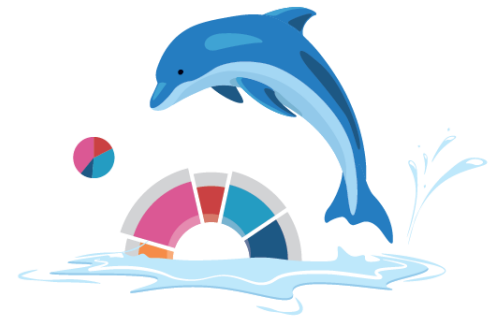


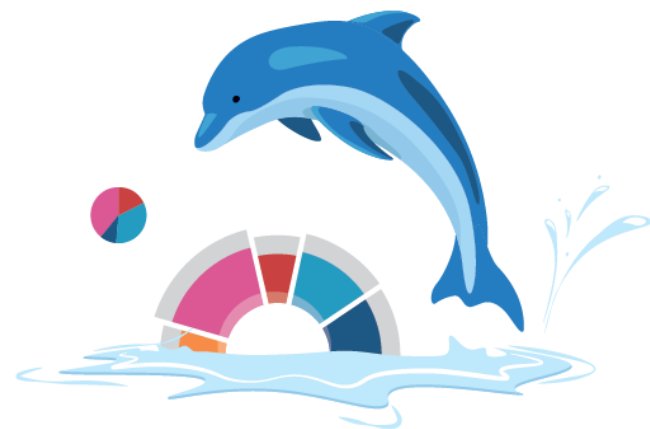
난생처음

MySQL



Chapter 02

SQL의 기본 질의문과 연산자



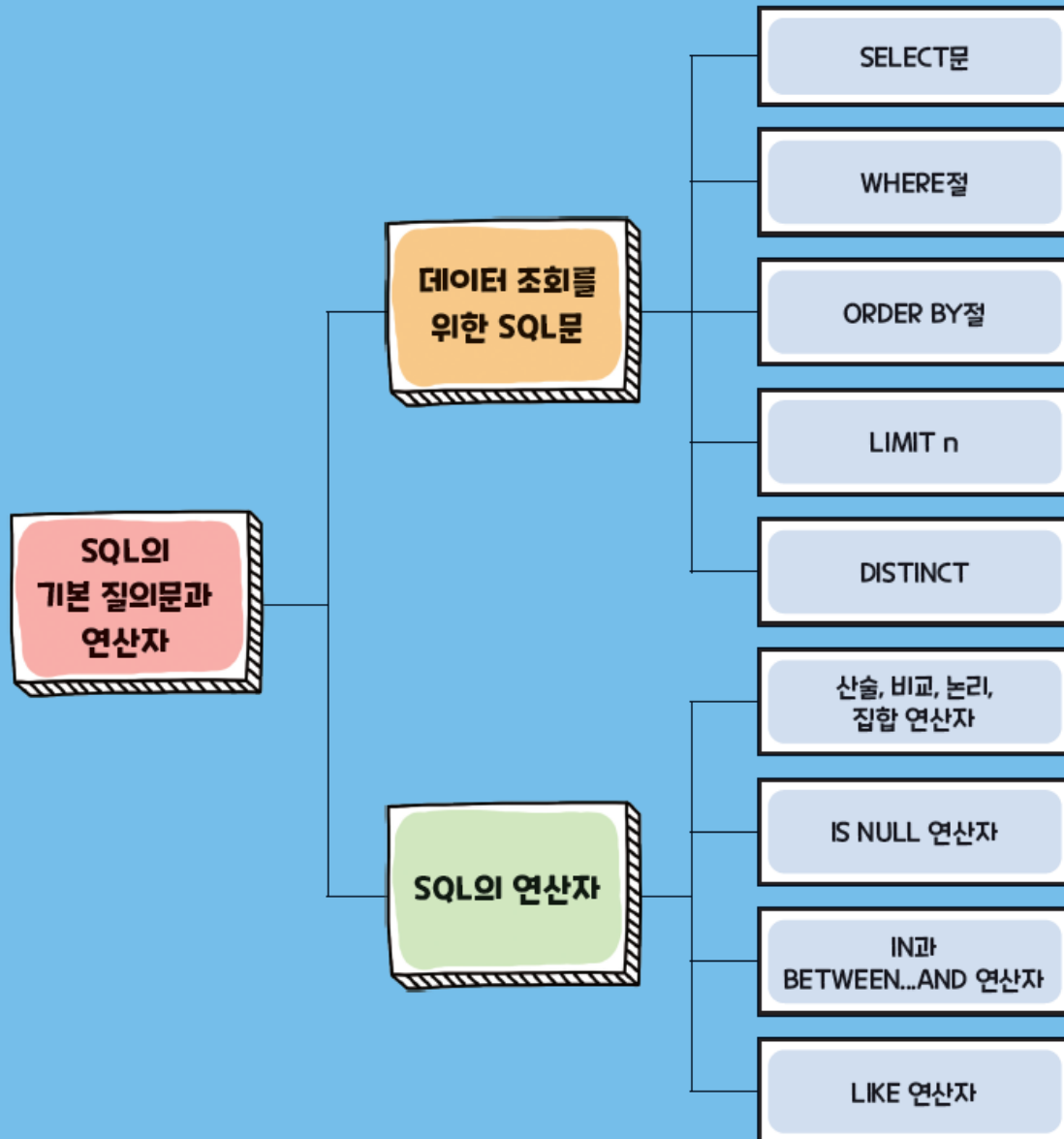
목차

1. 데이터 조회를 위한 SQL문
2. SQL 연산자

학습목표

- 데이터 조회를 위한 SQL 문법을 이해할 수 있습니다.
- SELECT절, WHERE절, ORDER BY절 및 LIMIT n에 대한 문법을 이해할 수 있습니다.
- SQL문에서 연산자를 활용할 수 있습니다

Preview



Section 01

데이터 조회를 위한 SQL문

1. SELECT문

■ SELECT문

- 형식

```
SELECT      컬럼1 [AS 별명1]
            , 컬럼2 [별명2]
            , 컬럼3
FROM        테이블명
[WHERE      조건]
[ORDER BY  컬럼]
[LIMIT     n];
```

Diagram illustrating the syntax of a SELECT statement with annotations:

- 컬럼1** [AS 별명1]: Column name (highlighted in yellow). A red line points to it with the label **검색할 컬럼명** (Column name to be searched).
- 테이블명**: Table name (highlighted in yellow). A red line points to it with the label **데이터를 검색할 테이블명** (Table name to search for data).

