

```

1  --1. 다음을 수행한 후 결과를 확인하시오.
2  --1). 테이블을 다음차트를 기반으로 DEPARTMENT 테이블을 생성하고, 생성한 후 테이블 구조를 확인하시오.
3  --열이름 :          id          name
4  --데이터유형 :   NUMBER          VARCHAR2
5  --길이 :          7              25
6  CREATE TABLE department
7  (
8      id      NUMBER(7),
9      name    VARCHAR2(25)
10 );
11 --2). 테이블을 다음차트를 기반으로 EMPLOY 테이블을 생성하고, 생성한 후 테이블 구조를 확인하시오.
12 --열이름 :          id          last name      first name      dept id
13 --데이터유형 : NUMBER VARCHAR2    VARCHAR2    NUMBER
14 --길이 :          7              25              25              7
15 CREATE TABLE employ
16 (
17     id      NUMBER(7),
18     last_name VARCHAR2(25),
19     first_name VARCHAR2(25),
20     dept_id  NUMBER(7)
21 );
22
23 --3). DEPARTMENT 및 EMPLOY 테이블 모두 삭제하시오.
24 --DROP TABLE department;
25 --DROP TABLE employ;
26
27
28 --2. 다음을 수행한 후 결과를 확인하시오.
29 --1) dept 테이블에서 부서번호가 10번인 부서의 이름을 'Oracle'로 변경하시오.
30 --UPDATE dept
31 --SET dname = 'Oracle'
32 --WHERE deptno = 10;
33 --2) 90번 부서, 부서이름은 'Database', 위치는 'Seoul'인 레코드를 입력하시오.
34 --INSERT INTO dept
35 --VALUES(90, 'Database', 'Seoul');
36 --3) SAVEPOINT를 지정하시오.
37 --SAVEPOINT aaa;
38 --4) 부서번호가 90번인 레코드를 삭제하시오.
39 --DELETE FROM dept
40 --WHERE deptno = 90;
41 --5) 3)번에 지정된 SAVEPOINT로 ROLLBACK하시오.
42 --ROLLBACK TO aaa;
43 --6) 전체 ROLLBACK 하시오.
44 --ROLLBACK;
45 --7) dept 테이블을 조회하시오.
46
47 --3. DEPT table 및 EMP table의 모든 제약 조건을 확인하는 질의를 각각 작성하되, 제약조건이름, 제약조건타입,
48 상태를 나타내시오.
49 --SELECT constraint name, constraint type, status
50 --FROM user constraints
51 --WHERE table_name IN ('EMP', 'DEPT');
52
53 --4. emp 테이블에 있는 모든 CONSTRAINT를 조회하시오.
54 --SELECT constraint_name, constraint_type, status
55 --FROM user constraints
56 --WHERE table_name = 'EMP';
57
58 --5. emp 테이블에 sal, comm을 제외한 모든 column과 행을 포함하는 emp_demo 테이블을 생성하는 SQL문을 작성하시오.
59 --CREATE TABLE emp_demo
60 --(empno, ename, job, mgr, hiredate, deptno)
61 --AS
62 --SELECT empno, ename, job, mgr, hiredate, deptno
63 --FROM emp;
64
65 --6. emp 및 dept table을 이용하여 사번, 사원이름, 업무, 부서이름, 위치를 갖는 emp_dept 테이블을 생성하시오.
66 --CREATE TABLE emp_dept
67 --AS
68 --SELECT empno, ename, job, dname, loc
69 --FROM emp, dept
70 --WHERE emp.deptno = dept.deptno;
71
72 --7. emp 및 salgrade 테이블을 이용하여 사번, 사원이름, 업무, 급여, 보너스, 등급을 포함하는 emp_grade
73 테이블을 생성하시오.
74 --CREATE TABLE emp_grade
75 --(empno, ename, job, sal, comm, grade)
76 --AS
77 --SELECT empno, ename, job, sal, comm, grade
78 --FROM emp, salgrade
79 --WHERE sal BETWEEN losal AND hisal;
80
81 --8. dept 테이블에서 PRIMARY KEY를 비활성화하는 SQL문을 작성하시오.
