어떤 문제인지 정확히 기억이 나지는 않지만 1:1은 아닐겁니다.  
정규화 관점에서는 1:1 관계인 엔터티들은 통합하는게 원칙입니다.  
1:1이라는 뜻은 식별자가 완전히 동일하고 속성만 다르다는 것으로 원래는 하나의 엔터티에 속해 있었다는 것을 의미하기 때문입니다.  
하나의 테이블을 다시 1:1 관계로 분할하는 시점은 정규화가 포함된 논리모델링을 마치고 물리모델링을 할 때 입니다.  
  
  
오늘 sqld시험 봤습니다...  
개인적으로 1과목이 너무 어렵고 헷갈려서 문제 복원을 해보고 과락이면 다시 준비하려고 합니다.  
10문제중 6개정도는 대충 기억나는데, 기억나시는분 좀 채워주세요...  
  
1. 속성의 명명  
2. 개념 스키마 설명  
3. 주식별자 도출 기준 (추가)  
4. 관계 체크사랑 가나다라  
5. 본식별자 인조식별자(추가)  
6. 엔터티 인스턴스 (추가)  
7. 관계 읽는 방법  
8.  
9. 비식별자관계가 안되는 것  
10. 2차 정규화 1:M  
  
  
오라클과 sql 서버 간의 쿼리 결과 차이를 묻는 문제가 3문제 정도 됐던걸로 기억합니다. Null과 '' 가지고 낸 문제들..  
  
  
식별자에대한내용이랑 인조식별자 나오지않았나요?  
  
  
객체지향에서는 1개인 경우가 있는데 보기?에서도 그럴수있다...는 뉘앙스로 나왔었어요  
  
  
과목1과 뒷편에 나오는 주관식부터 빨리 풀고 과목2를 푸는게 좋습니다.  
  
  
주관식 문제 중 sum 구하는 문제요.  
테이블 a 가 (아이디, 값)  
1 10  
2 10  
3 10  
4 10  
5 10 이고  
  
테이블 b 가 (아이디, 값)  
1 10  
2 10  
3 10  
4 10 이었던것 같습니다.  
  
조인 조건이 a.id >= b.id 이면 결과가..  
1 10  
2 10  
3 10  
4 10  
5 10  
  
2 10  
3 10  
4 10  
5 10  
  
3 10  
4 10  
5 10  
  
4 10  
5 10  
이렇게 되어서  
  
id별로 sum하면  
1 10  
2 20  
3 30  
4 40  
5 40  
이렇게 나오지 않나요?  
  
그래서 최종 sum한 값은 140 나오구요..  
  
db에 테이블 생성까지 해서 확인해 봤는데.. ;  
혹시 잘못된 부분 있으면 말씀주세요.  
  
  
문제가 b.id >= a.id로 알고있어요  
그래서 100이구요  
  
  
  
1. 구현, 동작 이라는 말이 들어간 문항  
2. 모델링 특징이었나? 데이터베이스 사전작업의 의미 관련된 문항이 틀렸다고 골랐습니다.  
3. UML ERD 관련 문제 - 잘 몰라서 3,4 골랐습니다.  
4. 관계차수 정답으로 골랐습니다.  
5. 1-2차 정규형 문항 골랐습니다.  
6. 1:1로 테이블 분리한다는 문항 골랐습니다.  
7. PK 순서없는 문항 보기 1번 정답으로 했습니다.  
8. 세개의 테이블 유지하고, 통합한다는 문항? 답으로 하였습니다.  
9. 2차 정규 , 관세번호 (pk 하나로 있던 문항) 골랐습니다.  
10. 거의 실시간 업무라는 항목 골랐습니다.  
  
11. hash join 관련 문제 - 대량 데이터 문항 골랏습니다  
12. Partition 넘을 수 없다 골랏습니다  
13. null관련 공백문자 혹은 0 들어간 문항 골랏습니다  
14. cnt 주문 이런 테이블이 보기에 있는 쿼리문제고 데이터 존재x 골랐습니다  
15. 널값 나타날 수 없다 라는 문항 골랐습니다.  
  
16. 상이한 뷰 정의하여 사용가눙  
17. truncate  
18. 실제 처리 건수  
19. any 들어간 보기  
23. 인덱스 엑세스 항상 유리 이런 문항이 틀렸다고 골랐습니다  
24. dml tcl 1.2번  
25. hash 큰 테이블 선행 ~~ 이런 보기 골랏음  
29. 유사개수칼럼은~ 상품분류코드별 -10000~+10000 이런 보기 골랐음  
31. grant auser to buser 이런 보기 골랏음  
32. nl조인 관련 문제 - Random 엑세스 위주 들어간 문항 골랐음  
  
34.varchar(30) not null 이런 보기 들어간 문항 골랏움  
38. 10시 10분  
40. intersect 랑 조인 한거랑 건수가 동일 하다라는 문항 골랐음  
  
hash nl 조인 이부분은 거의 틀린느낌이고 나머지고 엄청 틀렸네요 ㅠㅠ 틀린거 같은 문항 댓글 남겨주세요  
  
  
  
24번 언어랑 함수?가 알맞개 이어진거 고르는건데  
1. Dml - select  
2. Tcl - ???  
3. ???  
4. Dml - ???  
  
Select는 insert, update, delete와 함께 ddl문 아니었나요?  
  