- SELECT절에는 컬럼명 뿐만 아니라 수식이나 함수가 들어갈 수도 있음

1. SELECT문

■ SELECT문

- 컬럼에는 별명(Alias)을 붙여줄 수 있음
 - ✓ 별명은 컬럼명을 실제로 변경하는 것이 아니라 현재 SELECT 문장에서만 적용되는 컬럼의 제목
 - ✓ AS 다음에 별명을 입력하거나 AS를 생략하고 바로 별명을 입력해도 됨
 - ✓ 만약 별명에 공백이 있는 경우 또는 특수문자가 포함된 경우에는 반드시 쌍따옴표(" ")나 작은따옴표(' ') 안에 별명을 넣어줌



그림 2-1 별명을 붙이는 이유

1. SELECT문

■ [예제 2-1] 고객 테이블에 있는 모든 데이터를 조회하시오

```
SELECT *
FROM 고객;
```

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
ACDDR	굿모닝서울	이은광	영업 과장	송파구 잠실동 220	서울특별시		(02)978-1984	15911
ADHTR	엘케이 상사	김병현	영업 사원	동작구 흑석 3동 140-3	서울특별시		(02)345-1945	406
AIHTR	씨엔그룹	김성민	영업 사원	가학동 301	광명시	경기도	(02)31-0345	8788
ANKFR	오리안무역	최지수	마케팅 과장	성북구 길음동 136-11	서울특별시		(02)123-0345	6119
ANRFR	남해종합식품	강준	마케팅 과장	수택동 435	구리시	경기도	(034)756-2091	106
ANSFR	서보엠엔에프	김민경	영업 사원	동작구 대방동 36-8	서울특별시		(02)211-2221	729
ATRAN	대정 인터내셔널	백순웅	대표 이사	서초구 방배동 883-11	서울특별시		(02)681-6889	128790
AUSBL	제원인터내셔널	박지혜	영업 사원	서대문구 남가좌 1동 121	서울특별시		(02)811-1234	47865
BEVBS	케이리칼리스	오선주	영업 사원	구산동 17-111	김해시	경상남도	(0525)342-3333	8758
BQQZA	비큐큐푸드	황수영	대표 이사	서초구 서초동 120-3	서울특별시		(02)456-9876	100
CARRI	진가식품	이두원	영업 사원	비전 1동 118-36	평택시	경기도	(033)945-1932	122
CCOPI	진산푸드	김효정	영업 과장	구로구 오류 1동 101	서울특별시		(02)100-1670	1034
CORLA	한송F&B	이상훈	영업 사원	구로구 가리봉 1동 22-1	서울특별시		(02)123-7981	1331
CSURI	유토포에이지	이소영	영업 과장	서대문구 창천동 167-3	서울특별시		(02)712-8931	6608
CTEVI	한국맥널트 무역	김유환	영업 사원	중구 답동 78	인천광역시		(032)58-1984	939
CTUCA	유성물산교역	우지선	영업 사원	강남구 대치동 315-11	서울특별시		(02)943-4486	7232
DOSGO	프루트코리아	김태욱	영업 과장	중구 광희동 1가 188-3	서울특별시		(02)545-8911	713
DWOOL	하이론 식품	김예진	영업 사원	도봉구 미아 7동 334-7	서울특별시		(02)774-1290	6935
	조운트사		마케팅 과장		서울특별시		6766	1066

1. SELECT문

- [예제 2-2] 고객 테이블에서 고객번호, 담당자명, 고객회사명, 마일리지, 10% 인상된 마일리지를 조회하시오. 이때 마일리지는 '포인트'로, 인상된 마일리지는 '10%인상된 마일리지'로 별명을 붙입니다.

```
SELECT 고객번호
      ,담당자명
      ,고객회사명
      ,마일리지 AS 포인트
      ,마일리지 * 1.1 AS "10%인상된 마일리지"
FROM 고객;
```

▶ 실행결과

고객번호	담당자명	고객회사명	포인트	10%인상된 마일리지
ACDDR	이은광	굿모닝서울	15911	17502.1
ADHTR	김병현	열케이 상사	406	446.6
AIHTR	김성민	씨엔그룹	8788	9666.8
ANKFR	최지수	오리안무역	6119	6730.9
ANRFR	강준	남해종합식품	106	116.6
ANSFR	김민경	서보엠엔애프	729	801.9
ATRAN	배순용	대정 인터내셔널	128790	141669.0
AUSBL	박지혜	저원인덕내쇼날	47865	52651.5
BEVBS	오선주	케이리칼리스	8758	9633.8
BQQZA	황수영	비큐큐푸드	100	110.0
CARRI	이두원	진가식품	122	134.2
CCORT	김효정	진가식품	1034	1137.4

컬럼명인 마일리지 대신
별명인 '포인트'가 보여짐

2. WHERE절

- [예제 2-3] 고객 테이블에서 마일리지가 100,000점 이상인 고객의 고객번호, 담당자명, 마일리지를 조회하시오.

```
SELECT 고객번호  
      ,담당자명  
      ,마일리지  
FROM 고객  
WHERE 마일리지 >= 100000;
```

▶ 실행결과

고객번호	담당자명	마일리지
ATRAN	배순용	128790
PRDSU	한현구	116898

3. ORDER BY절

■ ORDER BY

- 레코드를 순서대로 정렬하고자 할 때 사용함
- 컬럼명 이외에도 수식이나 함수 또는 컬럼의 순번이 올 수도 있음
- 오름차순 정렬(ASC)
 - ✓ 작은 값부터 순서대로 보여줌
 - ✓ ASC는 생략 가능함
- 내림차순 정렬(DESC)
 - ✓ 큰 값부터 순서대로 보여줌
- ORDER절에서 컬럼명 대신 별명이나 컬럼의 순서를 넣을 수도 있음

```
ORDER BY 포인트 DESC; (O)
```

```
ORDER BY 4 DESC; (O)
```

