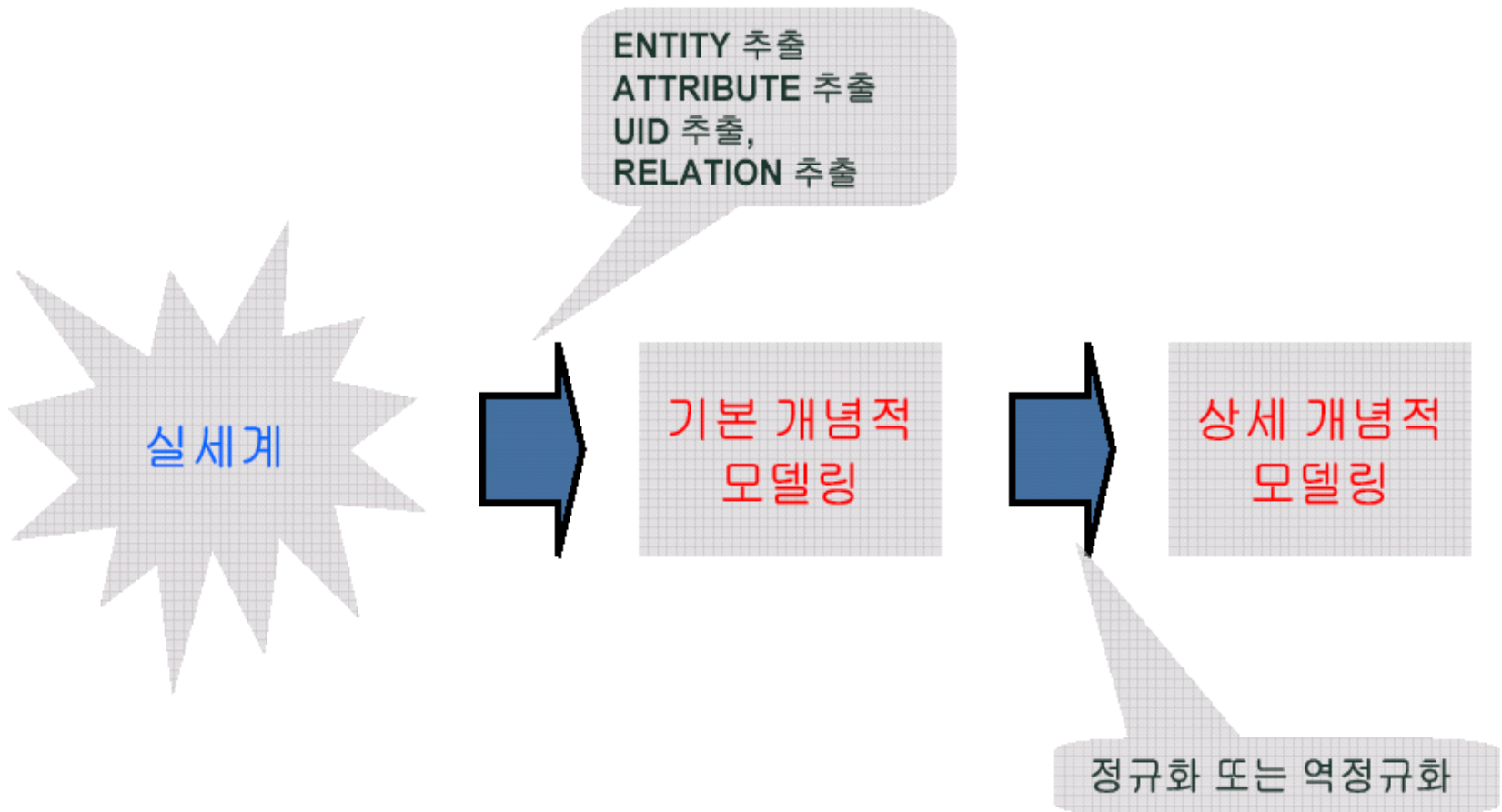


상세 개념적 모델링



정규화를 하는 이유

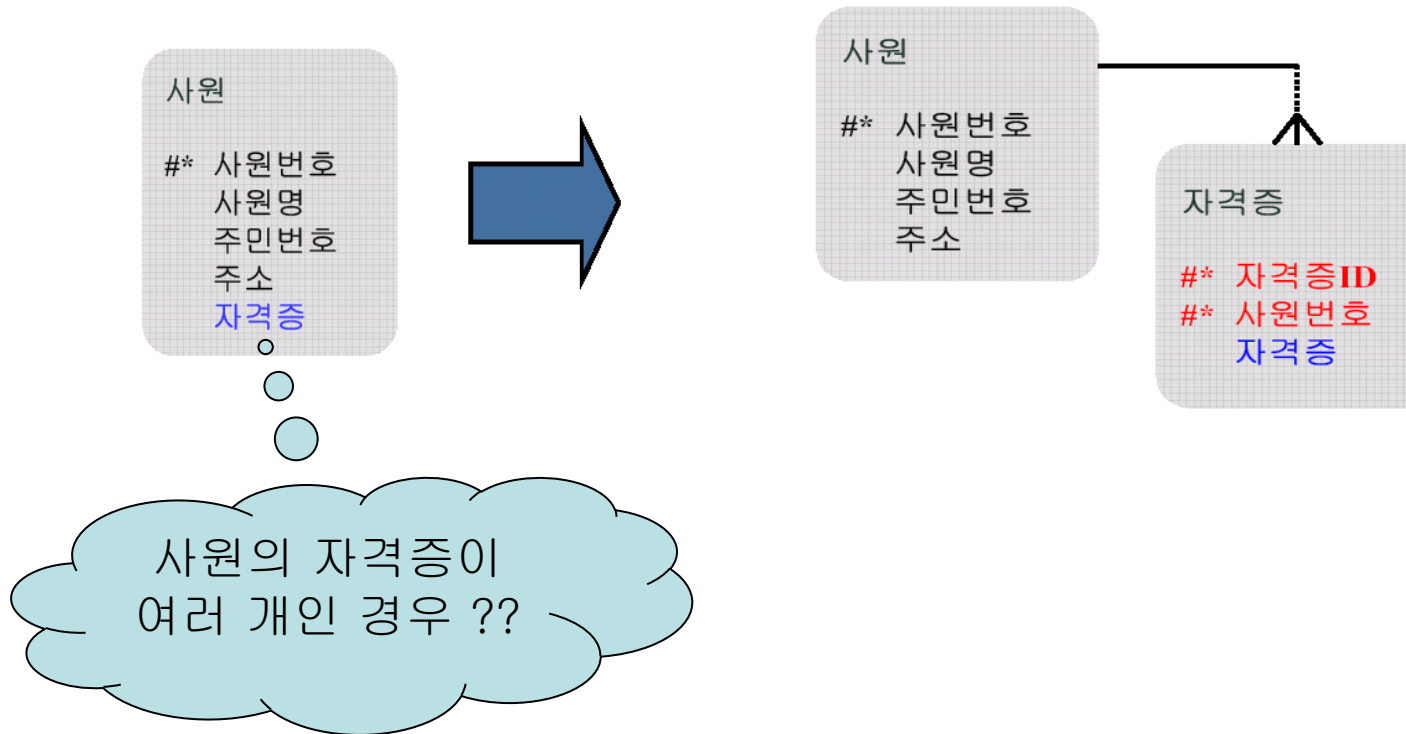
- 데이터의 중복성 제거
- 데이터 모형의 단순화
- Entity, Attribute의 누락 여부검증
- 데이터 모형의 안전성 검증

