

2019-1 DB 과제#2 - 20142697 권민수

- Q1 : 이름이 John B. Smith인 사원의 생년월일과 주소를 검색하시오

```
SELECT Bdate, address  
FROM employee  
WHERE CONCAT(Fname, Minit, Lname) = 'JohnBSmith';
```

| Result Grid | | Filter Rows: | Export: | Wrap Cell Content: |
|-------------|------------|--------------------------|---------|--------------------|
| | Bdate | address | | |
| ▶ | 1965-01-09 | 731 Fondren, Houston, TX | | |

- Q2 : 부서번호가 5인 부서에 근무하는 모든 사원의 모든 컬럼을 검색하시오

```
SELECT *  
FROM employee  
WHERE Dno = 5;
```

| | Ssn | Fname | Minit | Lname | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|-----------|----------|-------|---------|------------|--------------------------|------|----------|-----------|------|
| ▶ | 123456789 | John | B | Smith | 1965-01-09 | 731 Fondren, Houston, TX | M | 30000.00 | 333445555 | 5 |
| | 333445555 | Franklin | T | Wong | 1955-12-08 | 638 Voss, Houston, TX | M | 40000.00 | 888665555 | 5 |
| | 453453453 | Joyce | A | English | 1972-07-31 | 5631 Rice, Houston, TX | F | 25000.00 | 987654321 | 5 |
| * | 666884444 | Ramesh | K | Narayan | 1962-09-15 | 975 Fire Oak, Humble, TX | M | 38000.00 | 333445555 | 5 |
| * | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

- Q3 : 'Research' 부서에 근무하는 모든 사원의 이름, 그리고 주소를 검색하시오

```
SELECT e.Fname, e.Minit, e.Lname, e.address  
FROM employee e JOIN department d ON e.Dno = d.Dnumber  
WHERE d.Dname = 'Research';
```

| Result Grid | | | | | Filter Rows: | Export: | Wrap Cell Content: |
|-------------|----------|-------|---------|--------------------------|--------------|---------|--------------------|
| | Fname | Minit | Lname | address | | | |
| ▶ | John | B | Smith | 731 Fondren, Houston, TX | | | |
| | Franklin | T | Wong | 638 Voss, Houston, TX | | | |
| | Joyce | A | English | 5631 Rice, Houston, TX | | | |
| | Ramesh | K | Narayan | 975 Fire Oak, Humble, TX | | | |

- Q4 : 'Stafford'에 위치한 모든 프로젝트에 대해서, 프로젝트 번호, 담당부서 번호, 부서관리자의 이름을 검색하시오

```
SELECT p.Pnumber, p.Dnum, e.Fname, e.Minit, e.Lname  
FROM project p JOIN department d ON d.Dnumber = p.Dnum  
JOIN employee e ON d.Mgr_ssn = e.Ssn  
WHERE p.Plocation = 'Stafford';
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

| | Pnumber | Dnum | Fname | Minit | Lname |
|---|---------|------|----------|-------|---------|
| ▶ | 10 | 4 | Jennifer | S | Wallace |
| | 30 | 4 | Jennifer | S | Wallace |

- Q5 : 각 사원에 대해 사원의 이름과 성별, 그리고 직속 상사의 이름과 성별을 검색하시오. 단, 직속 상사가 없는 직원도 검색하시오. 출력 컬럼은 사원의 이름과 성별, 그리고 직속 상사의 이름과 성별 순으로 하며, 테이블 데이터는 사원 이름 중 Fname의 오름차순으로 나타내시오

```
SELECT e.Fname, e.Minit, e.Lname, e.sex, s.Fname, s.Minit, s.Lname, s.sex
FROM employee e LEFT JOIN employee s ON e.Super_ssn = s.Ssn
ORDER BY e.Fname;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

| | Fname | Minit | Lname | sex | Fname | Minit | Lname | sex |
|---|----------|-------|---------|-----|----------|-------|---------|------|
| ▶ | Ahmad | V | Jabbar | M | Jennifer | S | Wallace | F |
| | Alicia | J | Zelaya | F | Jennifer | S | Wallace | F |
| | Franklin | T | Wong | M | James | E | Borg | M |
| | James | E | Borg | M | NULL | NULL | NULL | NULL |
| | Jennifer | S | Wallace | F | James | E | Borg | M |
| | John | B | Smith | M | Franklin | T | Wong | M |
| | Joyce | A | English | F | Jennifer | S | Wallace | F |
| | Ramesh | K | Narayan | M | Franklin | T | Wong | M |