  
그 문항 보기는 기억안나는데 1.2 번 정답으로 하였고,  
DML 이 select update insert delete 입니다  
  
  
  
PRIOR관리자=사원;  
LIKE 'S%', LIKE '\_L%'  
ADD , FOREIGN KEY  
RENAME TABLE TO AFTER\_TABLE;  
BETWEEN PROCEDING..( 이건 틀려서 잘 모르겠구요)  
또 머가있었지.. 기억이 안나네.. 시험본지 하루밖에 안됐는데..ㅋㅋ  
주관식은 한문제 빼고는 다 맞은듯 싶은데..  
[출처] 어제 SQLD 시험 잘 보셨나요?? (데이터베이스(DB) 전문가 포럼) |작성자 TopClass  
  
  
LIKE 'S%', LIKE '\_L%'  
PRIOR 관리자사번=사원사번  
AVG() 구하는 건 NULL을 제외한 값만 계산 대상입니다. 2번째는 NULL  
unbounded preceding (?)  
RENAME TABLE STADIUM TO STADIUM\_JPS;  
LAG(SAL,2,0)  
ADD CONSTRATINT, FOREIGN KEY  
나머지 한개는 기억이 안나네요? ^^;  
[출처] 어제 SQLD 시험 잘 보셨나요?? (데이터베이스(DB) 전문가 포럼) |작성자 TopClass  
  
  
  
1. prior 관리자 = 사원;  
2. left outer join  
3. 30,null,10  
4. unbounded preceding  
5. like 'S%' like '\_L%'  
6. lag(sal,2,0)  
7. rename STADIUM to STADIUM\_JPS  
8. constraint , foreign key  
[출처] 어제 SQLD 시험 잘 보셨나요?? (데이터베이스(DB) 전문가 포럼) |작성자 TopClass  
  
  
  
select nvl(ceil(0.1),2) from dual where '123'='    123' ;  
  
select ~~ from dual where ''=null ;  
  
  
  
1. CHAR VS VARCHR  
  CHAR에서는 물자열을 비교할 때 공백(BLANK)을 채워서 비교하는 방법을 사용한다. 공백 채우기  
  비교에서는 우선 짧은 쪽의 끝에 공백을 추가하여 2개의 데이터가 같은 길이가 되도록 한다.  
  그리고 앞에서부터 한 문자씩 비교한다. 그렇기 때문에 끝의 공백만 다른 문자열은 같다고 판단한다.  
  그에 반해 VARCHAR 유형에서는 맨 처음부터 한 문자씩 비교하고 공백도 하나의 문자로 취급하므로  
  끝의 공백이 다르면 다른 문자로 판단한다.  
  ex) CHAR 유형        'AA' = 'AA   '  
       VARCHAR 유형  'AA' <> 'AA   '  
  
2. CHECK 제약조건  
  CHECK(COL1 IN (1,2,3))  
  CHECK 제약조건이 걸린 컴럼이 NOT NULL인 경우 NULL 값이 들어갈 수 없지만 NOT NULL이  
  아니면 NULL 값 허용  
  
3. SELECT 문장을 통한 테이블 생성 사례  
  1) ORACLE  
    CTAS : CREATE TABLE ~ AS SELECT ~  
  2) SQL SERVER  
    SELECT \* INTO TABLE1FROM TABLE2  
  ※ 기존 테이블의 제약조건 중에 NOT NULL만 새로운 복제 테이블에 적용되고, 기본키, 고유키,  
     외래키 , CHECK 등의 다른 제약조건은 없어진다.  
  
4. RENAME COLUMN  
  1) ORACLE  
    ALTER TABLE XPDLQMFAUD  
    RENAME COLUMN 변경해야 할 컬럼명 TO 새로운 컬럼명;  
  2) SQL SERVER  
    SP\_RENAME 변경해야할 컬럼명, 새로운 컬럼명, 'COLUMN';  
  
5. RENAME TABLE  
  1) ORACLE  
    RENAME 변경 전 테이블 명 TO 변경 후 테이블명;  
  2) SQL SERVER  
    SP\_RENAME 변경 전 테이블명, 변경 후 테이블명;  
  
6. 연산자의 종류  
  1) 비교 연산자 : =, >, >=, <, <=  
  2) SQL 연산자 : BETWEEN A AND B, IN(LIST), LIKE '비교문자열', IS NULL  
  3) 논리 연산자 : AND, OR, NOT  
  
7. 연산자의 우선순위  
  괄호() -> NOT 연산자 -> 비교 연산자, SQL 비교연산자 -> AND -> OR  
  
8. ROWNUM  
  WHERE 절에서 행의 개수를 제한하는 목적으로 사용  
  한 행만 가져오고 싶을 때  WHERE ROWNUM = 1;  
                                      WHERE ROWNUM <= 1;  
                                      WHERE ROWNUM < 2;  
  두건 이상의 N행을 가져오고 싶을 때  WHERE ROWNUM = N; (X)  
                                                     WHERE ROWNUM <= N; (O)  
                                                     WHERE ROWNUM < N+1; (O)  
  
9. 단일행 함수의 종류  
  1) 문자형 함수 : LOWER, UPPER, SUBSTR / SUBSTRING, LENGTH / LEN, LTRIM, RTRIM,  
                        TRIM, ASCII  
  2) 숫자형 함수 : ABS, MOD, ROUND, TRUNC, SIGN, CHR / CHAR, CEIL / CEILING,  
                        FLOOR, EXP, LOG, LN, POWER, SIN, COS, TAN  
  3) NULL 관련 함수 : NVL / ISNULL, NULLIF, COALESCE  
    ※ IS NULL은 함수가 아니라 연산자이다.  
  