3. ORDER BY절

- [예제 2-4] '서울특별시'에 사는 고객에 대해 고객번호, 담당자명, 도시, 마일리지를 조회하시오. 이때 마일리지가 많은 고객부터 순서대로 보입니다.

```
SELECT 고객번호  
      ,담당자명  
      ,도시  
      ,마일리지 AS 포인트  
FROM 고객  
WHERE 도시 = '서울특별시'  
ORDER BY 마일리지 DESC;
```

▶ 실행결과

고객번호	담당자명	도시	포인트
ATRAN	배순용	서울특별시	128790
PRDSU	한현구	서울특별시	116898
HMSLE	조효주	서울특별시	78001
AUSBL	박지혜	서울특별시	47865
LMKWI	조가영	서울특별시	36960
LASLI	지승환	서울특별시	22260
ACDDR	이은광	서울특별시	15911

4. LIMIT n

■ LIMIT n

- 사용하여 반환되는 레코드의 개수를 지정할 수 있음
- 1행부터 시작하여 n개의 레코드를 가져오기 위해서는 문장의 맨 마지막에 LIMIT n을 추가함
- 시작행의 위치를 지정할 수도 있음
- ORDER BY절 뒤에 LIMIT와 가져올 레코드의 수를 넣으면 상위 또는 하위 n개의 레코드를 조회할 수 있음

4. LIMIT n

- [예제 2-5] 고객 테이블에서 1행부터 시작하여 3개의 고객 정보를 조회하시오입니다.

```
SELECT *  
FROM 고객  
LIMIT 3;
```

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
ACDDR	굿모닝서울	이은광	영업 과장	송파구 잠실동 220	서울특별시		(02)978-1984	15911
ADHTR	엘케이 상사	김병현	영업 사원	동작구 흑석 3동 140-3	서울특별시		(02)345-1945	406
AIHTR	씨엔그룹	김성민	영업 사원	가학동 301	광명시	경기도	(02)31-0345	8788

- [예제 2-6] 마일리지가 많은 고객부터 상위 3명의 고객에 대한 모든 정보를 조회하시오.

```
SELECT *  
FROM 고객  
ORDER BY 마일리지 DESC  
LIMIT 3;
```

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
ATRAN	대정 인터내셔널	배순웅	대표 이사	서초구 방배동 883-11	서울특별시		(02)681-6889	128790
PRDSU	황인터프라이시스	한현구	회계 과장	용산구 서빙고동 111-16	서울특별시		(02)934-1984	116898
EBITH	그린로더스	홍소현	마케팅 과장	남동구 간석동 264-11	인천광역시		(032)945-1846	81479

5. DISTINCT

■ DISTINCT

- 데이터를 조회할 때 중복된 데이터를 한 번만 보이고자 할 때 사용함
- SELECT절에 DISTINCT를 넣으면 중복된 값은 제거한 결과를 보여줌

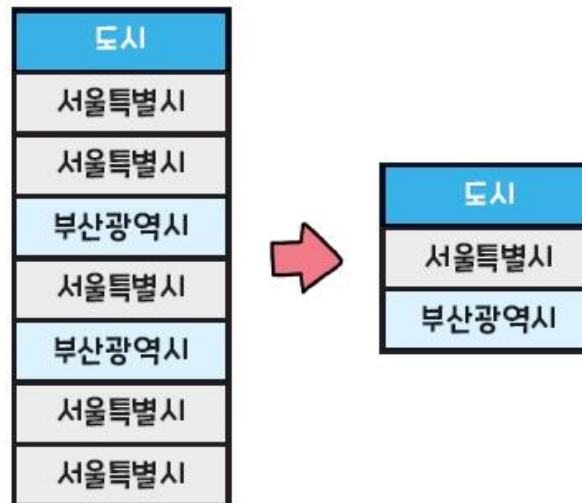


그림 2-2 DISTINCT 사용으로 중복 데이터 제거

5. DISTINCT

- [예제 2-7] 고객 테이블의 도시 컬럼에 들어있는 값 중 중복되는 도시 데이터를 한 번씩만 보이시오.

```
SELECT DISTINCT 도시  
FROM 고객;
```

▶ 실행결과

도시
서울특별시
광명시
구리시
김해시
공택시
인천광역시
대전광역시
상주시

5. DISTINCT

확인문제

다음 빈칸에 들어갈 단어를 채우시오.

중복된 값을 제거하기 위해서는 SELECT 다음에 ① 을 넣어줍니다. 레코드의 순서를 지정하기 위해서는 ② 절을 사용합니다. 이때 작은 값부터 순서대로 보이고 싶으면 ③ 를, 큰 값부터 보이고 싶으면 ④ 를 넣어줍니다. ③ 는 생략이 가능합니다.

① _____ ② _____ ③ _____ ④ _____

정답

① DISTINCT ② ORDER BY ③ ASC ④ DESC

Section 02

SQL 연산자

1. 산술 연산자

■ 연산자

- 변수나 값의 연산을 위해 사용되는 부호
- SQL에서 연산자는 단독으로 사용할 수 없음
- 컬럼이나 상수 등과 함께 사용하여 여러 가지 계산 작업을 수행할 수 있음

■ 산술 연산자

- 더하기(+), 빼기(-), 곱하기(*), 나누기(/), 나머지(%)
- DIV 연산자 : 나누기에서 정수 결과를 얻을 때 사용
- % 또는 MOD 연산자 : 나머지 구하기

1. 산술 연산자

- [예제 2-8] 두 개의 숫자 23과 5로 산술 연산자 $+$, $-$, $*$, $/$, $\%$ 를 사용한 결과를 나타내시오. DIV, MOD 연산자의 사용 결과도 함께 확인해봅니다.

```
SELECT 23 + 5 AS 더하기
, 23 - 5 AS 빼기
, 23 * 5 AS 곱하기
, 23 / 5 AS 실수나누기
, 23 DIV 5 AS 정수나누기
, 23 % 5 AS 나머지1
, 23 MOD 5 AS 나머지2;
```

테이블에서 데이터를 가져오지 않는 경우에는 FROM절을 생략하고 SELECT절만 작성합니다.

