

ORACLE[®]

DATABASE



1. 오라클 데이터베이스
2. 웹페이지에서 디비연동

디비연동

CART LIST



주문날짜	제품명	단가	수량	금액
2021-9-10	달고나	100	1	100
2021-9-10	눈갈사탕	200	1	200
300				주문하기

JOIN US

아이디(5자 이내)

비밀번호(5자 이내)

이름

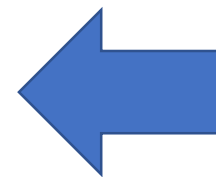
주소

전화번호(000-1111-2222)

join us

CART LIST

주문날짜	제품명	단가	수량	금액
2021-9-10	달고나	100	1	100
2021-9-10	눈깔사탕	200	1	200
300				주문하기



JOIN US

아이디(5자이내)

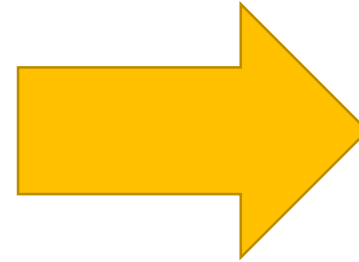
비밀번호(5자이내)

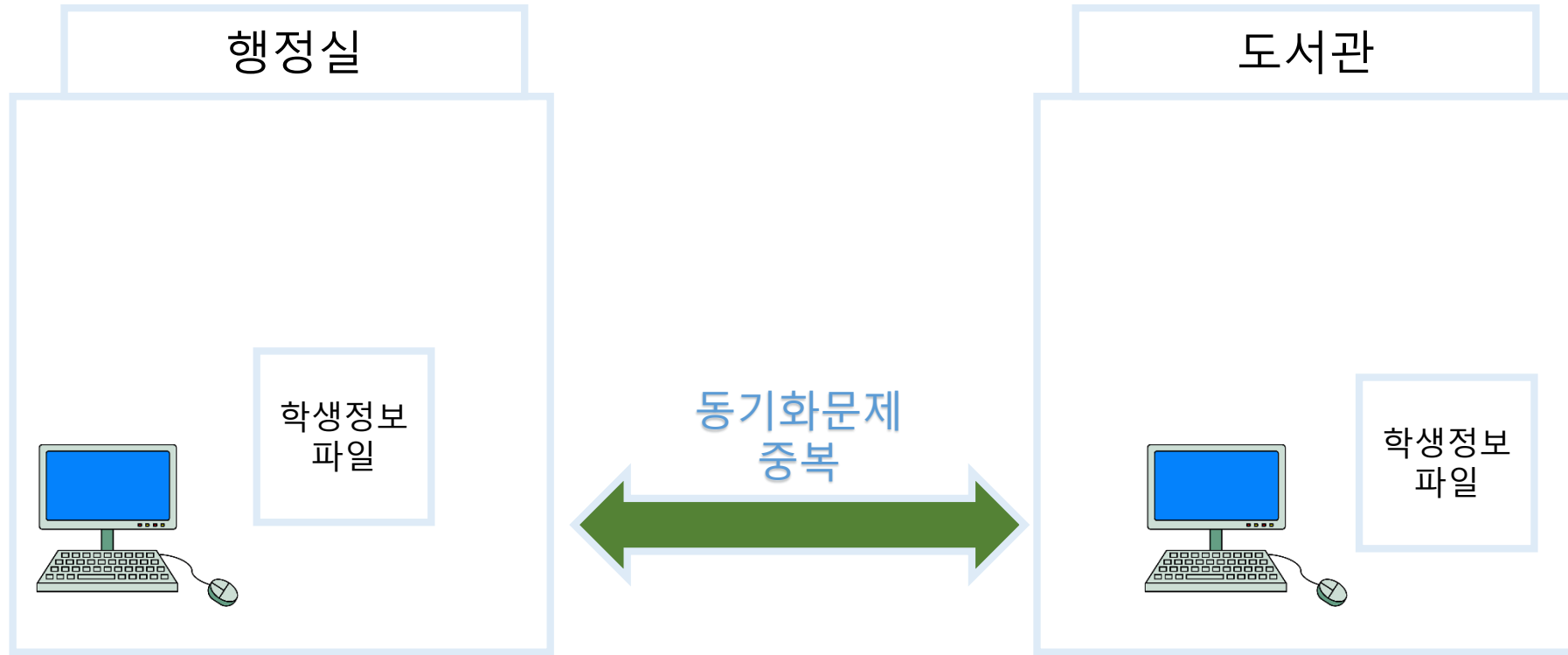
이름

주소

전화번호(000-1111-2222)

join us

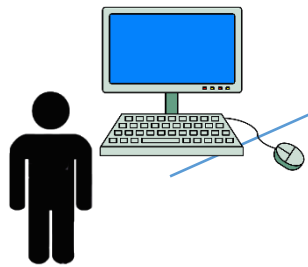
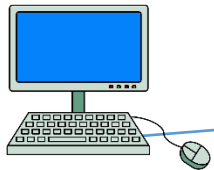