10. CEIL / CEILING() VS FLOOR()  
  1) CEIL / CEILING() : 숫자보다 크거나 같은 최소 정수를 리턴한다.  
    ex) CEIL(38.123) / CEILING(38.123)  ->  39  
          CEILING(-38.123)  ->  38  
  2) FLOOR() : 숫자보다 작거나 같은 최대 정수를 리턴한다.  
    ex) FLOOR(38.123)  ->  38  
         FLOOR(-38.123)  ->  -39  
  ※ CEILING은 천장, FLOOR은 바닥을 연상해서 외우면 좋을 듯 싶다.  
  
11. NVL / ISNULL 함수를 이용해 공집합을 9999로 바꾸기  
  SELECT NVL(MGR, 9999) MGR FROM EMP WHERE ENAME = 'JSC';  
  데이터를 찾을 수 없다.  
  -> NVL / ISNULL 함수는 NULL 값을 대상으로 다른 값으로 바꾸는 함수이지 공집합을  
     대상으로 하지 않는다.  
  SELECT NVL(MAX(MGR), 9999) MGR FROM EMP WHERE ENAME = 'JSC';  
  -> 집계함수를 인수로 한 NVL / ISNULL 함수를 이용해서 공집합인 경우에도 빈칸이 아닌 999로  
     출력하게 한다. 다른 함수와 달리 집계함수나 SCALAR SUBQUERY의 경우는 인수의 결과 값이  
     공집합인 경우에도 NULL을 출력한다.  
  
12. NULLIF 함수  
  NULLIF(EXPR1, EXPR2)  
  EXPR1이 EXPR2와 같으면 NULL, 다르면 EXPR1을 리턴  
  
13. COALESCE 함수  
  COALESCE(EXPR1, EXPR2, ...)  
  임의의 개수 EXPR에서 NULL이 아닌 최초의 EXPR을 나타낸다.  
  만약 모든 EXPR이 NULL이면 NULL을 리턴  
  
14. 집계함수  
  COUNT(\*) : NULL 값을 포함한 행의 수  
  COUNT(표현식) : 표현식의 값이 NULL 값인 거을 제외한 행의 수  
  SUM() : NULL 값을 제외한 합계  
  AVG() : NULL 값을 제외한 평균  
  ※ 조건절에 해당하는 데이터가 없을 때 COUNT(\*)의 결과 값은 0  
  
COL1  
COL2  
COL3  
COL4  
NULL  
NULL  
50  
30  
30  
20  
10  
30  
NULL  
10  
NULL  
NULL  
  
  
  
  
  
  
  
  SUM(COL1) + SUM(COL2 + COL3) + SUM(COL4)의 값은?  
  1.120    2.180    3.NULL    4.90  
  답은 제일 밑에 ㅋ  
  
15. GROUP BY 절  
  1) GROUP BY 절에서는 ALIAS명을 사용할 수 없다.  
  2) WHERE 절은 전체 데이터를 GROUP으로 나누기 전에 행들을 미리 제거시킨다.  
  3) HAVING 절은 일반적으로 GROUP BY 절 뒤에 위치한다.  
  4) GROUP BY, HAVING 절에는 SELECT 절에 정의되지 않은 컬럼은 사용 못함  
    ※ ORDER BY 절을 SELECT 절에 정의되지 않은 컬럼 사용 가능  
  
16. NULL의크기  
  ORACLE에서는 NULL 값을 가장 큰 값으로 간주  
  SQL SERVER에서는 NULL 값을 가장 작은 값으로 간주  
  
17. NATURAL JOIN  
  1) 두 테이블 간의 동일한 이름을 갖는 모든 칼럼들에 대해 EQUI(=) JOIN을 수행  
  2) NATURAL JOIN이 명시되면 추가로 USING 조건절, ON 조건절, WHERE 절에서 JOIN 조건을  
    정의할 수 없다.  
  3) JON에 사용된 컬럼들은 같은 데이터 유형이어야 하며, ALIAS나 텡블명과 같은 접두사를  
    붙일 수 없다.  
  
18. USING 조건절  
  FROM 절에 USING 조건절을 이용하면 같은 이름을 가진 컬럼들 중에서 원하는 컬럼에 대해서만  
  선택적으로 EQUI JOIN을 할 수 있다.  
  SELECT \* FROM DEPT JOIN DEPT\_TEMP  
  USING (DEPTNO);  
  ※ JOIN 칼럼에 대해서는 ALIAS나 테이블 이름과 같은 접두사를 붙일 수 없다.  
  
19. UNION VS UNION ALL  
  1) UNION : 여러개의 SQL의 결과에 대한 합집합으로 결과에서 모든 중복된 행을 하나의 행으로  
                 만든다.  
  2) UNION ALL : 중복된 행도 그대로 결과에 표시한다.  
  
20. 서브쿼리  
  1) 서브쿼리란 하나의 SQL문안에 포함되어 있는 또 다른 SQL문을 말한다.  
  2) 서브쿼리는 메인커리의 칼럼을 모두 사용할 수 있지만 메인쿼리는 서브쿼리의 칼럼을 사용할 수  
    없다. 질의 결과에 서브쿼리의 칼럼을 표시해야 한다면 조인방식으로 변환하거나 함수,  
    스칼라 서브쿼리 등을 사용해야 한다.  
  3) 단일행 서브쿼리  
    서브쿼리가 단일행 비교 연산자 (=, <, <=, >, >=, <>)와 함께 사용할 때는 서브쿼리의 결과건수가  
    반드시 1건 이하이어야 한다.  
  4) 다중행 서브쿼리  
    서브쿼리의 결과가 2건 이상 반환될 수 있다면 반드시 다중행 비교연산자 (IN, ALL, ANY, SOME)와  
    함께 사용해야 한다.  
    ex) IN (서브쿼리), ALL(서브쿼리), ANY(서브쿼리), EXISTS(서브쿼리)  
  5) 서브쿼리는 서브쿼리 레벨과 상관없이 항상 메인쿼리 레벨로 결과집합이 생성된다.  
    예를들어, 메인쿼리로 조직(1), 서브쿼리로 사원(M) 테이블을 사용하면 결과집합은  
    조직(1) 레벨이 된다.  
  6) 서브쿼리에서는 ORDER BY 절을 사용하지 못한다. ORDER BY 절은 SELECT 절에서 오직 한개만  
    올 수 있기 때문에 ORDER BY 절은 메인쿼리의 마지막 문장에 위치해야 한다.  
  
