

위치	오류유형	수정 전	수정 후
70p 번호 : 15	문제-보기(지문)	<p>15번 문제 보기</p> <p>주문</p> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <p>주문번호</p> <p>상품코드</p> </div> <div style="border: 1px solid black; padding: 5px;"> <p>주문일시</p> <p>주문수량</p> <div style="border: 2px solid red; padding: 2px; display: inline-block;"> <p>상품명</p> <p>회원번호</p> </div> </div>	<p>15번 문제 보기</p> <p>주문</p> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <p>주문번호</p> <p>상품코드</p> </div> <div style="border: 1px solid black; padding: 5px;"> <p>주문일시</p> <p>주문수량</p> <p style="color: red;">회원번호</p> <p style="color: red;">회원명</p> </div>

위치	오류유형	수정 전	수정 후																																																		
99p ④	개념,공식-설명	<p>수정 전</p> <p>④ NVL2(인수1, 인수2, 인수3) 인수1이 NULL이 아닌 경우 인수2를 반환하고 NULL인 경우 인수3을 반환하는 함수이다.</p> <p>예 NVL2(REVIEW_SCORE, '리뷰있음', '리뷰없음')</p> <p>→ REVIEW_SCORE(컬럼) 데이터가 NULL이 아닐 경우 '리뷰있음'을 반환하고 NULL일 경우 '리뷰없음'을 반환</p> <pre>1 SELECT * FROM REVIEW;</pre> <table border="1"> <thead> <tr> <th>#</th> <th>ABC PRODUCT_ID</th> <th>ABC MEMBER_NO</th> <th>123 REVIEW_SCORE</th> <th>ABC REVIEW</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>100001</td> <td>1</td> <td>5</td> <td>넘나 좋다는 것!</td> </tr> <tr> <td>2</td> <td>100001</td> <td>3</td> <td>0</td> <td>[NULL]</td> </tr> <tr> <td>3</td> <td>100001</td> <td>4</td> <td>1</td> <td>별로..</td> </tr> <tr> <td>4</td> <td>100001</td> <td>7</td> <td>3</td> <td>그냥저냥 쓸만 했음</td> </tr> </tbody> </table> <pre>1 SELECT MEMBER_NO, 2 NVL2(REVIEW_SCORE, '리뷰있음', '리뷰없음') AS REVIEW_CHECK 3 FROM REVIEW;</pre>	#	ABC PRODUCT_ID	ABC MEMBER_NO	123 REVIEW_SCORE	ABC REVIEW	1	100001	1	5	넘나 좋다는 것!	2	100001	3	0	[NULL]	3	100001	4	1	별로..	4	100001	7	3	그냥저냥 쓸만 했음	<p>수정 후</p> <p>④ NVL2(인수1, 인수2, 인수3) 인수1이 NULL이 아닌 경우 인수2를 반환하고 NULL인 경우 인수3을 반환하는 함수이다.</p> <p>예 NVL2(REVIEW , '리뷰있음', '리뷰없음')</p> <p>→ REVIEW (컬럼) 데이터가 NULL이 아닐 경우 '리뷰있음'을 반환하고 NULL일 경우 '리뷰없음'을 반환</p> <pre>1 SELECT * FROM REVIEW;</pre> <table border="1"> <thead> <tr> <th>#</th> <th>ABC PRODUCT_ID</th> <th>ABC MEMBER_NO</th> <th>123 REVIEW_SCORE</th> <th>ABC REVIEW</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>100001</td> <td>1</td> <td>5</td> <td>넘나 좋다는 것!</td> </tr> <tr> <td>2</td> <td>100001</td> <td>3</td> <td>0</td> <td>[NULL]</td> </tr> <tr> <td>3</td> <td>100001</td> <td>4</td> <td>1</td> <td>별로..</td> </tr> <tr> <td>4</td> <td>100001</td> <td>7</td> <td>3</td> <td>그냥저냥 쓸만 했음</td> </tr> </tbody> </table> <pre>1 SELECT MEMBER_NO, 2 NVL2(REVIEW , '리뷰있음', '리뷰없음') AS REVIEW_CHECK 3 FROM REVIEW;</pre>	#	ABC PRODUCT_ID	ABC MEMBER_NO	123 REVIEW_SCORE	ABC REVIEW	1	100001	1	5	넘나 좋다는 것!	2	100001	3	0	[NULL]	3	100001	4	1	별로..	4	100001	7	3	그냥저냥 쓸만 했음
#	ABC PRODUCT_ID	ABC MEMBER_NO	123 REVIEW_SCORE	ABC REVIEW																																																	
1	100001	1	5	넘나 좋다는 것!																																																	
2	100001	3	0	[NULL]																																																	
