SQL 문의 개요

SELECT 문

■ SQL(Structured Query Language, 구조화된 질의 언어)

- 데이터베이스에서 사용되는 일종의 공통 언어
- NCITS(국제표준화위원회)에서 ANSI/ISO SQL이라는 명칭의 SQL 표준을 관리하고 있음
- 1992년에 제정된 ANSI-92 SQL과 1999년에 제정된 ANSI-99 SQL을 대부분의 DBMS 회사에서 SQL 표준으로 사용하고 있음
- 각 회사는 ANSI-92/99 SQL의 표준을 준수하면서도 자신의 제품 특성을 반영한 SQL에 별도의 이름을 붙임
- MySQL에서는 그냥 SQL, 오라클에서는 PL/SQL, SQL Server에서는 Transact SQL(T-SQL) 사용

1/55

고객 테이블에서 고객아이디, 고객이름, 등급 속성을 검색해보자. 결과 테이블 \$ ** 고객이데(***) 고객이를 (** 등급**) 1 apple 정소화 gold 2 banana 김선우 vip 3 carrot 고명석 gold 4 orange 1 Age 8 silver 5 melon 성원용 gold 6 peach 오령준 silver 7 pear 채광주 silver

SELECT 문

_ 기본 검색문

- SELECT 키워드와 함께 검색하고 싶은 속성의 이름을 나열
- FROM 키워드와 함께 검색하고 싶은 속성이 있는 테이블의 이름을 나열
- 검색 결과는 테이블 형태로 반환됨

SELECT [ALL | DISTINCT] 속성_리스트 FROM 테이블_리스트;

- ALL
 - 결과 테이블이 레코드의 중복을 허용하도록 지정, 생략 가능
- DISTINCT
 - 결과 테이블이 레코드의 중복을 허용하지 않도록 지정

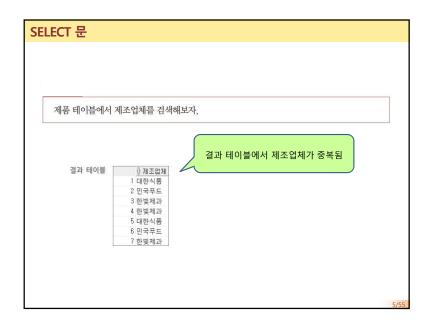
2/5

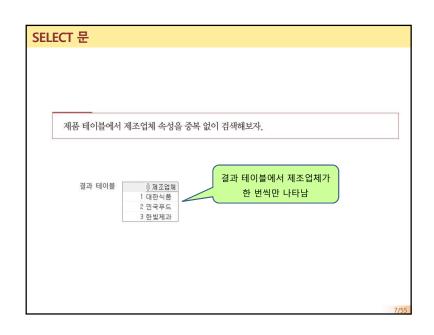
SELECT 문

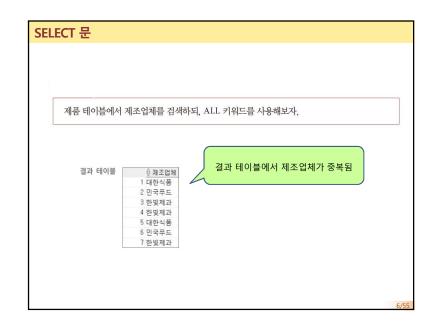
고객 테이블에 존재하는 모든 속성을 검색해보자.

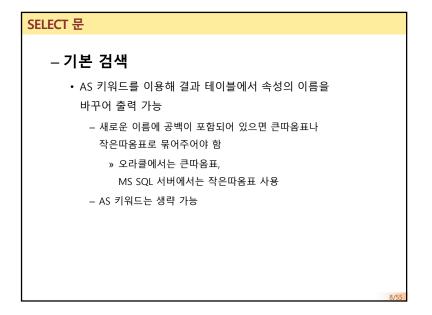
결과 테이블

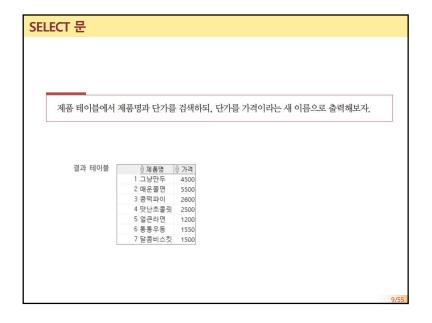
	♦ 고객아이디	◈ 고객이름	\$ LHOI	◈ 등급	♦ 직업	♦ 적립금
-1	apple	정소화	20	gold	학생	1000
2	banana	김선우	25	vip	간호사	2500
3	carrot	고명석	28	gold	교사	4500
4	orange	김용욱	22	silver	학생	0
5	melon	성원용	35	gold	회사원	5000
6	peach	오형준	(null)	silver	의사	300
7	pear	채광주	31	silver	회사원	500