21. ORACLE의 INDEX 액세스 기법의 종류  
  INDEX UNIQU SCAN, INDEX RANGE SCAN, INDEX RANGE SCAN DESCNDING,  
  INDEX FULL SCAN, FAST FULL INDEX SCAN, INDEX SKIP SCAN  
  
22. 계층형 질의  
  1) START WITH 절은 계층구조 전개이 시작 위치를 지정하는 구문이다.  
  2) CONNECT BY 절은 다음에 전개될 자식 데이터를 지정하는 구문이다.  
  3) 루트 데이터는 LEVEL 1이다. (0이 아님)  
  
  
14번의 답은 1번 120  
  
  
  
SQLD 부분만 보면 책 반정도 될꺼에요.  
앞에 모델링 부분 좀 헷깔리게 나오던데 잘 봐두시면 좋을꺼같고  
null과 관련해서 sum(\*), sum(col1), sum(col1+col2), sum(col1) + sum(col2)  
요런것도 잘봐두면 도움되실꺼에요  
  
  
  
  
안녕하세요.  
설명회때 보니 SQLD, SQLP 등에 대한 정보가 많이 없어서 좀 도움이 될까해서...  
뭐, 도움이 안될지도.ㅋㅋ  
  
1. SQLD & SQLP 비교  
- 범위  
   + SQLD = 1과목 + 2과목  
   + SQLP = SQLD  + 3과목 + 실기 2문제  
  
=> 분량으로 보면 1/3 정도가 3과목이라고 할 수 있어서 그닥 범위가 많지 않다고 생각하기 쉬운데, 실제로  
1과목, 2과목은 책 내용이 다라고 보시면 되지만 3과목은 오라클 성능 고도화의 원리와 해법 1,2 두권이 요약  
된 형태로 오히려 분량 자체가 1,2과목을 넘어선다고 보면 될듯 하네요  
  
=> 각 과목 난이도를 보면,  
    .1과목 : 데이터모델링의 기초적인 이해로 한번도 관련 내용을 접하지 않더라도 책만 읽어도 어느정도 이해가  
                되는 난이도입니다. 물론 더 자세히 들어가면 DAP 에 준하는 만큼 어렵지만 SQLD/P 에서의 1과목은  
                책만 읽어도 무난할 듯 하네요.  
    
    .2과목 : SQL 에 대한 기본적인 이해와 설명에 대한 내용으로 한번도 SQL 을 작성하지 않았다면 처음에 익숙해지는  
               시간이 필요하지만 책에 있는 내용 정도와 예제를 직접 작성해 보고 실행해 본다면 무리없이 이해할수 있을듯.  
               2과목은 경험자와 비경험자의 이해 습득의 정도가 조금 차이가 있을 뿐 큰 차이는 없을 듯 하네요.  
  
    .3과목 : SQL 활용으로 실제 SQL 실행 원리부터 시작해서 궁극적으로 SQL 튜닝으로 이어지는 내용인데 이 과목  
               하나만 따로 공부해도 1,2과목 전체 공부하는 시간보다 더 많은 시간이 소요됩니다. 이해도 어려울 뿐 아니라  
               이론적인 내용을 이해해도 실제 SQL 튜닝을 진행하는 부분도 상당한 시간이 소요되는 과목이네요.  
               책의 요약적인 내용으로 겉을 훑어 가는 방식으로 진행한다면 못 따라갈 정도는 아니지만, 경험자와 비경험자  
               의 차이가 좀 날 것으로 보입니다.  
               1,2과목 두과목을 합친 이상의 난이도로 시간을 투자하면 못할 건 없지만, 이번 시험까지 시간적 여유가 있을  
               까 싶네요.  
  
=> 3과목 범위의 실제 난이도  
    . 3과목은 실제 SQL 튜닝에 대한 전반적인 내용으로 해당 내용만으로도 몇개월의 스터디를 할 정도입니다.  
    . SQLP 에서는 실제 그 정도의 내용이 나오지는 않으나(객관식에서 뜬금없이 좀 어려운게 나오기는 하지만.ㅋ) 전체  
      적인 내용을 이해하는 정도는 요구하는 듯 하네요.  
    . 이해정도라고 하면 1,2과목의 두배정도의 시간을 소요한다고 생각하면 될듯 합니다.  
  
  
- 먼저 SQLD 부터? 아니면 SQLP 로?  
  
 => 전적으로 개인적인 생각임을 전제로!  
 => SQL 을 처음 접하시는 분이 이번 SQLP 를 붙는다는건 요행이라고 보시면 될듯 합니다. 못 붙을 정도는 아니지만  
     (많은 시간을 투자하고 열심히 한다면) 붙는다고 해도 개인의 실력 향상은 그닥 없을 것으로 보입니다. 내용자체가  
     방대하기도 하지만 실기문제의 난이도가 매회 뒤죽박죽이라 한번의 시험으로 붙는 것보다는 두어번 정도 준비하고  
     자격증을 취득하는게 실제 개인 실력 향상에 가장 도움이 되지 않을까 합니다.  
  