정규화

1. 제1정규화 : 엔티티내의 모든 속성은 반드시 하나의 값을 가져야 한다.
2. 제2정규화 : 복합키를 식별자로 가지는 경우 복합키의 일부분에만 종속적인 항목을 분류하는 작업.
3. 제3정규화 : 엔티티 내의 식별자를 제외한 모든 속성은 종속될 수 없다.

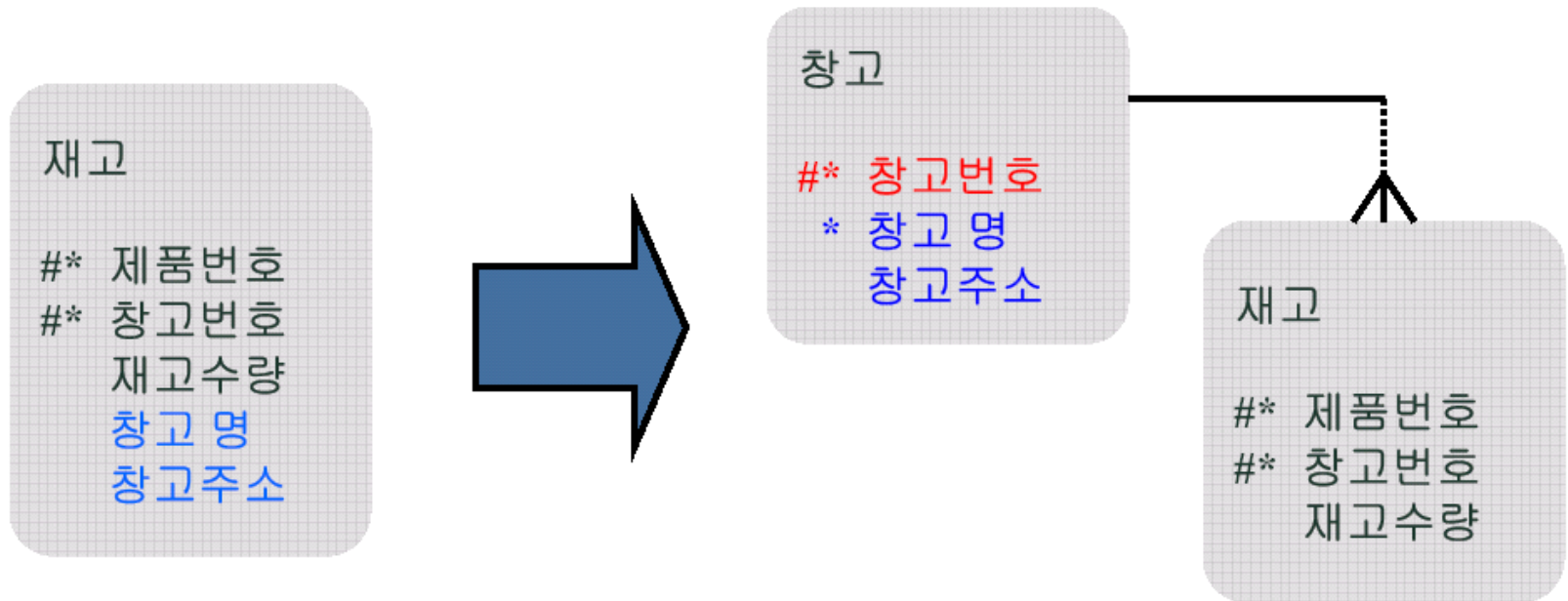
제1 정규화

- 새로운 엔티티를 생성하고 UID를 부여한 다음 1:N의 관계를 부여한다.



