# Database Systems
(CS 355 / CE 373)

Dr. Umer Tariq
Assistant Professor,
Dhanani School of Science & Engineering,
Habib University

# Acknowledgements


- Many slides have been borrowed from the official lecture slides accompanying the textbook:

  Database System Concepts, (2019), Seventh Edition,

  Avi Silberschatz, Henry F. Korth, S. Sudarshan

  McGraw-Hill, ISBN 9780078022159

  The original lecture slides are available at:

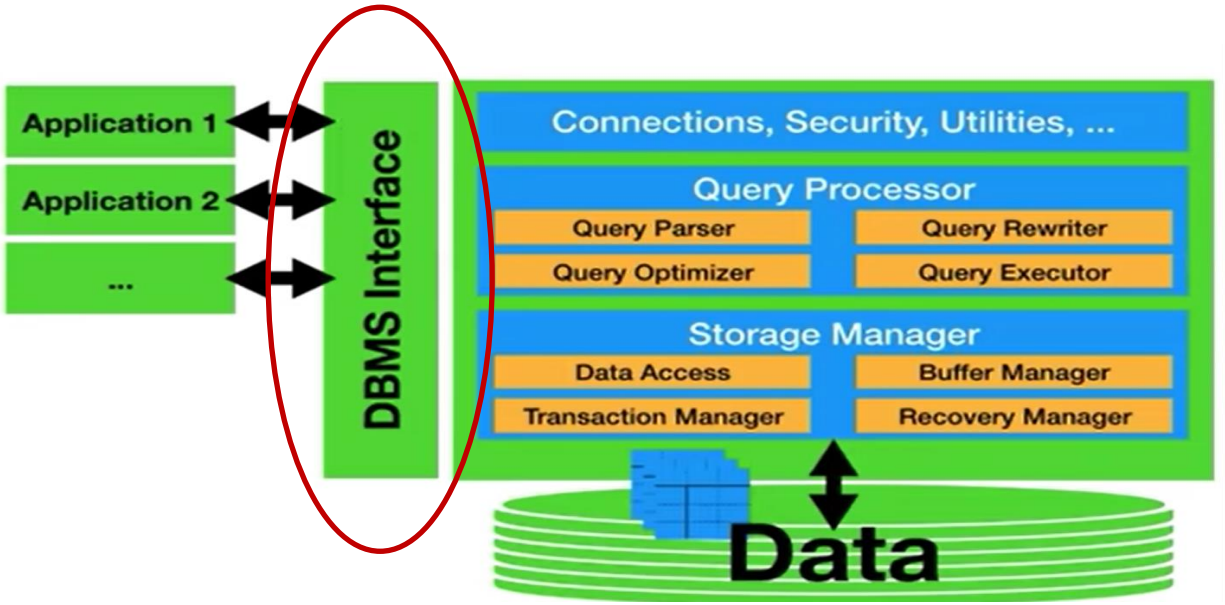
  https://www.db-book.com/

- Some of the slides have been borrowed from the lectures by Dr. Immanuel Trummer (Cornell University). Available at: (www.itrummer.org)

# Outline: Week 7

- SQL: Major Clauses in an SQL Query
- SQL: Aggregate Functions
- SQL: Joins

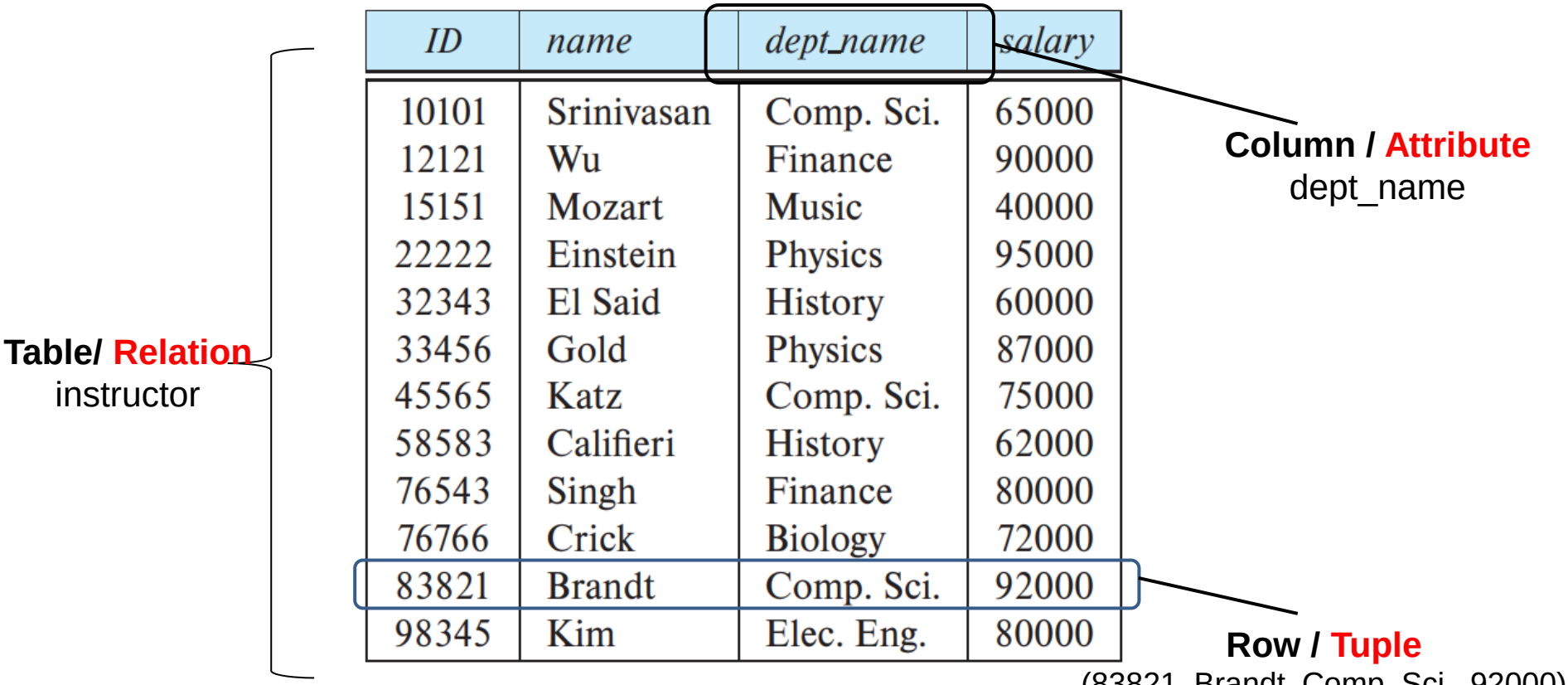
# DBMS-based Approach




- Data Model
  - A collection of conceptual tools for describing data, data relationships, data semantics, and consistency constraints.