 => 그럼 SQLD 를 먼저 하는게 나은지, SQLP 를 먼저 하는게 나을지를 생각해 보면...솔직히 별 의미가 없습니다.  
      SQLD 는 SQLP 일부분이라, SQLP 를 먼저 보냐, SQLD 를 먼저 보냐는 순전히 중간에 한단계 밟고 가느냐  
      아니냐의 차이정도라고 보시면 됩니다.  
  
 => 오히려 더 생각해야 할건 이번 시험까지 과연 SQLP 까지의 범위를 다 볼수 있냐, 없냐를 판단하시면 될듯 합니다.  
      SQL 을 처음 접하시는 분들은 이번 시험까지 SQLP 범위를 다 못 볼 확률이 높으니 SQLD 시험을 먼저 보는게  
      낫다라고 대부분 이야기를 하시는거 같네요.  
  
 => 만약에 이번 시험이 아니라 다다음 시험을 본다면 SQLP 를 바로 보시는게 낫습니다. 문제유형도 파악할겸 대충  
      자신의 위치를 알수도 있으니깐요.  
  
 => 대충 결론을 말하자면 SQL 첨 접하시는 분은 SQLP 를 목표로 열심히 하시구 진도가 SQLP 범위까지 못 갈거 같으면  
      SQLD 를 먼저 보는게 나을듯 하네요. 그리고 시험 결과에 상관없이 다음번 시험에 SQLP 를 보는게 좋을듯 합니다.  
  
  
2. 스터디 방식 이야기.  
- 스터디 방법  
  
=> 스터디를 몇번 해봤는데, 역시나 스터디 방법은 맴버들의 의견을 잘 조율해서 제일 좋을 듯 하네요~  
=> 자주 쓰는 방법 중 발표, 토의, 발표 + 토의, 기타 에 대해서 간단히 정리를 해봤습니다.  
  
1) 발표  
    - 발표자를 정하고 적절한 범위를 발표자가 정리한 내용을 발표하는 방식  
    - 장점 : 발표자가 내용을 숙지하기 좋고, 진행하기 좋음  
    - 단점 : 발표자 외 맴버들의 내용 숙지가 어려움. 약간의 지루함을 동반. 발표자 부재시 난감  
 2) 토의  
    - 특정 범위를 읽고와 다양한 방식으로 내용에 대한 토의를 함  
    - 발표자를 선정하고 발표를 하면서 토의를 하기도 함  
    - 장점 : 맴버들 전체가 참여를 하는 방식이 될 수 있음  
    - 단점 : 진도 빼기가 난감함. 산만해짐. 특정 맴버만 이야기를 하는 현상 발생 가능  
 3) 발표 + 토의 , 기타  
    - 위에 내용을 짬뽕하거나, 다른 방식을 쓴다거나 함.  
    - 기타  
       . 문제 작성 : 각 맴버가 지정된 범위내에서 문제를 만들어와 풀어보고 스터디 진행  
       . 책 읽기 : 책을 읽는 시간을 20~30분 정도 부여하고 스터디를 진행함  
       . 기타 등등  
  
  
3. 기타  
- SQLP 스터디  
=> SQL 을 쓰는 분들이라면 SQLP 스터디는 업무상에서 바로 스터디에 대한 효과를 누릴수 있을 듯 하네요.  
=> SQLP 스터디를 통해서 기본적인 SQL 튜닝에 대해 이해할수 있을듯 하지만 그렇다고 SQL 튜닝을 '잘'  
    한다고 보기는 힘들거 같네요. 대신 '잘' 할 수 있는 기반을 제공해 줄듯 합니다.  
=> SQLP 스터디는 한번에 그치지 말고 반복 학습이 큰 도움이 될듯.  
  
\*\*\* 쓰다보니 좀 길어졌네요. 도움이 되었기를..  
\*\*\* 주말만 오후가 생긴다면 매주는 아니더라도 매주 참여해서 같이 공부할수있을듯.^^  
\*\*\* 즐거운 하루 되세요.  
  
  
  
SAL        (       )  
----       ------  
100            0  
200            0  
300           100  
400           200  
500           300  
  
대충 이런 문제였던 걸로 기억합니다.  
빈칸에 들어갈 정답은 LAG(SAL,2,0) 입니다.  
  
  
  
문제는 기출문제중에 10%정도 그대로 나왔었고. 30%정도는 조금 응용 형태였는대 단순 사실형 보다는 이해를 기반으로 분석이 필요한 문제 위주로 나왔습니다. ( ex : 쿼리 실행 순서, 실행 결과 값, 조인과 쇼팅 특성 등 )  
  
  
  
주관식은 sql 위주로 나오고, 서술형이 아닌 단답형입니다.  
결과값이나, 전문용어, 함수 등이 나옵니다  
  
  
  
  
  
  
\*번호는 임의로 지정했습니다. 다 기억이 안나네요.\*  
  
\*틀린게 많을 수 있어요. 다른 의견 주시고.. '의견 많으면 9월 접수'합니다 ㅠㅠ\*  
벌써 -5부터시작합니다. (앞으로 15개 더틀리면 접수합니다.) 정보처리기사가 훨 쉽겠네요. SQLD는 2회도전중인데  ㅠㅠ  
  
파란색 오류 가능성 큰 답  
  
객관식  
  
1. 개념. 내부 사상 (물리적 메핑)  
14. "모든 ALIAS를 제가하면 에러가 난다"  
20. Domain  
23. 테이블의 속성은 살리면서, 데이터를 지우고자 한다  
 TRUNCATE table  
  
