

조인

질의 3-23 고객의 이름과 고객이 주문한 도서의 판매가격을 검색하시오.

```
SELECT  name, saleprice
FROM    Customer, Orders
WHERE   Customer.custid=Orders.custid;
```

NAME	SALEPRICE
박지성	21000
박지성	12000
박지성	6000
김연아	7000
김연아	8000
장미란	6000
장미란	13000
장미란	12000
추신수	13000
추신수	20000

질의 3-24 고객별로 주문한 모든 도서의 총 판매액을 구하고, 고객별로 정렬하시오.

```
SELECT  name, SUM(saleprice)
FROM    Customer, Orders
WHERE   Customer.custid=Orders.custid
GROUP BY Customer.name
ORDER BY Customer.name;
```

NAME	SUM(SALEPRICE)
김연아	15000
박지성	39000
장미란	31000
추신수	33000

조인

다음과 같은 질의는 어떻게 구할 수 있는가?

- “고객의 이름과 고객이 주문한 도서의 이름을 구하시오”

고객의 이름과 고객이 주문한 도서의 이름을 구하시오.

CUSTID	NAME	ADDRESS	PHONE
1	박지성	영국 맨체스터	000-5000-0001
2	김연아	대한민국 서울	000-6000-0001
3	장미란	대한민국 강원도	000-7000-0001
4	추신수	미국 캘리포니아	000-8000-0001
5	박세리	대한민국 대전	(null)

BOOKID	BOOKNAME	PUBLISHER	PRICE
1	속구의 역사	굿스포츠	7000
2	속구하는 여자	나무수	13000
3	속구의 이해	대한미디어	22000
4	골프 바이블	대한미디어	35000
5	피겨 교본	굿스포츠	8000
6	역도 단계별기술	굿스포츠	6000
7	야구의 추억	이상미디어	20000
8	야구를 부탁해	이상미디어	13000
9	올림픽 이야기	삼성당	7500
10	Olympic Champions	Pearson	13000

ORDERID	CUSTID	BOOKID	SALEPRICE	ORDERDATE
1	1	1	6000	20/07/01
2	1	3	21000	20/07/03
3	2	5	8000	20/07/03
4	3	6	6000	20/07/04
5	4	7	20000	20/07/05
6	1	2	12000	20/07/07
7	4	8	13000	20/07/07
8	3	10	12000	20/07/08
9	2	10	7000	20/07/09
10	3	8	13000	20/07/10

그림 3-14 마당서점 데이터 간의 연결

조인

질의 3-25 고객의 이름과 고객이 주문한 도서의 이름을 구하시오.

```
SELECT Customer.name, book.bookname
FROM Customer, Orders, Book
WHERE Customer.custid=Orders.custid AND Orders.bookid=Book.bookid;
```

NAME	BOOKNAME
박지성	축구의 역사
박지성	축구하는 여자
박지성	축구의 이해
김연아	피겨 교본
장미란	역도 단계별기술
추신수	야구의 추억
장미란	야구를 부탁해
추신수	야구를 부탁해
김연아	Olympic Champions
장미란	Olympic Champions

질의 3-26 가격이 20,000원인 도서를 주문한 고객의 이름과 도서의 이름을 구하시오.

```
SELECT Customer.name, book.bookname
FROM Customer, Orders, Book
WHERE Customer.custid=Orders.custid AND Orders.bookid=Book.bookid
AND Book.price=20000;
```

NAME	BOOKNAME
추신수	야구의 추억

조인

- 외부조인

질의 3-27 도서를 구매하지 않은 고객을 포함하여 고객의 이름과 고객이 주문한 도서의 판매가격을 구하시오.

```
SELECT  Customer.name, saleprice
FROM    Customer LEFT OUTER JOIN Orders
        ON Customer.custid=Orders.custid;
```

NAME	SALEPRICE
박지성	21000
박지성	12000
박지성	6000
김연아	7000
김연아	8000
장미란	6000
장미란	13000
장미란	12000
추신수	13000
추신수	20000
박세리	(null)