# The Relational Model

- The relational model uses a collection of tables to represent both data and the relationships among those data.




| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 12121 | Wu | Finance | 90000 |
| 15151 | Mozart | Music | 40000 |
| 22222 | Einstein | Physics | 95000 |
| 32343 | El Said | History | 60000 |
| 33456 | Gold | Physics | 87000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 58583 | Califieri | History | 62000 |
| 76543 | Singh | Finance | 80000 |
| 76766 | Crick | Biology | 72000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |

**Figure 2.1** The *instructor* relation.

# A Sample Relational Model

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 22222 | Einstein | Physics | 95000 |
| 12121 | Wu | Finance | 90000 |
| 32343 | El Said | History | 60000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 98345 | Kim | Elec. Eng. | 80000 |
| 76766 | Crick | Biology | 72000 |
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 58583 | Califieri | History | 62000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 15151 | Mozart | Music | 40000 |
| 33456 | Gold | Physics | 87000 |
| 76543 | Singh | Finance | 80000 |

(a) The *instructor* table

| *dept_name* | *building* | *budget* |
|---|---|---|
| Comp. Sci. | Taylor | 100000 |
| Biology | Watson | 90000 |
| Elec. Eng. | Taylor | 85000 |
| Music | Packard | 80000 |
| Finance | Painter | 120000 |
| History | Painter | 50000 |
| Physics | Watson | 70000 |

(b) The *department* table

# Utilizing Relational DBMS: Lifecycle

1) • Design relational schema (and Tell the DBMS about it) } DDL

2) • Populate tables/relations

3) • Write "queries" to get information back from tables } DML

# Structured Query Language (SQL)

- The standard to access/retrieve/manipulate data in a **relational database**
- Examples of a Data Definition Language (DDL) Component

```
create table department
    (dept_name      char (20),
     building       char (15),
     budget         numeric (12,2));
```

- Examples of a Data Manipulation Language (DML) Component

```
select instructor.name
from instructor
where instructor.dept_name = 'History';
```

```
select instructor.ID, department.dept_name
from instructor, department
where instructor.dept_name= department.dept_name and
        department.budget > 95000;
```

# SQL Queries: University Database (1/4)

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 12121 | Wu | Finance | 90000 |
| 15151 | Mozart | Music | 40000 |
| 22222 | Einstein | Physics | 95000 |
| 32343 | El Said | History | 60000 |
| 33456 | Gold | Physics | 87000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 58583 | Califieri | History | 62000 |
| 76543 | Singh | Finance | 80000 |
| 76766 | Crick | Biology | 72000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |

**Figure 2.1** The *instructor* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* |
|---|---|---|---|---|
| 10101 | CS-101 | 1 | Fall | 2017 |
| 10101 | CS-315 | 1 | Spring | 2018 |
| 10101 | CS-347 | 1 | Fall | 2017 |
| 12121 | FIN-201 | 1 | Spring | 2018 |
| 15151 | MU-199 | 1 | Spring | 2018 |
| 22222 | PHY-101 | 1 | Fall | 2017 |
| 32343 | HIS-351 | 1 | Spring | 2018 |
| 45565 | CS-101 | 1 | Spring | 2018 |
| 45565 | CS-319 | 1 | Spring | 2018 |
| 76766 | BIO-101 | 1 | Summer | 2017 |
| 76766 | BIO-301 | 1 | Summer | 2018 |
| 83821 | CS-190 | 1 | Spring | 2017 |
| 83821 | CS-190 | 2 | Spring | 2017 |
| 83821 | CS-319 | 2 | Spring | 2018 |
| 98345 | EE-181 | 1 | Spring | 2017 |

**Figure 2.7** The *teaches* relation.

# SQL Queries: University Database (2/4)

| *dept_name* | *building* | *budget* |
|---|---|---|
| Biology | Watson | 90000 |
| Comp. Sci. | Taylor | 100000 |
| Elec. Eng. | Taylor | 85000 |
| Finance | Painter | 120000 |
| History | Painter | 50000 |
| Music | Packard | 80000 |
| Physics | Watson | 70000 |

**Figure 2.5** The *department* relation.

| *course_id* | *sec_id* | *semester* | *year* | *building* | *room_number* | *time_slot_id* |
|---|---|---|---|---|---|---|
| BIO-101 | 1 | Summer | 2017 | Painter | 514 | B |
| BIO-301 | 1 | Summer | 2018 | Painter | 514 | A |
| CS-101 | 1 | Fall | 2017 | Packard | 101 | H |
| CS-101 | 1 | Spring | 2018 | Packard | 101 | F |
| CS-190 | 1 | Spring | 2017 | Taylor | 3128 | E |
| CS-190 | 2 | Spring | 2017 | Taylor | 3128 | A |
| CS-315 | 1 | Spring | 2018 | Watson | 120 | D |
| CS-319 | 1 | Spring | 2018 | Watson | 100 | B |
| CS-319 | 2 | Spring | 2018 | Taylor | 3128 | C |
| CS-347 | 1 | Fall | 2017 | Taylor | 3128 | A |
| EE-181 | 1 | Spring | 2017 | Taylor | 3128 | C |
| FIN-201 | 1 | Spring | 2018 | Packard | 101 | B |
| HIS-351 | 1 | Spring | 2018 | Painter | 514 | C |
| MU-199 | 1 | Spring | 2018 | Packard | 101 | D |
| PHY-101 | 1 | Fall | 2017 | Watson | 100 | A |

**Figure 2.6** The *section* relation.

# SQL Queries: University Database (3/4)

| *course_id* | *prereq_id* |
|---|---|
| BIO-301 | BIO-101 |
| BIO-399 | BIO-101 |
| CS-190 | CS-101 |
| CS-315 | CS-101 |
| CS-319 | CS-101 |
| CS-347 | CS-101 |
| EE-181 | PHY-101 |

**Figure 2.3** The *prereq* relation.

| *course_id* | *title* | *dept_name* | *credits* |
|---|---|---|---|
| BIO-101 | Intro. to Biology | Biology | 4 |
| BIO-301 | Genetics | Biology | 4 |
| BIO-399 | Computational Biology | Biology | 3 |
| CS-101 | Intro. to Computer Science | Comp. Sci. | 4 |
| CS-190 | Game Design | Comp. Sci. | 4 |
| CS-315 | Robotics | Comp. Sci. | 3 |
| CS-319 | Image Processing | Comp. Sci. | 3 |
| CS-347 | Database System Concepts | Comp. Sci. | 3 |
| EE-181 | Intro. to Digital Systems | Elec. Eng. | 3 |
| FIN-201 | Investment Banking | Finance | 3 |
| HIS-351 | World History | History | 3 |
| MU-199 | Music Video Production | Music | 3 |
| PHY-101 | Physical Principles | Physics | 4 |

**Figure 2.2** The *course* relation.

# SQL Queries: University Database (4/4)

| *ID* | *name* | *dept_name* | *tot_cred* |
|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 |
| 12345 | Shankar | Comp. Sci. | 32 |
| 19991 | Brandt | History | 80 |
| 23121 | Chavez | Finance | 110 |
| 44553 | Peltier | Physics | 56 |
| 45678 | Levy | Physics | 46 |
| 54321 | Williams | Comp. Sci. | 54 |
| 55739 | Sanchez | Music | 38 |
| 70557 | Snow | Physics | 0 |
| 76543 | Brown | Comp. Sci. | 58 |
| 76653 | Aoi | Elec. Eng. | 60 |
| 98765 | Bourikas | Elec. Eng. | 98 |
| 98988 | Tanaka | Biology | 120 |

**Figure 4.1** The *student* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|
| 00128 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.2** The *takes* relation.