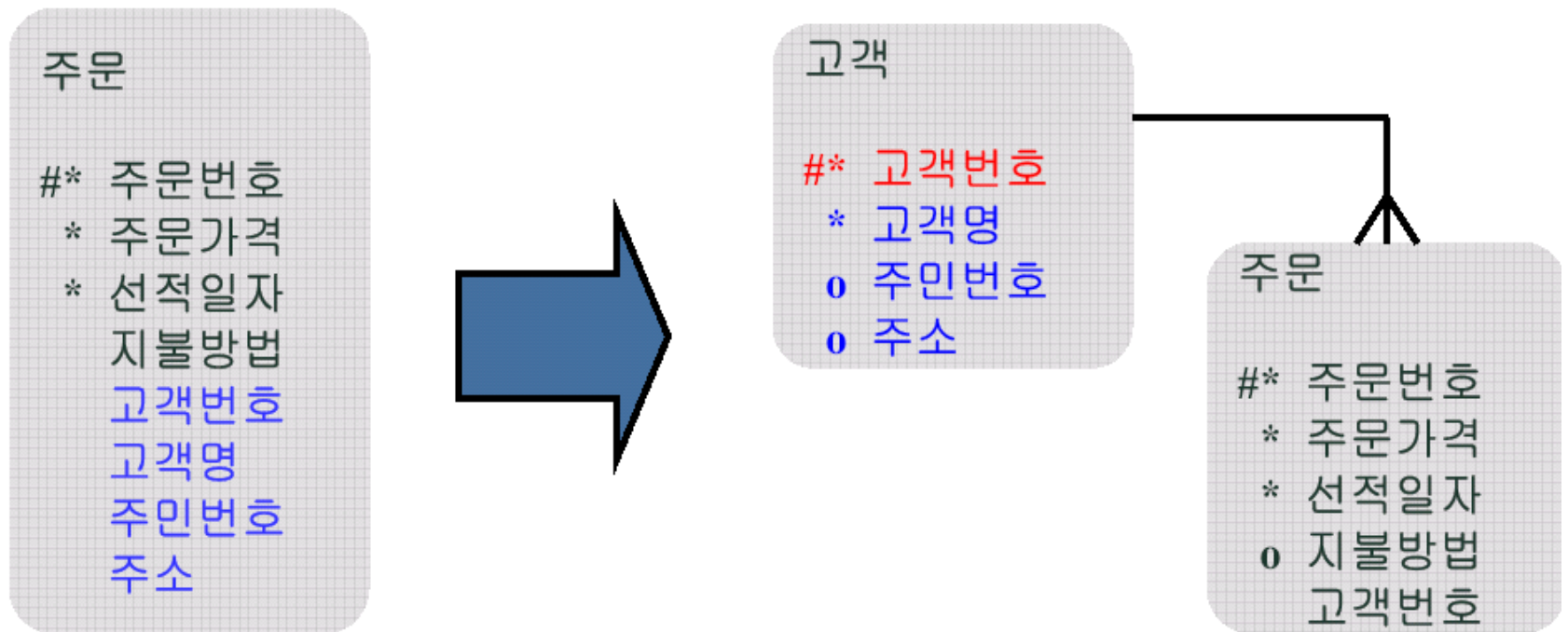
제2정규화

- 새로운 엔티티를 생성하고 UID를 부여한 다음 1:N의 관계를 부여한다.



제3정규화

- 새로운 엔티티를 생성하고 UID를 부여한 다음 1:N의 관계를 부여한다.



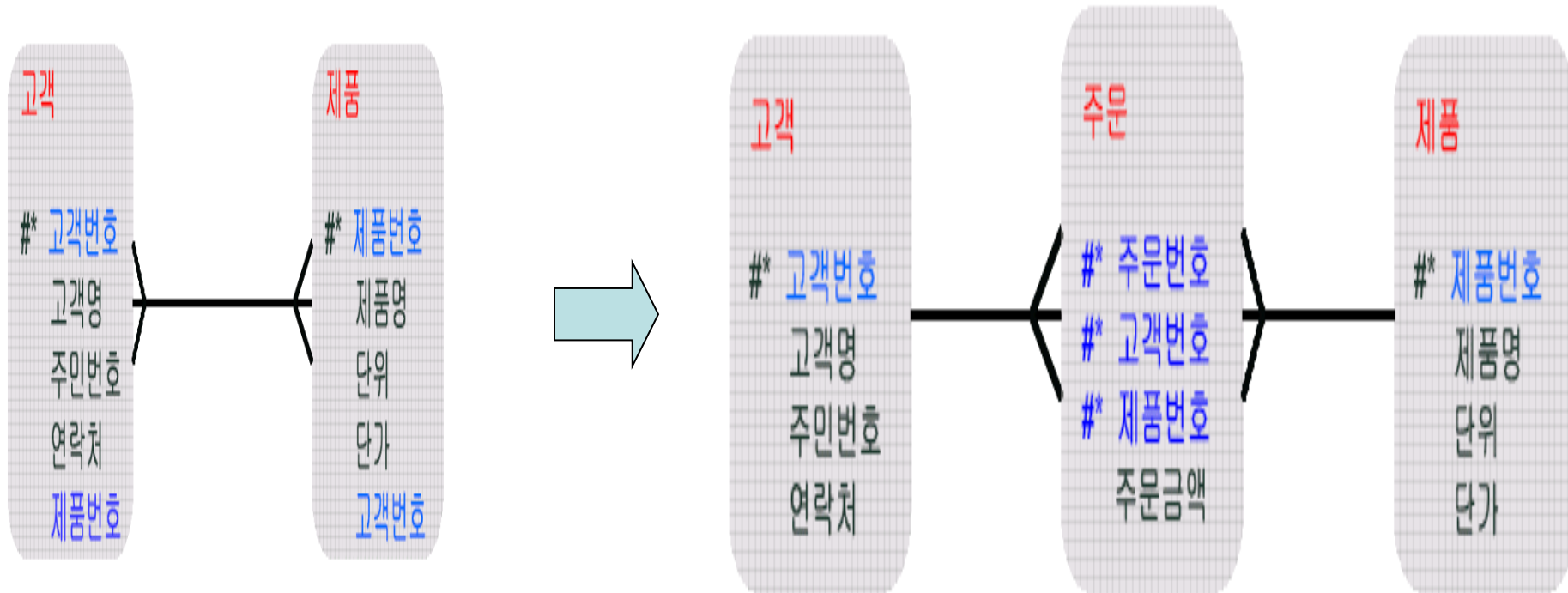
정규화의 장단점

정규화의 정도가 높은 경우	정규화의 정도가 낮은 경우
유연한 데이터의 구축	데이터의 결합처리가 감소
데이터의 정확성이 높다.	물리적 접근이 단순
물리적 접근이 복잡	데이터에 많은 LOCK이 발생
길이가 짧은 데이터가 생김	길이가 긴 데이터가 생길 수 있다.

