

[문제 1] 사원정보(EMPLOYEES) 테이블에서 사원의 성과 이름은 Name, 업무는 Job, 급여는 Salary, 연봉에 \$100 보너스를 추가하여 계산한 값은 Increased Ann_Salary, 급여에 \$100 보너스를 추가하여 계산한 연봉은 "Increased Salary"라는 별칭으로 출력하시오

[문제 2] 사원정보(EMPLOYEE) 테이블에서 모든 사원의 이름(last_name)과 연봉을 "이름: 1 Year Salary = \$연봉" 형식으로 출력하고, "1 Year Salary"라는 별칭을 붙여 출력하시오

[문제 3] HR 부서에서는 급여(salary)와 수당율(commission_pct)에 대한 지출 보고서를 작성하려고 한다. 이에 수당을 받는 모든 사원의 성과 이름(Name으로 별칭), 급여, 업무, 수당율을 출력하시오. 이때 급여가 큰 순서대로 정렬하되, 급여가 같으면 수당율이 큰 순서대로 정렬하시오.

[문제 4] 이번 분기에 60번 IT 부서에서는 신규 프로그램을 개발하고 보급하여 회사에 공헌하였다. 이에 해당 부서의 사원 급여를 12.3% 인상하기로 하였다. 60번 IT 부서 사원의 급여를 12.3% 인상하여 정수만(반올림) 표시하는 보고서를 작성하시오. 출력 형식은 사번, 이름과 성 (Name으로 별칭), 급여, 인상된 급여(Increased Salary로 별칭)순으로 출력한다.

[문제 5] 각 이름이 's'로 끝나는 직원들의 이름과 업무를 아래의 예와 같이 출력하고자 한다. 출력 시 성과 이름은 첫 글자가 대문자, 업무는 모두 대문자로 출력하고 머리글은 "Employee JOBs."로 표시하시오.

예) Sigal Tobias is a PU_CLERK

[문제 6] 모든 사원의 연봉을 표시하는 보고서를 작성하려고 한다. 보고서에 사원의 성과 이름 (Name으로 별칭), 급여, 수당여부에 따른 연봉을 포함하여 출력하시오. 수당여부는 수당이 있으면 "Salary + Commission", 수당이 없으면 "Salary only"라고 표시하고, 별칭은 적절히 붙인다. 또한 출

력 시 연봉이 높은 순으로 정렬한다.

<Hint> IFNULL() 함수, IF() 함수를 이용하여 NULL이었는지 아니었는지를 식별할 수 있다.

[문제 7] 각 사원이 소속된 부서별로 급여 합계, 급여 평균, 급여 최대값, 급여 최소값을 집계하고자 한다. 계산된 출력값은 6자리와 세 자리 구분기호, \$ 표시와 함께 출력하고 부서번호의 오름차순 정렬하시오. 단, 부서에 소속되지 않은 사원에 대한 정보는 제외하고 출력시 머리글은 아래 예시처럼 별칭(alias) 처리하시오.

[문제 8] 직원들의 업무별 전체 급여 평균이 \$10,000보다 큰 경우를 조회하여 업무, 급여 평균을 출력하시오. 단 업무에 사원(CLERK)이 포함된 경우는 제외하고 전체 급여 평균이 높은 순서대로 출력하시오.

[문제 9] hr 스키마에 존재하는 Employees, Departments, Locations 테이블의 구조를 파악한 후 Oxford에 근무하는 사원의 성과 이름(Name으로 별칭), 업무, 부서명, 도시명을 출력하시오. 이때 첫 번째 열은 회사명인 'Han-Bit'이라는 상수값이 출력되도록 하시오.

[문제 10] HR 스키마에 있는 Employees, Departments 테이블의 구조를 파악한 후 직원수가 5명 이상인 부서의 부서명과 직원수를 출력하시오. 이때 직원수가 많은 순으로 정렬하시오.

[문제 11] 각 사원의 급여에 따른 급여 등급을 보고하려고 한다. 급여 등급은 JOB_GRADES 테이블에 표시된다. 해당 테이블의 구조를 살펴본 후 사원의 성과 이름(Name으로 별칭), 업무, 부서명, 입사일, 급여, 급여등급을 출력하시오.

[문제 12] HR 부서의 어떤 사원은 급여정보를 조회하는 업무를 맡고 있다. Tucker 직원(last_name)보다 급여를 많이 받고 있는 사원의 성과 이름(Name으로 별칭), 업무, 급여를 출력하시오.

[문제 13] 사원의 급여 정보 중 업무별 최소 급여를 받고 있는 사원의 성과 이름(Name으로 별칭), 업무, 급여, 입사일을 출력하시오.