# Structured Query Language (SQL)

- The standard to access/retrieve/manipulate data in a **relational database**
- Examples of a Data Definition Language (DDL) Component

```
create table department
    (dept_name      char (20),
     building       char (15),
     budget         numeric (12,2));
```

- Examples of a Data Manipulation Language (DML) Component

```
select instructor.name
from instructor
where instructor.dept_name = 'History';
```

```
select instructor.ID, department.dept_name
from instructor, department
where instructor.dept_name= department.dept_name and
        department.budget > 95000;
```

# SQL: Data Manipulation

- Basic query structure

- Major clauses in an SQL query
  - SELECT
  - FROM
  - WHERE
  - GROUP BY
  - HAVING

3 basic clauses

# Remember: Order of Execution in SQL Queries

In general, the meaning of an SQL query can be understood as follows:

1. Generate a Cartesian product of the relations listed in the **from** clause.
2. Apply the predicates specified in the **where** clause on the result of Step 1.
3. For each tuple in the result of Step 2, output the attributes (or results of expressions) specified in the **select** clause.

# Remember: Updated Order of Execution in SQL Queries

The meaning of a query containing aggregation, **group by**, or **having** clauses is defined by the following sequence of operations:

1. As was the case for queries without aggregation, the **from** clause is first evaluated to get a relation.
2. If a **where** clause is present, the predicate in the **where** clause is applied on the result relation of the **from** clause.
3. Tuples satisfying the **where** predicate are then placed into groups by the **group by** clause if it is present. If the **group by** clause is absent, the entire set of tuples satisfying the **where** predicate is treated as being in one group.
4. The **having** clause, if it is present, is applied to each group; the groups that do not satisfy the **having** clause predicate are removed.
5. The **select** clause uses the remaining groups to generate tuples of the result of the query, applying the aggregate functions to get a single result tuple for each group.

Find the names of all instructors in "Comp. Sci" ?

SELECT
FROM
WHERE

# SQL Queries: University Database (1/4)

| ID | name | dept_name | salary |
|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 12121 | Wu | Finance | 90000 |
| 15151 | Mozart | Music | 40000 |
| 22222 | Einstein | Physics | ~~95000~~ null |
| 32343 | El Said | History | 60000 |
| 33456 | Gold | Physics | 87000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 58583 | Califieri | History | ~~62000~~ Null |
| 76543 | Singh | Finance | 80000 |
| 76766 | ~~Crick~~ | ~~Biology~~ | 72000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |

**Figure 2.1** The *instructor* relation.

| ID | course_id | sec_id | semester | year |
|---|---|---|---|---|
| 10101 | CS-101 | 1 | Fall | 2017 |
| 10101 | CS-315 | 1 | Spring | 2018 |
| 10101 | CS-347 | 1 | Fall | 2017 |
| 12121 | FIN-201 | 1 | Spring | 2018 |
| 15151 | MU-199 | 1 | Spring | 2018 |
| 22222 | PHY-101 | 1 | Fall | 2017 |
| 32343 | HIS-351 | 1 | Spring | 2018 |
| 45565 | CS-101 | 1 | Spring | 2018 |
| 45565 | CS-319 | 1 | Spring | 2018 |
| 76766 | BIO-101 | 1 | Summer | 2017 |
| 76766 | BIO-301 | 1 | Summer | 2018 |
| 83821 | CS-190 | 1 | Spring | 2017 |
| 83821 | CS-190 | 2 | Spring | 2017 |
| 83821 | CS-319 | 2 | Spring | 2018 |
| 98345 | EE-181 | 1 | Spring | 2017 |

**Figure 2.7** The *teaches* relation.

3) SELECT ID, name
1) FROM instructor
2) WHERE dept_name = 'Comp. Sci'

SELECT dept_name
FROM instructor

WHERE salary is not null

# Aggregate Functions: Group By

- Find the average salary of instructors in each department

1) FROM INSTRUCTOR

2) WHERE avg(salary) ≥ 42000

# Aggregate Functions: Group By

- Find the average salary of instructors in each department

~~2~~. SELECT

3.

1. FROM instructor

2) WHERE salary is not NULL

3. GROUP BY dept_name

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 12121 | Wu | Finance | 90000 |
| 15151 | Mozart | Music | 40000 |
| 22222 | Einstein | Physics | 95000 |
| 32343 | El Said | History | 60000 |
| 33456 | Gold | Physics | 87000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 58583 | Califieri | History | 62000 |
| 76543 | Singh | Finance | 80000 |
| 76766 | Crick | Biology | 72000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |

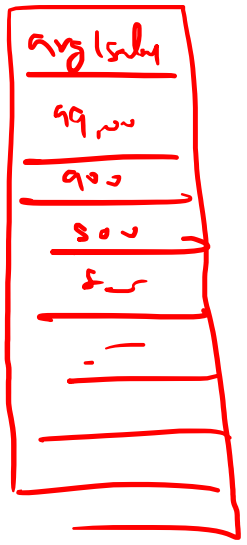
**Figure 2.1** The *instructor* relation.

# Aggregate Functions: Group By

- Find the average salary of instructors in each department

4) SELECT avg (salary)
1) **from** *instructor*
2) WHERE
3) **group by** *dept_name*;

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 76766 | Crick | Biology | 72000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |
| 12121 | Wu | Finance | 90000 |
| 76543 | Singh | Finance | 80000 |
| 32343 | El Said | History | 60000 |
| 58583 | Califieri | History | 62000 |
| 15151 | Mozart | Music | 40000 |
| 33456 | Gold | Physics | 87000 |
| 22222 | Einstein | Physics | 95000 |



# Aggregate Functions: Group By

- Find the average salary of instructors in each department

```
select dept_name, avg (salary) as avg_salary
from instructor
group by dept_name;
```

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 76766 | Crick | Biology | 72000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |
| 12121 | Wu | Finance | 90000 |
| 76543 | Singh | Finance | 80000 |
| 32343 | El Said | History | 60000 |
| 58583 | Califieri | History | 62000 |
| 15151 | Mozart | Music | 40000 |
| 33456 | Gold | Physics | 87000 |
| 22222 | Einstein | Physics | 95000 |

| *dept_name* | *avg_salary* |
|---|---|
| Biology | 72000 |
| Comp. Sci. | 77333 |
| Elec. Eng. | 80000 |
| Finance | 85000 |
| History | 61000 |
| Music | 40000 |
| Physics | 91000 |

# Aggregate Functions: Group By + Having

- Find the names and average salaries of all departments whose average salary is greater than 42000

# Aggregate Functions: Group By + Having

- Find the names and average salaries of all departments whose average salary is greater than 42000

1) FROM instructor
2) WHERE
3) GROUP BY dept_name
4) HAVING

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 12121 | Wu | Finance | 90000 |
| 15151 | Mozart | Music | 40000 |
| 22222 | Einstein | Physics | 95000 |
| 32343 | El Said | History | 60000 |
| 33456 | Gold | Physics | 87000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 58583 | Califieri | History | 62000 |
| 76543 | Singh | Finance | 80000 |
| 76766 | Crick | Biology | 72000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |

**Figure 2.1** The *instructor* relation.

# Aggregate Functions: Group By + Having

- Find the names and average salaries of all departments whose average salary is greater than 42000

1) **from** instructor
2) WHERE
3) **group by** dept_name
4) HAVING avg(salary) ≥ 42000

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 76766 | Crick | Biology | 72000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |
| 12121 | Wu | Finance | 90000 |
| 76543 | Singh | Finance | 80000 |
| 32343 | El Said | History | 60000 |
| 58583 | Califieri | History | 62000 |
| 15151 | Mozart | Music | 40000 |
| 33456 | Gold | Physics | 87000 |
| 22222 | Einstein | Physics | 95000 |

# Aggregate Functions: Group By + Having

- Find the names and average salaries of all departments whose average salary is greater than 42000

5) SELECT dept_name, avg(salary)
1) **from** instructor
2) where
3) **group by** dept_name
4) **having** avg (salary) > 42000;