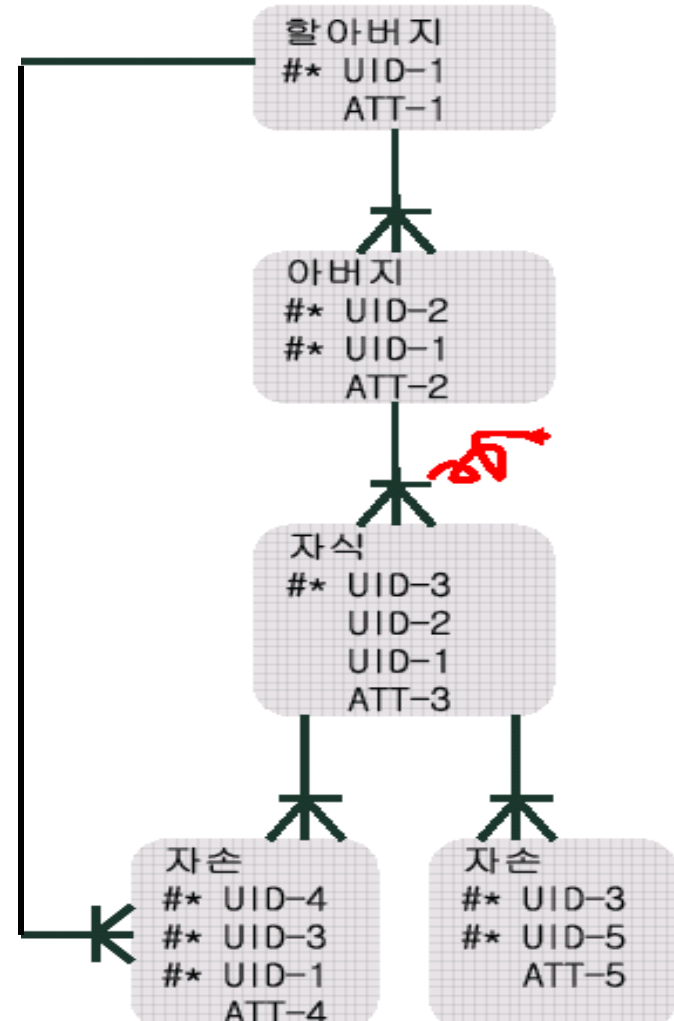
과도한 정규화 작업을 피해야 하는 경우

- 대상 엔티티가 검색만으로 사용되는 경우
- 다른 엔티티의 Attribute 와 함께 사용되는 빈도가 적은 경우
- 소규모의 데이터를 저장하는 엔티티이거나 드물게 변경되는 경우

N:M 관계의 해소



식별자바



논리적 데이터베이스 설계

관계형 데이터베이스의 개념

Primary
key

Column

Table

PNO	PNAME	JOB	MANAGER	STARTDATE	PAY	BONUS	DNO
1111	SMITH	MANAGER	1001	17-DEC-90	1000		10
1112	ALLY	SALESMAN	1116	20-FEB-91	1600	500	30
1113	WORD	SALESMAN	1116	24-FEB-92	1450	300	30
1114	JAMES	MANAGER	1001	12-APR-90	3975		20
1001	BILL	PRESIDENT		10-JAN-89	7000		10
1116	JOHNSON	MANAGER	1001	01-MAY-91	3550		30
1118	MARTIN	ANALYST	1111	09-SEP-91	3450		10
1121	KIM	CLEARC	1114	08-DEC-90	4000		20
1123	LEE	SALESMAN	1116	23-SEP-91	1200	0	30
1123	PARK	ANALYST	1111	03-JAN-90	2500		10

Row
10

Field

Not
Null

Null

PNO PNAME	JOB	MANAGER	STARTDATE	PAY	BONUS	DNO	DNO DNAME	PHONE	POSITION
1111 SMITH	MANAGER	1001	17-DEC-90	1000		10	10 FINANCE	032-277-0411	INCHON
1112 ALLY	SALESMAN	1116	20-FEB-91	1600	500	30	20 RESEARCH	061-535-1212	BUSAN
1113 WORD	SALESMAN	1116	24-FEB-92	1450	300	30	30 SALES	02-555-4985	SEOUL
1114 JAMES	MANAGER	1001	12-APR-90	3975		20	40 MARKETING	031-284-3800	SUWON
1001 BILL	PRESIDENT		10-JAN-89	7000		10			
1116 JOHNSON	MANAGER	1001	01-MAY-91	3550		30			
1118 MARTIN	ANALYST	1111	09-SEP-91	3450		10			
1121 KIM	CLERK	1114	08-DEC-90	4000		20			
1123 LEE	SALESMAN	1116	23-SEP-91	1200	0	30			
1123 PARK	ANALYST	1111	03-JAN-90	2500		10			

Foreign
key

Primary
key

매핑

개념적 모델링	논리적 DB 설계
Entity	Table
Attribute	Column
UID	Primary key
Relationship	Foreign key
Attribute의 Mandatory Optional	Not null null

Entity -> Table

- 테이블명은 Naming-rule에 의해 작성하되 너무 길지 않도록
- ERD의 entity명을 테이블명으로 사용하는 것이 좋다.

