SET OPERATIONS

SQL set operations are

- Union operation (**UNION**),
- Set difference (**EXCEPT**)
- Intersection operation (**INTERSECT**)

Duplicate tuples are eliminated from the result

Requires *union compatible relations*

SET OPERATIONS

Make a list of Pname's for projects that involve an employee whose last name is 'Smith'

as a worker **OR**

as a manager of the department that controls the project.

EMPLOYEE

FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO

DEPARTMENT

DNAME	DNUMBER	MGRSSN	MGRSTARTDATE

PROJECT

PNAME	PNUMBER	PLOCATION	DNUM

DEPT_LOCATIONS

DNUMBER	DLOCATION

WORKS_ON

ESSN	PNO	HOURS

SET OPERATIONS

Make a list of Pname's for projects that involve an employee whose last name is 'Smith'

as a worker **OR**

as a manager of the department that controls the project.

```
(SELECT  PNAME
FROM     (PROJECT JOIN WORKS_ON ON PNUMBER=PNO ) JOIN
                  EMPLOYEE ON ESSN=SSN
WHERE    LNAME='Smith')
UNION
(SELECT  PNAME
         FROM     (PROJECT JOIN DEPARTMENT ON DNUM=DNUMBER ) JOIN
                           EMPLOYEE ON MGRSSN=SSN
         WHERE    LNAME='Smith')
```

EXAMPLE: Retrieve the SSN of employees who have no dependents.

ALL_EMPS ← π SSN(EMPLOYEE)

EMPS_WITH_DEPS(SSN) ← π ESSN(DEPENDENT)

EMPS_WITHOUT_DEPS ← (ALL_EMPS - EMPS_WITH_DEPS)

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

EXAMPLE: Retrieve the SSN of employees who have no dependents.

```
SELECT SSN FROM EMPLOYEE
EXCEPT
SELECT ESSN FROM DEPENDENT
```

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

EXAMPLE: Retrieve the names of employees who have no dependents.

ALL_EMPS ← π SSN(EMPLOYEE)

EMPS_WITH_DEPS(SSN) ← π ESSN(DEPENDENT)

EMPS_WITHOUT_DEPS ← (ALL_EMPS - EMPS_WITH_DEPS)

RESULT ← π LNAME, FNAME (EMPS_WITHOUT_DEPS * EMPLOYEE)

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

EXAMPLE: Retrieve the SSN of employees who has a daughter as well as a son

```
SELECT ESSN
FROM DEPENDENT
WHERE Relationship = 'Daughter' and Relationship = 'SON'
```

WHATS WRONG ??

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

EXAMPLE: Retrieve the SSN of employees who has a daughter as well as a son

```
SELECT ESSN
FROM DEPENDENT
WHERE Relationship = 'Daughter'

INTERSECT

SELECT ESSN
FROM DEPENDENT
WHERE Relationship = 'SON'
```

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

SET OPERATIONS

Find SSN of employees who work on project named ProductX **and** as well as on project named ProductY

EMPLOYEE

FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO

DEPARTMENT

DNAME	DNUMBER	MGRSSN	MGRSTARTDATE

PROJECT

PNAME	PNUMBER	PLOCATION	DNUM

DEPT_LOCATIONS

DNUMBER	DLOCATION

WORKS_ON

ESSN	PNO	HOURS

ALIASES

- **In SQL**, we can use the **same name for two or more attributes** as long as the attributes are in ***different relations***

A query that refers to two attributes with the same name must *prefix* the relation's name to the attribute name

Example:

EMPLOYEE.DNO, DEPARTMENT.DNO

ALIASES

For each employee, retrieve the employee's name, and the name of his or her immediate supervisor.