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 76766 | Crick | Biology | 72000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |
| 12121 | Wu | Finance | 90000 |
| 76543 | Singh | Finance | 80000 |
| 32343 | El Said | History | 60000 |
| 58583 | Califieri | History | 62000 |
| 15151 | Mozart | Music | 40000 |
| 33456 | Gold | Physics | 87000 |
| 22222 | Einstein | Physics | 95000 |

# Aggregate Functions: Group By + Having

- Find the names and average salaries of all departments whose average salary is greater than 42000

```
select dept_name, avg (salary) as avg_salary
from instructor
group by dept_name
having avg (salary) > 42000;
```

| dept_name | avg_salary |
|---|---|
| Biology | 72000 |
| Comp. Sci. | 77333.33333333333 |
| Elec. Eng. | 80000 |
| Finance | 85000 |
| History | 61000 |
| Physics | 91000 |

# Aggregate Functions: Group By + Having

- Find the names and average salaries of all departments whose average salary is greater than 42000

```
select dept_name, avg (salary) as avg_salary
from instructor
group by dept_name
having avg (salary) > 42000;
```

6, order by avg_salary ↓

| dept_name | avg_salary |
|---|---|
| Biology | 72000 |
| Comp. Sci. | 77333.33333333333 |
| Elec. Eng. | 80000 |
| Finance | 85000 |
| History | 61000 |
| Physics | 91000 |

# SQL: Various Join Operations

- Inner Join
- Left Outer Join
- Right Outer Join
- Full Outer Join

# Basic Operations of Relational Algebra: **Join (⋈)**

- The **join** operation allows us to combine a select operation and a **c**artesian product operation into a single operation.
  - Notation: Consider relations $r(R)$ and $s(S)$. Let $\theta$ be a predicate on attributes in the schema R ∪ S. The join operation $r \bowtie_\theta s$ is defined as follows:

$$r \bowtie_\theta s = \sigma_\theta (r \times s)$$

- Thus $\sigma_{instructor.id = teaches.id}$ (*instructor* x *teaches*) can equivalently be written as:

***instructor*** $\bowtie_{Instructor.id = teaches.id}$ ***teaches.***

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 12121 | Wu | Finance | 90000 |
| 15151 | Mozart | Music | 40000 |
| 22222 | Einstein | Physics | 95000 |
| 32343 | El Said | History | 60000 |
| 33456 | Gold | Physics | 87000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 58583 | Califieri | History | 62000 |
| 76543 | Singh | Finance | 80000 |
| 76766 | Crick | Biology | 72000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |

**Figure 2.1** The *instructor* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* |
|---|---|---|---|---|
| 10101 | CS-101 | 1 | Fall | 2017 |
| 10101 | CS-315 | 1 | Spring | 2018 |
| 10101 | CS-347 | 1 | Fall | 2017 |
| 12121 | FIN-201 | 1 | Spring | 2018 |
| 15151 | MU-199 | 1 | Spring | 2018 |
| 22222 | PHY-101 | 1 | Fall | 2017 |
| 32343 | HIS-351 | 1 | Spring | 2018 |
| 45565 | CS-101 | 1 | Spring | 2018 |
| 45565 | CS-319 | 1 | Spring | 2018 |
| 76766 | BIO-101 | 1 | Summer | 2017 |
| 76766 | BIO-301 | 1 | Summer | 2018 |
| 83821 | CS-190 | 1 | Spring | 2017 |
| 83821 | CS-190 | 2 | Spring | 2017 |
| 83821 | CS-319 | 2 | Spring | 2018 |
| 98345 | EE-181 | 1 | Spring | 2017 |

**Figure 2.7** The *teaches* relation.

| *instructor.ID* | *name* | *dept_name* | *salary* | *teaches.ID* | *course_id* | *sec_id* | *semester* | *year* |
|---|---|---|---|---|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 | 10101 | CS-101 | 1 | Fall | 2017 |
| 10101 | Srinivasan | Comp. Sci. | 65000 | 10101 | CS-315 | 1 | Spring | 2018 |
| 10101 | Srinivasan | Comp. Sci. | 65000 | 10101 | CS-347 | 1 | Fall | 2017 |
| 12121 | Wu | Finance | 90000 | 12121 | FIN-201 | 1 | Spring | 2018 |
| 15151 | Mozart | Music | 40000 | 15151 | MU-199 | 1 | Spring | 2018 |
| 22222 | Einstein | Physics | 95000 | 22222 | PHY-101 | 1 | Fall | 2017 |
| 32343 | El Said | History | 60000 | 32343 | HIS-351 | 1 | Spring | 2018 |
| 45565 | Katz | Comp. Sci. | 75000 | 45565 | CS-101 | 1 | Spring | 2018 |
| 45565 | Katz | Comp. Sci. | 75000 | 45565 | CS-319 | 1 | Spring | 2018 |
| 76766 | Crick | Biology | 72000 | 76766 | BIO-101 | 1 | Summer | 2017 |
| 76766 | Crick | Biology | 72000 | 76766 | BIO-301 | 1 | Summer | 2018 |
| 83821 | Brandt | Comp. Sci. | 92000 | 83821 | CS-190 | 1 | Spring | 2017 |
| 83821 | Brandt | Comp. Sci. | 92000 | 83821 | CS-190 | 2 | Spring | 2017 |
| 83821 | Brandt | Comp. Sci. | 92000 | 83821 | CS-319 | 2 | Spring | 2018 |
| 98345 | Kim | Elec. Eng. | 80000 | 98345 | EE-181 | 1 | Spring | 2017 |

# More about Join Operation: **Inner Join (⋈)**

- The **join** operation allows us to combine a select operation and a **c**artesian product operation into a single operation.
  - Notation: Consider relations $r(R)$ and $s(S)$. Let $\theta$ be a predicate on attributes in the schema R ∪ S. The join operation $r \bowtie_\theta s$ is defined as follows:

$$r \bowtie_\theta s = \sigma_\theta (r \times s)$$

- Thus $\sigma_{instructor.id = teaches.id}$ (*instructor* x *teaches*) can equivalently be written as: *instructor* $\bowtie_{Instructor.id = teaches.id}$ *teaches*.
- This is also known as **Inner Join.**

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 12121 | Wu | Finance | 90000 |
| 15151 | Mozart | Music | 40000 |
| 22222 | Einstein | Physics | 95000 |
| 32343 | El Said | History | 60000 |
| 33456 | Gold | Physics | 87000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 58583 | Califieri | History | 62000 |
| 76543 | Singh | Finance | 80000 |
| 76766 | Crick | Biology | 72000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |

**Figure 2.1** The *instructor* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* |
|---|---|---|---|---|
| 10101 | CS-101 | 1 | Fall | 2017 |
| 10101 | CS-315 | 1 | Spring | 2018 |
| 10101 | CS-347 | 1 | Fall | 2017 |
| 12121 | FIN-201 | 1 | Spring | 2018 |
| 15151 | MU-199 | 1 | Spring | 2018 |
| 22222 | PHY-101 | 1 | Fall | 2017 |
| 32343 | HIS-351 | 1 | Spring | 2018 |
| 45565 | CS-101 | 1 | Spring | 2018 |
| 45565 | CS-319 | 1 | Spring | 2018 |
| 76766 | BIO-101 | 1 | Summer | 2017 |
| 76766 | BIO-301 | 1 | Summer | 2018 |
| 83821 | CS-190 | 1 | Spring | 2017 |
| 83821 | CS-190 | 2 | Spring | 2017 |
| 83821 | CS-319 | 2 | Spring | 2018 |
| 98345 | EE-181 | 1 | Spring | 2017 |