Attribute -> Column

- 예약어를 컬럼명 사용말것
- 가능하면 짧게

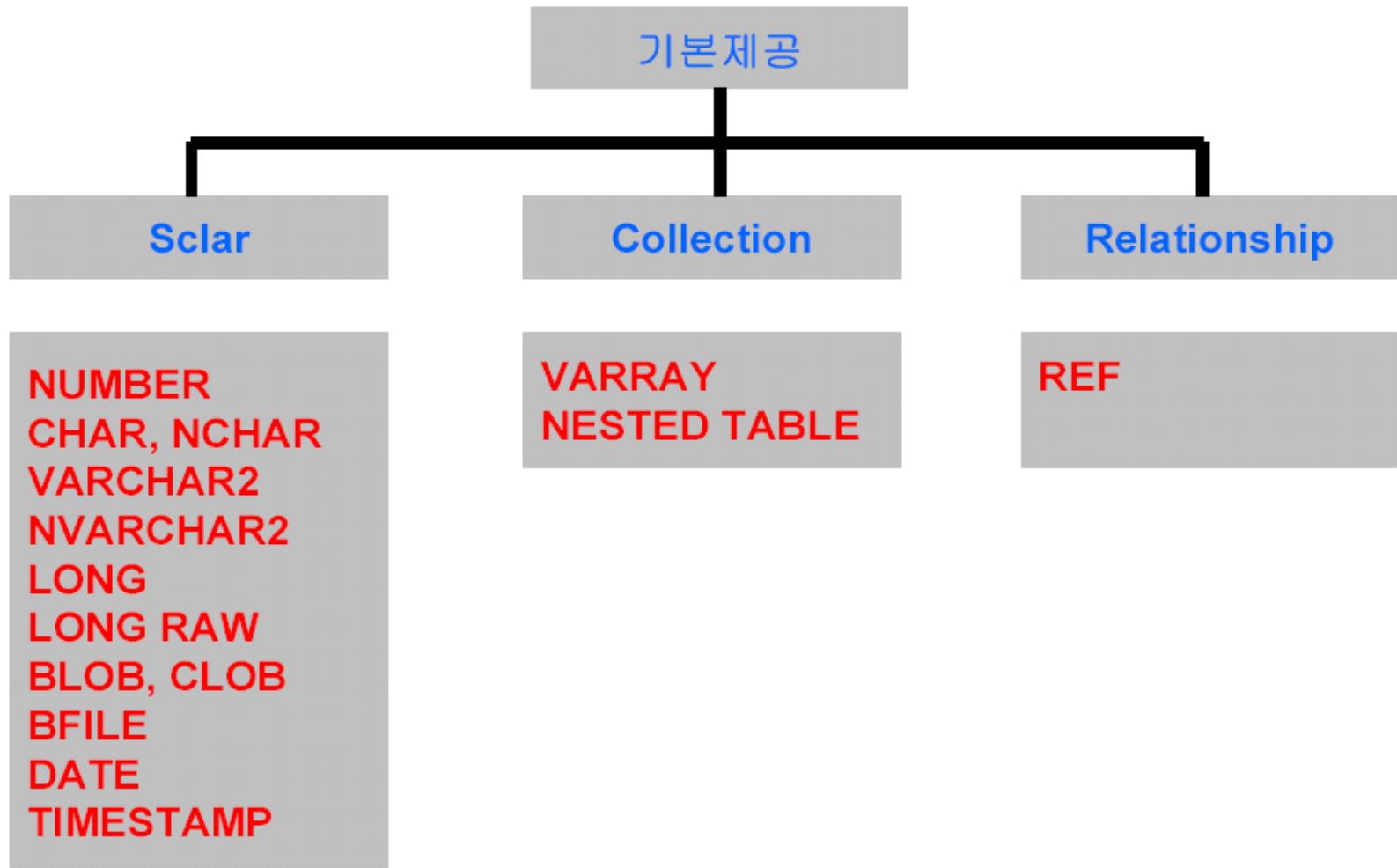
식별자 -> Primary key

- 여러 개의 컬럼으로 구성할 때는 제약명에 u1,u2... 등으로 표기

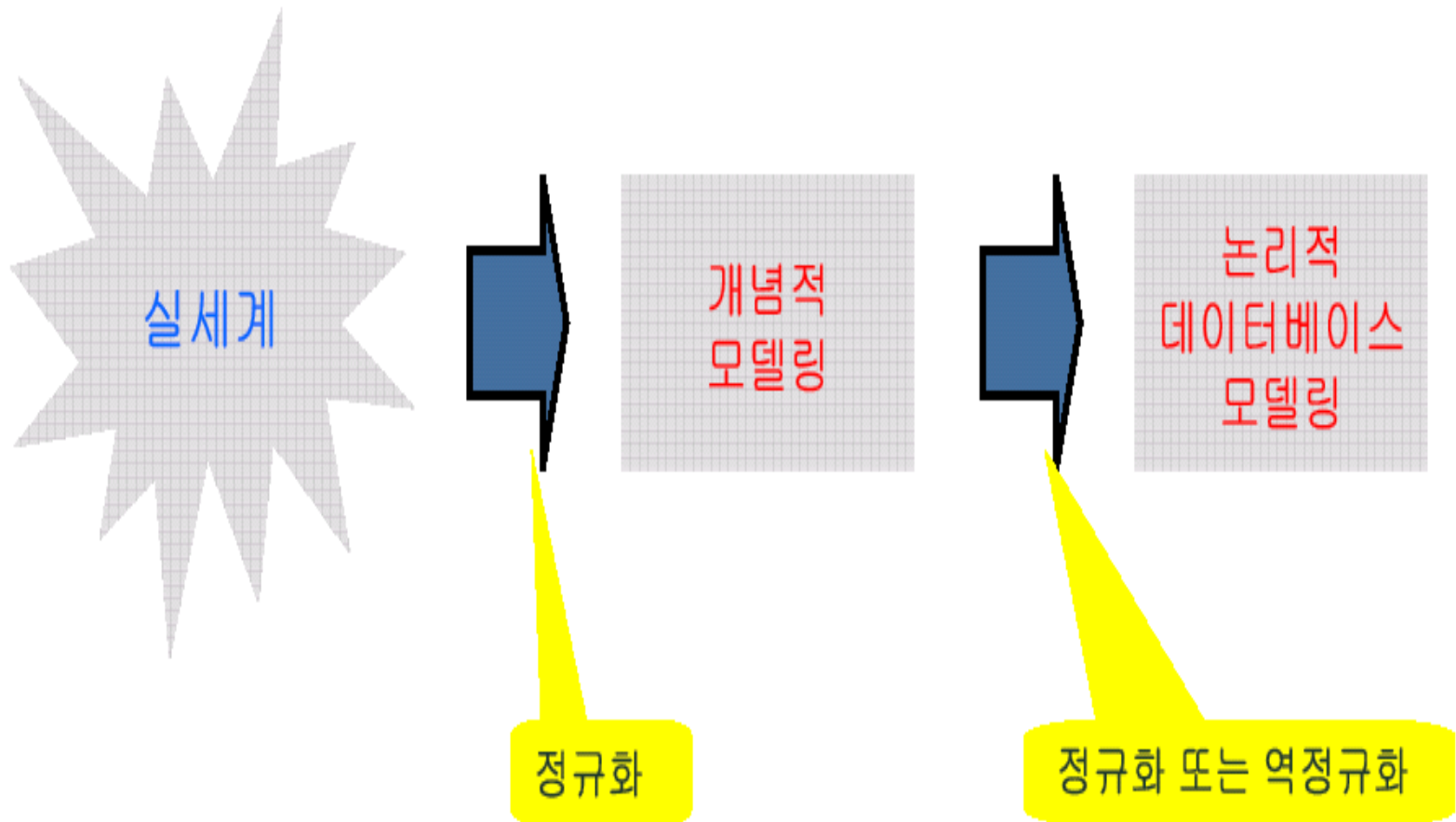
Relation → Foreign Key

- 1:N관계
- 1:1관계에서 필수인 쪽에 Primary Key
- 1:1관계에서 모두 선택인 경우 자주 사용되는 것이 Primary Key

데이터 타입



역정규화



역정규화

- 데이터의 중복을 통해 과도한 논리적 결합 현상을 피한다.
- 유도된 컬럼을 추가하여 불필요한 연산을 피한다. (예) 판매금액 = 수량 * 단가