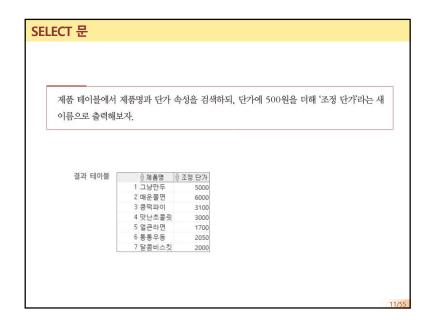












SELECT 문

_ 산술식을 이용한 검색

- SELECT 키워드와 함께 산술식 제시
 - 산술식: 속성의 이름과 +, -, *, / 등의 산술 연산자와 상수로 구성
- 속성의 값이 실제로 변경되는 것은 아니고 결과 테이블에서만 계산된 결과 값이 출력됨

10 /EI

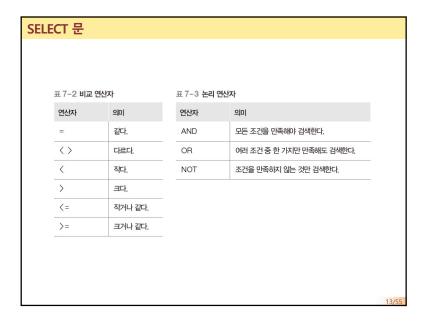
SELECT 문

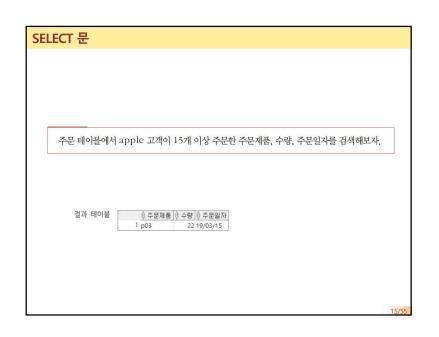
– 조건 검색

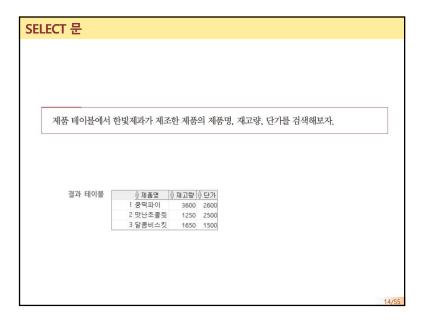
• 조건을 만족하는 데이터만 검색

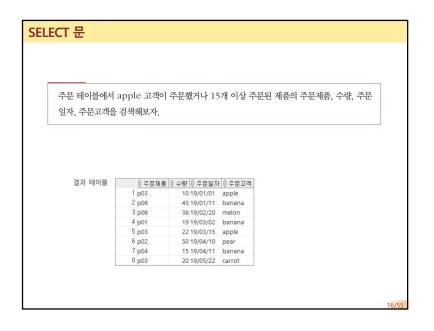
SELECT [ALL | DISTINCT] 속성_리스트 FROM 테이블_리스트 [WHERE 조건];

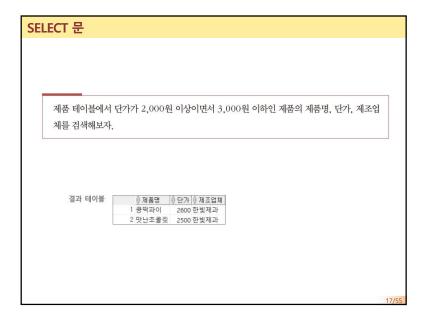
- WHERE 키워드와 함께 비교 연산자와 논리 연산자를 이용한 검색 조건 제시
 - 숫자뿐만 아니라 문자나 날짜 값을 비교하는 것도 가능
 - » 예) 'A' < 'C'
 - » 예) '2019-12-01' < '2019-12-02'
 - 조건에서 문자나 날짜 값은 작은따옴표로 묶어서 표현

