2015학년도 ( 2 )학기 과제물(온라인제출용)

**교과목명 : 데이터처리와활용**

**학 번 : 201435-341832**

**성 명 : 지 용 기**

**연 락 처 : 010-9828-0332, braveji@hanmail.net**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

o 과제유형 : ( 공통 ) 형

o 과 제 명 : 데이터처리와활용

**1장 연습문제 : 6번 기본키의 중요한 이유를 설명하고, 실생활 데이터에서 기본키로 이용할 수 있는 예제 데이터를 제시하라.**

|  |
| --- |
| Primary key란 행을 고유하게 구분해 주는 최소의 정보입니다. 모든 테이블에는 primary key가 있어야 하며, 오직 하나의 primary key만 존재할 수 있습니다. 그리고 그 하나의 primary key는 단일 컬럼으로 구성될 수도 있고 둘 이상의 다중의 컬럼들로 구성될 수도 있습니다.  만일 어떤 하나의 테이블에 primary key 역할을 할 수 있는 컬럼 또는 컬럼들의 그룹이 여러 개 있다면 그 컬럼 또는 컬럼들의 그룹을 candidate key라고 합니다. 하나의 테이블에 여러 개의 candidate key들이 존재할 수 있습니다.  가령 회원 테이블에 각각의 회원에게 고유하게 부여되는 (회원번호) 컬럼과 회원의 (주민등록번호) 컬럼, 그리고 회원이 회원가입 시에 선택하면서 회원별로 고유한 회원의 (로그인ID) 컬럼이 존재한다면 이 테이블에는 세 개의 candidate key가 존재하는 겁니다. 그렇지만 이 세 개의 candidate key 중에서 오직 하나의 candidate key 만이 primary key로 선택될 수 있습니다.  참고로 (회원번호) 컬럼을 primary key로 선택했다면 (회원번호) 컬럼은 회원 테이블의 primary key가 되는 것이고, primary key로 뽑히지 못한 (주민등록번호) 컬럼과 (로그인ID) 컬럼은 alternate key가 됩니다.  학교에서 학생 관리 데이터베이스를 만들 때, 학생의 테이블의 기본키는 **학번**이 될 수 있으며, 학번 대신에 **주민번호**도 후보가 될 수 있습니다. |

**2장 연습문제 : 7번 학교에서 성적정보를 데이터베이스를 구축하고자 한다. 성적과 관련있는 정보는 다음과 같다고 가장하자.**

|  |
| --- |
| * 학생에 관련된 정보 * 교과목에 관련된 정보 * 담당교수에 관련된 정보 * 취득학점 및 성적에 관한 정보 |

(1) 이와 같은 데이터를 위한 E-R 다임어그램을 설계하라.



(2) 데이터베이스 테이블을 설계하라.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 테이블 논리명 | 교수 정보 |  |  |  |  |
| 테이블 물리명 | TB\_PROF |  |  |  |  |
| 속성 | 속성명 | 타입 | 길이 | 제약사항 | 설명 |
| 교수번호 | PID | char | 8 | PK | 부임년도+순서번호로 구성됨 예) YYYY0000 |
| 이름 | PNAME | varchar | 20 | NOT NULL |  |
| 소속학과번호 | Pdept\_ID | varchar | 6 | NOT NULL |  |
| 소속학과명 | Pdept | varchar | 20 | NOT NULL |  |
| 부임년도 | Reg\_Time | char | 8 | NOT NULL | YYYYMMDD |
| 교수등급 | Grade | int | 4 | NOT NULL | 1 : 정교수, 2:부교수, 3:조교수 |
|  |  |  |  |  |  |
| 테이블 논리명 | 학생 정보 |  |  |  |  |
| 테이블 물리명 | TB\_STUDENT |  |  |  |  |
| 속성 | 속성명 | 타입 | 길이 | 제약사항 | 설명 |
| 학번 | SID | char | 13 | PK | 입학년도+소속학과+순서번호로 구성됨 예) YYYYDeptID000 |
| 이름 | Sname | varchar | 20 | NOT NULL |  |
| 소속학과번호 | Pdept\_ID | varchar | 6 | NOT NULL |  |
| 소속학과명 | Pdept | varchar | 20 | NOT NULL |  |
| 학년 | Grade | int |  | NOT NULL |  |
| 휴학여부 | Absence | char | 1 | NOT NULL | Y:휴학, N: 재학 |
|  |  |  |  |  |  |
| 테이블 논리명 | 교과목 정보 |  |  |  |  |
| 테이블 물리명 | TB\_CURRICULUM |  |  |  |  |
| 속성 | 속성명 | 타입 | 길이 | 제약사항 | 설명 |
| 교과목번호 | SubjectID | char | 6 | PK |  |
| 교과목명 | SubjectName | varchar2 | 40 | NOT NULL |  |
| 전공여부 | IS\_MAJOR | boolean |  | NOT NULL |  |
| 담당교수번호 | MAIN\_PID | char | 8 | NOT NULL |  |
|  |  |  |  |  |  |
| 테이블 논리명 | 성적 정보 |  |  |  |  |
| 테이블 물리명 | TB\_SCORE |  |  |  |  |
| 속성 | 속성명 | 타입 | 길이 | 제약사항 | 설명 |
| 학번 | SID | char | 13 | PK |  |
| 교과목번호 | SubjectID | char | 6 | PK |  |
| 교수번호 | PID | char | 8 | PK |  |
| 취득년도 | MAIN\_PID | char | 4 | PK | YYYY |
| 학점 | Score | int |  | NOT NULL |  |