- Q6 : 사원 'Franklin Wong'이 직접 관리하는 사원의 이름을 검색하시오. 테이블 데이터는 사원 이름 중 Fname의 오름차순으로 나타내시오

```
SELECT e.Fname, e.Minit, e.Lname
FROM employee e JOIN employee s ON e.Super_ssn = s.Ssn
WHERE CONCAT(s.Fname, " ", s.Lname) = 'Franklin Wong'
ORDER BY e.Fname;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

| | Fname | Minit | Lname |
|---|--------|-------|---------|
| ▶ | John | B | Smith |
| | Ramesh | K | Narayan |

- Q7 : 사원의 SSN과 부서의 Dname에 대한 모든 조합을 생성하시오. 출력 컬럼은 Ssn, Dname 순으로 나열하며, 테이블 데이터는 Ssn과 Dname의 오름차순으로 출력하시오

```
SELECT e.Ssn, d.Dname
FROM employee e CROSS JOIN department d
ORDER BY e.Ssn, d.Dname;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| | Ssn | Dname |
|---|-----------|----------------|
| ▶ | 123456789 | Administration |
| | 123456789 | Headquarters |
| | 123456789 | Research |
| | 333445555 | Administration |
| | 333445555 | Headquarters |
| | 333445555 | Research |
| | 453453453 | Administration |
| | 453453453 | Headquarters |
| | 453453453 | Research |
| | 666884444 | Administration |
| | 666884444 | Headquarters |
| | 666884444 | Research |
| | 888665555 | Administration |
| | 888665555 | Headquarters |
| | 888665555 | Research |
| | 987654321 | Administration |
| | 987654321 | Headquarters |
| | 987654321 | Research |
| | 987987987 | Administration |
| | 987987987 | Headquarters |
| | 987987987 | Research |
| | 999887777 | Administration |
| | 999887777 | Headquarters |
| | 999887777 | Research |

- Q8 : 성이 'Wong'인 사원이 일하는 프로젝트, 혹은 성이 'Wong'인 사원이 관리하는 부서에서 진행하는 프로젝트의 번호를 검색하시오. 테이블 데이터는 프로젝트 번호의 오름차순으로 나타내시오

```

SELECT w.Pno
FROM employee e JOIN works_on w ON e.Ssn = w.Essn
WHERE e.Lname = 'Wong'
UNION
SELECT p.Pnumber
FROM employee e JOIN department d ON e.Ssn = d.Mgr_ssn JOIN project p ON d.Dnumber = p.Dnum
WHERE e.Lname = 'Wong'
ORDER BY Pno;
    
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| | Pno |
|---|-----|
| ▶ | 1 |
| | 2 |
| | 3 |
| | 10 |
| | 20 |

- Q9 : 주소에 'Houston, TX'이 들어있는 모든 사원의 이름을 검색하시오. 출력 컬럼은 사원의 이름, 주소 순으로 하며, 테이블 데이터는 이름 중 Fname의 오름차순으로 나타내시오.

```

SELECT Fname, Minit, Lname, address
FROM employee
WHERE address LIKE '%Houston, TX'
ORDER BY Fname;

```

| | Fname | Minit | Lname | address |
|---|----------|-------|---------|--------------------------|
| ▶ | Ahmad | V | Jabbar | 980 Dallas, Houston, TX |
| | Franklin | T | Wong | 638 Voss, Houston, TX |
| | James | E | Borg | 450 Stone, Houston, TX |
| | John | B | Smith | 731 Fondren, Houston, TX |
| | Joyce | A | English | 5631 Rice, Houston, TX |

- Q10 : 'ProductX' 프로젝트에 참여하는 모든 사원의 이름, 그리고 그들의 급여를 10% 올린 경우의 급여를 구하시오

```

SELECT e.Fname, e.Minit, e.Lname, (e.Salary*1.1) AS 'Salary(10% up)'
FROM works_on w JOIN project p ON p.Pnumber = w.Pno JOIN employee e ON e.Ssn = w.Essn
WHERE p.Pname = 'ProductX';

```

| | Fname | Minit | Lname | Salary(10% up) |
|---|-------|-------|---------|----------------|
| ▶ | John | B | Smith | 33000.000 |
| | Joyce | A | English | 27500.000 |

- Q11 : 모든 부서 이름, 부서에 소속한 사원의 이름, 그리고 각 사원이 진행하는 프로젝트 이름의 리스트를 검색하시오. 테이블 데이터는 부서 이름의 내림차순, 그리고 각 부서 내에서 사원 이름의 오름차순, 프로젝트 이름의 오름차순으로 나타내시오