82 --ALTER TABLE dept
83 --DISABLE CONSTRAINT dept deptno pk CASCADE;
84
85 --9. 위 문제 6에서 생성한 emp_dept 테이블에 empno를 PRIMARY KEY 로 설정하는 SQL문을 작성하시오.
86 --ALTER TABLE emp_dept
87 --ADD CONSTRAINT emp_dept empno pk PRIMARY KEY(empno);
88
89 --10. 위 문제 7에서 생성한 emp_grade 테이블에서 sal의 사이즈를 정수부문 12자리, 소수이하 4자리를 기억할수
90 있게 변경하는 SQL문을 작성하시오.
91 --ALTER TABLE emp_grade
92 --MODIFY (sal NUMBER(12,4));
93
94 --11. 위 문제 6에서 생성한 emp_dept 테이블의 내용과 기억 장소를 모두 해제하는 SQL문을 작성하시오.
95 --TRUNCATE TABLE emp_dept;
96

```

```

93
94 --12. 위 문제 7에서 생성한 emp grade 테이블을 삭제하는 SQL문을 작성하시오.
95 --DROP TABLE emp_grade;
96
97 --13. 아래의 구조를 만족하는 my_data 테이블을 생성하시오. 단, id가 PRIMARY KEY이다. 그리고 Data
Dictionary에서 확인하시오.
98 --      열이름      :      id      name      userid      salary
99 --데이터유형 : NUMBER      VARCHAR2      VARCHAR2      NUMBER
100 --길이 :      4      10      30      10,2
101
102 --CREATE TABLE my_data
103 -- (id      NUMBER(4)      CONSTRAINT my_data id pk PRIMARY KEY,
104 -- name      VARCHAR2(10),
105 -- userid      VARCHAR2(30),
106 -- salary      NUMBER(10,2)
107 -- );
108
109
110 --14. 위 문제 13에서 생성한 my_data테이블에 아래의 값을 입력하시오.
111 --      id      name      userid      salary
112 --      1      SCOTT      sscott      10,000.00
113 --      2      FORD      fford      13,000.00
114 --      3      PATEL      ppatel      33,000.00
115 --      4      REPORT      rreport      23,500.00
116 --      5      GOOD      ggood      44,450.00
117
118 --INSERT INTO my_data
119 --VALUES(1, 'SCOTT', 'ssott', 10000);
120 --INSERT INTO my_data
121 --VALUES(2, 'FORD', 'fford', 13000);
122 --INSERT INTO my_data
123 --VALUES(3, 'PATEL', 'ppatel', 33000);
124 --INSERT INTO my_data
125 --VALUES(4, 'REPORT', 'rreport', 23500);
126 --INSERT INTO my_data
127 --VALUES(5, 'GOOD', 'ggood', 44450);
128
129 --15. 위 문제에서 입력한 자료를 확인하시오.
130 --SELECT * FROM my_data;
131
132 --16. 위 문제 14에서 입력한 자료를 영구적으로 데이터베이스에 등록하시오.
133 --COMMIT;
134
135 --17. my_data테이블에서 id가 3번인 사람의 급여를 65,000.00으로 갱신하고 영구적으로 데이터베이스에 반영하시오.
136 --UPDATE my_data
137 --SET salary = 65000
138 --WHERE id=3;
139 --COMMIT;
140
141 --18. 이름이 FORD인 사원을 영구 제명하시오.
142 --DELETE FROM my_data
143 --WHERE name=UPPER('ford');
144 --COMMIT;
145
146 --19. 급여가 15,000 이하인 사람의 급여를 15,000으로 변경하시오.
147 --UPDATE my_data
148 --SET salary = 15000
149 --WHERE salary <=15000;
150
151 --20. my_data 테이블을 삭제하시오.
152 --DROP TABLE my_data
153
154
155 --21. BONUS table에 etc column을 추가하라. 단 자료형은 VARCHAR2(50)을 사용하라.
156 ALTER TABLE bonus
157 ADD
158 (
159      etc VARCHAR2(50)
160 );

```