조인

표 3-7 조인 문법

명령	문법
내부조인	SELECT <속성들> FROM 테이블 1, 테이블 2 WHERE <조인조건> AND <검색조건>
	SELECT <속성들> FROM 테이블 1 INNER JOIN 테이블 2 ON <조인조건> WHERE <검색조건>
외부조인	SELECT <속성들> FROM 테이블 1 {LEFT RIGHT FULL [OUTER]} JOIN 테이블 2 ON <조인조건> WHERE <검색조건>

부속질의

질의 3-28 가장 비싼 도서의 이름을 보이시오.

```
SELECT bookname
FROM Book
WHERE price=(SELECT MAX(price)
              FROM Book);
```

BOOKNAME
골프 바이블

BOOKID	BOOKNAME	PUBLISHER	PRICE
1	축구의 역사	굿스포츠	7000
2	축구하는 여자	나무수	13000
3	축구의 이해	대한미디어	22000
4	골프 바이블	대한미디어	35000
5	피겨 교본	굿스포츠	8000
6	역도 단계별기술	굿스포츠	6000
7	야구의 추억	이상미디어	20000
8	야구를 부탁해	이상미디어	13000
9	올림픽 이야기	삼성당	7500
10	Olympic Champions	Pearson	13000

①

가장 비싼
도서의 가격은
→ 35,000원

②

BOOKID	BOOKNAME	PUBLISHER	PRICE
1	축구의 역사	굿스포츠	7000
2	축구하는 여자	나무수	13000
3	축구의 이해	대한미디어	22000
4	골프 바이블	대한미디어	35000
5	피겨 교본	굿스포츠	8000
6	역도 단계별기술	굿스포츠	6000
7	야구의 추억	이상미디어	20000
8	야구를 부탁해	이상미디어	13000
9	올림픽 이야기	삼성당	7500
10	Olympic Champions	Pearson	13000

그림 3-15 부속질의의 실행 순서

부속질의

질의 3-29 도서를 구매한 적이 있는 고객의 이름을 검색하시오.

```
SELECT    name
FROM      Customer
WHERE     custid IN (SELECT custid
                     FROM Orders);
```

NAME
박지성
김연아
장미란
추신수

질의 3-30 대한미디어에서 출판한 도서를 구매한 고객의 이름을 보이시오.

```
SELECT    name
FROM      Customer
WHERE     custid IN(SELECT    custid
                        FROM    Orders
                        WHERE    bookid IN(SELECT    bookid
                                              FROM      Book
                                              WHERE     publisher='대한미디어'));
```

NAME
박지성

부속질의

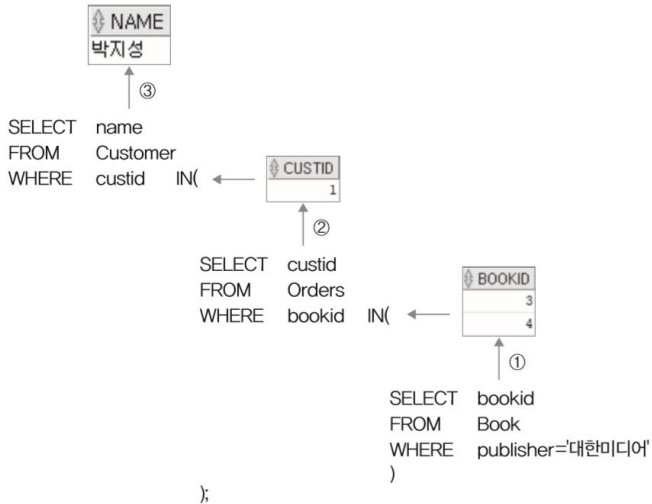


그림 3-16 3단계 부속질의의 실행 순서

부속질의

'대한미디어'에서 출판한 도서를 구매한 고객의 이름을 보이시오.