▶ 실행결과

더하기	빼기	곱하기	실수나누기	정수나누기	나머지1	나머지2
28	18	115	4.6000	4	3	3

2. 비교 연산자

■ 비교 연산자

- 두 값을 비교하여 조건에 맞는 결과를 얻고자 할 때 사용하는 연산자
- 1(True)이나 0(False) 값이 반환됨
- 크거나 같다($>=$), 작거나 같다($<=$), 크다($>$), 작다($<$), 같다($=$), 같지 않다(\neq 또는 $<>$)가 있음

2. 비교 연산자

- [예제 2-9] 두 개의 숫자 23과 5로 비교 연산자 \geq , \leq , $>$, $<$, $=$, \neq , $<>$ 를 사용한 결과를 보이시오.

```
SELECT 23 >= 5  
       ,23 <= 5  
       ,23 > 23  
       ,23 < 23  
       ,23 = 23  
       ,23 != 23  
       ,23 <> 23;
```

▶ 실행결과

23 >= 5	23 <= 5	23 > 23	23 < 23	23 = 23	23 != 23	23 <> 23
1	0	0	0	1	0	0

- [예제 2-10] 담당자가 '대표 이사'가 아닌 고객의 모든 정보를 보이시오.

```
SELECT *  
FROM 고객  
WHERE 담당자직위 <> '대표 이사';
```

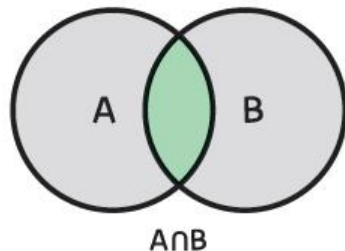
▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
ACDDR	굿모닝서울	이은광	영업 과장	송파구 잠실동 220	서울특별시		(02)978-1984	15911
ADHTR	엘케이 상사	김병현	영업 사원	동작구 흑석 3동 140-3	서울특별시		(02)345-1945	406
AIHTR	씨엔그룹	김성민	영업 사원	가학동 301	광명시	경기도	(02)31-0345	8788
ANKFR	오리안무역	최지수	마케팅 과장	성북구 길음동 136-11	서울특별시		(02)123-0345	6119
ANRFR	남해종합식품	강준	마케팅 과장	수택동 435	구리시	경기도	(034)756-2091	106

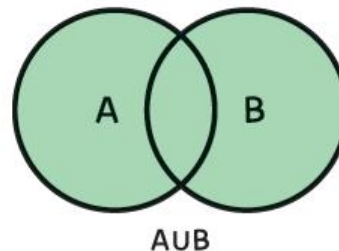
3. 논리 연산자

■ 논리 연산자

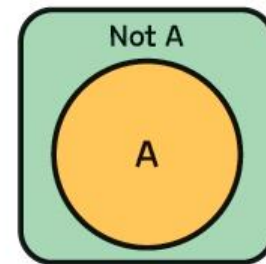
- 표현식이 참인지 거짓인지를 확인하기 위해 사용하는 연산자
- AND
 - ✓ 조건이 두 개 이상 있는 경우에 사용
 - ✓ 모든 조건이 참(True)인 레코드를 반환함
- OR
 - ✓ 여러 조건 중에서 하나 이상의 조건이 참인 레코드를 반환함
- NOT
 - ✓ 부정을 의미하는 연산자
- AND, OR, NOT 연산자를 교집합, 합집합, 여집합으로 대응시킬 수 있음



(a) AND 연산



(b) OR 연산



(c) NOT 연산

그림 2-3 논리 연산자의 집합 표현

3. 논리 연산자

- [예제 2-11] 도시가 '부산광역시'이면서 마일리지가 1,000점보다 작은 고객의 모든 정보를 보이시오.

```
SELECT *  
FROM 고객  
WHERE 도시 = '부산광역시'  
AND 마일리지 < 1000;
```

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
LLIWE	태산무역	박소영	영업 과장	부산진구 양정 1동 462-...	부산광역시		(051)555-5111	806

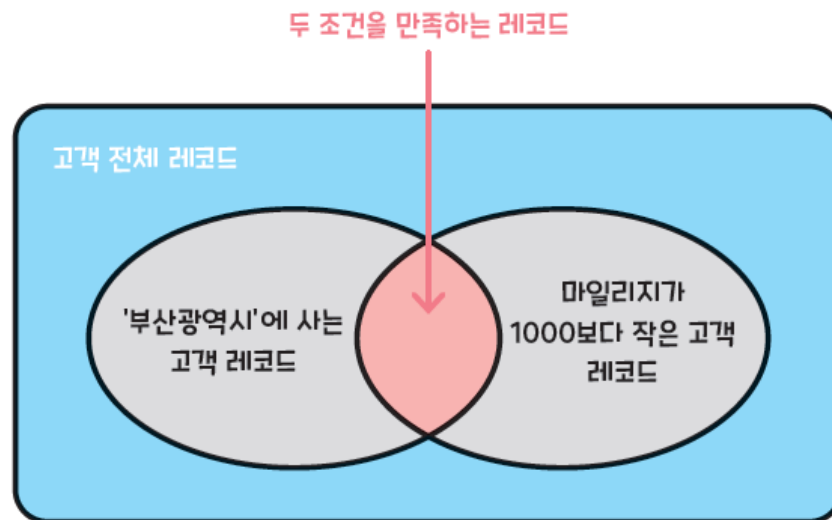


그림 2-4 [예제 2-11]의 두 가지 조건

4. 집합 연산자

■ 집합 연산자

- MySQL은 집합 연산자 중에서 합집합에 대한 연산자만 제공함
- 집합 연산자를 사용한 쿼리문에는 두 개 이상의 SELECT문이 들어있어야 함
- 두 개의 SELECT문 사이에 집합 연산자를 넣어줌
- UNION
 - ✓ 합집합 연산자
 - ✓ UNION ALL : 중복된 레코드까지 모두 다 나타내는 합집합 연산자
- UNION 연산자를 사용할 때는 주의할 점
 - ① 각 SELECT문에서 컬럼의 개수는 동일해야 한다.
 - ② 각 SELECT문에서 같은 위치에 존재하는 컬럼의 데이터 타입은 동일하거나, 상호 호환이 가능해야 함