**Figure 2.7** The *teaches* relation.

| *instructor.ID* | *name* | *dept_name* | *salary* | *teaches.ID* | *course_id* | *sec_id* | *semester* | *year* |
|---|---|---|---|---|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 | 10101 | CS-101 | 1 | Fall | 2017 |
| 10101 | Srinivasan | Comp. Sci. | 65000 | 10101 | CS-315 | 1 | Spring | 2018 |
| 10101 | Srinivasan | Comp. Sci. | 65000 | 10101 | CS-347 | 1 | Fall | 2017 |
| 12121 | Wu | Finance | 90000 | 12121 | FIN-201 | 1 | Spring | 2018 |
| 15151 | Mozart | Music | 40000 | 15151 | MU-199 | 1 | Spring | 2018 |
| 22222 | Einstein | Physics | 95000 | 22222 | PHY-101 | 1 | Fall | 2017 |
| 32343 | El Said | History | 60000 | 32343 | HIS-351 | 1 | Spring | 2018 |
| 45565 | Katz | Comp. Sci. | 75000 | 45565 | CS-101 | 1 | Spring | 2018 |
| 45565 | Katz | Comp. Sci. | 75000 | 45565 | CS-319 | 1 | Spring | 2018 |
| 76766 | Crick | Biology | 72000 | 76766 | BIO-101 | 1 | Summer | 2017 |
| 76766 | Crick | Biology | 72000 | 76766 | BIO-301 | 1 | Summer | 2018 |
| 83821 | Brandt | Comp. Sci. | 92000 | 83821 | CS-190 | 1 | Spring | 2017 |
| 83821 | Brandt | Comp. Sci. | 92000 | 83821 | CS-190 | 2 | Spring | 2017 |
| 83821 | Brandt | Comp. Sci. | 92000 | 83821 | CS-319 | 2 | Spring | 2018 |
| 98345 | Kim | Elec. Eng. | 80000 | 98345 | EE-181 | 1 | Spring | 2017 |

# SQL: **Inner Join (⋈)**

```
select *
from instructor, teaches
where instructor.ID=teaches.ID
```

```
select *
from instructor join teaches  on instructor.ID=teaches.ID
```

```
select *
from instructor inner join teaches  on instructor.ID=teaches.ID
```

| *ID* | *name* | *dept_name* | *salary* |
|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 12121 | Wu | Finance | 90000 |
| 15151 | Mozart | Music | 40000 |
| 22222 | Einstein | Physics | 95000 |
| 32343 | El Said | History | 60000 |
| 33456 | Gold | Physics | 87000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 58583 | Califieri | History | 62000 |
| 76543 | Singh | Finance | 80000 |
| 76766 | Crick | Biology | 72000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 98345 | Kim | Elec. Eng. | 80000 |

**Figure 2.1** The *instructor* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* |
|---|---|---|---|---|
| 10101 | CS-101 | 1 | Fall | 2017 |
| 10101 | CS-315 | 1 | Spring | 2018 |
| 10101 | CS-347 | 1 | Fall | 2017 |
| 12121 | FIN-201 | 1 | Spring | 2018 |
| 15151 | MU-199 | 1 | Spring | 2018 |
| 22222 | PHY-101 | 1 | Fall | 2017 |
| 32343 | HIS-351 | 1 | Spring | 2018 |
| 45565 | CS-101 | 1 | Spring | 2018 |
| 45565 | CS-319 | 1 | Spring | 2018 |
| 76766 | BIO-101 | 1 | Summer | 2017 |
| 76766 | BIO-301 | 1 | Summer | 2018 |
| 83821 | CS-190 | 1 | Spring | 2017 |
| 83821 | CS-190 | 2 | Spring | 2017 |
| 83821 | CS-319 | 2 | Spring | 2018 |
| 98345 | EE-181 | 1 | Spring | 2017 |

**Figure 2.7** The *teaches* relation.

| *instructor.ID* | *name* | *dept_name* | *salary* | *teaches.ID* | *course_id* | *sec_id* | *semester* | *year* |
|---|---|---|---|---|---|---|---|---|
| 10101 | Srinivasan | Comp. Sci. | 65000 | 10101 | CS-101 | 1 | Fall | 2017 |
| 10101 | Srinivasan | Comp. Sci. | 65000 | 10101 | CS-315 | 1 | Spring | 2018 |
| 10101 | Srinivasan | Comp. Sci. | 65000 | 10101 | CS-347 | 1 | Fall | 2017 |
| 12121 | Wu | Finance | 90000 | 12121 | FIN-201 | 1 | Spring | 2018 |
| 15151 | Mozart | Music | 40000 | 15151 | MU-199 | 1 | Spring | 2018 |
| 22222 | Einstein | Physics | 95000 | 22222 | PHY-101 | 1 | Fall | 2017 |
| 32343 | El Said | History | 60000 | 32343 | HIS-351 | 1 | Spring | 2018 |
| 45565 | Katz | Comp. Sci. | 75000 | 45565 | CS-101 | 1 | Spring | 2018 |
| 45565 | Katz | Comp. Sci. | 75000 | 45565 | CS-319 | 1 | Spring | 2018 |
| 76766 | Crick | Biology | 72000 | 76766 | BIO-101 | 1 | Summer | 2017 |
| 76766 | Crick | Biology | 72000 | 76766 | BIO-301 | 1 | Summer | 2018 |
| 83821 | Brandt | Comp. Sci. | 92000 | 83821 | CS-190 | 1 | Spring | 2017 |
| 83821 | Brandt | Comp. Sci. | 92000 | 83821 | CS-190 | 2 | Spring | 2017 |
| 83821 | Brandt | Comp. Sci. | 92000 | 83821 | CS-319 | 2 | Spring | 2018 |
| 98345 | Kim | Elec. Eng. | 80000 | 98345 | EE-181 | 1 | Spring | 2017 |

# More about Join Operation: **Natural Join (*)**

- Suppose two relations $R_1$ and $R_2$ have a number of attributes that have the same name and the same domain.
- A **natural join** of $R_1$ and $R_2$ (denoted by *) includes only those tuples where the values of common attributes of $R_1$ and $R_2$ are the same.
- In this case, tuples with same values from $R_2$ need not be included.

| *ID* | *name* | *dept_name* | *tot_cred* |
|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 |
| 12345 | Shankar | Comp. Sci. | 32 |
| 19991 | Brandt | History | 80 |
| 23121 | Chavez | Finance | 110 |
| 44553 | Peltier | Physics | 56 |
| 45678 | Levy | Physics | 46 |
| 54321 | Williams | Comp. Sci. | 54 |
| 55739 | Sanchez | Music | 38 |
| 70557 | Snow | Physics | 0 |
| 76543 | Brown | Comp. Sci. | 58 |
| 76653 | Aoi | Elec. Eng. | 60 |
| 98765 | Bourikas | Elec. Eng. | 98 |
| 98988 | Tanaka | Biology | 120 |

**Figure 4.1** The *student* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|
| 00128 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.2** The *takes* relation.

| *ID* | *name* | *dept_name* | *tot_cred* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | Zhang | Comp. Sci. | 102 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | Shankar | Comp. Sci. | 32 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | Shankar | Comp. Sci. | 32 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | Shankar | Comp. Sci. | 32 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | Shankar | Comp. Sci. | 32 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | Brandt | History | 80 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | Chavez | Finance | 110 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | Peltier | Physics | 56 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | Levy | Physics | 46 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | Levy | Physics | 46 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | Levy | Physics | 46 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | Williams | Comp. Sci. | 54 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | Williams | Comp. Sci. | 54 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | Sanchez | Music | 38 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | Brown | Comp. Sci. | 58 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | Brown | Comp. Sci. | 58 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | Aoi | Elec. Eng. | 60 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | Bourikas | Elec. Eng. | 98 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | Bourikas | Elec. Eng. | 98 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | Tanaka | Biology | 120 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | Tanaka | Biology | 120 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.3** The natural join of the *student* relation with the *takes* relation.

