

데이터베이스 기초

9장 과제1, 10장 실습과제1,2,4

<9장_과제1>

(1) emp와 dept 테이블로 부터 다음과 같은 조건을 만족하는 뷰를 생성하시오

```
CREATE VIEW empd
AS select empno, ename, hiredate, sal, job, dname, loc
   from emp, dept
   where emp.deptno = dept.deptno;
```

(2) 생성된 뷰를 이용하여 다음의 질의에 대한 sql 문을 작성하시오

① 모든 사원의 이름, 부서명을 보이시오

```
select ename, dname
from empd;
```

② 급여가 2500 보다 많은 사원의 직원번호, 이름, 급여, 부서위치를 보이시오

```
select empno, ename, sal, loc
from empd
where sal > 2500;
```

③ 부서위치가 BOSTON 인 사원의 이름과 입사일자를 보이시오
(Boston에 근무하는 사람이 없어서 New York으로 바꿈)

```
select ename, hiredate
from empd
where loc = 'NEW YORK';
```

④ SMITH 사원의 부서위치를 보이시오

```
select loc
from empd
where ename = 'SMITH';
```

(3) emp 테이블에서 BOSTON 에 근무하는 사원들만 추려서 emp_boston 뷰를 생성하시오. emp_boston의 내용을 보이시오.
(Boston에 근무하는 사람이 없어서 Chicago로 바꿈)

```
CREATE VIEW emp_chicago
AS select *
   from emp
  where ename in (select ename
                  from emp e, dept d
                 where e.deptno = d.deptno
                  and loc = 'Chicago');
```

(4) 다음과 같은 구조를 갖는 뷰 v_emp 를 emp 와 dept 테이블로 부터 생성하고 내용을 보이시오 (컬럼 이름을 표와 동일하게 해야 함)

```
CREATE VIEW v_emp
AS select dname as dept_name,
        count(*) as tot_emp,
        sum(sal) as tot_sal,
        avg(sal) as avg_sal
from emp e, dept d
where e.deptno = d.deptno
group by dname;
```

	dept_name	tot_emp	tot_sal	avg_sal
▶	RESEARCH	5	16962.50	3392.500000
	SALES	6	30340.00	5056.666667
	ACCOUNTING	3	9625.00	3208.333333

<10장_실습과제1>

1. 직원번호를 매개변수로 입력 받아 직원번호, 이름, 담당업무, 연봉, 소속 부서명을 보여주는 stored procedure 를 작성하시오 (up_emp_sel_1)

```
1 • CREATE PROCEDURE up_emp_sel_1(id int)
2 BEGIN
3     select empno, ename, job, sal, dname
4     from emp e, dept d
5     where e.deptno = d.deptno
6     and empno = id;
7 END
```

```
1 • call my_db.up_emp_sel_1(7369);
2
```

Result Grid					
Filter Rows: <input type="text"/>					
Export: 					
	empno	ename	job	sal	dname
▶	7369	SMITH	CLERK	5880.00	RESEARCH

2. 부서번호, 부서명, 위치를 매개변수로 입력 받아 새로운 부서 정보를 생성하는 stored procedure 를 작성하시오 (up_dept_insert_1)

```
up_dept_insert_1

1 • CREATE DEFINER=`root`@`localhost` PROCEDURE `up_dept_insert_1`(  
2     new_deptno decimal(2,0),  
3     new_dname varchar(14),  
4     new_loc varchar(13)  
5 )  
6 BEGIN  
7     insert into dept(deptno, dname, loc)  
8     values (new_deptno, new_dname, new_loc);  
9 END
```

- 1 • call my_db.up_dept_insert_1(50, 'MARKETING', 'KOREA');
- 2 • select * from dept;

Result Grid			
Filter Rows:			
	DEPTNO	DNAME	LOC
▶	10	ACCOUNTING	NEW YORK
	20	RESEARCH	DALLAS
	30	SALES	CHICAGO
	40	OPERATIONS	BOSTON
	50	MARKETING	KOREA
*	HULL	HULL	HULL

```
DELIMITER //
```

```
CREATE PROCEDURE up_emp_update_1(  
    eno int,  
    enm varchar(20))  
  
BEGIN  
  
    update emp  
    set  ename=enm  
    where empno=eno;  
  
END  
  
//  
  
DELIMITER ;
```

[illegible]

<10장_실습과제2>

1. 사원번호를 매개변수로 입력 받아 사원의 담당업무가 'ENGINEER' 이 면 급여를 20% 올리고, 아닌 경우는 10%를 올리는 stored procedure 를 작성하시오
(up_emp_update_2)

```
DELIMITER //
CREATE PROCEDURE up_emp_update_2(eno int)
BEGIN
    DECLARE tmp_job varchar(10) ; # 변수선언
    select job into tmp_job
    from emp
    where empno=eno;

    IF (tmp_job="engineer") THEN
        update emp
        set sal = sal*1.2
        where empno = eno ;
    ELSE
        update emp
        set sal = sal*1.1
        where empno = eno ;
    END IF ;
END
//
DELIMITER ;
```