3	100001	4	1	별로..																																																	
4	100001	7	3	그냥저냥 쓸만 했음																																																	
#	ABC PRODUCT_ID	ABC MEMBER_NO	123 REVIEW_SCORE	ABC REVIEW																																																	
1	100001	1	5	넘나 좋다는 것!																																																	
2	100001	3	0	[NULL]																																																	
3	100001	4	1	별로..																																																	
4	100001	7	3	그냥저냥 쓸만 했음																																																	

위치	오류유형	수정 전	수정 후
132~133p Level Up Test	문제-보기(지문)	<p>① Ⓐ : INNER JOIN 이벤트 응모 B ON A.회원번호 = B.회원번호 Ⓑ : A.이름, A.회원번호 Ⓒ : HAVING COUNT(B.응모순번) >= 10</p> <p>② Ⓐ : INNER JOIN 이벤트 응모 B ON A.회원번호 = B.회원번호 Ⓑ : A.이름, A.핸드폰 번호 Ⓒ : HAVING COUNT(B.응모순번) >= 10</p> <p>③ Ⓐ : INNER JOIN 이벤트 응모 B ON A.회원번호 = B.회원번호 Ⓑ : A.이름, A.핸드폰 번호 Ⓒ : HAVING SUM(B.응모순번) >= 10</p> <p>④ Ⓐ : LEFT OUTER JOIN 이벤트 응모 B ON A.회원번호 = B.회원번호 Ⓑ : A.이름, A.핸드폰 번호 Ⓒ : HAVING COUNT(B.응모순번) >= 10</p> <p>[해설] . .</p> <p>COUNT(응모순번) >= 10이라는 조건을 충족시켜야 한다. 그리고 SELECT 절에 A.이름, A.핸드폰 번호 컬럼이 출력되었으므로 GROUP BY 또한 같은 컬럼으로 정의되어야 한다.</p>	<p>① Ⓐ : INNER JOIN 이벤트 응모 B ON A.회원번호 = B.회원번호 Ⓑ : A.이름, A.회원번호 Ⓒ : HAVING COUNT(A.회원번호) >= 10</p> <p>② Ⓐ : INNER JOIN 이벤트 응모 B ON A.회원번호 = B.회원번호 Ⓑ : A.이름, A.핸드폰 번호 Ⓒ : HAVING COUNT(A.회원번호) >= 10</p> <p>③ Ⓐ : INNER JOIN 이벤트 응모 B ON A.회원번호 = B.회원번호 Ⓑ : A.이름, A.핸드폰 번호 Ⓒ : HAVING SUM(B.응모순번) >= 10</p> <p>④ Ⓐ : LEFT OUTER JOIN 이벤트 응모 B ON A.회원번호 = B.회원번호 Ⓑ : A.이름, A.핸드폰 번호 Ⓒ : HAVING COUNT(A.회원번호) >= 10</p> <p>[해설] . .</p> <p>COUNT(A.회원번호) >= 10이라는 조건을 충족시켜야 한다. 그리고 SELECT 절에 A.이름, A.핸드폰 번호 컬럼이 출력되었으므로 GROUP BY 또한 같은 컬럼으로 정의되어야 한다.</p>

위치	오류유형	수정 전	수정 후																																																
221p Level Up Test	문제-보기(지문)	수정 전 [결과] <table border="1"> <thead> <tr> <th>주문 연월</th> <th>주문상품</th> <th>주문수량</th> </tr> </thead> <tbody> <tr><td>202105</td><td>전체</td><td>50</td></tr> <tr><td>202106</td><td>전체</td><td>32</td></tr> <tr><td>202107</td><td>전체</td><td>43</td></tr> <tr><td>전체</td><td>모니터</td><td>25</td></tr> <tr><td>전체</td><td>마우스</td><td>30</td></tr> <tr><td>전체</td><td>노트북</td><td>20</td></tr> <tr><td>전체</td><td>키보드</td><td>50</td></tr> </tbody> </table>	주문 연월	주문상품	주문수량	202105	전체	50	202106	전체	32	202107	전체	43	전체	모니터	25	전체	마우스	30	전체	노트북	20	전체	키보드	50	[결과] <table border="1"> <thead> <tr> <th>주문 연월</th> <th>주문상품</th> <th>주문수량</th> </tr> </thead> <tbody> <tr><td>202105</td><td>전체</td><td>50</td></tr> <tr><td>202106</td><td>전체</td><td>32</td></tr> <tr><td>202107</td><td>전체</td><td>43</td></tr> <tr><td>전체</td><td>노트북</td><td>20</td></tr> <tr><td>전체</td><td>마우스</td><td>30</td></tr> <tr><td>전체</td><td>모니터</td><td>25</td></tr> <tr><td>전체</td><td>키보드</td><td>50</td></tr> </tbody> </table>	주문 연월	주문상품	주문수량	202105	전체	50	202106	전체	32	202107	전체	43	전체	노트북	20	전체	마우스	30	전체	모니터	25	전체	키보드	50
		주문 연월	주문상품	주문수량																																															
202105	전체	50																																																	