데이터베이스 이전

중복을 제거하는 방법
으로 결함을 줄임

행정실

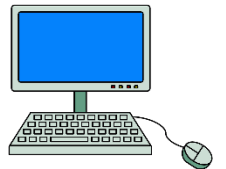


학생

등록금납입

도서대여

도서관

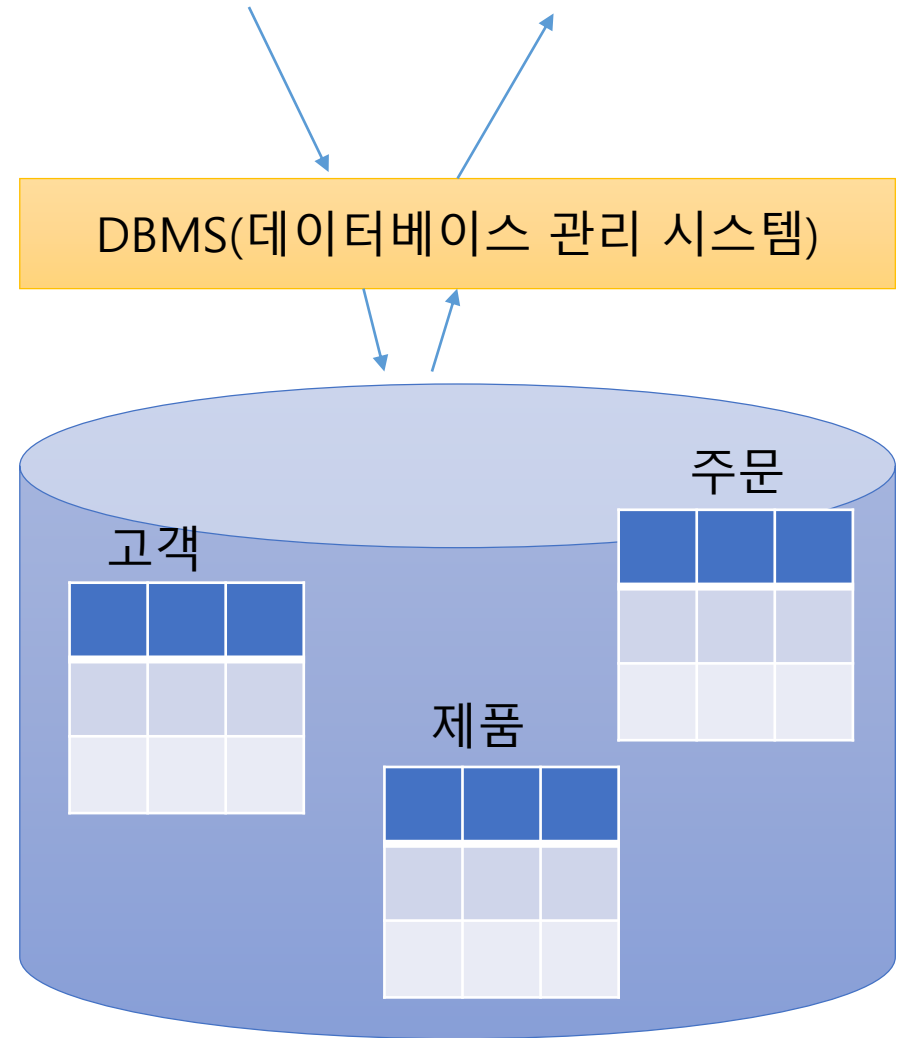


공용데이터(공유)
통합데이터

데이터베이스

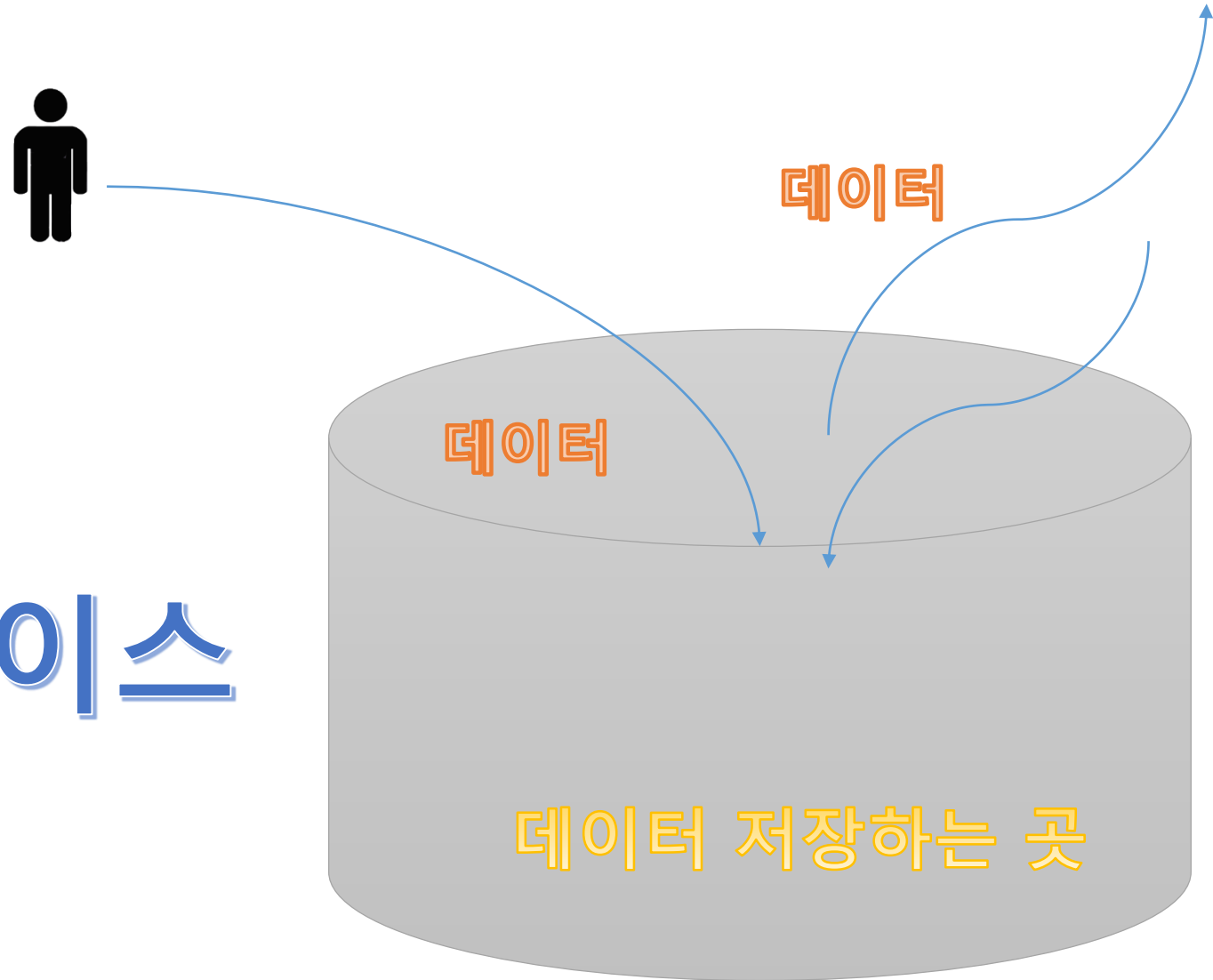
줄여서 DB라고도 하며, 특정 다수의 이용자들에게 필요한 정보를 제공한다든지 조직 내에서 필요로 하는 정보를 체계적으로 축적하여 그 조직 내의 이용자에게 필요한 정보를 제공함