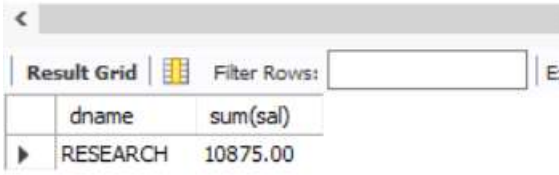

- ```
1 • call my_db.up_emp_update_2(7369);
2 • select * from emp where empno = 7369;
```

[illegible]

2. 사원번호를 매개변수로 입력 받은 후에 그 사원이 속한 부서 사람들의 연봉 합계를 구하여 출력하는 stored procedure 를 작성하시오 (p\_emp\_sel\_2)

```
DELIMITER //
CREATE PROCEDURE p_emp_sel_2(eno int)
BEGIN
 select sum(sal)
 from emp
 where deptno=(select deptno from emp where empno=eno);
END
//
```

```
1 • call my_db.p_emp_sel_2(7369);
2
```



The screenshot shows a SQL IDE interface. At the top, a SQL statement is entered: `call my_db.p_emp_sel_2(7369);`. Below the editor, there is a 'Result Grid' tab. The grid has two columns: 'dname' and 'sum(sal)'. There is one row of data: 'RESEARCH' with a value of '10875.00'.

| dname    | sum(sal) |
|----------|----------|
| RESEARCH | 10875.00 |

3. 사원번호를 매개변수로 입력 받은 후에 사원의 급여가 평균급여 이상 이면 해당 사원의 근무지를 출력 하고, 그렇지 않으면 사원의 직무를 출력하는 stored procedure 를 작성 하시오. (p\_emp\_sel\_3)

```
DELIMITER //
CREATE PROCEDURE p_emp_sel_3(eno int)
BEGIN
 declare avg_sal int;
 declare emp_sal int;
 select avg(sal) into avg_sal
 from emp;
 select sal into emp_sal
 from emp
 where empno=eno;

 IF (emp_sal>=avg_sal) THEN
 select loc
 from emp e, dept d
 where e.deptno=d.deptno and empno=eno;
 ELSE
 select job
 from emp
 where empno=eno;
 END IF ;
END
//
DELIMITER ;
```

## <10장\_실습과제2>

(1) 사원이름을 입력하면 사원의 매니저 이름을 리턴하는 함수를 작성하 시오 (f\_mgr)  
다음과 같이 함수를 테스트한다.

```
select empno, ename, f_mgr(ename) as manager
from emp
```

```
f_mgr

1 • CREATE DEFINER='root'@'localhost' FUNCTION `f_mgr` (
2 value varchar(10)
3) RETURNS varchar(10) CHARSET utf8mb4
4 BEGIN
5 declare mgrname varchar(10);
6 set mgrname = (select m.ename
7 from emp e, emp m
8 where e.mgr = m.empno
9 and e.ename = value);
10 RETURN mgrname;
11 END
```

```
1 • select empno, ename, f_mgr(ename) as manager
2 from emp;
```

| Result Grid |        |         |  |
|-------------|--------|---------|--|
| empno       | ename  | manager |  |
| 7369        | SMITH  | FORD    |  |
| 7499        | ALLEN  | BLAKE   |  |
| 7521        | WARD   | BLAKE   |  |
| 7566        | JONES  | KING    |  |
| 7654        | MARTIN | BLAKE   |  |
| 7698        | BLAKE  | KING    |  |
| 7782        | CLARK  | KING    |  |
| 7788        | SCOTT  | JONES   |  |
| 7839        | KING   | NOTE    |  |
| 7844        | TURNER | BLAKE   |  |
| 7876        | ADAMS  | SCOTT   |  |
| 7900        | JAMES  | BLAKE   |  |
| 7902        | FORD   | JONES   |  |
| 7934        | MILLER | CLARK   |  |

(2) 부서번호를 입력하면 부서의 위치를 출력하는 함수를 작성하시오 (f\_emp2) 다음과 같이 함수를 테스트한다.

```
select empno, ename, job, f_emp2(deptno) as loc
from emp
```

```
f_emp2

1 • CREATE DEFINER=`root`@`localhost` FUNCTION `f_emp2`(
2 value decimal(2,0)
3) RETURNS varchar(13) CHARSET utf8mb4
4 BEGIN
5 declare deptloc varchar(13);
6 set deptloc = (select loc
7 from dept
8 where deptno = value);
9 RETURN deptloc;
10 END

1 • select my_db.f_emp2(30);
2
```

|                  |  |
|------------------|--|
| <                |  |
| Result Grid      |  |
| Filter Rows:     |  |
| my_db.f_emp2(30) |  |
| CHICAGO          |  |