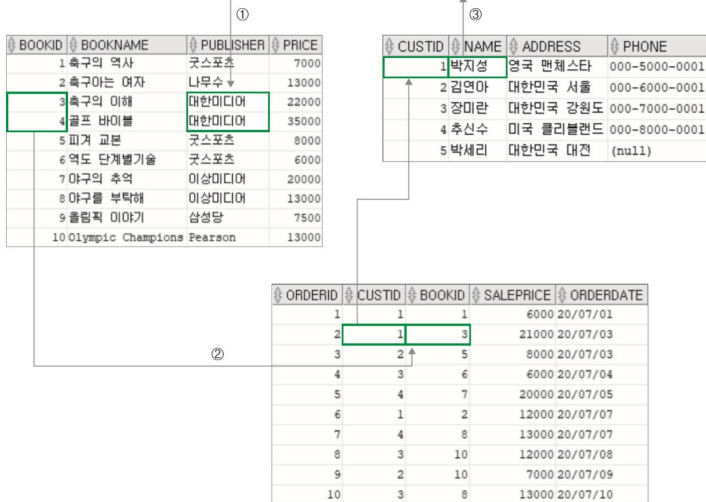


그림 3-17 3단계 부속질의의 실행 순서와 데이터 예

부속질의

- **상관 부속질의(correlated subquery)**

- 상위 부속질의의 튜플을 이용하여 하위 부속질을 계산함.
- 상위 부속질의와 하위 부속질의가 독립적이지 않고 서로 관련을 맺고 있음.

질의 3-31 출판사별로 출판사의 평균 도서 가격보다 비싼 도서를 구하시오.

```
SELECT  b1.bookname
FROM    Book b1
WHERE   b1.price > (SELECT  avg(b2.price)
                   FROM    Book b2
                   WHERE    b2.publisher=b1.publisher);
```

BOOKNAME
골프 바이블
피겨 교본
야구의 추억

부속질의

Book 테이블: b1으로 나타냄

BOOKID	BOOKNAME	PUBLISHER	PRICE
1	축구의 역사	굿스포츠	7000
2	축구하는 여자	나무수	13000
3	축구의 이해	대한미디어	22000
4	골프 바이블	대한미디어	35000
5	피겨 교본	굿스포츠	8000
6	역도 단계별기술	굿스포츠	6000
7	야구의 추억	이상미디어	20000
8	야구를 부탁해	이상미디어	13000
9	올림픽 이야기	삼성당	7500
10	Olympic Champions	Pearson	13000

Book 테이블: b2로 나타냄

BOOKID	BOOKNAME	PUBLISHER	PRICE
1	축구의 역사	굿스포츠	7000
2	축구하는 여자	나무수	13000
3	축구의 이해	대한미디어	22000
4	골프 바이블	대한미디어	35000
5	피겨 교본	굿스포츠	8000
6	역도 단계별기술	굿스포츠	6000
7	야구의 추억	이상미디어	20000
8	야구를 부탁해	이상미디어	13000
9	올림픽 이야기	삼성당	7500
10	Olympic Champions	Pearson	13000

b1 테이블의
투플 t에
해당하는
출판사를
b2 테이블로
가져가서,
같은 출판사를
가진 투플들의
price 평균을
구한다.

avg(b2.price)

28500

$b1.price > avg(b2.price)$

그림 3-18 상관 부속질의의 데이터 예

집합연산

- 합집합 **UNION**, 차집합 **MINUS**, 교집합 **INTERSECT**

{도서를 주문하지 않은 고객} = {모든 고객} - {도서를 주문한 고객}

질의 3-32 도서를 주문하지 않은 고객의 이름을 보이시오.

```
SELECT    name
FROM      Customer
MINUS
SELECT    name
FROM      Customer
WHERE     custid IN (SELECT custid FROM Orders);
```

 NAME
박세리

- 주의할점: Oracle은 차집합을 **MINUS**로 하지만 SQL 표준에서는 **EXCEPT** 를 사용

EXISTS

● EXISTS

- 원래 단어에서 의미하는 것과 같이 조건에 맞는 튜플이 존재하면 결과에 포함시킴
- 즉 부속질의문의 어떤 행이 조건에 만족하면 참임
 - 반면 NOT EXISTS는 부속질의문의 모든 행이 조건에 만족하지 않을 때만 참임.

질의 3-33 주문이 있는 고객의 이름과 주소를 보이시오.

```
SELECT  name, address
FROM    Customer cs
WHERE   EXISTS (SELECT *
                FROM    Orders od
                WHERE    cs.custid=od.custid);
```

NAME	ADDRESS
박지성	영국 맨체스터
김연아	대한민국 서울
장미란	대한민국 강원도
추신수	미국 클리블랜드