Oracle
Mysql



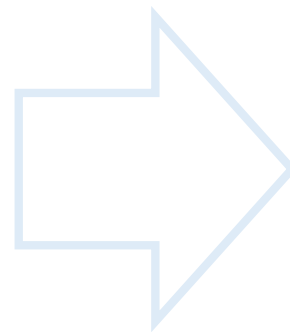
데이터베이스

데이터베이스



데이터베이스 관리시스템의 필요성

여러 사용자가 공용으로 사용을 함
으로써
동시성
성능
보완
문제를 해결하기 위함



DBMS

데이터베이스 관리
자역할



데이터

데이터

엄격한 관리자

ESTJ-T



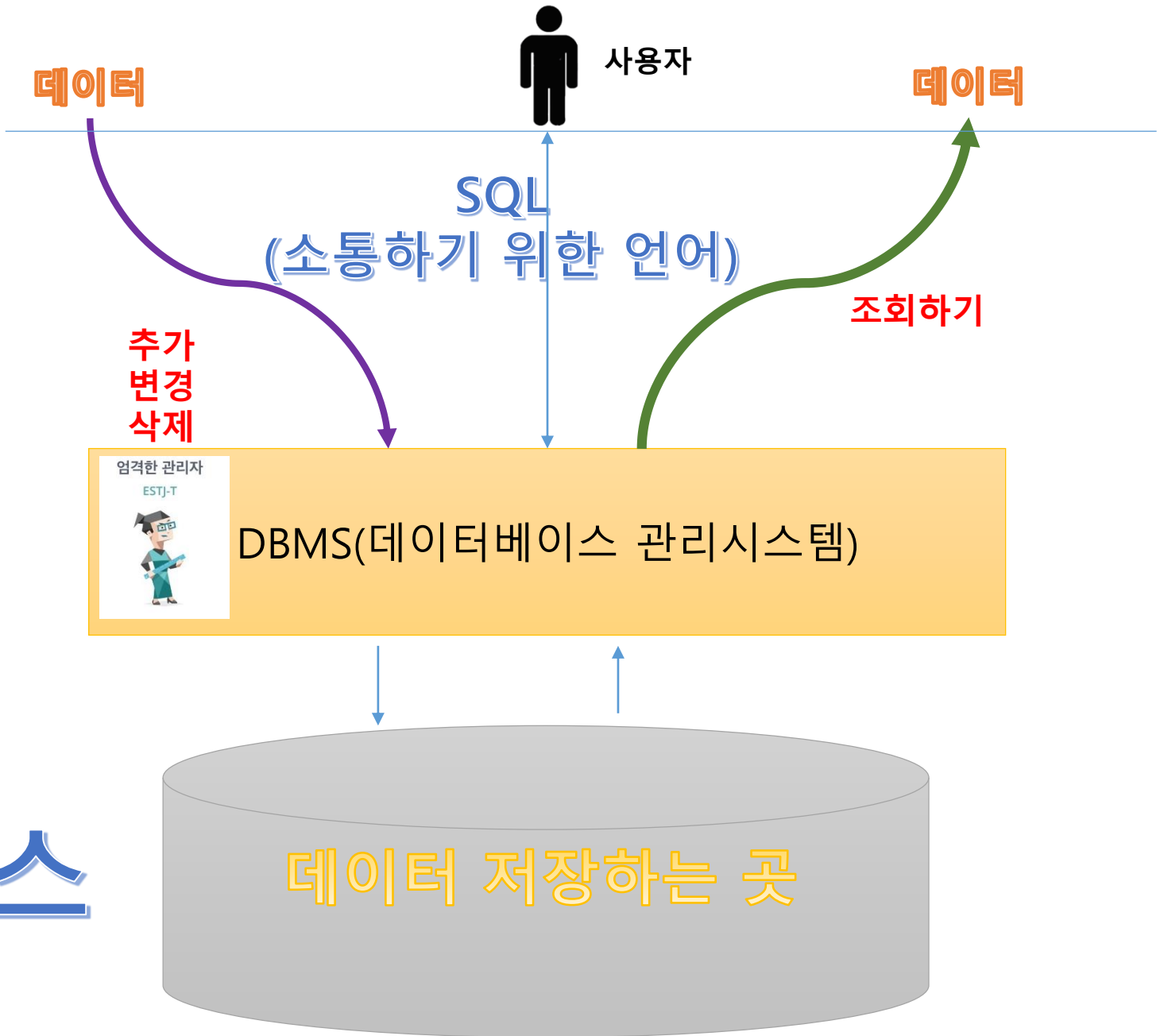
DBMS(데이터베이스 관리시스템)