**3장 연습문제 : 3번 다음 테이블을 참조하여 아래의 물음에 답하여라.**

(1) 위의 두 테이블을 조인하여 학번, 학생이름, 학년, 지도교수이름 필드의 데이터를 검색하기 위한 SQL문장을 작성하여라.

|  |
| --- |
| SELECT  S.SID AS 학번,  S.Name AS 학생이름,  TO\_NUMBER( TO\_CHAR( sysdate, 'YY' ) ) - TO\_NUMBER( SUBSTR(S.ID, 1,2) ) AS 학년,  P.Name As 지도교수이름  FROM Student S INNER JOIN Prof P ON S.PID = P.PID |

(2) 위의 두 테이블로부터 다음과 같은 결과가 나올 수 있는 SQL문장을 작성하여라.

|  |
| --- |
| SELECT  Name,  Dept  FROM Student  UNION  SELECT  Name,  Dept  FROM Prof |

**4장 연습문제 : 6번 다음 테이블을 참조하여 아래의 물음에 답하여라.**

(1) 위의 두 테이블에서 생일이 같은 학생들을 검색하기 위한 SQL문장을 작성하여라.

|  |
| --- |
| SELECT s1.학번, s2.학번  FROM Student1 s1, Student2 s2  WHERE  S1.생년월일 = (s2.출생연도 || “/” ||s2.출생월 || “/” || s2.출생일) |

(2) 두 테이블에서 학생들을 출생월별로 집계하고, 월별 출생학생들의 상대적인 비율을 추출하는 SQL문을 작성해 보아라.

|  |
| --- |
| SELECT 학번, 출생월,  PERCENT\_RANK() OVER( ORDER BY 출생월 ) 비율  FROM (  SELECT 학번, SUBSTR(생년월일, ,4, 5) AS 출생월  FROM Student1  UNION  SELECT 학번, 출생월  FROM Student2 ) AS AllStudent |

(3) 두 테이블에서 나이가 가장 많은 학생을 검색하는 SQL문자을 작성하여라.

|  |
| --- |
| SELECT A.\*  FROM (  SELECT 학번, 출생월,  ROW\_NUMBER() OVER( ORDER BY 출생월 DESC ) , RANK  FROM (  SELECT 학번, 생년월일  FROM Student1  UNION  SELECT 학번, ( s2.출생연도 || “/” ||s2.출생월 || “/” || s2.출생일 ) AS 생년월일  FROM Student2 ) AS AllStudent  ) A  WHER A.RANK = 1 |

(4) 두 테이블에서 학생들의 평균아이를 구하는 SQL문을 작성하여라.

|  |
| --- |
| SELECT AVG( 나이 )  FROM (  SELECT TO\_NUMBER( TO\_CHAR( sysdate, 'YY' ) ) - TO\_NUMBER( SUBSTR(생년월일, 1,2) ) AS 나이  FROM Student1  UNION  SELECT \_NUMBER( TO\_CHAR( sysdate, 'YY' ) ) - TO\_NUMBER(출생월) AS 나이  FROM Student2 ) AS AllStudent |

**5장 연습문제 : 3번 테이블 Customer\_Info와 Customer\_OnlineInfo에서 고객번호와 이름, 전화번호, 전자우편주소를 고객번호 순으로 엑셀 시트로 가지와 보아라.**