4. 집합 연산자

■ 집합 연산자

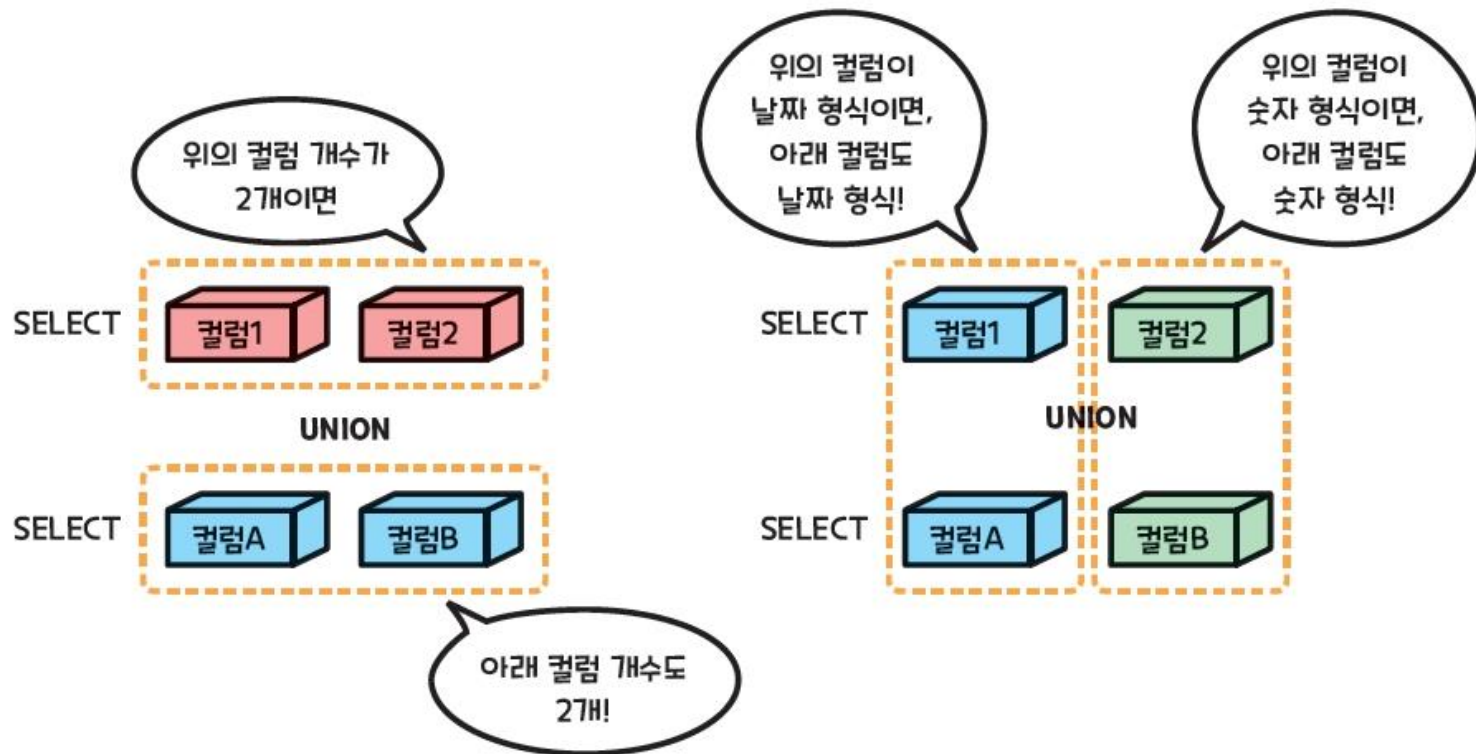


그림 2-5 UNION 연산자 사용 시 주의사항

4. 집합 연산자

- [예제 2-12] '부산광역시'에 살거나 마일리지가 1,000점보다 작은 고객에 대하여 고객번호, 담당자명, 마일리지, 도시를 보이시오. 이때 결과는 고객번호 순으로 정렬합니다.

```
SELECT 고객번호
      ,담당자명
      ,마일리지
      ,도시
FROM 고객
WHERE 도시 = '부산광역시'
UNION
SELECT 고객번호
      ,담당자명
      ,마일리지
      ,도시
FROM 고객
WHERE 마일리지 < 1000
ORDER BY 1; ● ORDER BY절은 제일 마지막에 넣어줌
```

▶ 실행결과

고객번호	담당자명	마일리지	도시
	주현욱		의정부시
TTCRA	홍지현	892	의정부시
TTMBO	안수인	7329	부산광역시
TTRAN	김희원	448	서울특별시
URLGO	김하경	865	성남시
VESSE	전혜연	414	서울특별시
ZYKLA	장재성	261	원주시

A

B

4. 집합 연산자

- [예제 2-12] '부산광역시'에 살거나 마일리지가 1,000점보다 작은 고객에 대하여 고객번호, 담당자명, 마일리지, 도시를 보이시오. 이때 결과는 고객번호 순으로 정렬합니다.
 - OR 연산자를 사용한 해결법

```
SELECT 고객번호  
      ,담당자명  
      ,마일리지  
      ,도시  
FROM 고객  
WHERE 도시 = '부산광역시'  
OR 마일리지 < 1000  
ORDER BY 1;
```

5. IS NULL 연산자

■ NULL(Unknown Value)

- NULL은 알 수 없는 값을 의미함
- 0이나 빈 문자열(Empty String)과는 다른 의미함
- 컬럼에 값이 들어있지 않은 데이터를 검색하기 위해서는 IS NULL을 사용해야 함
 - ✓ = NULL이 아님

5. IS NULL 연산자

- [예제 2-13] 지역에 값이 들어있지 않는 고객의 정보를 보이시오

```
SELECT *  
FROM 고객  
WHERE 지역 IS NULL;
```

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

- 실행결과 0개의 레코드가 나옴
- 빈 문자열을 검색했을 때는 레코드가 검색됨

```
SELECT *  
FROM 고객  
WHERE 지역 = '';
```

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
ACDDR	굿모닝서울	이은광	영업 과장	송파구 잠실동 220	서울특별시		(02)978-1984	15911
ADHTR	엘케이 상사	김병현	영업 사원	동작구 흑석 3동 140-3	서울특별시		(02)345-1945	406
ANKFR	오리안무역	최지수	마케팅 과장	성북구 길음동 136-11	서울특별시		(02)123-0345	6119
ANSFR	서보엠엔에프	김민경	영업 사원	동작구 대방동 36-8	서울특별시		(02)211-2221	729
ATRAN	대정 인터내셔널	배순용	대표 이사	서초구 방배동 883-11	서울특별시		(02)681-6889	128790
AUSBL	제원인터내셔널	박지혜	영업 사원	서대문구 남가좌 1동 121	서울특별시		(02)811-1234	47865
BOGZA	비류큐푸드	최수영	대표 이사	서초구 서호동 120-3	서울특별시		(02)456-9876	100