Using **AS** keyword to specify aliases

```
SELECT  E.FNAME, E.LNAME, S.FNAME, S.LNAME
FROM    EMPLOYEE AS E, EMPLOYEE AS S
WHERE   E.SUPERSSN=S.SSN
```

```
SELECT  E.FNAME, E.LNAME, S.FNAME, S.LNAME
FROM    EMPLOYEE E S
WHERE   E.SUPERSSN=S.SSN
```

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address
John	B	Smith	123456789	1965-01-09	731 Fondren, Ho
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houst
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Sp
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bella
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, H
Joyce	A	English	453453453	1972-07-31	5631 Rice, Hous
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Hou
James	E	Borg	888665555	1937-11-10	450 Stone, Hous

E.Fname	E.Lname	S.Fname	S.Lname
John	Smith	Franklin	Wong
Franklin	Wong	James	Borg
Alicia	Zelaya	Jennifer	Wallace
Jennifer	Wallace	James	Borg
Ramesh	Narayan	Franklin	Wong
Joyce	English	Franklin	Wong
Ahmad	Jabbar	Jennifer	Wallace

ARITHMETIC OPERATIONS

Arithmetic operators '+', '-'. '*', and '/') can be applied to numeric values in an SQL query result

Give all employees who work on the 'ProductX' project a 10% raise.

```
SELECT    FNAME, LNAME, 1.1*SALARY
FROM      (WORKS_ON JOIN PROJECT ON PNO=PNUMBER)
          JOIN EMPLOYEE ON ESSN=SSN
WHERE     PNAME='ProductX'
```

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong				M	40000	888665555	5
Alicia	J	Zelaya				F	25000	987654321	4
Jennifer	S	Wallace				F	43000	888665555	4
Ramesh	K	Narayan				M	38000	333445555	5
Joyce	A	English				F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

PROJECT

PNAME	PNUMBER	PLOCATION	DNUM

WORKS_ON

ESSN	PNO	HOURS

ORDER BY

The **ORDER BY** clause **sort** the ***tuples*** in a query result

Retrieve a list of employees and the projects each works in, ordered by the employee's department number

```
SELECT    DNO, LNAME, FNAME, PNO
FROM      EMPLOYEE JOIN WORKS_ON ON SSN=ESSN
ORDER BY  DNO
```

EMPLOYEE

FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO

DEPARTMENT

DNAME	DNUMBER	MGRSSN	MGRSTARTDATE

PROJECT

PNAME	PNUMBER	PLOCATION	DNUM

DEPT_LOCATIONS

DNUMBER	DLOCATION

WORKS_ON

ESSN	PNO	HOURS

ORDER BY

The **ORDER BY** clause **sort** the ***tuples*** in a query result

Retrieve a list of employees and the projects each works in, ordered by the employee's department number

```
SELECT    DNO, LNAME, FNAME, PNO
FROM      EMPLOYEE JOIN WORKS_ON ON SSN=ESSN
ORDER BY  DNO
```

The default order is in ***ascending order*** of values

We can specify the keyword **DESC** if we want a descending order

- **ORDER BY** Dname **DESC,** Lname **ASC**

ORDER BY

Retrieve a list of Male employees and the projects each works in, ordered by the employee's department **name**, **and within each department ordered alphabetically by employee last name, then first name.**

```
SELECT    DNAME, LNAME, FNAME, PNAME
FROM      ((DEPARTMENT JOIN EMPLOYEE ON DNUMBER=DNO )
               JOIN WORKS_ON ON SSN=ESSN )
               JOIN PROJECT ON PNO=PNUMBER)
WHERE     SEX = 'MALE'
ORDER BY  DNAME, LNAME, FNAME
```

SQL QUERIES

- Retrieve the names of all employees who do not have supervisors.
- **SELECT FNAME, LNAME**
  FROM EMPLOYEE
  WHERE SUPERSSN IS NULL

- **Note: If a join condition is specified, tuples with NULL values for the join attributes are not included in the result**

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

SUBSTRING COMPARISON

LIKE operator is used to compare partial strings

Two reserved characters are used:

- '%' (or '*' in some implementations) replaces an arbitrary number of characters, and
- '_' replaces a single arbitrary character

Retrieve all employees whose address is in Houston, Texas.

- **SELECT FNAME, LNAME**
  FROM EMPLOYEE
  WHERE ADDRESS LIKE '%Houston,TX%'

SUBSTRING COMPARISON

Retrieve all employees who were born during the 1950s.

- **SELECT** **FNAME, LNAME**
  FROM **EMPLOYEE**
  WHERE **BDATE LIKE** '195_',.