# SQL: **Natural Join (∗)**

```
select *
from student natural join takes
```

- not supported by MS SQL Server (used in lab)

| *ID* | *name* | *dept_name* | *tot_cred* |
|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 |
| 12345 | Shankar | Comp. Sci. | 32 |
| 19991 | Brandt | History | 80 |
| 23121 | Chavez | Finance | 110 |
| 44553 | Peltier | Physics | 56 |
| 45678 | Levy | Physics | 46 |
| 54321 | Williams | Comp. Sci. | 54 |
| 55739 | Sanchez | Music | 38 |
| 70557 | Snow | Physics | 0 |
| 76543 | Brown | Comp. Sci. | 58 |
| 76653 | Aoi | Elec. Eng. | 60 |
| 98765 | Bourikas | Elec. Eng. | 98 |
| 98988 | Tanaka | Biology | 120 |

**Figure 4.1** The *student* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|
| 00128 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.2** The *takes* relation.

| *ID* | *name* | *dept_name* | *tot_cred* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | Zhang | Comp. Sci. | 102 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | Shankar | Comp. Sci. | 32 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | Shankar | Comp. Sci. | 32 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | Shankar | Comp. Sci. | 32 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | Shankar | Comp. Sci. | 32 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | Brandt | History | 80 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | Chavez | Finance | 110 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | Peltier | Physics | 56 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | Levy | Physics | 46 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | Levy | Physics | 46 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | Levy | Physics | 46 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | Williams | Comp. Sci. | 54 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | Williams | Comp. Sci. | 54 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | Sanchez | Music | 38 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | Brown | Comp. Sci. | 58 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | Brown | Comp. Sci. | 58 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | Aoi | Elec. Eng. | 60 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | Bourikas | Elec. Eng. | 98 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | Bourikas | Elec. Eng. | 98 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | Tanaka | Biology | 120 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | Tanaka | Biology | 120 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.3** The natural join of the *student* relation with the *takes* relation.

# Why do we Need Outer Joins?

- Consider *student* and *takes* relations. If we combine their information through *inner join* or *natural join*, the output does not include students who have not taken any courses.
- We can address this issue through the use of *outer joins*.
  - An outer join is used where the user wants to keep all the tuples in R, or all those in S, or all those in both relations in the result of the JOIN, regardless of whether or not they have matching tuples in the other relation.

| *ID* | *name* | *dept_name* | *tot_cred* |
|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 |
| 12345 | Shankar | Comp. Sci. | 32 |
| 19991 | Brandt | History | 80 |
| 23121 | Chavez | Finance | 110 |
| 44553 | Peltier | Physics | 56 |
| 45678 | Levy | Physics | 46 |
| 54321 | Williams | Comp. Sci. | 54 |
| 55739 | Sanchez | Music | 38 |
| 70557 | Snow | Physics | 0 |
| 76543 | Brown | Comp. Sci. | 58 |
| 76653 | Aoi | Elec. Eng. | 60 |
| 98765 | Bourikas | Elec. Eng. | 98 |
| 98988 | Tanaka | Biology | 120 |

**Figure 4.1** The *student* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|
| 00128 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.2** The *takes* relation.

| *ID* | *name* | *dept_name* | *tot_cred* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | Zhang | Comp. Sci. | 102 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | Shankar | Comp. Sci. | 32 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | Shankar | Comp. Sci. | 32 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | Shankar | Comp. Sci. | 32 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | Shankar | Comp. Sci. | 32 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | Brandt | History | 80 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | Chavez | Finance | 110 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | Peltier | Physics | 56 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | Levy | Physics | 46 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | Levy | Physics | 46 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | Levy | Physics | 46 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | Williams | Comp. Sci. | 54 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | Williams | Comp. Sci. | 54 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | Sanchez | Music | 38 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | Brown | Comp. Sci. | 58 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | Brown | Comp. Sci. | 58 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | Aoi | Elec. Eng. | 60 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | Bourikas | Elec. Eng. | 98 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | Bourikas | Elec. Eng. | 98 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | Tanaka | Biology | 120 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | Tanaka | Biology | 120 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.3** The natural join of the *student* relation with the *takes* relation.

# More about Join Operation: **Left Outer Join** (⟕)

- A **left outer join** operation keeps every tuple in the first, or left, relation *R* in $R \text{ ⟕ } S$; if no matching tuple is found in *S*, then the attributes of *S* in the join result are filled or padded with NULL values.

| *ID* | *name* | *dept_name* | *tot_cred* |
|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 |
| 12345 | Shankar | Comp. Sci. | 32 |
| 19991 | Brandt | History | 80 |
| 23121 | Chavez | Finance | 110 |
| 44553 | Peltier | Physics | 56 |
| 45678 | Levy | Physics | 46 |
| 54321 | Williams | Comp. Sci. | 54 |
| 55739 | Sanchez | Music | 38 |
| 70557 | Snow | Physics | 0 |
| 76543 | Brown | Comp. Sci. | 58 |
| 76653 | Aoi | Elec. Eng. | 60 |
| 98765 | Bourikas | Elec. Eng. | 98 |
| 98988 | Tanaka | Biology | 120 |

**Figure 4.1** The *student* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|
| 00128 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.2** The *takes* relation.

| *ID* | *name* | *dept_name* | *tot_cred* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | Zhang | Comp. Sci. | 102 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | Shankar | Comp. Sci. | 32 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | Shankar | Comp. Sci. | 32 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | Shankar | Comp. Sci. | 32 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | Shankar | Comp. Sci. | 32 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | Brandt | History | 80 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | Chavez | Finance | 110 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | Peltier | Physics | 56 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | Levy | Physics | 46 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | Levy | Physics | 46 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | Levy | Physics | 46 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | Williams | Comp. Sci. | 54 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | Williams | Comp. Sci. | 54 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | Sanchez | Music | 38 | MU-199 | 1 | Spring | 2018 | A- |
| 70557 | Snow | Physics | 0 | *null* | *null* | *null* | *null* | *null* |
| 76543 | Brown | Comp. Sci. | 58 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | Brown | Comp. Sci. | 58 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | Aoi | Elec. Eng. | 60 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | Bourikas | Elec. Eng. | 98 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | Bourikas | Elec. Eng. | 98 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | Tanaka | Biology | 120 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | Tanaka | Biology | 120 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.4** Result of *student* **natural left outer join** *takes*.