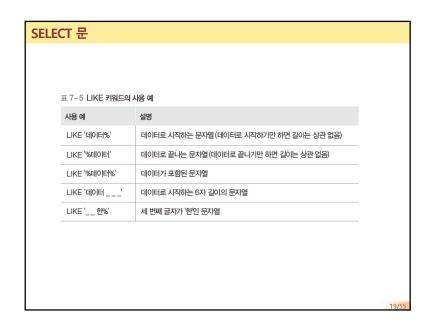




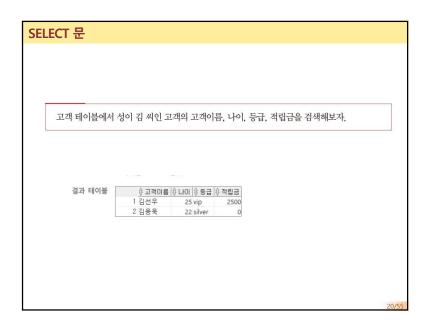


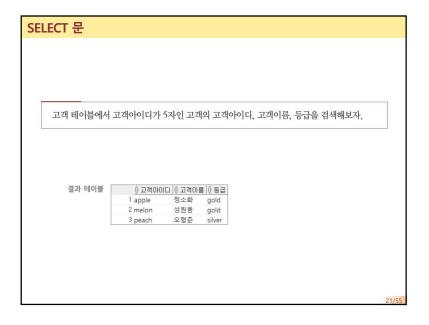






SELECT 문 - LIKE를 이용한 검색 • LIKE 키워드를 이용해 부분적으로 일치하는 데이터를 검색 • 문자열을 이용하는 조건에만 LIKE 키워드 사용 가능 표7-4 LIKE 키워드와함께 사용할수있는 기호 기호 설명 % 이개 이상의 문자 (문자의 내용과 개수는 상관 없음) 1개의 문자 (문자의 내용은 상관 없음)





SELECT 문	
고객 테이블에서 나이가 아직 입력되지 않은 고객의 고객이름을 검색해보자.	
결과 테이블 <u>\$ 고역이름</u> 1 오형준	
	23/55

SELECT 문

– NULL을 이용한 검색

- IS NULL 키워드를 이용해 특정 속성의 값이 널 값인지를 비교
- IS NOT NULL 키워드를 이용해 특정 속성의 값이 널 값이 아닌지를 비교
- 검색 조건에서 널 값은 다른 값과 크기를 비교하면 결과가 모두 거짓이 됨

22

ELECT 문				
고객 테이블에서	나시기 시미 이글		ㄹㅇ 기계케터기	
고객 테이들에서	나이가 이미 입덕	크 고객의 고객이	금을 검색애보사.	
결과 테이블	∜ 고객이름			
241 -11-12	1 정소화			
	2 김선우			
	3 고명석			
	4 김용욱			
	5 성원용			
	6 채광주			
L	V 410 T			

SELECT 문

_ 정렬 검색

- ORDER BY 키워드를 이용해 결과 테이블 내용을 사용자가 원하는 순서로 출력
- ORDER BY 키워드와 함께 정렬 기준이 되는 속성과 정렬 방식을 지정
 - 오름차순(기본): ASC / 내림차순: DESC
 - 널 값은 오름차순에서는 맨 마지막에 출력되고, 내림차순에서는 맨 먼저 출력됨
 - 여러 기준에 따라 정렬하려면 정렬 기준이 되는 속성을 차례대로 제시 SELECT [ALL ! DISTINCT] 소설 링스트

SELECT [ALL | DISTINCT] 속성_리스트 FROM 테이블_리스트

[WHERE 조건]

[ORDER BY 속성_리스트 [ASC | DESC]];

25/55

SELECT 문

주문 테이블에서 수량이 10개 이상인 주문의 주문고객, 주문제품, 수량, 주문일자를 검색해 보자. 단, 주문제품을 기준으로 오름차순 정렬하고, 동일 제품은 수량을 기준으로 내림차순 정렬해보자.

결과 테이블

	♦ 주문고객	♦ 주문제품	♦ 수량	♦ 주문일자
1	banana	p01	19	19/03/02
2	pear	p02	50	19/04/10
3	apple	p03	22	19/03/15
4	carrot	p03	20	19/05/22
5	apple	p03	10	19/01/01
6	banana	p04	15	19/04/11
7	banana	p06	45	19/01/11
8	melon	p06	36	19/02/20

SELECT 문

고객 테이블에서 고객이름, 등급, 나이를 검색하되, 나이를 기준으로 내림차순 정렬해보자.