빈 문자열

5. IS NULL 연산자

■ [예제 2-13] 지역에 값이 들어있지 않는 고객의 정보를 보이시오

- IS NULL을 사용했을 때 검색되지 않은 이유
 - ✓ NULL과 빈 문자열은 의미가 서로 다르기 때문
 - ✓ CSV 파일에서 테이블로 가져오기를 했을 때 값이 들어있지 않은 셀은 NULL이 아닌 빈 문자열로 저장됨
 - ✓ 따라서 IS NULL을 사용하여 레코드를 조회하면 결과가 나오지 않음
- 해결 방법) UPDATE문으로 문자열을 NULL로 변경하기

```
UPDATE 고객
SET 지역 = NULL
WHERE 지역 = '';
```

- 지역 컬럼의 값이 NULL인 레코드를 다시 검색하기

```
SELECT *
FROM 고객
WHERE 지역 IS NULL;
```

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
ACDDR	굿모닝서울	이은광	영업 과장	송파구 잠실동 220	서울특별시	NULL	(02)978-1984	15911
ADHTR	엘케이 상사	김병현	영업 사원	동작구 흑석 3동 140-3	서울특별시	NULL	(02)345-1945	406
ANKFR	오리안무역	최지수	마케팅 과장	성북구 길음동 136-11	서울특별시	NULL	(02)123-0345	6119
ANSFR	서보엠엔에프	김민경	영업 사원	동작구 대방동 36-8	서울특별시	NULL	(02)211-2221	729
ATRAN	대정 인터내셔널	배순용	대표 이사	서초구 방배동 883-11	서울특별시	NULL	(02)681-6889	128790
AUSBL	제원인터내셔널	박지혜	영업 사원	서대문구 남가좌 1동 121	서울특별시	NULL	(02)811-1234	47865
7007A	비큐푸드	홍수영	대표 이사	서초구 서초동 100-2	서울특별시	NULL	(02)456-9876	100

5. IS NULL 연산자

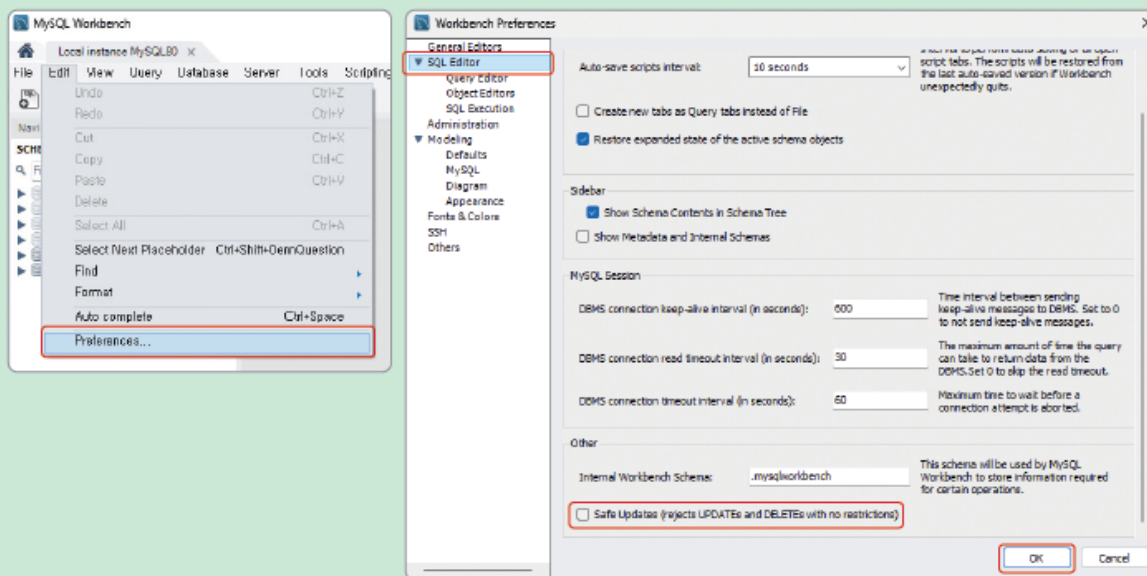
하나 더 알기

UPDATE나 DELETE가 제대로 동작하지 않는 경우

MySQL에서는 테이블에서 키를 이용한 UPDATE 또는 DELETE만을 허용합니다. 즉, UPDATE나 DELETE문에서 WHERE절이 없거나 WHERE절에 키 컬럼 외의 컬럼을 사용한 조건문이 사용되면 오류가 발생합니다.

Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a KEY column. To disable safe mode, toggle the option in Preferences -> SQL Editor and reconnect.

이를 해결하기 위해서는 Safe Mode 옵션을 해제해주어야 합니다. 워크bench의 [Edit]-[Preferences] 메뉴에서 [SQL Editor]를 선택한 다음, 체크되어 있는 [Safe Updates] 항목을 클릭하여 체크 해제하고 <OK> 버튼을 클릭한 후 워크bench를 재실행합니다.



만일 일시적으로 Safe Mode 해제 옵션을 적용하려면 다음과 같이 작성합니다.

```
SET SQL_SAFE_UPDATES = 0;
```

6. IN과 BETWEEN...AND 연산자

■ IN과 BETWEEN...AND 연산자

- WHERE절을 조금 더 쉽고 효율적으로 사용할 수 있게 도와주는 연산자

■ IN

- 동일한 컬럼에 대해서 OR을 수행해야 하는 경우 더 간단히 해결해주는 연산자

■ BETWEEN...AND

- 동일한 컬럼에 대해서 '~이상 ~이하'라는 조건이 있는 경우에 더 간단히 해결해주는 연산자

6. IN과 BETWEEN...AND 연산자

- [예제 2-14] 담당자직위가 '영업 과장'이거나 '마케팅 과장'인 고객에 대하여 고객번호, 담당자명, 담당자직위를 보이시오.

```
SELECT 고객번호  
      ,담당자명  
      ,담당자직위  
FROM 고객  
WHERE 담당자직위 = '영업 과장'  
OR 담당자직위 = '마케팅 과장';
```