202106	전체	32																																																	
202107	전체	43																																																	
전체	모니터	25																																																	
전체	마우스	30																																																	
전체	노트북	20																																																	
전체	키보드	50																																																	
주문 연월	주문상품	주문수량																																																	
202105	전체	50																																																	
202106	전체	32																																																	
202107	전체	43																																																	
전체	노트북	20																																																	
전체	마우스	30																																																	
전체	모니터	25																																																	
전체	키보드	50																																																	
318p (3) DELETE	개념,공식-설명	(3) DELETE 이미 저장된 데이터를 삭제하고 싶을 때 사용하는 명령어이다. WHERE 절이 없으면 테이블의 모든 Row가 삭제되니 주의해야 한다. DELETE FROM 테이블명 (WHERE 수정할 데이터에 대한 조건);	(3) DELETE 이미 저장된 데이터를 삭제하고 싶을 때 사용하는 명령어이다. WHERE 절이 없으면 테이블의 모든 Row가 삭제되니 주의해야 한다. DELETE FROM 테이블명 (WHERE 삭제할 데이터에 대한 조건);																																																
357p 번호 : 25	해설	DDL(CREATE, ALTER, DROP, RENAME)의 경우 자동 커밋(Auto Commit)이 되기 때문에 ROLLBACK이 불가능하다.	DDL(CREATE, ALTER, DROP, RENAME)의 경우 자동 커밋(Auto Commit)이 되기 때문에 ROLLBACK이 불가능하다.																																																
377p 번호 : 27	해설	④ CASE 0 WHEN 0 THEN 1 ELSE NULL : 0 = 0이면 1 아니면 NULL ⇒ 1	④ CASE 0 WHEN 0 THEN 1 ELSE NULL : 0 = 0이면 1 아니면 NULL ⇒ 1																																																
385p 문제 보기 번호 : 45	문제-보기(지문)	[SQL] SELECT A.COL1 AS A1, A.COL2 AS A2, B. COL1 AS B1, B.COL2 AS B2 FROM SAMPLE1 A () SAMPLE2 B ON A.COL1 = B.COL2;	[SQL] SELECT A.COL1 AS A1, A.COL2 AS A2, B. COL1 AS B1, B.COL2 AS B2 FROM SAMPLE1 A () SAMPLE2 B ON A.COL1 = B.COL1;																																																

위치	오류유형	수정 전	수정 후																																																																		
395p 번호 : 31	문제-문항	<p>[해설]</p> <p>④ 연산자 실행 순서는 괄호, NOT, 비교연산자, AND, OR 순이다.</p>	<p>[해설]</p> <p>④ 연산자 실행 순서는 괄호, 비교연산자, NOT, AND, OR 순이다.</p>																																																																		
417p 번호 : 31	문제-보기(지문)	<p>③</p> <table border="1"> <thead> <tr> <th>EMP_ID</th> <th>NAME</th> <th>MGR_ID</th> </tr> </thead> <tbody> <tr> <td>1003</td> <td>나미</td> <td>NULL</td> </tr> <tr> <td>1005</td> <td>초파</td> <td>1003</td> </tr> <tr> <td>1006</td> <td>우습</td> <td>1003</td> </tr> <tr> <td>1000</td> <td>루피</td> <td>NULL</td> </tr> <tr> <td>1002</td> <td>상디</td> <td>1000</td> </tr> <tr> <td>1001</td> <td>조로</td> <td>1000</td> </tr> </tbody> </table> <p>④</p> <table border="1"> <thead> <tr> <th>EMP_ID</th> <th>NAME</th> <th>MGR_ID</th> </tr> </thead> <tbody> <tr> <td>1003</td> <td>나미</td> <td>NULL</td> </tr> <tr> <td>1005</td> <td>초파</td> <td>1003</td> </tr> <tr> <td>1006</td> <td>우습</td> <td>1003</td> </tr> </tbody> </table>	