데이터베이스

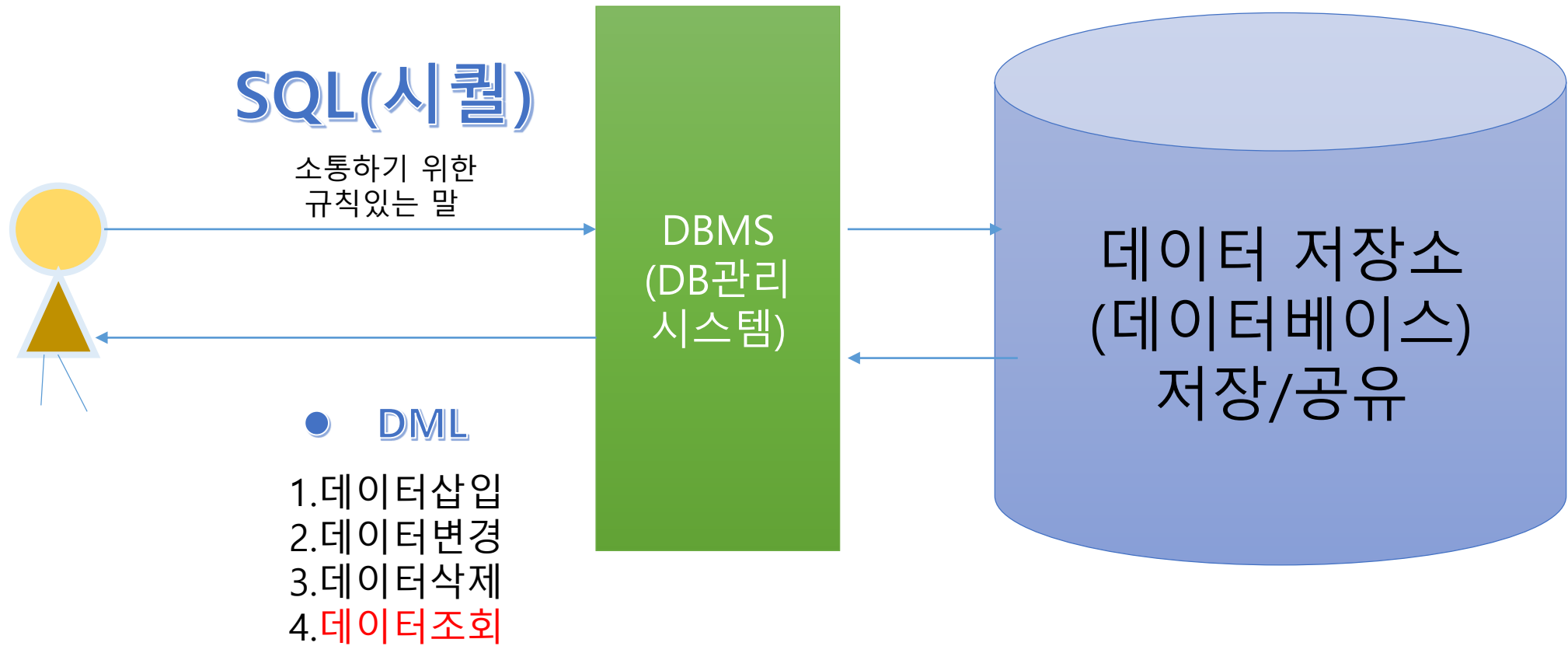
데이터 저장하는 곳



데이터베이스



데이터베이스관리자 역할



SQL(시퀀)

- DDL (데이터정의 언어)
- DML (데이터조작 언어)
- DCL (데이터제어 언어)

■ DDL

- CREATE : 테이블 생성
- DROP : 테이블 삭제
- ALTER : 테이블 수정

■ DML (CRUD)

- CREATE : 데이터 생성, 추가, INSERT
- READ : 데이터 조회, SELECT
- UPDATE : 데이터 변경(갱신)
- DELETE : 데이터 삭제



Select(조회)
Insert(삽입)
Update(수정)
Delete(삭제)

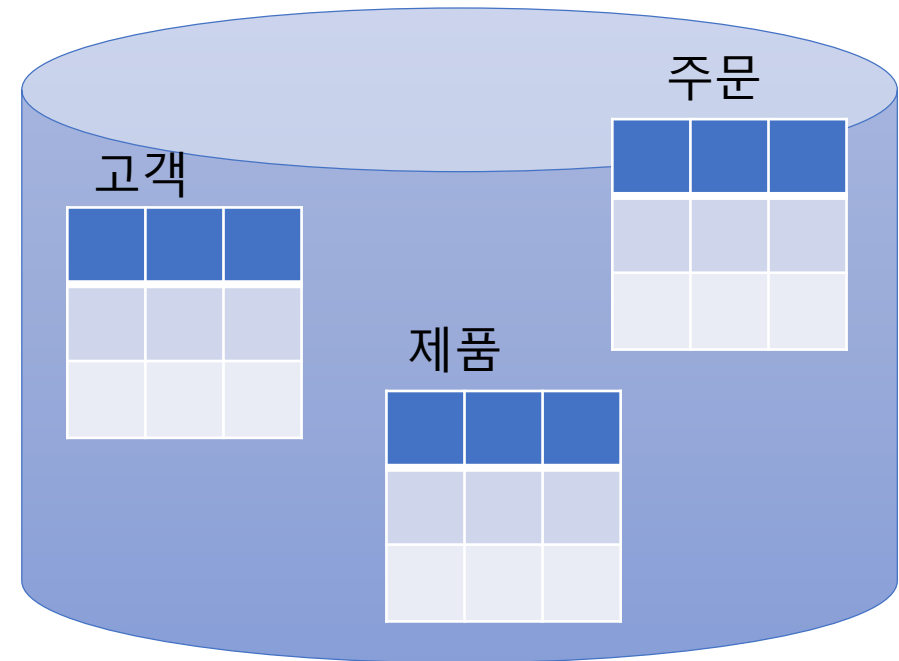
oracle

DBMS(데이터베이스 관리 시스템)

SQL (query language)