```

SELECT d.Dname, e.Fname, e.Minit, e.Lname, p.Pname
FROM employee e JOIN department d ON e.Dno = d.Dnumber
JOIN works_on w ON e.Ssn = w.Essn
JOIN project p ON p.Pnumber = w.Pno
ORDER BY d.Dname DESC, e.Fname ASC, p.Pname ASC;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: A

| | Dname | Fname | Minit | Lname | Pname |
|---|----------------|----------|-------|---------|-----------------|
| ▶ | Research | Franklin | T | Wong | Computerization |
| | Research | Franklin | T | Wong | ProductY |
| | Research | Franklin | T | Wong | ProductZ |
| | Research | Franklin | T | Wong | Reorganization |
| | Research | John | B | Smith | ProductX |
| | Research | John | B | Smith | ProductY |
| | Research | Joyce | A | English | ProductX |
| | Research | Joyce | A | English | ProductY |
| | Research | Ramesh | K | Narayan | ProductZ |
| | Headquarters | James | E | Borg | Reorganization |
| | Administration | Ahmad | V | Jabbar | Computerization |
| | Administration | Ahmad | V | Jabbar | Newbenefits |
| | Administration | Alicia | J | Zelaya | Computerization |
| | Administration | Alicia | J | Zelaya | Newbenefits |
| | Administration | Jennifer | S | Wallace | Newbenefits |
| | Administration | Jennifer | S | Wallace | Reorganization |

- Q12 : 5번 부서에 근무하는 사원 중에서, ProjectX 프로젝트에 주당 10시간 이상 일하는 사원의 이름과 주당근무시간을 검색하시오

```
SELECT DISTINCT e.Fname, e.Minit, e.Lname, w.Hours
FROM employee e JOIN department d ON e.Dno = d.Dnumber
                  JOIN project p ON d.Dnumber = p.Dnum
                  JOIN works_on w ON e.Ssn = w.Essn AND p.Pnumber = w.Pno
WHERE p.Pname = 'ProductX' AND e.Dno = '5' AND w.Hours >= 10;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: A

| | Fname | Minit | Lname | Hours |
|---|-------|-------|---------|-------|
| ▶ | John | B | Smith | 32.5 |
| | Joyce | A | English | 20.0 |

- Q13 : 배우자가 있는 사원 수를 검색하시오

```
SELECT COUNT(e.Ssn)
FROM employee e JOIN dependent d ON e.Ssn = d.Essn
WHERE d.Relationship LIKE 'Spouse';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: A

| | COUNT(e.Ssn) |
|---|--------------|
| ▶ | 3 |

- Q14 : 부양가족이 있는 사원에 대해, 부양가족 수를 구하시오. 출력으로 사원의 이름과 부양가족 수(컬럼 이름은 NumOfDependents로 할 것)를 나열하시오

```
SELECT e.Fname, e.Minit, e.Lname, COUNT(*) AS NumOfDependents
FROM employee e JOIN dependent d ON e.Ssn = d.Essn
GROUP BY e.Ssn;
```

| | Fname | Minit | Lname | NumOfDependents |
|---|----------|-------|---------|-----------------|
| ▶ | John | B | Smith | 3 |
| | Franklin | T | Wong | 3 |
| | Jennifer | S | Wallace | 1 |

- Q15 : 모든 사원에 대해, 부양가족의 이름과 관계를 구하시오. 이때 부양가족이 없는 사원도 포함합니다. 출력 컬럼은 사원의 이름, 부양가족과의 관계, 부양가족의 이름 순으로 나열하시오. 테이블 데이터는 사원 이름 중 Fname, Minit, Lname의 오름차순으로 나타내시오

```
SELECT e.Fname, e.Minit, e.Lname, d.Relationship, d.Dependent_name
FROM employee e LEFT JOIN dependent d ON e.Ssn = d.Essn
ORDER BY e.Fname, e.Minit, e.Lname;
```

| | Fname | Minit | Lname | Relationship | Dependent_name |
|---|----------|-------|---------|--------------|----------------|
| ▶ | Ahmad | V | Jabbar | NULL | NULL |
| | Alicia | J | Zelaya | NULL | NULL |
| | Franklin | T | Wong | Spouse | Joy |
| | Franklin | T | Wong | Son | Theodore |
| | Franklin | T | Wong | Daughter | Alice |
| | James | E | Borg | NULL | NULL |
| | Jennifer | S | Wallace | Spouse | Abner |
| | John | B | Smith | Daughter | Alice |
| | John | B | Smith | Spouse | Elizabeth |
| | John | B | Smith | Son | Michael |
| | Joyce | A | English | NULL | NULL |
| | Ramesh | K | Narayan | NULL | NULL |