결과 테이블

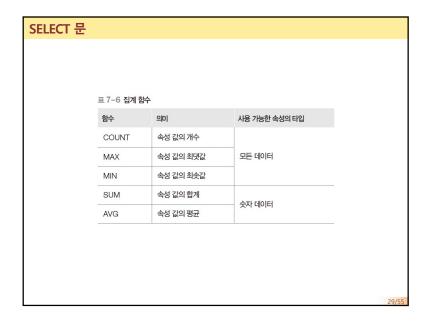
	◈ 고객이름	♦ 등급	↓ LHOI
1	오형준	silver	(null)
2	성원용	gold	35
3	채광주	silver	31
4	고명석	gold	28
5	김선우	vip	25
6	김용욱	silver	22
7	정소화	gold	20

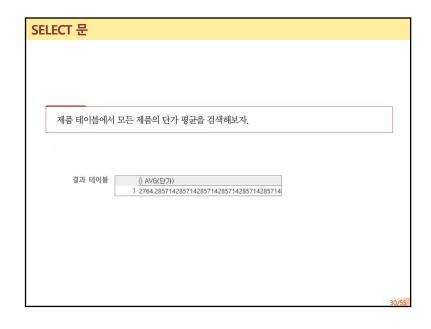
26/55

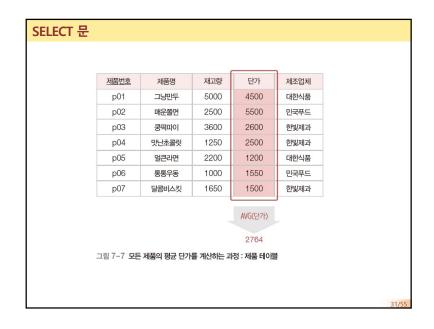
SELECT 문

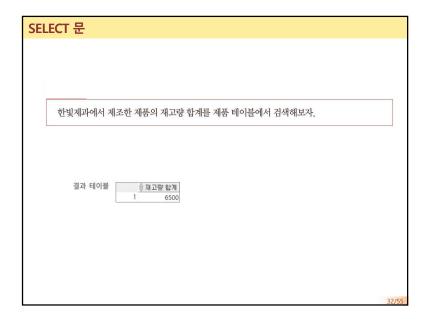
_ 집계 함수를 이용한 검색

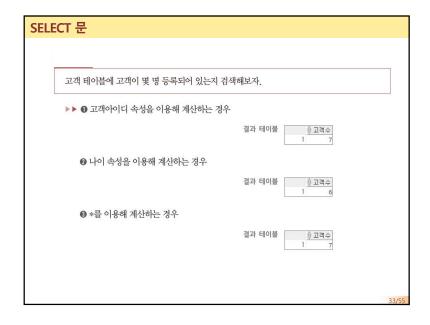
- 특정 속성 값을 통계적으로 계산한 결과를 검색하기 위해 집계 함수를 이용
 - 집계 함수(aggregate function)
 - » 열 함수(column function)라고도 함
 - » 개수, 합계, 평균, 최댓값, 최솟값의 계산 기능을 제공
- 집계 함수 사용 시 주의 사항
 - 집계 함수는 널인 속성 값은 제외하고 계산함
 - 집계 함수는 WHERE 절에서는 사용할 수 없고, SELECT 절이나 HAVING 절에서만 사용 가능