- IN을 사용한 해결 방법

```
SELECT 고객번호  
      ,담당자명  
      ,담당자직위  
FROM 고객  
WHERE 담당자직위 IN ('영업 과장', '마케팅 과장');
```

▶ 실행결과

고객번호	담당자명	담당자직위
ACDDR	이은광	영업 과장
ANKFR	최지수	마케팅 과장
ANRFR	강준	마케팅 과장
CCOPI	김효정	영업 과장
CSURI	이소영	영업 과장

6. IN과 BETWEEN...AND 연산자

- [예제 2-15] 마일리지가 100,000점 이상 200,000점 이하인 고객에 대해 담당자명, 마일리지를 보이시오.

```
SELECT 담당자명  
      ,마일리지  
FROM 고객  
WHERE 마일리지 >= 100000  
AND 마일리지 <= 200000;
```

- BETWEEN...AND를 사용한 해결 방법

```
SELECT 담당자명  
      ,마일리지  
FROM 고객  
WHERE 마일리지 BETWEEN 100000 AND 200000;
```

▶ 실행결과

담당자명	마일리지
배순용	128790
한현구	116898

7. LIKE 연산자

■ LIKE 연산자

- 특정 문자열이 지정된 패턴과 일치하는지 확인할 때 사용함
- 와일드카드 문자(% , _)와 함께 사용하여 원하는 결과를 얻을 수 있음
- LIKE와 와일드카드 문자를 함께 사용하지 않으면 = 기호를 사용한 것과 동일한 결과가 나옴

표 2-1 와일드카드 문자와 사용 예

와일드카드 문자	설명	예시 문제	SQL 코드
%	0개 이상의 문자를 가진 문자열을 포함	주소 컬럼 값 중 '흑석'이 들어가는 것	주소 LIKE '%흑석%'
		도시 컬럼 값 중 '광역시'로 끝나는 것	도시 LIKE '%광역시'
_(밑줄)	정확히 한 개의 문자를 포함	고객번호 두 번째가 'C'인 것	고객번호 LIKE '_C%'
		고객번호 세 번째가 'C'인 것	고객번호 LIKE '##_C%'
		고객번호가 5글자이면서 세 번째가 'C'인 것	고객번호 LIKE '##_C_#'

하나 더 알기 Y

와일드카드 문자란?

SQL에서 와일드카드(Wildcard) 문자는 문자열을 비교하거나 검색할 때 패턴 매칭을 위해 사용되는 특수 문자를 나타냅니다. 와일드카드 문자를 사용하면 일부 문자열을 일치시키거나 패턴에 따라 문자열을 검색할 수 있어 데이터 검색 및 필터링에 유용합니다.



7. LIKE 연산자

- [예제 2-16] 도시가 '광역시'이면서 고객번호 두 번째 글자 또는 세 번째 글자가 'C'인 고객의 모든 정보를 보이시오.

```
SELECT *  
FROM 고객  
WHERE 도시 LIKE '%광역시'  
AND (고객번호 LIKE '_C%' OR 고객번호 LIKE '__C%');
```

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
ICKOU	인터비스	정대광	회계 과장	서구 연희동 400-16	인천광역시	NULL	(032)44-1114	857
NTCCE	코스리아 무역	유석희	마케팅 과장	남동구 간석 3동 270-8	인천광역시	NULL	(032)944-3687	112
RICPE	뉴가든상사	김지호	영업 사원	금정구 청룡동 168-14	부산광역시	NULL	(051)12-1122	1177

7. LIKE 연산자

■ 조건의 우선순위

- WHERE절에서 조건의 우선순위를 지정해야 하는 경우 소괄호(())를 사용함
- 우선순위를 표현하지 않은 경우에는 잘못된 결과가 반환될 수 있음
- [예] [예제 2-16]에서 괄호를 넣지 않으면 A가 먼저 수행된 후에 B가 수행됨

WHERE 도시 LIKE '%광역시' AND 고객번호 LIKE '_C%' OR 고객번호 LIKE '___C%';

The diagram illustrates the execution order of the SQL query without parentheses. A blue arrow labeled 'A' points to the condition '도시 LIKE '%광역시' AND 고객번호 LIKE '_C%'' which is enclosed in a solid blue box. A red arrow labeled 'B' points to the condition 'OR 고객번호 LIKE '___C%'' which is enclosed in a dashed red box. This indicates that condition A is evaluated first, followed by condition B.

- 따라서 C와 같이 괄호를 사용하여 우선순위를 명시적으로 지정해야 함

WHERE 도시 LIKE '%광역시' AND (고객번호 LIKE '_C%' OR 고객번호 LIKE '___C%');

The diagram illustrates the execution order of the SQL query with parentheses. An orange arrow labeled 'C' points to the entire condition '(고객번호 LIKE '_C%' OR 고객번호 LIKE '___C%')' which is enclosed in a solid orange box. This indicates that the entire parenthesized condition is evaluated as a single unit after the first part of the WHERE clause.

7. LIKE 연산자

하나 더 알기 Y

REGEXP 연산자

REGEXP 연산자 또는 REGEXP_LIKE 함수를 사용하면 LIKE보다 좀 더 정교한 패턴 매칭을 통해 데이터를 검색할 수 있습니다. 예를 들어 '푸드'가 들어가 있는 고객회사명을 검색해보겠습니다. REGEXP 연산자를 사용하면 컬럼명 REGEXP 패턴 형식으로 작성합니다.

```
SELECT *  
FROM 고객  
WHERE 고객회사명 REGEXP '푸드';
```

REGEXP_LIKE 함수를 사용하면 함수명 안에 컬럼명과 패턴을 함께 넣습니다.

```
SELECT *  
FROM 고객  
WHERE REGEXP_LIKE(고객회사명, '푸드');
```

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
BQQZA	비류큐푸드	황수영	대표 이사	서초구 서초동 120-3	서울특별시		(02)456-9876	100
CCOPI	진산푸드	김효정	영업 과장	구로구 오류 1동 101	서울특별시		(02)100-1670	1034
ECRTH	플라워푸드	권덕현	영업 사원	종로구 내자동 190-53	서울특별시		(02)954-1397	364

7. LIKE 연산자

[표 2-2]를 통해 몇 가지 패턴에 대한 사용 방법을 확인해봅시다.