- 하나의 테이블을 2개 이상의 테이블로 분리하여 불필요한 컬럼의 검색을 피한다.
- 자주 사용되는 테이블의 논리적 결합을 피하기 위해 집계 테이블을 생성한다.

주문번호	고객ID	고객명	주소	상품코드	주문상품명	수량
------	------	-----	----	------	-------	----

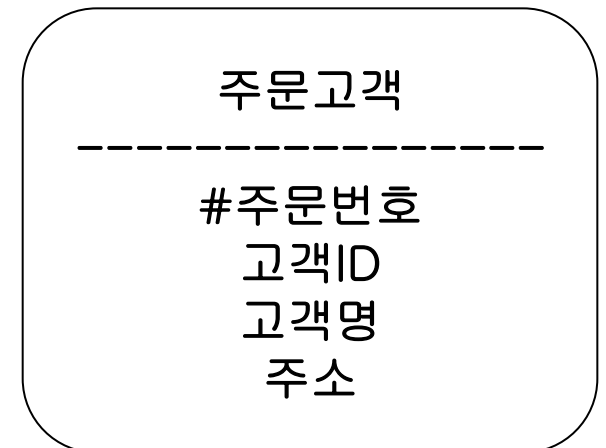
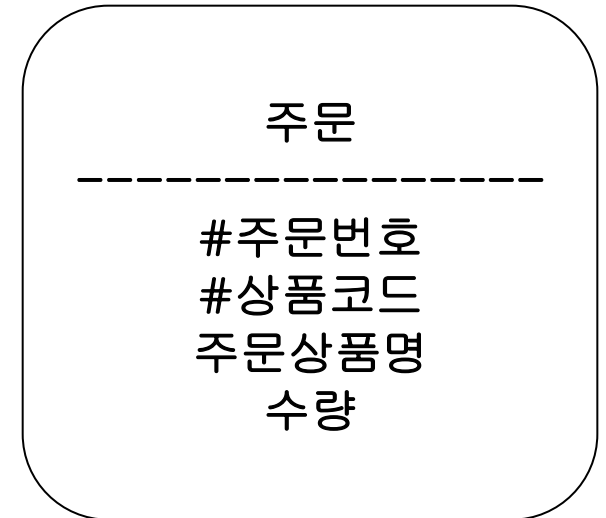
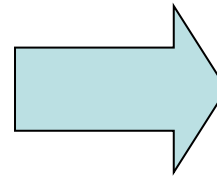
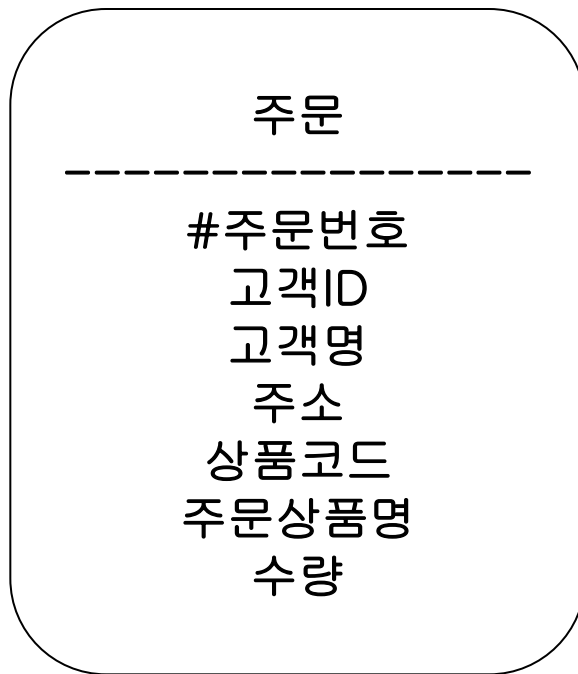
1	1001	홍길동	서울	2001	세탁기	1
1	1001	홍길동	서울	2002	스피커	2
1	1001	홍길동	서울	2003	핸드백	1
2	1002	성실해	부산	2002	스피커	1