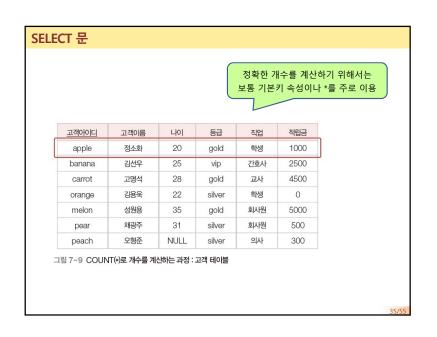


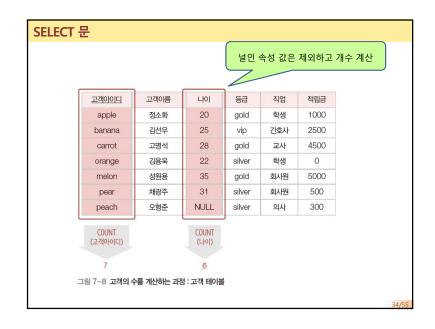


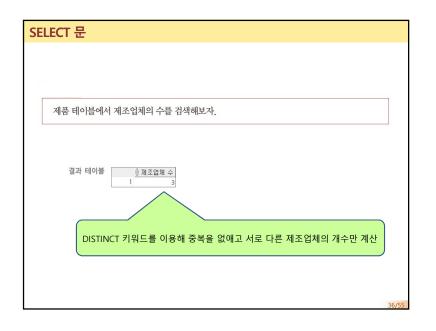










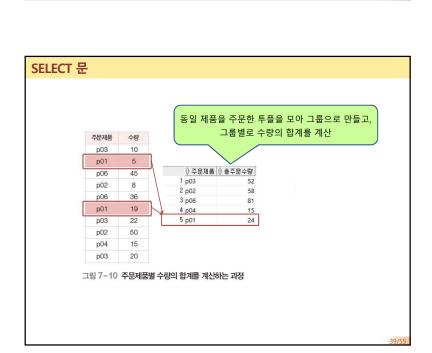


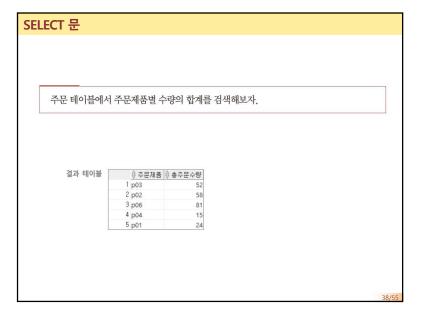
SELECT 문

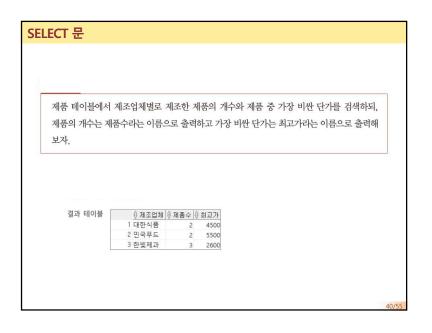
_ 그룹별 검색

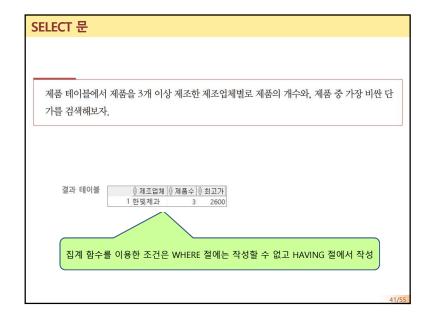
SELECT [ALL | DISTINCT] 속성_리스트
FROM 테이블_리스트
[WHERE 조건]
[GROUP BY 속성_리스트 [HAVING 조건]]
[ORDER BY 속성_리스트 [ASC | DESC]];

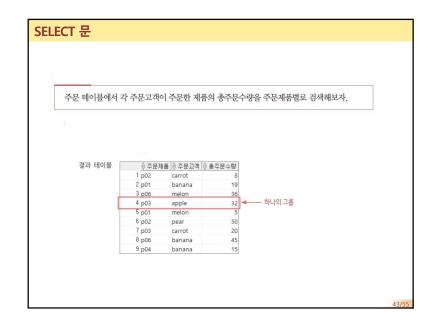
- GROUP BY 키워드를 이용해 특정 속성의 값이 같은 투플을 모아 그룹을 만들고, 그룹별로 검색
 - GROUP BY 키워드와 함께 그룹을 나누는 기준이 되는 속성을 지정
- HAVING 키워드를 함께 이용해 그룹에 대한 조건을 작성
- 그룹을 나누는 기준이 되는 속성을 SELECT 절에도 작성하는 것이 좋음