- Q16 : 모든 사원에 대해, 부양가족 수를 구하고, 이때 부양가족이 없는 사원은 부양가족수를 0으로 표시하시오. 출력 컬럼은 사원의 이름과 부양가족 수(NumOfDependents) 순으로 나열하시오. 테이블 데이터는 부양가족수의 내림차순, 이름 중 Fname의 오름차순으로 나타내시오.

```
SELECT e.Fname, e.Minit, e.Lname, COALESCE(COUNT(d.Relationship), 0) AS NumOfDependents
FROM employee e LEFT JOIN dependent d ON e.Ssn = d.Essn
GROUP BY e.Ssn
ORDER BY NumOfDependents DESC, e.Fname ASC;
```

| | Fname | Minit | Lname | NumOfDependents |
|---|----------|-------|---------|-----------------|
| ▶ | Franklin | T | Wong | 3 |
| | John | B | Smith | 3 |
| | Jennifer | S | Wallace | 1 |
| | Ahmad | V | Jabbar | 0 |
| | Alicia | J | Zelaya | 0 |
| | James | E | Borg | 0 |
| | Joyce | A | English | 0 |
| | Ramesh | K | Narayan | 0 |

- Q17 : 부양가족이 없는 사원의 이름을 구하시오. 테이블 데이터는 이름 중 Fname의 오름차순으로 나타내시오

```

SELECT e.Fname, e.Minit, e.Lname
FROM employee e LEFT JOIN dependent d ON e.Ssn = d.Essn
WHERE ISNULL(d.Essn)
GROUP BY e.Ssn
ORDER BY e.Fname ASC;

```

| | Fname | Minit | Lname |
|---|--------|-------|---------|
| ▶ | Ahmad | V | Jabbar |
| | Alicia | J | Zelaya |
| | James | E | Borg |
| | Joyce | A | English |
| | Ramesh | K | Narayan |

- Q18 : 자녀가 있는 사원에 대해, 부양가족 수를 구하시오. 출력 컬럼으로 사원의 이름과 부양가족수 (NumOfDependents) 순으로 나열하시오. 테이블 데이터는 이름 중 Fname의 오름차순으로 나타내시오

```

SELECT Fname, Minit, Lname, COUNT(*) AS NumOfDependents
FROM employee e JOIN dependent d ON e.Ssn = d.Essn
WHERE e.Ssn IN (
    SELECT DISTINCT e.Ssn
    FROM employee e JOIN dependent d ON e.Ssn = d.Essn
    WHERE d.Relationship IN ('Son', 'Daughter')
)
GROUP BY e.Ssn
ORDER BY e.Fname ASC;

```

| | Fname | Minit | Lname | NumOfDependents |
|---|----------|-------|-------|-----------------|
| ▶ | Franklin | T | Wong | 3 |
| | John | B | Smith | 3 |

- Q19 : 각 부서에 대해, 부서의 위치와 같은 곳에서 진행되는 프로젝트를 검색하시오. 출력 컬럼은 부서 명칭, 부서 위치, 프로젝트 명칭, 프로젝트 위치 순으로 나열하시오. 테이블 데이터는 부서 명칭과 프로젝트 명칭은 오름차순으로 나타내시오

```

SELECT d.Dname, dl.Dlocation, p.Pname, p.Plocation
FROM department d JOIN dept_locations dl USING (Dnumber)
JOIN project p ON dl.Dlocation = p.Plocation AND p.Dnum = d.Dnumber
ORDER BY d.Dname ASC, p.Pname ASC;

```

| | Dname | Dlocation | Pname | Plocation |
|---|----------------|-----------|-----------------|-----------|
| ▶ | Administration | Stafford | Computerization | Stafford |
| | Administration | Stafford | Newbenefits | Stafford |
| | Headquarters | Houston | Reorganization | Houston |
| | Research | Bellaire | ProductX | Bellaire |
| | Research | Sugarland | ProductY | Sugarland |
| | Research | Houston | ProductZ | Houston |