# SQL: **Left Outer Join (⟕)**

```
select *
from student left outer join takes on student.ID=takes.ID
```

| ID | name | dept_name | tot_cred |
|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 |
| 12345 | Shankar | Comp. Sci. | 32 |
| 19991 | Brandt | History | 80 |
| 23121 | Chavez | Finance | 110 |
| 44553 | Peltier | Physics | 56 |
| 45678 | Levy | Physics | 46 |
| 54321 | Williams | Comp. Sci. | 54 |
| 55739 | Sanchez | Music | 38 |
| 70557 | Snow | Physics | 0 |
| 76543 | Brown | Comp. Sci. | 58 |
| 76653 | Aoi | Elec. Eng. | 60 |
| 98765 | Bourikas | Elec. Eng. | 98 |
| 98988 | Tanaka | Biology | 120 |

**Figure 4.1** The *student* relation.

| ID | course_id | sec_id | semester | year | grade |
|---|---|---|---|---|---|
| 00128 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.2** The *takes* relation.

| ID | name | dept_name | tot_cred | course_id | sec_id | semester | year | grade |
|---|---|---|---|---|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | Zhang | Comp. Sci. | 102 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | Shankar | Comp. Sci. | 32 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | Shankar | Comp. Sci. | 32 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | Shankar | Comp. Sci. | 32 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | Shankar | Comp. Sci. | 32 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | Brandt | History | 80 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | Chavez | Finance | 110 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | Peltier | Physics | 56 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | Levy | Physics | 46 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | Levy | Physics | 46 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | Levy | Physics | 46 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | Williams | Comp. Sci. | 54 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | Williams | Comp. Sci. | 54 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | Sanchez | Music | 38 | MU-199 | 1 | Spring | 2018 | A- |
| 70557 | Snow | Physics | 0 | *null* | *null* | *null* | *null* | *null* |
| 76543 | Brown | Comp. Sci. | 58 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | Brown | Comp. Sci. | 58 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | Aoi | Elec. Eng. | 60 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | Bourikas | Elec. Eng. | 98 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | Bourikas | Elec. Eng. | 98 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | Tanaka | Biology | 120 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | Tanaka | Biology | 120 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.4** Result of *student* **natural left outer join** *takes*.

# More about Join Operation: **Right Outer Join (⟖)**

- A similar operation **right outer join**, denoted by ⟖, keeps every tuple in the second, or right, relation *S* in the result of *R* ⟖ *S*; if no matching tuple is found in *R*, then the attributes of R in the join result are filled or padded with NULL values.

| *ID* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|
| 00128 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.2** The *takes* relation.

| *ID* | *name* | *dept_name* | *tot_cred* |
|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 |
| 12345 | Shankar | Comp. Sci. | 32 |
| 19991 | Brandt | History | 80 |
| 23121 | Chavez | Finance | 110 |
| 44553 | Peltier | Physics | 56 |
| 45678 | Levy | Physics | 46 |
| 54321 | Williams | Comp. Sci. | 54 |
| 55739 | Sanchez | Music | 38 |
| 70557 | Snow | Physics | 0 |
| 76543 | Brown | Comp. Sci. | 58 |
| 76653 | Aoi | Elec. Eng. | 60 |
| 98765 | Bourikas | Elec. Eng. | 98 |
| 98988 | Tanaka | Biology | 120 |

**Figure 4.1** The *student* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* | *grade* | *name* | *dept_name* | *tot_cred* |
|---|---|---|---|---|---|---|---|---|
| 00128 | CS-101 | 1 | Fall | 2017 | A | Zhang | Comp. Sci. | 102 |
| 00128 | CS-347 | 1 | Fall | 2017 | A- | Zhang | Comp. Sci. | 102 |
| 12345 | CS-101 | 1 | Fall | 2017 | C | Shankar | Comp. Sci. | 32 |
| 12345 | CS-190 | 2 | Spring | 2017 | A | Shankar | Comp. Sci. | 32 |
| 12345 | CS-315 | 1 | Spring | 2018 | A | Shankar | Comp. Sci. | 32 |
| 12345 | CS-347 | 1 | Fall | 2017 | A | Shankar | Comp. Sci. | 32 |
| 19991 | HIS-351 | 1 | Spring | 2018 | B | Brandt | History | 80 |
| 23121 | FIN-201 | 1 | Spring | 2018 | C+ | Chavez | Finance | 110 |
| 44553 | PHY-101 | 1 | Fall | 2017 | B- | Peltier | Physics | 56 |
| 45678 | CS-101 | 1 | Fall | 2017 | F | Levy | Physics | 46 |
| 45678 | CS-101 | 1 | Spring | 2018 | B+ | Levy | Physics | 46 |
| 45678 | CS-319 | 1 | Spring | 2018 | B | Levy | Physics | 46 |
| 54321 | CS-101 | 1 | Fall | 2017 | A- | Williams | Comp. Sci. | 54 |
| 54321 | CS-190 | 2 | Spring | 2017 | B+ | Williams | Comp. Sci. | 54 |
| 55739 | MU-199 | 1 | Spring | 2018 | A- | Sanchez | Music | 38 |
| 70557 | *null* | *null* | *null* | *null* | *null* | Snow | Physics | 0 |
| 76543 | CS-101 | 1 | Fall | 2017 | A | Brown | Comp. Sci. | 58 |
| 76543 | CS-319 | 2 | Spring | 2018 | A | Brown | Comp. Sci. | 58 |
| 76653 | EE-181 | 1 | Spring | 2017 | C | Aoi | Elec. Eng. | 60 |
| 98765 | CS-101 | 1 | Fall | 2017 | C- | Bourikas | Elec. Eng. | 98 |
| 98765 | CS-315 | 1 | Spring | 2018 | B | Bourikas | Elec. Eng. | 98 |
| 98988 | BIO-101 | 1 | Summer | 2017 | A | Tanaka | Biology | 120 |
| 98988 | BIO-301 | 1 | Summer | 2018 | *null* | Tanaka | Biology | 120 |

**Figure 4.5** The result of *takes* **natural right outer join** *student*.

# More about Join Operation: **Right Outer Join (⟖)**

```
select *
from takes right outer join student on takes.ID=student.ID
```

| *ID* | *course_id* | *sec_id* | *semester* | *year* | *grade* |
|---|---|---|---|---|---|
| 00128 | CS-101 | 1 | Fall | 2017 | A |
| 00128 | CS-347 | 1 | Fall | 2017 | A- |
| 12345 | CS-101 | 1 | Fall | 2017 | C |
| 12345 | CS-190 | 2 | Spring | 2017 | A |
| 12345 | CS-315 | 1 | Spring | 2018 | A |
| 12345 | CS-347 | 1 | Fall | 2017 | A |
| 19991 | HIS-351 | 1 | Spring | 2018 | B |
| 23121 | FIN-201 | 1 | Spring | 2018 | C+ |
| 44553 | PHY-101 | 1 | Fall | 2017 | B- |
| 45678 | CS-101 | 1 | Fall | 2017 | F |
| 45678 | CS-101 | 1 | Spring | 2018 | B+ |
| 45678 | CS-319 | 1 | Spring | 2018 | B |
| 54321 | CS-101 | 1 | Fall | 2017 | A- |
| 54321 | CS-190 | 2 | Spring | 2017 | B+ |
| 55739 | MU-199 | 1 | Spring | 2018 | A- |
| 76543 | CS-101 | 1 | Fall | 2017 | A |
| 76543 | CS-319 | 2 | Spring | 2018 | A |
| 76653 | EE-181 | 1 | Spring | 2017 | C |
| 98765 | CS-101 | 1 | Fall | 2017 | C- |
| 98765 | CS-315 | 1 | Spring | 2018 | B |
| 98988 | BIO-101 | 1 | Summer | 2017 | A |
| 98988 | BIO-301 | 1 | Summer | 2018 | *null* |