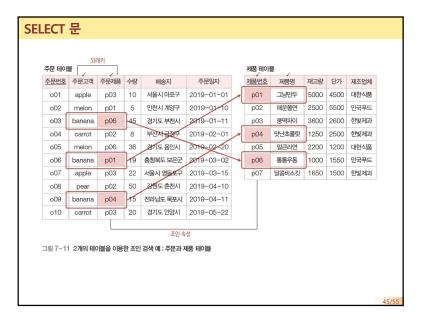


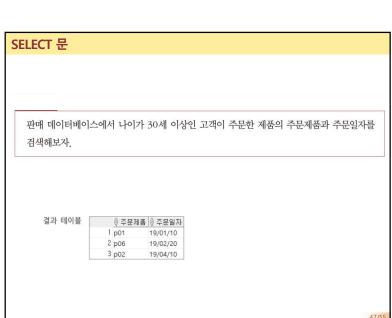


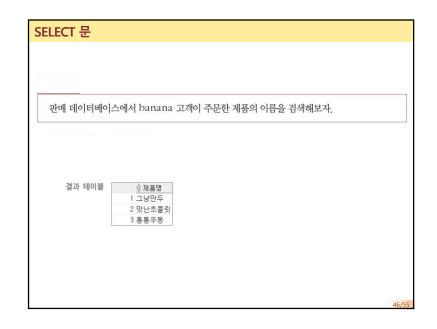
SELECT 문 고객 테이블에서 적립금 평균이 1,000원 이상인 등급에 대해 등급별 고객수와 적립금 평균을 검색해보자. 결과 테이블 ● 등급 ◈ 고객수 ◈ 평균적립금 1 gold 3 3500 2 vip 1 2500

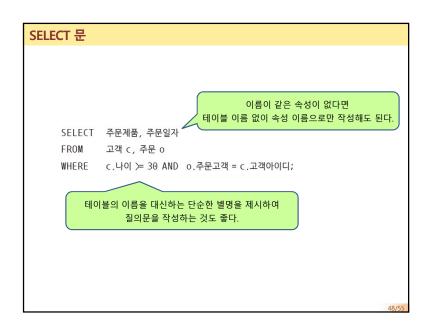
SELECT 문

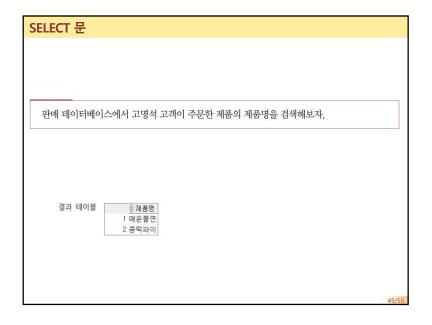
- _ 여러 테이블에 대한 조인 검색
 - 조인 검색: 여러 개의 테이블을 연결하여 데이터를 검색하는 것
 - 조인 속성: 조인 검색을 위해 테이블을 연결해주는 속성
 - 연결하려는 테이블 간에 조인 속성의 이름은 달라도 되지만도메인은 같아야 함
 - 일반적으로 외래키를 조인 속성으로 이용함
 - FROM 절에 검색에 필요한 모든 테이블을 나열
 - WHERE 절에 조인 속성의 값이 같아야 함을 의미하는 조인 조건을 제시
 - 같은 이름의 속성이 서로 다른 테이블에 존재할 수 있기 때문에 속성 이름 앞에 해당 속성이 소속된 테이블의 이름을 표시 예) 주문.주문고객

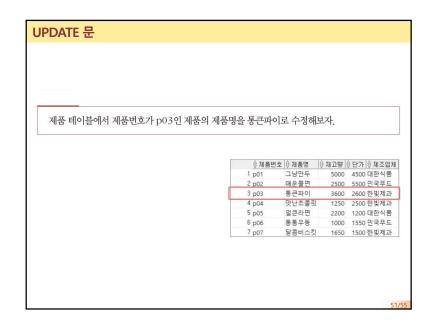












UPDATE 문

- 테이블에 저장된 투플에서 특정 속성의 값을 수정

UPDATE 테이블_이름
SET 속성_이름1 = 값1, 속성_이름2 = 값2, …
[WHERE 조건];

- SET 키워드 다음에 속성 값을 어떻게 수정할 것인지를 지정
- WHERE 절에 제시된 조건을 만족하는 레코드들에 대해서만 속성 값을 수정
 - WHERE 절을 생략하면 테이블에 존재하는 모든 레코드를 대상으로 수정

50/55

UPDATE 문

제품 테이블에 있는 모든 제품의 단가를 10% 인상해보자. 그런 다음 제품 테이블의 모든 내용을 검색하여 인상 내용을 확인해보자.

♦ 제품번호	♦ 제품명	♦ 재고량	♦ 단가	🖟 제 조업체
1 p01	그냥만두	5000	4950	대한식품
2 p02	매운쫄면	2500	6050	민국푸드
3 p03	통큰파이	3600	2860	한빛제과
4 p04	맛난초콜릿	1250	2750	한빛제과
5 p05	얼큰라면	2200	1320	대한식품
6 p06	통통우동	1000	1705	민국푸드
7 p07	달콤비스킷	1650	1650	한빛제과

13