중복되는 데이터

주문

#주문번호
고객ID
고객명
주소
상품코드
주문상품명
수량

1 정규화



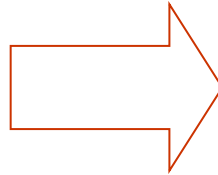
2 정규화

주문

#주문번호
#상품코드
주문상품명
수량

주문고객

#주문번호
고객ID
고객명
주소



주문

#주문번호
#상품코드
수량

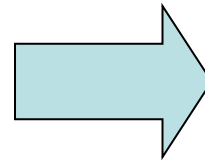
상품

#상품코드
상품명

주문고객

#주문번호
고객ID
고객명
주소

3 정규화



주문

#주문번호
#상품코드
수량

상품

#상품코드
상품명

주문고객

#주문번호
고객ID
고객명
주소

주문

#주문번호
#상품코드
수량

상품

#상품코드
상품명

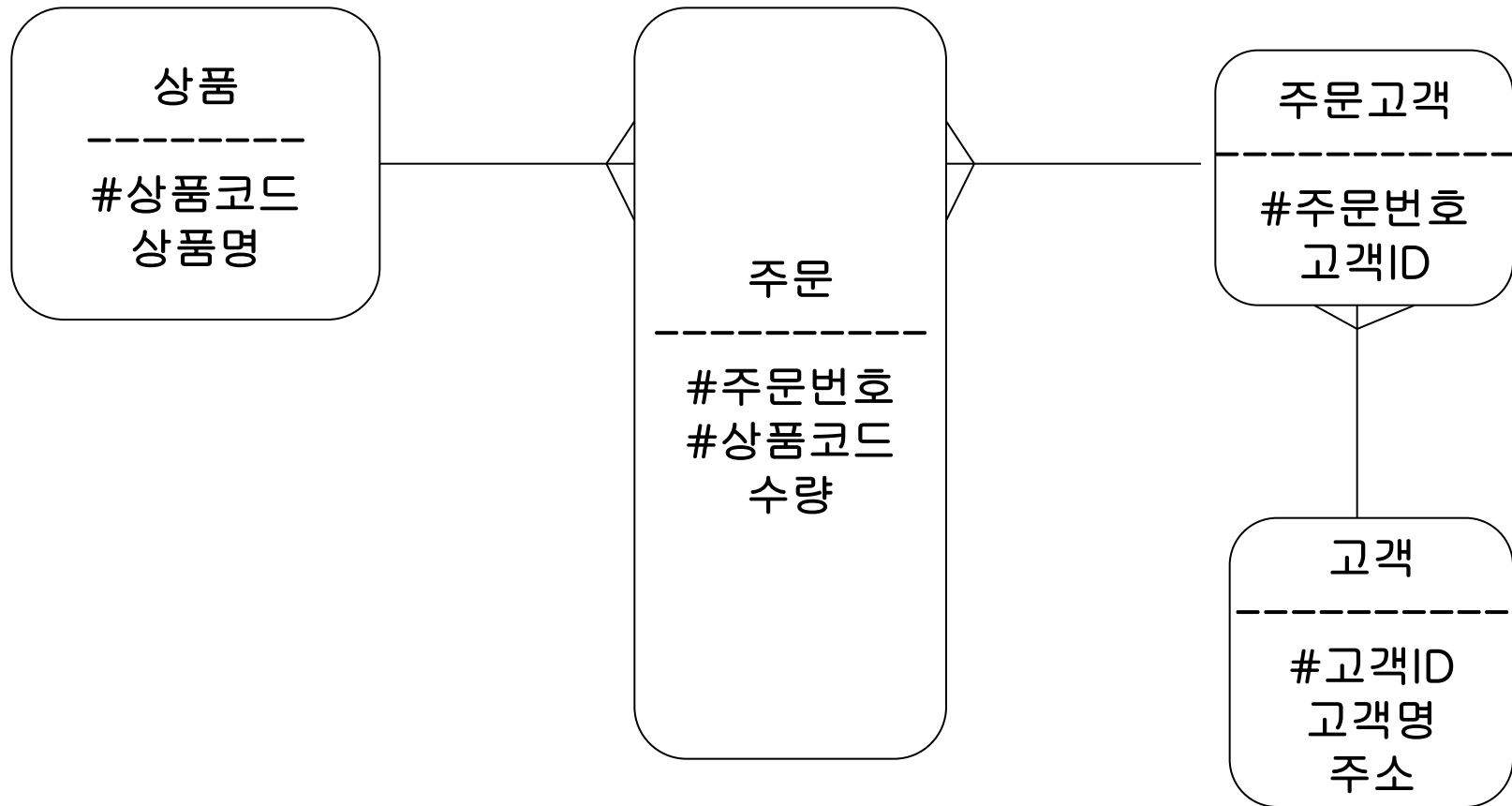
주문고객

#주문번호
고객ID

주문고객

#고객ID
고객명
주소

ERD

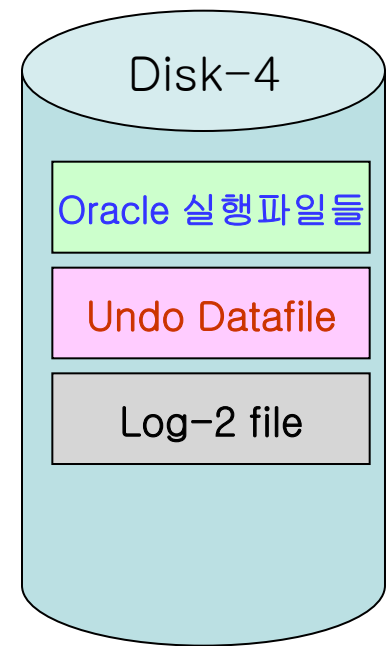
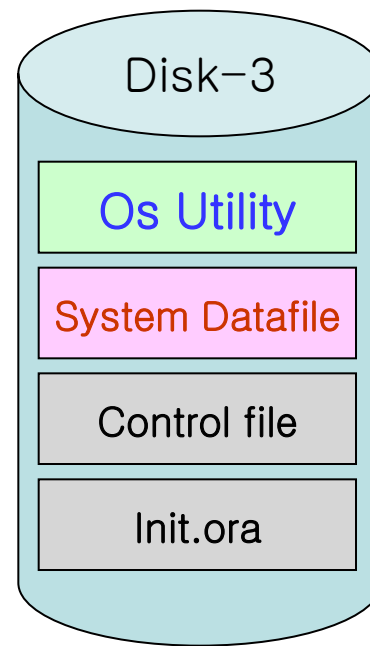
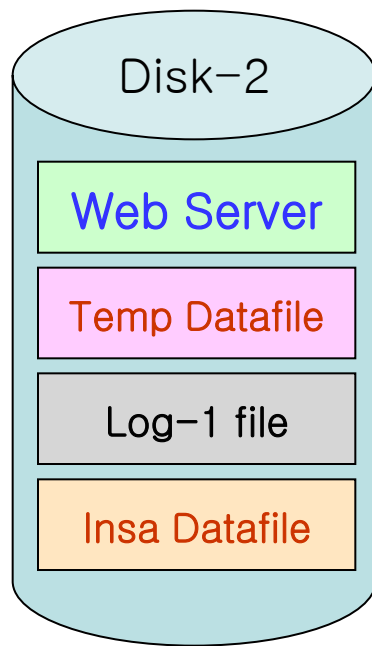
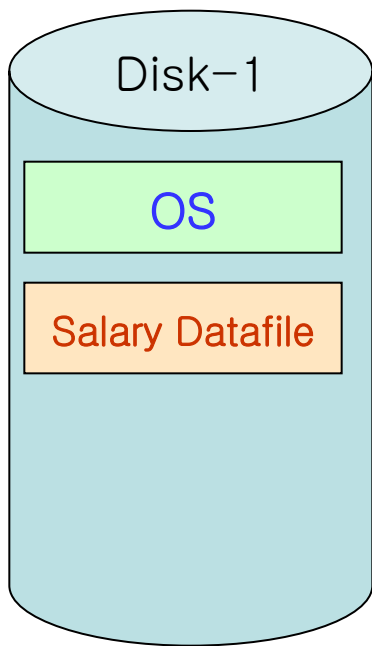