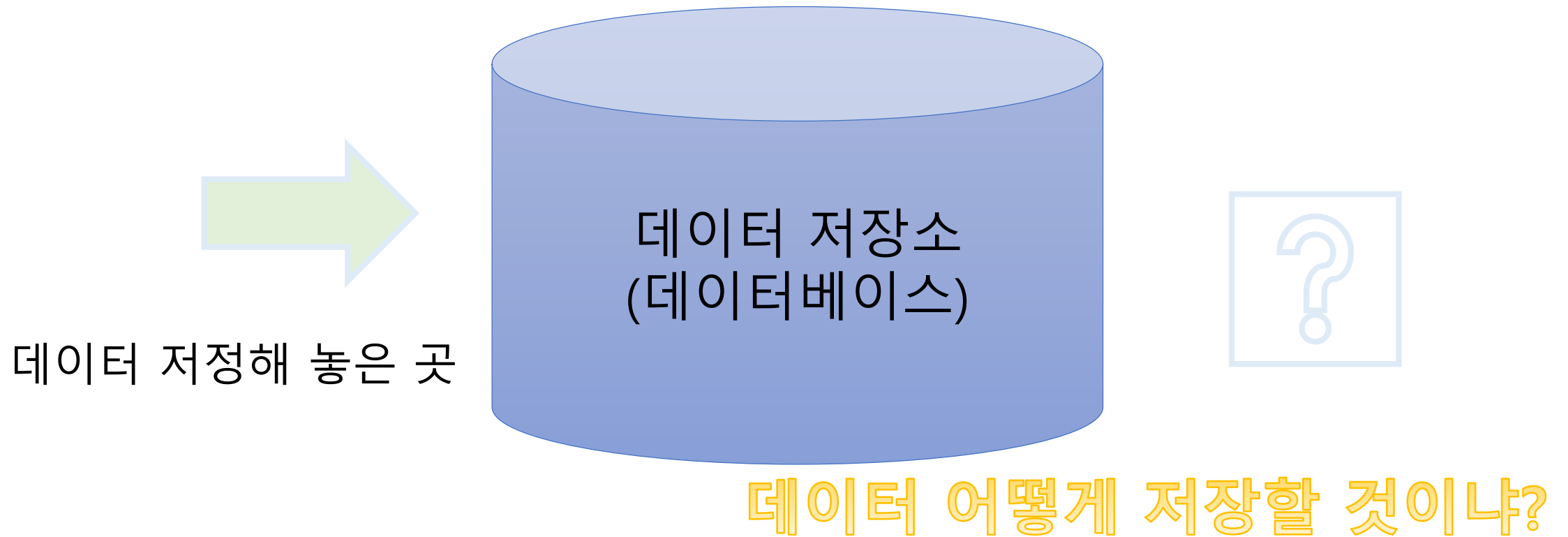
구조화된 질의 언어 :DBMS에게 질의 하는 언어

Query :문의[질의]하다

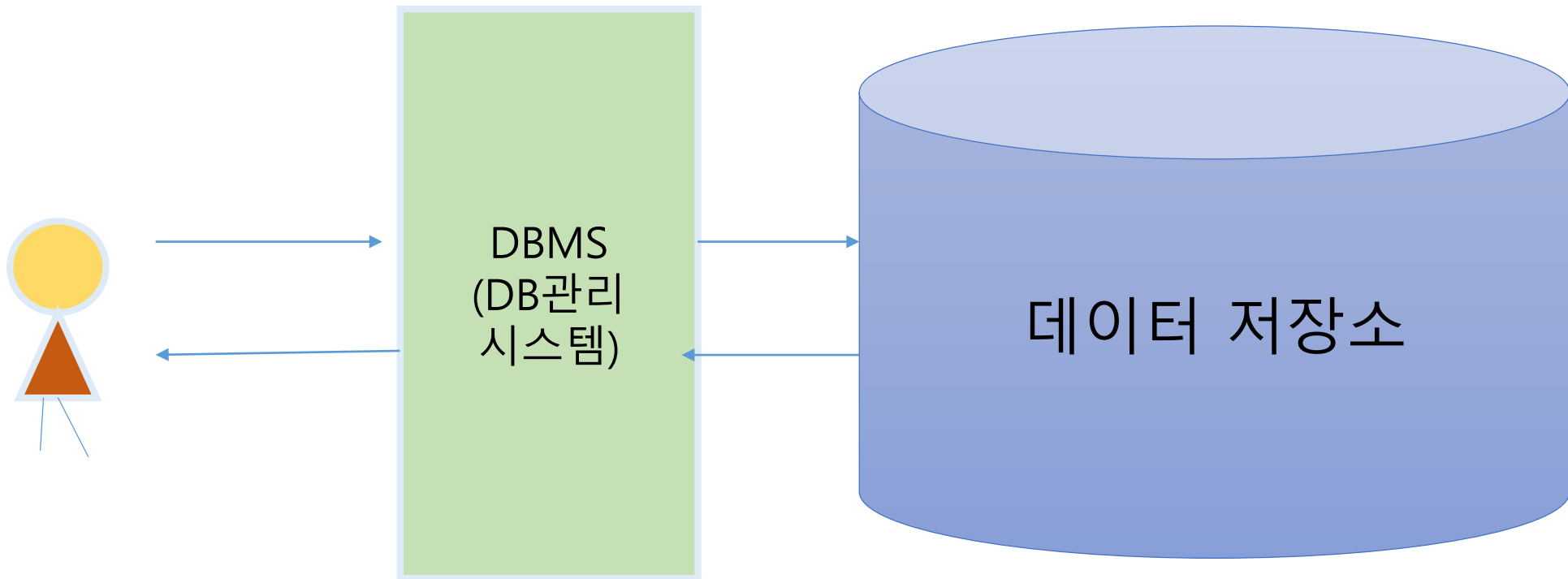


데이터베이스

관계형 데이터베이스 RDBMS



데이터베이스관리자 역할



주문정보

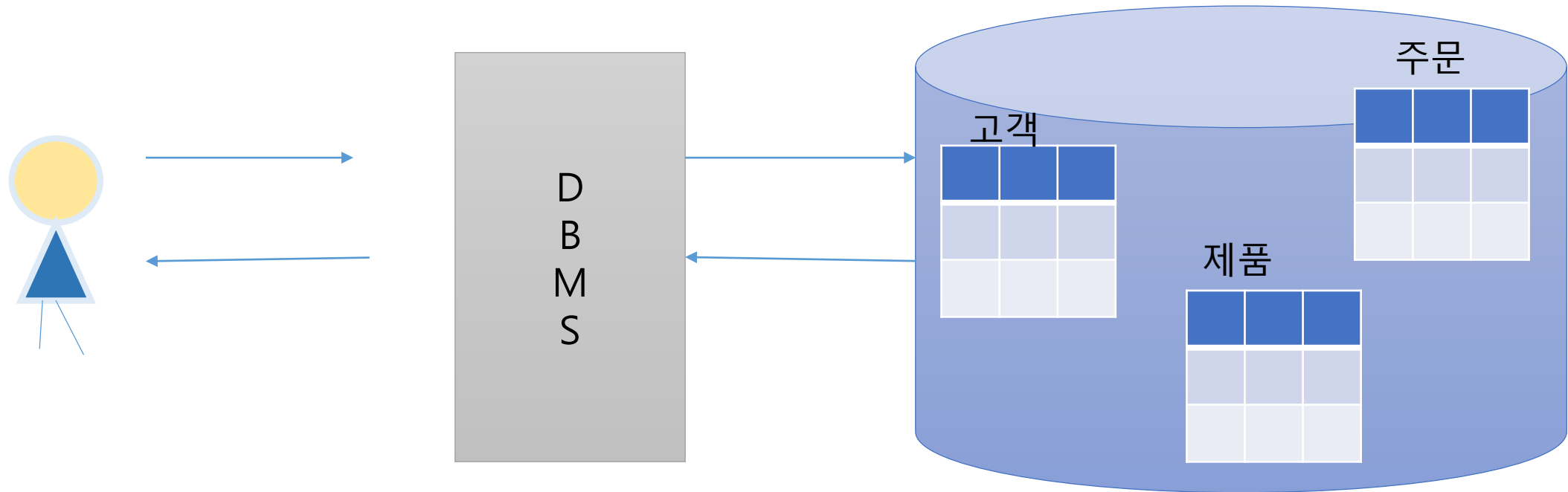
문제점?