[문제 14] 소속 부서의 평균 급여보다 많은 급여를 받는 사원에 대하여 사원의 성과 이름(Name으로 별칭), 급여, 부서번호, 업무를 출력하시오.

[문제 15] 직원들의 지역별 근무 현황을 조회하고자 한다. 도시 이름이 영문 'O'로 시작하는 지역에 살고 있는 사원의 사번, 이름, 업무, 입사일을 출력하시오.

[문제 16] 모든 사원의 소속부서 평균연봉을 계산하여 직원별로 성과 이름(Name으로 별칭), 업무, 급여, 부서번호, 부서 평균연봉(Department Avg Salary로 별칭)을 출력하시오.

[문제 17] 직원정보(Employees) 테이블에 JOB_ID는 사원의 현재 업무를 뜻하고, JOB_HISTORY에 JOB_ID는 사원의 이전 업무를 뜻한다. 이 두 테이블을 교차해보면 업무가 변경된 사원의 정보도 볼 수 있지만 이전에 한번 했던 같은 업무를 그대로 하고 있는 사원의 정보도 볼 수 있다. 이 전에 한번 했던 같은 업무를 보고 있는 사원의 사번과 업무를 출력하시오.

[문제 18] 위 결과를 이용하여 출력된 176번 사원의 업무 이력의 변경 날짜 이력을 조회하시오.

[샘플 문제] HR 부서에서는 신규 프로젝트의 성공으로 해당하는 각 업무 자들에 대한 급여 인상을 결정하고, 다음과 같이 업무 별 급여 인상에 대해 적용하고자 한다. 현재 107명의 사원은 19개의 업무에 소속되어 근무 중이다. (Distinct job_id) 이 중 5개의 업무 자들에 대한 급여 인상이 각각 결정되었고, 나머지 업무에 대해서는 인상이 동결되었다. HR_REP(10%), MK_REP(12%), PR_REP(15%), SA_REP(18%), IT_PROG(20%)

<Hint> CASE와 DECODE를 이용하여 위 조건을 만족하는 구문을 작성해 본다.

```
SELECT employee_id, CONCAT(last_name, ' ', first_name) as "Name", job_id, salary,
       CASE job_id WHEN 'HR_REP' THEN 1.10 * salary
                   WHEN 'MK_REP' THEN 1.12 * salary
                   WHEN 'PR_REP' THEN 1.15 * salary
                   WHEN 'SA_REP' THEN 1.18 * salary
                   WHEN 'IT_PROG' THEN 1.20 * salary
       ELSE salary
       END "New Salary"
FROM Employees;
```

[샘플 문제] HR 부서에서는 최상위 입사일에 해당하는 2001년부터 2003년까지 입사자들의 급여를 각각 5%, 3%, 1% 인상하여 지분에 따른 배당금으로 지급하고자 한다. 전체 사원들 중 해당 년도에 해당하는 사원들의 급여를 검색하여 적용하고, 입사일자에 따른 오름차순 정렬을 수행하시오.

<Hint> CASE 구문을 이용한 검색 조건 비교를 이용하여 해당 년도 별 조건 처리를 수행 할 수 있다.

```
SELECT employee_id, CONCAT(last_name, ' ', first_name) as "Name", hire_date, salary,
       CASE WHEN hire_date < STR_TO_DATE('1998-01-01', '%Y-%m-%d') THEN salary * 1.05
            WHEN hire_date < STR_TO_DATE('1999-01-01', '%Y-%m-%d') THEN salary * 1.03
            WHEN hire_date < STR_TO_DATE('2000-01-01', '%Y-%m-%d') THEN salary * 1.01
       ELSE salary
       END "New Salary"
FROM Employees
ORDER BY hire_date;
```

[문제 19] 부서별 급여 합계를 구하고, 그 결과를 가지고 다음과 같이 표현하시오.

Sum Salary > 100000 이면, "Excellent"

Sum Salary > 50000 이면, "Good"

Sum Salary > 10000 이면, "Medium"

Sum Salary <= 10000 이면, "Well"

<Hint> case 문을 사용하는 보통 방법을 사용할 수도 있고, inline view를 이용하여 우선 부서별 급여 합계를 구하고, 상위 쿼리에서 CASE 구문을 이용하여 위의 조건 비교를 통해 급여 합계에 따른 표현을 할 수 있다.

[문제 20] 2005년 이전에 입사한 사원 중 업무에 "MGR"이 포함된 사원은 15% 급여를 인상하고, "MAN"이 포함된 사원은 20% 급여 인상이 정해졌고, 또한 2005년부터 근무를 시작한 사원 중 "MGR"이 포함된 사원은 25% 급여 인상을 수행하는 쿼리를 작성하시오. 해당되는 사원들만 출력한다.