LIKE operator allows us to get around the fact that each value is considered atomic and indivisible

Hence, in SQL, character string attribute values are not atomic

AGGREGATE FUNCTIONS

- Include **COUNT**, **SUM**, **MAX**, **MIN**, and **AVG**
- Find the maximum salary, the minimum salary, and the average salary among all employees.

```
SELECT   MAX(SALARY), MIN(SALARY), AVG(SALARY)
FROM     EMPLOYEE
```

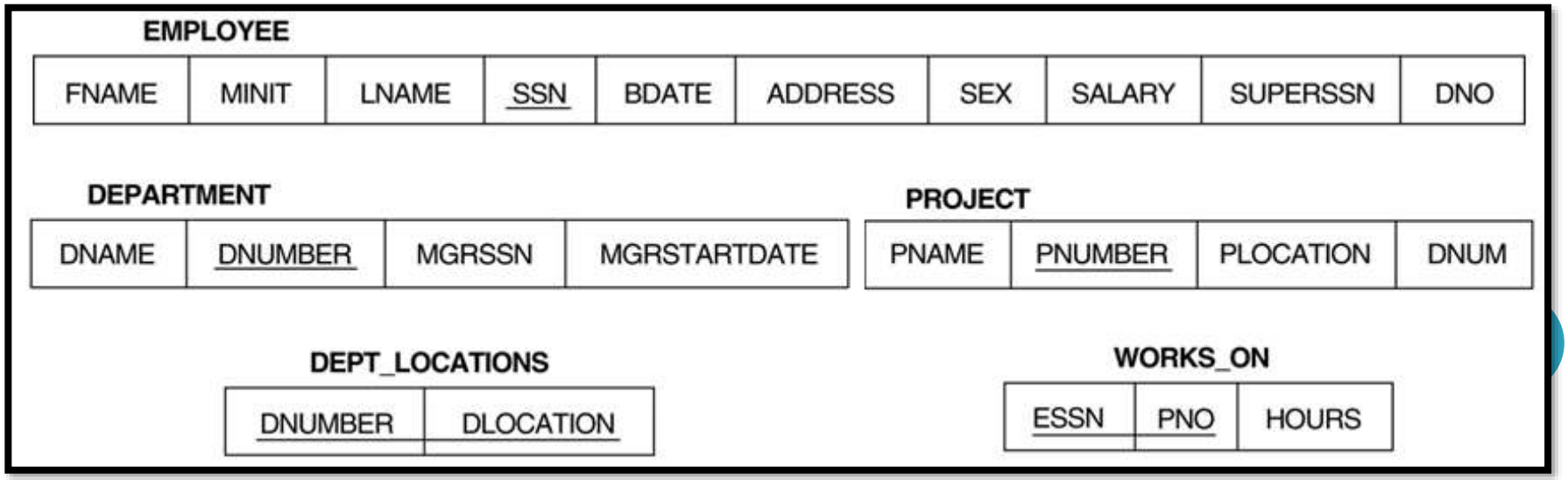
$\mathcal{F}_{\text{SUM Salary, AVERAGE Salary, MIN Salary}}$ (EMPLOYEE)

Some SQL implementations *may not allow more than one function* in the SELECT-clause

AGGREGATE FUNCTIONS

Retrieve the no. of employees in the ‘Administration' department

```
SELECT   COUNT (*)
FROM     EMPLOYEE JOIN DEPARTMENT ON DNO=DNUMBER
WHERE    DNAME=‘Administration’
```



Aggregate Functions $\mathcal{F}$

- $\mathcal{F}_{\text{MAX Salary}}$ (EMPLOYEE)
- $\mathcal{F}_{\text{MIN Salary}}$ (EMPLOYEE)
- $\mathcal{F}_{\text{SUM Salary, AVERAGE Salary}}$ (EMPLOYEE)
- $\mathcal{F}_{\text{COUNT SSN}}$ (EMPLOYEE)

COUNT (*) returns the no. of rows in the result of the query ***(it counts without removing duplicates)***

NULL values are **discarded** when aggregate functions are applied to a particular column (attribute).

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

AGGREGATE EXAMPLE

Count the number of distinct salary values in the database

```
SELECT  COUNT (DISTINCT Salary)
FROM    EMPLOYEE
```

NULL values are **discarded** when aggregate functions are applied to a particular attribute.

Aggregate functions are allowed only in the SELECT and the HAVING clause of a SQL statement.