에이콘몰

이름	주소	전화번호	주문일자	구매상품	수량	단가	금액
한준	라스베가스	010-1111-2222	2021-01-10	쫄면	2	1,000	2,000
조성민	L.A	010-2222-3333	2020-01-10	눈갈사탕	5	100	500
김두산	워싱턴D.C	010-3333-4444	2021-01-01	아폴로	2	200	400
김정아	뉴욕	010-4444-5555	2021-05-01	뽕튀기	1	2,000	2,000
주샛별	텍사스	010-5555-6666	2021-06-01	눈갈사탕	3	100	300
한준	라스베가스	010-1111-2222	2021-06-10	눈갈사탕	3	100	300
한준	라스베가스	010-1111-2222	2021-06-20	눈갈사탕	4	100	400
조성민	L.A	010-2222-3333	2021-06-11	쫄면	1	1,000	1,000

주문정보를 하나의 공간(테이블)에 저장하다면 어떤 문제점이 있을까요?

데이터를 어떻게 저장할까?



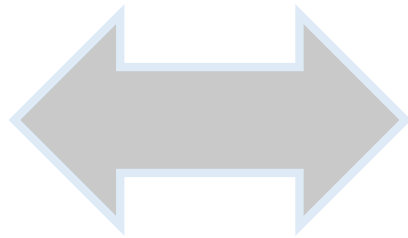
데이터베이스

관계형 데이터베이스 (RDBMS)

정규화

테이블 여러 개로 쪼개기

왜 쪼갤까?



조인

여러테이블을 하나로 합치기

왜 합칠까?

고객

아이디	이름	주소	전화번호
H001	한준	라스베가스	010-1111-2222
H002	조성민	L.A	010-2222-3333
H003	김두산	워싱턴D.C	010-3333-4444
H004	김정아	뉴욕	010-4444-5555
H005	주샛별	텍사스	010-5555-6666

제품

제품코드	구매상품	단가
P001	쫄면	1,000
P002	눈깔사탕	100
P003	아폴로	200
P004	빵튀기	2,000

주문

주문번호	주문날짜	고객코드	제품코드	수량
J001	2021-01-10	H001	P001	2
J002	2020-01-10	H002	P002	5
J003	2021-01-01	H003	P003	2
J004	2021-05-01	H004	P004	1
J005	2021-06-01	H005	P002	3
J006	2021-06-10	H001	P002	3
J007	2021-06-11	H002	P001	1
J008	2021-06-20	H001	P002	4

조인연습

고객

아아디	이름	주소	전화번호
H001	한준	라스베가스	010-1111-2222
H002	조성민	L.A	010-2222-3333
H003	김두산	워싱턴D.C	010-3333-4444
H004	김정아	뉴욕	010-4444-5555
H005	주샛별	텍사스	010-5555-6666

주문

주문번호	주문날짜		수량
J001	20210110		2
J002	20200110		5
J003	20210101		2
J004	20210501		1
J005	20210601		3
J006	20210610		3
J007	20210611		1
J008	20210620		4

2개 테이블 조인

주문내역에서 고객이름을 같이 보고 싶을때 어떻게 해야 할까요?

주문번호	주문날짜	고객코드	수량	이름
J001	2021-01-10	H001	2	한준
J002	2020-01-10	H002	5	조성민
J003	2021-01-01	H003	2	김두산
J004	2021-05-01	H004	1	김정아
J005	2021-06-01	H005	3	주샛별
J006	2021-06-10	H001	3	한준
J007	2021-06-11	H002	1	조성민
J008	2021-06-20	H001	4	한준

주문테이블에 고객테이블 조인하기

```
SELECT o.*, c.이름  
FROM 주문 o, 고객 c  
WHERE o.고객코드 = c.아이디
```

```
SELECT o.*, c.이름  
FROM 주문 o,  
      고객 c  
ON   o.고객코드 = c.아이디
```

고객

아이디	이름	주소	전화번호
H001	한준	라스베가스	010-1111-2222
H002	조성민	L.A	010-2222-3333
H003	김두산	워싱턴D.C	010-3333-4444
H004	김정아	뉴욕	010-4444-5555
H005	주삿별	텍사스	010-5555-6666

주문날짜	이름	수량
20210110	한준	2
20200110	조성민	5
20210101	김두산	2
20210501	김정아	1
20210601	주삿별	3
20210610	한준	3
20210611	조성민	1
20210620	주삿별	4

join

주문

주문번호	주문날짜	고객코드	수량
J001	20210110	H001	2
J002	20200110	H002	5
J003	20210101	H003	2
J004	20210501	H004	1
J005	20210601	H005	3
J006	20210610	H001	3
J007	20210611	H002	1
J008	20210620	H001	4

SELECT 주문날짜, 이름, 수량
FROM 주문
JOIN 고객
ON 주문. 고객코드 = 고객.아이디

SELECT 주문날짜, 이름, 수량
FROM 주문, 고객
WHERE 주문. 고객코드 = 고객.아이디

주문내역 쿼리 (3개의 테이블 조인)

```
SELECT name , addr , tel , odate , pname , sale_cnt, price , sale_cnt * price
FROM ORDER o
JOIN MEMBER m
ON o.id = m.id
JOIN GOODS g
ON o.pcode = g.pcode
```

```
SELECT name , addr , tel , odate , pname , sale_cnt, price , sale_cnt * price
FROM ORDER o , MEMBER m , GOODS g
WHERE o.id= m.id and o.pcode = g.pcode
```

데이터베이스관리시스템(DBMS)

- 정의기능(DDL)
- **조작기능(DML)**
- 제어기능(DCL)

DDL

CREATE / ALTER / DROP

DML

CRUD

C : INSERT(데이터 생성) / create

R : SELECT (데이터 조회) / read

U : UPDATE(데이터 수정) /update

D : DELETE(데이터 삭제) /delete



중요

테이블 정의하기 (만들기)

```
create table member(
```

```
  id          varchar2(50) primary key ,  
  pwd         varchar2(50) ,  
  name        varchar2(50),  
  gender      char(6) ,    //남성, 여성  
  age         number(3),  
  birthday    date ,  
  phone       char(13)
```

```
);
```

member

id	pwd	name	gender	age	birthday	phone
angel	1234	홍길동	남성	18	2021-08-10	010-1111-2222