  
24. SQL문에 대한 내용으로 옳은 것은?  
FIRST\_VALUE, LAST\_VALUE는 ORACLE에서만 지원하는 함수이다.  
25. WINDOW함수에 대한 설명으로 옳은 것은?(옳지 않은건지? 보기가 기억이 안나네요)  
1)ROWS는 레코드 값에 대해, RANGE(물리적 레코드)에 대한 내용이다. (틀린거죠)  
2)  
37.  
측정코드  
측정날  
최대  
최소  
NULL  
2014.01  
120  
5  
NULL  
2014.02  
105  
5  
NULL  
?0  
?  
?  
결과가 위의 테이블과 같이 나오려면 사용해야할 SQL문은?  
1)GROUP BY GROUPING SETES (측정코드, 측정날)  
HAVING 측정코드 IS NULL  
2)GROUP BY ROLLUP (측정날, 측정코드)  
HAVING 측정코드 IS NULL  
등 ... 보기가 여러 가지 있었습니다. 저는 1,4번했습니다..... 틀렸겠죠?  
  
  
35. 수학에서의 합집합과 같은 집합연산자는?  
UNION  
  
36. TRANSACTION의 특징으로 옳지 않은 것은?  
PROCEDURE 은 반드시 한 트렉잭션내에서 수행되어야한다?  
  
37.  
TABLE A : 회원넘버, 회원이름 , 휴대폰 넘버  
TABLE B: 휴대폰 넘버 운영체제 넘버,  
TABLE : 운영체제 넘버, 운영체제이름  
여기서 A-B조인// 할 때A-B조인시 “WHERE 회원넘버 IN (회원 넘버 1001,002)”를 한 SQL문에 대한 출력 결과를 고르시오.  
저는 1번했는데요..  
38. 인덱스칼럼 분포문제 ->  
col1 - 0.55  
col2 - 0.33  
col3 - 0.005??  
select col1, col2, col3 form table where col1 = ... and col3 = ...  
검색향상을 위해 사용해야할 index 생성을 가장 올바르게 한 것은?  
저는 col3, col2,col1,로 하였습니다. 분포도가 적은 순으로 해야 scan을 덜 할 것 같아서요.  
  
  
40. Rank, Dense\_rank, rownum, NTILE  
  
  
-주관식-  
43.lag(sal, 2, 0)  
44.530,510(부장20제외), 5(group by)  
45.hello world  
46.3,5,5,5,9 ( natural, left outer. Right outer, full outer, cross)  
47.OR  
48.고립성  
49. 5  
50.Role  
  
  
38번 참조하는게 col1과 col3 이라서 col1과 col3만 나온걸로했네요 저는..  
그리구 문제 minus 관련해서 나온거있는데 그건 3번했구요 하루지나서 기억이안나네요  
9월 다시 봐야죠...  
  
  
  
46번 크로스 조인은... 조건절을 가지고 간다는 전제가 있기 때문에 9가 되지 않습니다.  
  
  
  
46번 주관식 전 1,3,5,3,9로 했는데..  
문제가 natural, left, full, right, cross순서  
  
  
p88  
첨부한 사진에서  
서울7호에서 매각된 총매각금액, 총유찰금액을 산출하는 조회용  
SQL문장을 왜 저렇게 작성한건지 이해가 가질 않습니다.  
SELECT 총매각금액, 총유찰금액  
FROM 매각일자별매각내역  
WHERE 매각장소 = '서울 7호';  
하면 되는 것 아닌가요?  
네 잘 지적하셨네요. 교재에 표기 오류가 있었던것 같습니다. 매각일자별매각내역 엔터티에 매각장소가 없다고 생각하면 이해하는데 도움이 될것 같네요. 또한 같은 날짜에는 다른 장소에서 매각되지 않아야 말이 되겠네요.  
현재 모델링의 경우엔 당연히 작성해주신 쿼리가 맞는것 같아보입니다  
  
  
  
  
주관식에 SQL SERVER 함수문제도 나옵니다.  
그리고 쿼리문 문제같은경우 SQL SERVER나 오라클의 구분없이 출제가능성은 모두 존재하는것 같으며,  
실제 출제비율은 대략 오라클 7, SQL SERVER 3쯤 되는것 같네요.  
마지막으로 SQL SERVER 나 오라클 둘중 하나만 묻는건 보았지만, 하나의 문제에 대해서 오라클버전, SQL SERVER버전 두 개의 예제를 내주고 동시에 다 묻는경우는 못봤네요.  
  
  
  
실제 SQLD와 SQLP는 동일 범위내에서 (튜닝을 제외한 부분)는 수준이 거의 유사합니다.  
DB가이드넷 퀴즈에서 SQLD 범위내의 문제만 보시면 될 것 같습니다.  
그래도 합격점 점수가 낮아서 당연 SQLD가 SQLP보다는 훨씬 쉽겠죠?  2번정도 정독하시면 쉽게 합격하실것 같은데..  
NULL 관련해서 명확히 동작원리를 이해하시고,  
조인의 문법을 정확히 알면 시험치실때 많이 도움이 되실 겁니다.  
함수는 책에 나오는것만 정확히 알고 계심 됩니다.  
(특히 ceil / round / trunc 처럼 비슷하지만 조금씩 다른 함수들은..정확히 알지 못하면 단답형으로 나오면 쥐약입니다.)  
도움이 되었나 모르겠네요.  
  