표 2-2 자주 사용되는 패턴 문자와 예시

패턴 문자	설명	예시 문제 → SQL 코드
.	단일 문자	고객회사명이 5글자 이상 → WHERE 고객회사명 REGEXP '.....'
	OR	도시가 '과천시' 또는 '오산시' → WHERE 도시 REGEXP '과천시 오산시'
[]	[] 안에 나열된 패턴에 해당하는 문자열	도시가 '인천광역시' 또는 '부산광역시' 또는 '대전광역시' → WHERE 도시 REGEXP '[인천 부산 대전]광역시'
^	시작하는 문자열	'한'으로 시작하는 고객회사명 → WHERE 고객회사명 REGEXP '^한'
\$	끝나는 문자열	고객회사명이 5글자이면서 데이터 중 4번째 글자가 '푸'인 고객회사명 → WHERE 고객회사명 REGEXP '^...푸.\$'
*	0번 이상	'정'이 0번 이상 들어있는 담당자명 → WHERE 담당자명 REGEXP '정*'
+	1번 이상	'정'이 1번 이상 들어있는 담당자명 → WHERE 담당자명 REGEXP '정+'
[^문자]	괄호 안의 문자 제외	고객번호 맨 뒷자리에 'H'~'Z'가 들어가지 않는 고객번호 → WHERE 고객번호 REGEXP '[^H-Z]\$'
{m}	m회	고객회사명이 5글자 → WHERE 고객회사명 REGEXP '^.{5}\$'; 'T'나 'S'가 2회 나타나는 고객번호(TT, SS, TS, ST) → WHERE 고객번호 REGEXP '[TS]{2}';
{m,n}	m회 이상 n회 이하	'T'나 'S'가 1회 이상 2회 이하 나타나는 고객번호(T, S, TT, SS, TS, ST) → WHERE 고객번호 REGEXP '[TS]{1,2}';

패턴을 다양하게 응용하면 복잡한 조건의 데이터도 잘 매칭하여 검색할 수 있습니다.

7. LIKE 연산자

확인문제

다음 문제를 읽고 빈칸을 채우시오.

1. 전화번호 뒷자리가 '45'로 끝나는 고객 정보

```
SELECT *  
FROM 고객  
WHERE 전화번호 LIKE ;
```

2. 전화번호 중 뒤에서 세 번째 부터가 '45'인 고객 정보

```
SELECT *  
FROM 고객  
WHERE 전화번호 LIKE ;
```

3. 전화번호 중에 '45'가 들어간 고객 정보

```
SELECT *  
FROM 고객  
WHERE 전화번호 LIKE ;
```

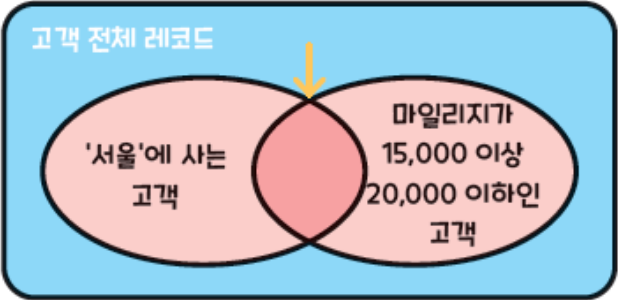
정답

1. '%45' 2. '%45_' 3. '%45%'

점검문제

문제 1

다음 실행 결과를 참고하여 '서울'에 사는 고객 중에 마일 리지가 15,000점 이상 20,000점 이하인 고객의 모든 정 보를 보이시오.



▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
ACDDR	굿모닝서울	이은팔	영업 과장	송파구 잠실동 220	서울특별시	NULL	(02)978-1984	15911

문제 2

한빛무역의 고객들은 어느 지역, 어느 도시에 사는지 지역과 도시를 한 번씩만 보이시오. 이때 결과를 지역 순으로 나타내고, 동일 지역에 대해서는 도시 순으로 나타냅니다.

▶ 실행결과

지역	도시
NULL	대전광역시
NULL	부산광역시
NULL	서울특별시
NULL	인천광역시
강원도	원주시
강원도	춘천시
경기도	과천시
경기도	광명시
경기도	구리시
경기도	동두천시

문제 3

‘춘천시’나 ‘과천시’ 또는 ‘광명시’에 사는 고객 중에서 담당자직위에 ‘이사’ 또는 ‘사원’이 들어가는 고객의 모든 정보를 보이시오.

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
AIHTR	씨엔그룹	김성민	영업 사원	가학동 301	광명시	경기도	(02)31-0345	8788
LAAHI	성민식품상사	신민주	영업 사원	중앙로 3가 150-1	춘천시	강원도	(0369)39-1243	382
LKOFO	진영무역	조희진	대표 이사	중앙동 427-10	과천시	경기도	(02)768-7688	8607
NSHCO	은혜식품	이승연	영업 사원	소하동 11-3	광명시	경기도	(02)489-9184	387

문제 4

‘광역시’나 ‘특별시’에 살지 않는 고객들 중에서 마일리지 많은 상위 고객 3명의 모든 정보를 보이시오.

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
LZAWO	오뚜락	주민영	대표 이사	건업동 111-16	제주시	제주도	(064)643-2871	8838
AIHTR	씨엔그룹	김성민	영업 사원	가학동 301	광명시	경기도	(02)31-0345	8788
BEVBS	케이리탈리스	오선주	영업 사원	구산동 17-111	김해시	경상남도	(0525)342-3333	8758

문제 5

지역에 값이 들어있는 고객 중에서 담당자직위가 '대표 이사'인 고객을 빼고 보이시오.

▶ 실행결과

고객번호	고객회사명	담당자명	담당자직위	주소	도시	지역	전화번호	마일리지
AIHTR	씨엔그룹	김성민	영업 사원	가학동 301	광명시	경기도	(02)31-0345	8788
ANRFR	남해종합식품	강준	마케팅 과장	수택동 435	구리시	경기도	(034)756-2091	106
BEVBS	케이리칼리스	오선주	영업 사원	구산동 17-111	김해시	경상남도	(0525)342-3333	8758
...	진가식품	...	영업 사원	(033)945-1923	...

Thank you!