Primary key설정(주키 설정)

데이터 형식

character	'Hong'
(문자)	'A'
	'123'
numeric	37
(숫자)	2.87
Date (날짜)	2012-02-12
timestamp	2009-10-23 10.11.23.00000

문자형데이터

CHAR	고정길이	CHAR(50) : 무조건 50 의미함 한글: 3byte
VARCHAR2	가변길이	VCHAR2(50) : 최대 50 의미함
NCHAR	고정길이	한글:2byte :한글저장시 char보다 더 효율적임
CLOB		대용량데이터 (최대 4G)

숫자형데이터

NUMBER(4)		9999 최대길이4인 숫자,
NUMBER(6,2)		소수점 2자리를 포함하는 최대6자리 (소수점 2째자리에서 반올림)
NUMBER(6,-2)		소수점 -2째 자리에서 반올림 하는 최대6자리

숫자형데이터

NUMBER(4)	9999 최대길이4인 숫자,
NUMBER(6,2)	소수점 2자리를 포함하는 최대6자리 (소수점 2째자리에서 반올림)
NUMBER(6,-2)	소수점 -2째 자리에서 반올림 하는 최대6자리

날짜형데이터

DATE	년-월-일
TIMESTAMP	년-월-일:시:분:초

테이블 수정

```
ALTER TABLE MEMBER MODIFY ID NVARCHAR2(50);
```

```
ALTER TABLE MEMBER ADD ( ADDR VARCHAR2(50) );
```

```
ALTER TABLE MEMBER RENAME COLUMN ID TO userID;
```

```
ALTER TABLE MEMBER DROP COLUMN ADDR;
```

TRUNCATE 명령

TRUNCATE TABLE MEMBER;

DROP 명령

DROP TABLE MEMBER;

NO	NAME	TEL
111	AAA	82/10/01
222	BBB	83/01/01
333	CCC	84/01/01

원본테이블



NO	NAME	TEL

DELETE결과



NO	NAME	TEL

TRUNCATE 결과

데이터를 전부 삭제하고 사용하고 있던 공간을 반납
(테이블자체가 지워지지 않음)



DROP 결과

테이블 만들기 테이블 수정하기 테이블삭제하기

MEMBER테이블의 스키마가 어떻게 되니?

```
CREATE TABLE MEMBER(  
  id varchar2(7) ,  
  name varchar2(10)  
);  
INSERT INTO MEMBER VALUES('aaa', 'test');
```

테이블만들기 (테이블구조 만들기) , 구조(스키마)

```
ALTER TABLE MEMBER MODIFY ID VARCHAR2(5);  
ALTER TABLE MEMBER ADD ( ADDR VARCHAR2(5) );  
ALTER TABLE MEMBER RENAME COLUMN ID TO user_id;  
ALTER TABLE MEMBER DROP COLUMN ADDR;
```

테이블수정하기
테이블 구조(스키마) 수정하기

```
SELECT * FROM MEMER; // 조회해 보면 id ->user_id 변경
```

```
DROP TABLE MEMBER;
```

테이블삭제하기

데이터 활용

데이터 조작(DML)

INSERT문

데이터 삽입 :

```
INSERT INTO 테이블명(필드명1, 필드명2...)  
VALUES ( 값1, 값2 ..) ;
```

데이터 삽입 :

```
INSERT INTO 테이블명  
VALUES ( 값1, 값2 ..) ;
```

주의 !!
테이블의 컬럼의
순서와 개수에 맞
춰 값을 정해줘야
함

```
INSERT INTO member ( user_id , name)  
VALUES( 'testid' , '홍길동') ;
```

```
INSERT INTO member  
VALUES( 'testid2' , '홍길순') ;
```

데이터조회(Read , SELECT)

```
SELECT * FROM 테이블명
```

```
SELECT 컬럼명1, 컬럼명2 FROM 테이블명
```

```
SELECT * FROM 테이블명 WHERE 컬럼명=조건
```

연산자 종류	설명
=	비교 대상에서 같은 조건을 검색(같은것)
!= , <>	비교 대상에서 같지 않은 조건을 검색 (같지않은것)
>	비교 대상에서 큰 조건을 검색 (큰것)
>=	비교 대상에서 크거나 같은 조건을 검색(이상)
<	비교 대상에서 작은 조건을 검색 (작은것)
<=	비교 대상에서 작거나 같은 조건을 검색 (이하)
BETWEEN a AND b	a와 b 사이에 있는 범위 값을 모두 검색(a에서b까지)
IN(a,b,c)	a이거나 b이거나 c인 조건을 검색(괄호안에 있는것만)
LIKE	특정패턴을 가지고 있는 조건을 검색('% , _ ') % :글자수에 제한이 없음, _ : 한글자만
IS NULL/IS NOT NULL	NULL값을 검색/ NULL이 아닌 값을 검색
A AND B	조건이 여러 개 일때 둘 다 만족하는 값 검색
A OR B	A조건이나 B조건 중 한가지라도 만족하는 값을 검색
NOT A	A가 아닌 모든 조건을 검색(조건부정)

조건절에서 사용되는 연산자

조회연습문제

1. 모든 정보조회 (* : 모든컬럼을 조회하는 것으로 정함)
2. 이름과 생일만 조회
3. 이름이 황가온인 학생 정보조회
4. 몸무게가 70키로 이상인 학생정보 조회
5. 키가 175 초과인 학생정보 조회
6. 몸무게가 65에서 70인 학생 정보 (65~70)
7. 주소가 텍사스, 뉴욕인 학생정보
8. 김씨 성을 가진 학생정보 조회
9. 김씨 성을 가지고 키가 177이상인 학생정보 조회
10. 김씨 성을 가지거나 키가 177이사인 학생정보 조회
11. 김씨 성을 가진 학생이 아닌 학생정보 조회
12. 몸무게가 NULL 인 학생정보 조회