  
  
- 주식별자 도출 기준?  
- 다음 설명과 관계있는 것은? 답)개념적 스키마  
- 엔터티에 관한 틀린 설명은? 답) 싱글톤패턴처럼 엔터티도 인스턴스 1개만 있는 것이 가능하다  
- PK설정하는 다음 쿼리문중 문법이 맞는 것은?  
- 속성명을 정하는 방법 틀린것은?  
- 비식별자관계로 지정하는 기준?  
- 리두로그의 성질에 대해 묻는 문제?  
- View Merging이 일어나지 않는 쿼리문은? 답모름) ㄱ)rownum~ ㄴ)~connect\_by~ ㄷ)~count(\*)~ group by~ ㄹ) ~row\_number() ~  
- With Anchor문 쿼리문 적어놓고 쿼리에 대한 설명 틀린 것은?  
- 관계명을 읽는 방법으로 부적절한 것은? 답) 기준(source)엔터티를 한개 또는 하나 이상으로 읽는다  
- 관계가 있는지 알아보는 체크사항으로 맞는 것은? 답) ㄱ ㄴ ㄷ ㄹ  
- 바인드변수 사용관련 틀린 설명은? 답)  히스토그램뿐만아니라 각종통계를 이용못한다  
- DDL,DCL 종류가 다른 쿼리는? 답)GRANT ~  
- 쿼리실행시 런타임 오류가 날가능성 있는 것은? 답) where col1= (select a from~) (단일행인지 확실하지 않은데 '='을 쓴 경우  
- group by dname,rollup(job), copy테이블이용한 소계 등 SQL3개를 놓고 결과를 비교하는 문제  
- 다음 설명에 관한 기법은? 답) 낙관적기법  
- 사용자 정의 함수(?)에 대한 설명 틀린것? 답) 트리거에서 커밋할수 있다  
- (기본서 p88) 매각물건,일자별매각물건, 매각일자별매각내역 관련 정규화 문제? 답) 2차정규화 어쩌구 저쩌구  
- (기본서 p442) FROM절에 정의된후 쿼리속에서 절차성을 주는 효과...? 답) Inline view  
- (기본서 p499) User Call 설명 틀린 것은? 답) 전체범위처리때 사용하고 어쩌구 저쩌구  
- (기본서 p517) 실행계획 보여주고 BCHR,전체읽은수,물리적 읽은수 계산하기  
- (기본서 p542) 트랜잭션 특징 4가지중 틀린것은? 답) Isolation을 독립성(Independent)으로 바꾼것  
- (기본서 p582) 실행계획을 보여주고 VIEW PUSHED PREDICATE과 관련있는 것? 답) 조인조건 Pushdown  
- (기본서 p588) 힌트 no\_expand에 대한 것(OR Expansion 방지)  
- (기본서 p626) 클러스터링 팩터가 가장 좋은 것 고르기 ? 답) [AAA] [BBB]  
- [col1+col2]결합인덱스일때 order by시 인덱스만 이용하지 못하는 것 2개 고르기?  
  ㄱ)order by col1,col2  
  ㄴ)order by col1,col2 nulls last  
  ㄷ)order by col1 desc,col2  
  ㄹ)order by col1 desc,col2 nulls first  
   
- merge into 는 이번에 아예 출제 안됨  
  
  
\*기억나는 단답형 문제(총10문항) 유형은?  
  
 1) 쿼리문을 주고 예상 결과를 적는 문제  
  예) select ~ from ~ row\_number() over (partion by deptno) as rnum....group by deptno  
     답) 6 7 9 10  
  예)  
  Create table parent1 (  
  ...  
  ON DELETE SET NULL  
  )  
   
  Create table parent2 (  
  ...  
  ON DELETE CASCADE  
  )  
   
  Create table child (  
  ...  
  FK설정  
  )  
   
  delete from parent1 where ~;  
  delete from parent2 where ~;  
  대충 위처럼 설정해놓고 삭제후에 테이블에 남아있는 데이터 빈칸 채우기  
  
  
 2) 쿼리문의 예상 건수를 적는 문제  
  예) 쿼리문 (having sal > ALL (select ~ )) --> 답 0  
  쿼리문 카티션곱과 일반조인 --> 답 2700건, 90건  
   
 3) 쿼리문 중간을 비워놓고 채우는 문제  
  예) PRIOR 관리번호=사원번호, RIGHT OUTER JOIN  
  
 4) 기타 책에 나오는 내용 중 빈칸 채우기 문제  
  예) Seria/Repeat~Read/Read Unc~/Read Comi~, ,Segment/Extent  
  
[출처] SQLP]제19회(2015.12.05) SQLP 시험 후기|작성자 수제비 김창훈  
  
  
################  
정규화, 조인(NL, SMJ, Hash) 관련 문제는 꼭 나오므로 차이를 꼭 외워서 가야한다.  
  
1차, 2차, 3차 정규화의 차이과 조인간의 차이를 꼭 알아야 한다.  
  
  
  
Grouping Set, Cube, Rollup 관련 문제가 3문제 정도 나왔는데,  
  
이부분은 실제로 쿼리를 작성해보고 결과의 차이를 알아두지 않으면 풀기 어려웠다.  
  
이 함수들도 꼭 출력해서 차이를 알아두고 가자.  
  
  
  
테이블간의 조인에서 동등조인은 잘 아는데, between 연산이나, 대소관계(>, <)를 이용한 조인은 헷갈린다.  
  
이부분도 데이터가 있는 테이블에 날려서 결과를 확인해보고 가도록 하자.  
  
  
  
또, 그룹합수 MIN, MAX 등의 결과가 null 과 공집합이 나오는 차이도 알아두어야 한다.  
  
Group by 를 함께 사용할 때 결과가 달라지는 것도 알아두어야 한다.  
  