물리적 데이터베이스 설계

- DBMS내에 객체들을 생성하는 단계

물리적 설계

- 디스크에 적절하게 IO가 발생할 수 있도록 분산
- 사용빈도가 많은 것과 적은 것끼리 짝지어 배치하는 것이 최선의 방법



O/S 관련파일

Oracle 관련파일

- 1)Operating System
- 2)Web Server
- 3)OS Utility

- 1)Oracle 실행파일들
- 2)System Datafile
- 3)UNDO Datafile
- 4)Temp Datafile
- 5)Salary Datafile
- 6)Insa Datafile
- 7)Log-1 file
- 8)Log-2 file
- 9)Control file
- 10)init.ora

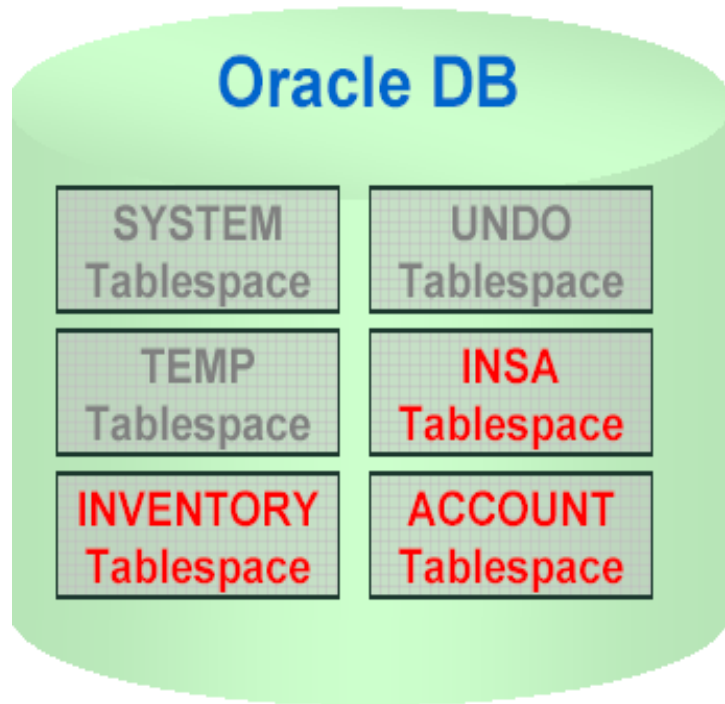
구조

- 오라클데이터베이스>테이블스페이스
(SYSTEM, UNDO, TEMP 등) >세그먼트
(테이블, 인덱스, 뷰 등)>익스텐트>블록

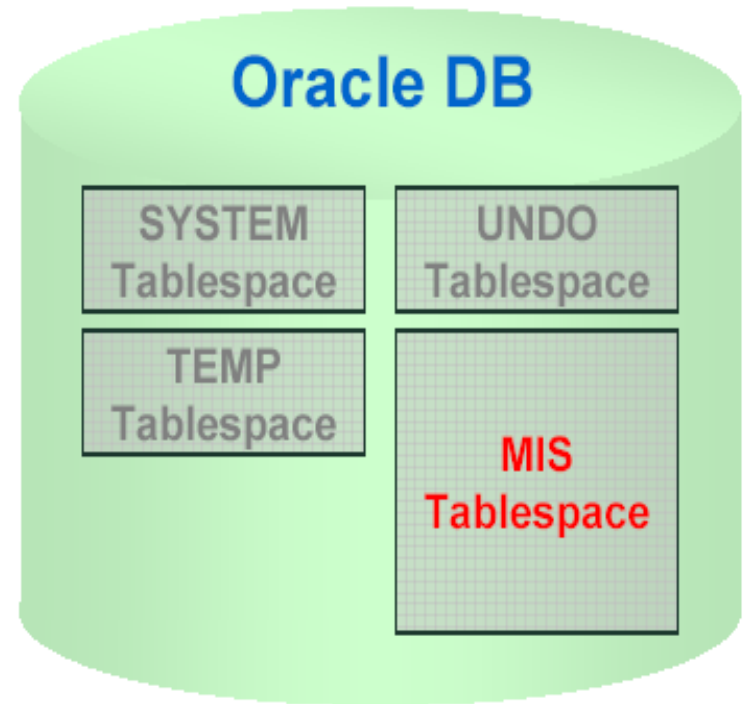
테이블스페이스 설계 기준

- 데이터 테이블 스페이스와 SYSTEM 테이블 스페이스는 분리
- 데이터 테이블 스페이스와 UNDO 테이블 스페이스는 분리
- 데이터 테이블 스페이스와 TEMP 테이블 스페이스는 분리
- 각 데이터 테이블 스페이스는 I/O 경합을 줄이기 위해 분리

사용자의 구조 설계



업무별로 사용자 생성



전체업무를 하나의 사용자로 생성

제약조건

- 데이터의 무결성 보장
 - Primary key: 하나의 테이블에 오직 하나만 존재
 - Foreign key
 - Check
 - Unique
 - Not null