EMP_ID	NAME	MGR_ID	1003	나미	NULL	1005	초파	1003	1006	우습	1003	1000	루피	NULL	1002	상디	1000	1001	조로	1000	EMP_ID	NAME	MGR_ID	1003	나미	NULL	1005	초파	1003	1006	우습	1003	<p>③</p> <table border="1"> <thead> <tr> <th>EMP_ID</th> <th>NAME</th> <th>MGR_ID</th> </tr> </thead> <tbody> <tr> <td>1003</td> <td>나미</td> <td>NULL</td> </tr> <tr> <td>1005</td> <td>초파</td> <td>1003</td> </tr> <tr> <td>1004</td> <td>우습</td> <td>1003</td> </tr> <tr> <td>1000</td> <td>루피</td> <td>NULL</td> </tr> <tr> <td>1002</td> <td>상디</td> <td>1000</td> </tr> <tr> <td>1001</td> <td>조로</td> <td>1000</td> </tr> </tbody> </table> <p>④</p> <table border="1"> <thead> <tr> <th>EMP_ID</th> <th>NAME</th> <th>MGR_ID</th> </tr> </thead> <tbody> <tr> <td>1003</td> <td>나미</td> <td>NULL</td> </tr> <tr> <td>1005</td> <td>초파</td> <td>1003</td> </tr> <tr> <td>1004</td> <td>우습</td> <td>1003</td> </tr> </tbody> </table>	EMP_ID	NAME	MGR_ID	1003	나미	NULL	1005	초파	1003	1004	우습	1003	1000	루피	NULL	1002	상디	1000	1001	조로	1000	EMP_ID	NAME	MGR_ID	1003	나미	NULL	1005	초파	1003	1004	우습	1003
EMP_ID	NAME	MGR_ID																																																																			
1003	나미	NULL																																																																			
1005	초파	1003																																																																			
1006	우습	1003																																																																			
1000	루피	NULL																																																																			
1002	상디	1000																																																																			
1001	조로	1000																																																																			
EMP_ID	NAME	MGR_ID																																																																			
1003	나미	NULL																																																																			
1005	초파	1003																																																																			
1006	우습	1003																																																																			
EMP_ID	NAME	MGR_ID																																																																			
1003	나미	NULL																																																																			
1005	초파	1003																																																																			
1004	우습	1003																																																																			
1000	루피	NULL																																																																			
1002	상디	1000																																																																			
1001	조로	1000																																																																			
EMP_ID	NAME	MGR_ID																																																																			
1003	나미	NULL																																																																			
1005	초파	1003																																																																			
1004	우습	1003																																																																			