  
  
날짜 연산의 결과와 그룹함수에서 Grouping Set의 결과에 대한 내용도 알아두면 좋겠다.  
  
  
##############  
예를 들어  ROLLUP ,CUBE, GROUPING SET  
  
의 결과를 각각 알고 있어야 한다  
  
[출처] 2015 9/5일 SQLD 시험 후기|작성자 StargazerCH  
  
  
  
))))))))))))))))))))))))))))))))))))  
  
  
벤더별로의 데이터베이스 특징이라던가  
  
DB역사 , 조인순서그림등 무심코 지나쳤던 내용들이  
  
문제로 많이 나와서 엄청 헷갈렸어요~  
  
[출처] 14920 SQLD 시험 후기 /시험 유형/|작성자 kwon\_min\_ji  
  
  
  
  
지금 생각나는 문제를 한번 생각해보자면..  
  
개념/내부 사상  
식별자의 특징(유일성이나 뭐 최소 한 개이상의 속성이 있어야 한다 이런 내용이 보기로 있었어요)  
주 식별자의 특징  
윈도우 함수 특징  
도메인  
반정규화 방법? 순서?  
ERD 주고 1:M관계 어쩌구 하는 연습문제랑 비슷한 유형의 문제  
TRUNCATE TABLE  
수학적 합집합 UNION  
조인 쿼리 문제 INNER JOIN, OUTER JOIN  
집합 연산자  
서브쿼리 특징 (단일행 서브쿼리, 다중행 서브쿼리, 연관 서브쿼리, 비연관 서브쿼리)  
그룹함수 문제들 (여기서 많이 나온것 같아요)  
트리거 특징  
규칙기반 옵티마이저 (Single row by rowid)  
쿼리 결과보고 추가해야하는 인덱스 고르는 문제  
트랜잭션의 특징 일관성이 답인거같아요..  
  
  
  
  
SQL의 역사에 대한 문제 : SQL이 몇년에 처음 나오고 객체형 관계형은 어떤것이며 벤더에는 어떤것이 있고.. 하는 등의 문제가 나왔었다.  
SQL 조인, 인덱스 처리에 대한 흐름도 순서 : NL JOIN HASH JOIN MERGE JOIN 등의 처리 흐름도의 그림을 보고서  흐름 순서를 쓰는 문제가 있었다  다행히 맞긴 했지만 의외였던 문제에 추가 ~  
  
  
SELECT, PROJECT DIVIDE 등이 SQL 벤더에 어떤 절로 표현되었는지 물어보는 문제가 나왔었다. 슥 지나갔던 초입 소개부분이라 당황했었다.  
  
  
등등.. 기억이 안난다  
  
  
  
  
2. 대부분의 문제유형들  
  
  
2과목의 대부분의 문제 유형은 테이블과 SQL문을 보고 값이나 테이블 형태를 맞추는 것이었다.  
  
  
복잡한것이 많았으나, SQL문의 핵심 함수를 이해했거나, 인라인 뷰 등을 잘 파악할 수 있어야 풀 수 있는 문제였다.  
  
  
1과목에서는 1장 데이터모델링의 이해 의 개념이 헷갈리게 나와서 1번부터 헷갈렸었다.  
  
  
걱정했던 2장 데이터모델링 성능은 의외로 단순하게 나왔다.   
  
  
5장의 최적화 관리는 많이 나오진 않았으나 각 개념과 내용에 대해 이해하고 있어야 풀 수 있는 문제들이었다.  
  
  
그래서 헷갈렸다.  
  
  
  
  
1. 과목 데이터 모델의 이해 및 분석  
  
- 1과목은 10문제 밖에 아니라서 깊이 있게 보지 않았는데 생각보다 난해한 문제들이 많이 있었다  
  
슈퍼-서브타입모델, 분산데이터 적용방법 등등 문제가 있었는데 보기가 헷갈린 것들이 많이 있었다는...  달달 외우기 보다는 전체적으로 이해를 하고 있어야 할것같다.  
  
대충 보다가는 보기 둘중에 고민하다 찍게되는 상황이... ㅠㅠ  
  
10문제 밖에 아니지만 과락이 있으므로 1과목도 대충 보지 말것!  
  
  
  
2. SQL 이해 및 활용  
  
-  요 과목은 전체적으로 이론적인 내용보다는 SQL 쿼리문에 관련된 문제가 대다수였다  
  
문제 유형은 보기로 주어진 SELECT 문을 수행했을때 도출되는 결과나 수행 결과를 보여주고 SQL문을 완성시키는 것이다.  
  
수행된 결과 행이 몇개인지, 수행 결과 값이 무엇인지, 보기와 같은 결과를 도출하기 위해 SELECT 문 중간의 빈 괄호안에 어떤 구문을 넣어야 하는지에 대한 문제이다.  
  
이 과목을 공부하기 위해서는 일단 이론적인 내용을 이해하고 책의 예제을 보고 ?실제로 SQL문을 만들어보고 결과를 확인해보는 것이 중요하다.  
  
  
  
대충 기억나는 문제는  
  
- ROLLUP  
  
- ?RANK()/DENSE\_RANK()/ROW\_NUMBER()의 차이점에 관련 문제  
  
- ?집합연산자를 이용한 SQL문의 결과값(중복배제)  
  
- ?집합연산자를 이용한 SQL문을 JOIN을 사용한 SQL로 바꾸었을때 괄호채우기  
  
- ORDER BY  결과값 순서 관련 문제  
  
- ?JOIN 수행원리 (NL JOIN의 동작 순서)  
  
- ?유저 권한 관련문제  
  
- ?트랜잭션 특성  
  
- ?데이터베이스의 발전 등등등.