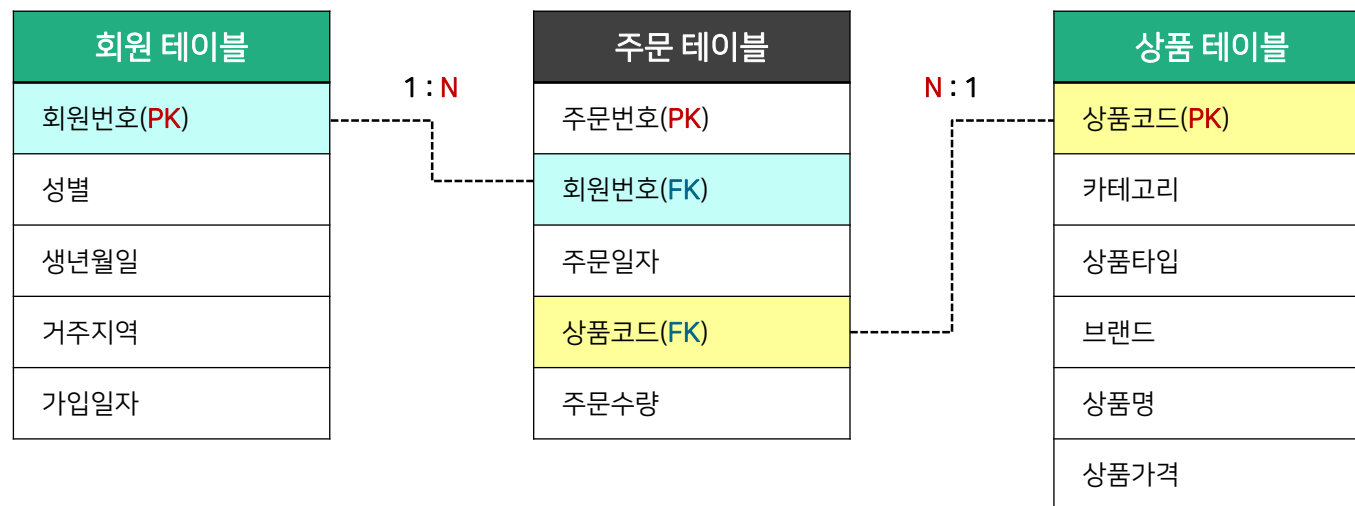
ORDER BY 절

: 열 **정렬**

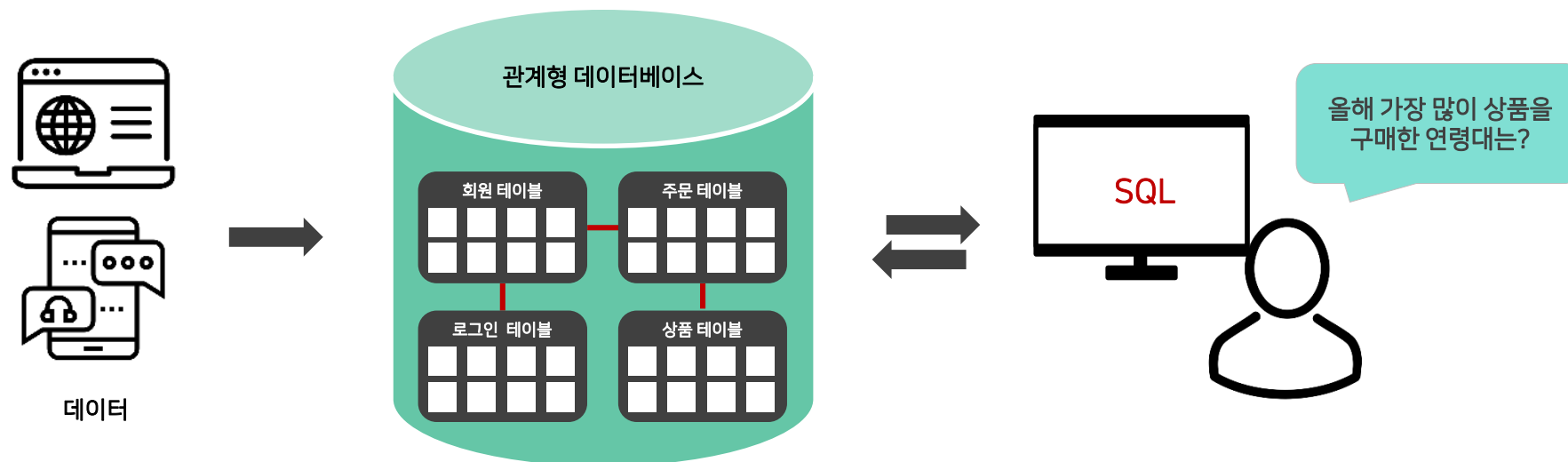
테이블 결합(JOIN)은 3가지 이상 테이블 결합이 가능합니다.

주문 및 상품 테이블은 상품코드 기준으로 N:1 관계이다.

ex. 1개의 상품이 여러 번(N) 주문이 가능하다.



테이블 결합(JOIN)을 통해, 여러 테이블을 활용하여 분석이 가능합니다.



# End of Document

-

# Thank you.