단일행함수

1. 문자함수

LOWER()함수

UPPER()함수

LENGTH()함수

LENGTHB()함수

CONCAT()함수 || 연산자와 동일

SUBSTR()함수

SUBSTRB()함수

INSTR()함수

LPAD()함수

RPAD()함수

LTRIM()함수

RTRIM()함수

REPLACE()함수

SUBSTR() : 문자열에서 특정길이의 문자만 골라낼 때 사용 (문자의 일부만을 활용할 때 사용)

SELECT SUBSTR('abcde', 3, 2) FROM dual;



=> 3번째에서 문자열길이 2만큼
(위치는 1부터 시작함)

SELECT SUBSTR('abcde', -3, 2) FROM dual ;



=> 마이너스(-)위치는 오른쪽에서 시작함

결과값: cd , cd

SUBSTRB() : (추출할 바이트수를 지정함)

SELECT SUBSTR('홍길동', 1, 2) from dual;

=> 홍길

SELECT SUBSTRB('홍길동', 1, 3) from dual;

=> 홍

LPAD() 함수 : 왼쪽을 기준으로 특정값으로 채운다

```
SELECT LPAD('star' , 10 , '*') FROM DUAL;
```

*******star**

RPAD() 함수 : 오른쪽 기준

```
SELECT RPAD('star' , 10 , '*') FROM DUAL;
```

star*****

2.숫자함수

1) ROUND() 함수 :반올림

문 법: ROUND(숫자 , 출력을 원하는 자리 수)

2) TRUNC() 함수 : 버림

이 함수는 ROUND 와 사용법은 동일하며 차이점은 무조건 버림을 한다는 것입니다.

문 법: TRUNC(숫자 , 원하는 자리수)

3) MOD() , CEIL() , FLOOR() 함수

3. 날짜함수

SYSDATE , SYSTIMESTAMP : 현재 날짜를 얻어 오는 함수 , 괄호없음 - 주의

MONTHS_BETWEEN() 함수

3. 형변환함수



SELECT TO_CHAR(SYSDATE, 'YYYY') FROM DUAL;

YYYY - 연도를 4자리로 표현 (예: 2014)

YY - 연도의 끝 2자리를 표현

MM : 월을 숫자2자리로 표현

MON : OCT : 영문3자리 표현

MONTH : 월을 뜻하는 이름 전체 표현

DD : 일을 숫자2자리로 표시

DAY : 요일에 해당하는 명칭 표시

```
SELECT 주문번호 , 주문수량 FROM 주문테이블 WHERE 주문일자  
BETWEEN  
'2021-08-01'  
AND  
'2021-08-31' ;
```

8월 한달동안의 주문
내역 조회

```
SELECT 주문번호 , 주문수량 FROM 주문테이블 WHERE 주문일자  
BETWEEN  
TO_DATE('2021-08-01', 'YYYY-MM-DD')  
AND  
TO_DATE('2021-08-31', 'YYYY-MM-DD') ;
```

TO_CHAR(1234, '99999')	=> 1234
TO_CHAR(1234, '099999')	=> 001234
TO_CHAR(1234, '\$9999')	=> \$1234
TO_CHAR(1234 , '9999.99')	=> 1234.00
TO_CHAR(12345, '99,999')	=> 12,345

TO_NUMBER() 함수

숫자가 아닌 **숫자처럼 생긴 문자**를 **숫자**로 바꾸어 주는 함수입니다.

```
SELECT TO_NUMBER('5') from dual;
```

Dual 테이블의 정의

1. 오라클 자체에서 제공되는 테이블
2. 간단하게 함수를 이용해서 계산 결과값을 확인 할 때 사용하는 테이블

함수에 대한 쓰임을 알고 싶을때 특정 테이블을 생성할 필요없이 dual 테이블을 이용하여 함수의 값을 리턴(return)받을 수 있다.

TO_DATE()함수

:날짜처럼 생긴 문자를 날짜로 바꿔 주는 함수이다.

```
SELECT TO_DATE('2014/05/31') from dual
```

일반함수

NVL()함수

NULL값을 만나면 다른 값으로 치환해서 출력하는 함수.

```
SELECT name, salary , nvl( comm , 0)  
FROM emp ;
```

NVL2()함수

NULL값이 아닐 경우 출력할 값을 지정할 수 있다.

comm이 null이 아니면 comm+400 null이면 100 출력함

```
SELECT name, salary , nvl2( comm , comm+400 , 100)  
FROM emp ;
```

DECODE()함수

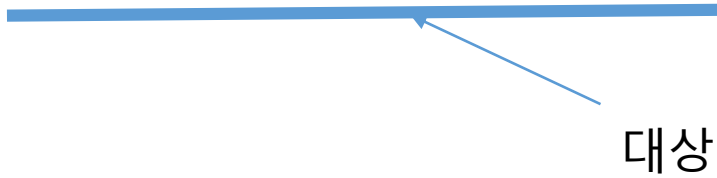
일반언어의 if문을 오라클에서 가져온 개념

```
SELECT name, DECODE(gender , 1 , '남성', 2, '여성' )  
FROM customer ;
```

id	name	gender
a001	김혜겸	1
b001	김세정	2
c001	아이유	2

CASE문

```
SELECT name, tel , CASE SUBSTR( tel, 1, INSTR(tel, ')' ) -1 ) WHEN '02' THEN 'SEOUL'
                                WHEN '031' THEN 'GYEONGGGI'
                                WHEN '051' THEN 'BUSAN'
                                WHEN '052' THEN 'ULSAN'
                                WHEN '055' THEN 'GYEONGNAM'
                                END LOC
FROM student
```



```
02)6255-9875
02)312-9839
053)738-1234
```

```
CREATE TABLE Score_tbl(  
    id varchar2(5) not null primary key,  
    name varchar2(20)  
    total number(3)  
);
```

```
INSERT INTO score_tbl values('G001', '홍길동', 80);
```

```
SELECT name, CASE WHEN total >= 90 THEN '수'  
                  WHEN total >=80 THEN '우'  
                  WHEN total >=70 THEN '미'  
                  ELSE '가'  
                END GRADE  
  
FROM score_tbl;
```

복수행함수 (그룹함수)

COUNT()함수

SUM()함수

AVG()함수

MAX()함수

MIN()함수

GROUP BY절

(특정 조건을 주고 해당 조건에 맞는 결과들을 모아서 조금 더 구체적인 결과 얻음)

부서별 월급 합계, 또는 부서별 월급 평균