**Figure 4.2** The *takes* relation.

| *ID* | *name* | *dept_name* | *tot_cred* |
|---|---|---|---|
| 00128 | Zhang | Comp. Sci. | 102 |
| 12345 | Shankar | Comp. Sci. | 32 |
| 19991 | Brandt | History | 80 |
| 23121 | Chavez | Finance | 110 |
| 44553 | Peltier | Physics | 56 |
| 45678 | Levy | Physics | 46 |
| 54321 | Williams | Comp. Sci. | 54 |
| 55739 | Sanchez | Music | 38 |
| 70557 | Snow | Physics | 0 |
| 76543 | Brown | Comp. Sci. | 58 |
| 76653 | Aoi | Elec. Eng. | 60 |
| 98765 | Bourikas | Elec. Eng. | 98 |
| 98988 | Tanaka | Biology | 120 |

**Figure 4.1** The *student* relation.

| *ID* | *course_id* | *sec_id* | *semester* | *year* | *grade* | *name* | *dept_name* | *tot_cred* |
|---|---|---|---|---|---|---|---|---|
| 00128 | CS-101 | 1 | Fall | 2017 | A | Zhang | Comp. Sci. | 102 |
| 00128 | CS-347 | 1 | Fall | 2017 | A- | Zhang | Comp. Sci. | 102 |
| 12345 | CS-101 | 1 | Fall | 2017 | C | Shankar | Comp. Sci. | 32 |
| 12345 | CS-190 | 2 | Spring | 2017 | A | Shankar | Comp. Sci. | 32 |
| 12345 | CS-315 | 1 | Spring | 2018 | A | Shankar | Comp. Sci. | 32 |
| 12345 | CS-347 | 1 | Fall | 2017 | A | Shankar | Comp. Sci. | 32 |
| 19991 | HIS-351 | 1 | Spring | 2018 | B | Brandt | History | 80 |
| 23121 | FIN-201 | 1 | Spring | 2018 | C+ | Chavez | Finance | 110 |
| 44553 | PHY-101 | 1 | Fall | 2017 | B- | Peltier | Physics | 56 |
| 45678 | CS-101 | 1 | Fall | 2017 | F | Levy | Physics | 46 |
| 45678 | CS-101 | 1 | Spring | 2018 | B+ | Levy | Physics | 46 |
| 45678 | CS-319 | 1 | Spring | 2018 | B | Levy | Physics | 46 |
| 54321 | CS-101 | 1 | Fall | 2017 | A- | Williams | Comp. Sci. | 54 |
| 54321 | CS-190 | 2 | Spring | 2017 | B+ | Williams | Comp. Sci. | 54 |
| 55739 | MU-199 | 1 | Spring | 2018 | A- | Sanchez | Music | 38 |
| 70557 | *null* | *null* | *null* | *null* | *null* | Snow | Physics | 0 |
| 76543 | CS-101 | 1 | Fall | 2017 | A | Brown | Comp. Sci. | 58 |
| 76543 | CS-319 | 2 | Spring | 2018 | A | Brown | Comp. Sci. | 58 |
| 76653 | EE-181 | 1 | Spring | 2017 | C | Aoi | Elec. Eng. | 60 |
| 98765 | CS-101 | 1 | Fall | 2017 | C- | Bourikas | Elec. Eng. | 98 |
| 98765 | CS-315 | 1 | Spring | 2018 | B | Bourikas | Elec. Eng. | 98 |
| 98988 | BIO-101 | 1 | Summer | 2017 | A | Tanaka | Biology | 120 |
| 98988 | BIO-301 | 1 | Summer | 2018 | *null* | Tanaka | Biology | 120 |

**Figure 4.5** The result of *takes* **natural right outer join** *student*.

# More about Join Operation: **Full Outer Join (⟗)**

- Finally, a **full outer join** denoted by ⟗, keeps all tuples in both the left and the right relations when no matching tuples are found, padding them with NULL values as needed.

**DEPARTMENT**

| Dname | Dnumber | Mgr_ssn | Mgr_start_date |
|---|---|---|---|
| Research | 5 | 333445555 | 1988-05-22 |
| Administration | 4 | 987654321 | 1995-01-01 |
| Headquarters | 1 | 888665555 | 1981-06-19 |

**DEPT_LOCATIONS**

| Dnumber | Dlocation |
|---|---|
| 1 | Houston |
| 4 | Stafford |
| 5 | Bellaire |
| 5 | Sugarland |
| 5 | Houston |

**DEPT_LOCS**

| Dname | Dnumber | Mgr_ssn | Mgr_start_date | Location |
|---|---|---|---|---|
| Headquarters | 1 | 888665555 | 1981-06-19 | Houston |
| Administration | 4 | 987654321 | 1995-01-01 | Stafford |
| Research | 5 | 333445555 | 1988-05-22 | Bellaire |
| Research | 5 | 333445555 | 1988-05-22 | Sugarland |
| Research | 5 | 333445555 | 1988-05-22 | Houston |

# More about Join Operation: **Full Outer Join (⟗)**

- Finally, a **full outer join** denoted by ⟗, keeps all tuples in both the left and the right relations when no matching tuples are found, padding them with NULL values as needed.

**DEPARTMENT**

| Dname | Dnumber | Mgr_ssn | Mgr_start_date |
|---|---|---|---|
| Research | 5 | 333445555 | 1988-05-22 |
| Administration | 4 | 987654321 | 1995-01-01 |
| Headquarters | 1 | 888665555 | 1981-06-19 |

Faculty 6 1111 1111 1994-01-01

**DEPT_LOCATIONS**

| Dnumber | Dlocation |
|---|---|
| 1 | Houston |
| 4 | Stafford |
| 5 | Bellaire |
| 5 | Sugarland |
| 5 | Houston |

7 Dallas
8 Austin

**DEPT_LOCS**

| Dname | Dnumber | Mgr_ssn | Mgr_start_date | Location |
|---|---|---|---|---|
| Headquarters | 1 | 888665555 | 1981-06-19 | Houston |
| Administration | 4 | 987654321 | 1995-01-01 | Stafford |
| Research | 5 | 333445555 | 1988-05-22 | Bellaire |
| Research | 5 | 333445555 | 1988-05-22 | Sugarland |
| Research | 5 | 333445555 | 1988-05-22 | Houston |

Faculty 6 1111 1994-01-01 Null
Null 7 Null Null Dallas
Null 8 Null Null Austin

# More about Join Operation: **Full Outer Join (⟗)**

```
select *
from department full outer join dept_locations on department.Dnumber=dept_locations.Dnumber
```

**DEPARTMENT**

| Dname | Dnumber | Mgr_ssn | Mgr_start_date |
|---|---|---|---|
| Research | 5 | 333445555 | 1988-05-22 |
| Administration | 4 | 987654321 | 1995-01-01 |
| Headquarters | 1 | 888665555 | 1981-06-19 |

**DEPT_LOCATIONS**

| Dnumber | Dlocation |
|---|---|
| 1 | Houston |
| 4 | Stafford |
| 5 | Bellaire |
| 5 | Sugarland |
| 5 | Houston |

**DEPT_LOCS**

| Dname | Dnumber | Mgr_ssn | Mgr_start_date | Location |
|---|---|---|---|---|
| Headquarters | 1 | 888665555 | 1981-06-19 | Houston |
| Administration | 4 | 987654321 | 1995-01-01 | Stafford |
| Research | 5 | 333445555 | 1988-05-22 | Bellaire |
| Research | 5 | 333445555 | 1988-05-22 | Sugarland |
| Research | 5 | 333445555 | 1988-05-22 | Houston |