418p 번호 : 32	문제-문항	<p>① LAG(2)</p> <p>② LAG(2, 0)</p> <p>③ LEAD(2)</p> <p>④ LEAD(2, 0)</p>	<p>① LAG(COL2, 2)</p> <p>② LAG(COL2, 2, 0)</p> <p>③ LEAD(COL2, 2)</p> <p>④ LEAD(COL2, 2, 0)</p>																																																																		

위치	오류유형	수정 전		수정 후																					
426p 번호 : 8	문제-보기(지문)	수정 전	<div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid black; padding: 5px;"> <p>고객</p> <p>고객번호</p> <hr/> <p>가입일자</p> <p>고객등급</p> <p>핸드폰번호</p> </div> <div style="border: 1px solid black; padding: 5px;"> <p>계좌</p> <p>계좌번호</p> <hr/> <p>계좌명</p> <p>관리지점</p> <p>고객번호</p> <p>계좌개설일</p> <p>예수금</p> </div> </div>																						
		수정 후	<div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid black; padding: 5px;"> <p>강사</p> <p>강사코드</p> <hr/> <p>강사명</p> <p>강의과목</p> <p>핸드폰번호</p> <p>주소</p> </div> <div style="border: 1px solid black; padding: 5px;"> <p>강의일정</p> <p>강의코드</p> <p>강의일자</p> <hr/> <p>강의장</p> <p>강의시작일</p> <p>강의종료일</p> <p>강사코드(FK)</p> </div> </div>																						
433~434p 번호 : 25	문제-본문	<p>③</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>COL1</th> <th>COL2</th> </tr> </thead> <tbody> <tr> <td>AA</td> <td>NULL</td> </tr> <tr> <td>BB</td> <td>NULL</td> </tr> <tr> <td>CC</td> <td>10</td> </tr> <tr> <td>DD</td> <td>10</td> </tr> </tbody> </table> <p>[해설] DEFAULT를 설정하면 해당 컬럼에 NULL이 입력될 경우 NULL 값 대신 DEFAULT로 정의된 값이 자동으로 입력되며 ALTER 명령어로 DEFAULT 지정 시 이후에 들어오는 데이터부터 적용이 된다.</p>		COL1	COL2	AA	NULL	BB	NULL	CC	10	DD	10	<p>③</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>COL1</th> <th>COL2</th> </tr> </thead> <tbody> <tr> <td>AA</td> <td>NULL</td> </tr> <tr> <td>BB</td> <td>NULL</td> </tr> <tr> <td>CC</td> <td>NULL</td> </tr> <tr> <td>DD</td> <td>10</td> </tr> </tbody> </table> <p>[해설] DEFAULT를 설정하면 해당 컬럼이 입력되지 않을 경우 DEFAULT로 정의된 값이 자동으로 입력되고, 해당 컬럼에 NULL이 입력될 경우 NULL 값 대신 DEFAULT로 정의된 값이 자동으로 입력되며 ALTER 명령어로 DEFAULT 지정 시 이후에 들어오는 데이터부터 적용이 된다.</p>		COL1	COL2	AA	NULL	BB	NULL	CC	NULL	DD	10
COL1	COL2																								
AA	NULL																								
BB	NULL																								
CC	10																								
DD	10																								
COL1	COL2																								
AA	NULL																								
BB	NULL																								
CC	NULL																								
DD	10																								

위치	오류유형	수정 전	수정 후
434p 번호 : 26	문제-본문	<p>26. 다음 중 오류가 발생할 수 있는 SQL을 고르시오.</p> <p>[해설] 문자형 컬럼의 비교값으로 숫자형 데이터가 주어질 경우 묵시적 형변환이 일어나 에러가 발생하지 않는다. 하지만 반대로 숫자형 컬럼의 비교값으로 문자형 데이터가 주어질 경우 문자형 데이터가 숫자로만 이루어졌다면 마찬가지로 묵시적 형변환이 일어나 에러가 나지 않지만 텍스트로 이루어졌다면 에러가 발생하게 된다.</p> <p>[정답] ②</p>	<p>26. 다음 중 오류가 발생하지 않는 SQL을 고르시오. (Oracle이라고 가정)</p> <p>[해설] 묵시적 형변환은 데이터가 필요에 따라 한 데이터 유형에서 다른 유형으로 자동으로 변환되는 과정을 의미한다. Oracle의 경우 숫자형 컬럼의 비교값으로 문자형 데이터(숫자로 이루어진)가 주어질 경우 묵시적 형변환이 일어나 에러가 발생하지 않지만, 반대로 문자형 컬럼의 비교값으로 숫자형 데이터가 주어질 경우 에러가 발생하게 된다.</p> <p>[정답] ④</p>
436p 번호 : 31	문제-본문	<p>① FROM 절에 ALIAS를 사용하기 위해 AS 키워드를 사용할 수 있다.</p>	<p>① SQL Server의 경우 FROM 절에 ALIAS를 사용하기 위해 AS 키워드를 사용할 수 있다.</p>

도서의 오류로 학습에 불편드린 점 진심으로 사과드립니다.
더 나은 도서를 만들기 위해 노력하는 시대교육그룹이 되겠습니다.