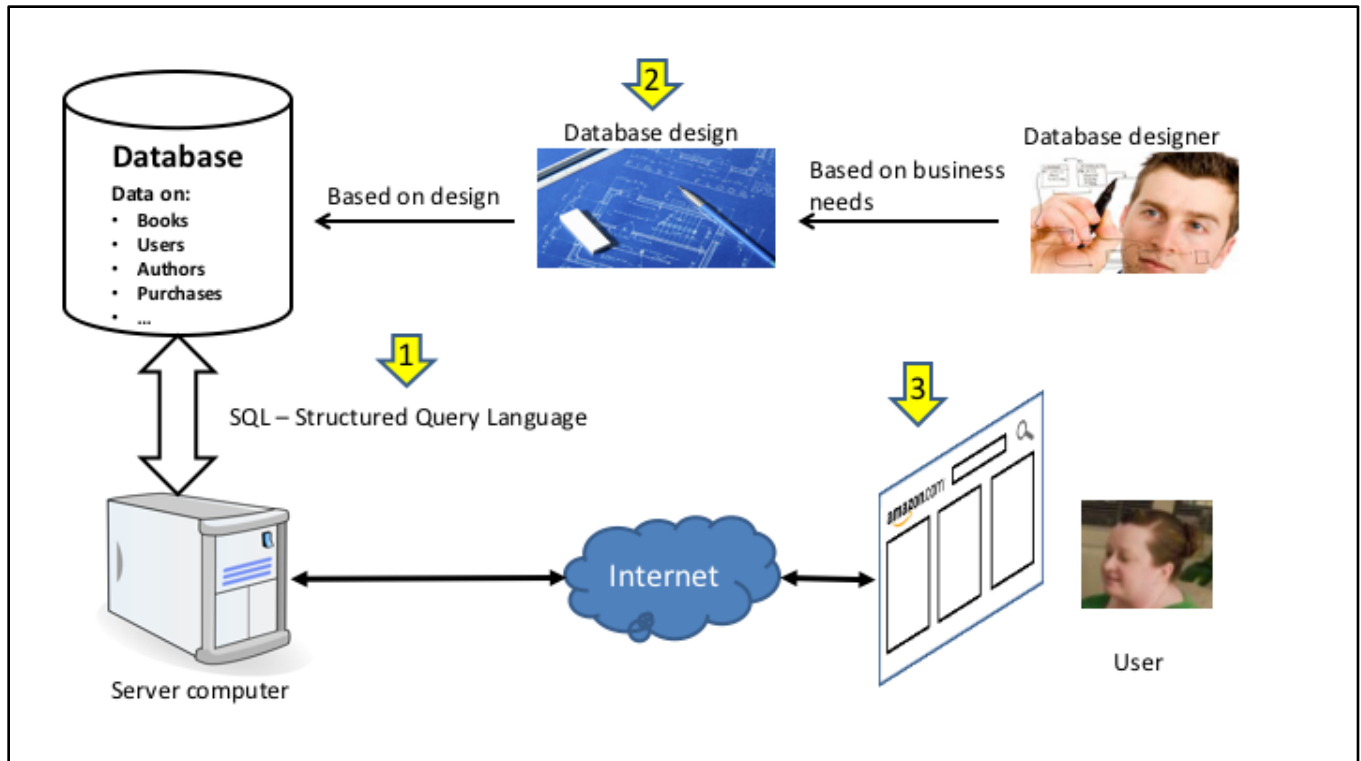


Business Information Modeling

Enterprise-wide Accounting Information Systems -- I

Week 4 – SQL: Part 3



We will use the college and sba databases for the examples

You already have these loaded and should try each query as we go along

Students

| student_id | firstname | lastname | height | weight | dob | state |
|------------|-----------|-------------|--------|--------|------------|-------|
| 1 | DARIUS | Brands | 76.2 | 197.5 | 6/8/1987 | MN |
| 2 | BRAIN | Kunsch | 74.3 | 185.4 | 7/26/1989 | MD |
| 3 | KENDRICK | Kafton | 72.9 | 181 | 2/21/1986 | VT |
| 4 | ARMAND | Lachat | 72.2 | 189.7 | 8/26/1991 | MT |
| 5 | ALI | Borio | 74.3 | 180.6 | 6/19/1989 | FL |
| 6 | JACKSON | Samay | 68.5 | 175 | 12/6/1985 | MN |
| 7 | DANE | Talahytewa | 69.9 | 199.4 | 3/2/1985 | CT |
| 8 | CLIFF | Casher | 73.9 | 201.7 | 1/22/1987 | ID |
| 9 | MARCEL | Jacot | 76.5 | 183.5 | 3/16/1990 | ND |
| 10 | MARCEL | Titze | 70 | 188.2 | 6/12/1985 | HI |
| 11 | RAPHAEL | Sitko | 72.9 | 196.4 | 9/29/1992 | ND |
| 12 | ALI | Aufderheide | 71.9 | 193.2 | 1/11/1984 | AL |
| 13 | CAREY | Dense | 73.7 | 180.8 | 5/16/1984 | FL |
| 14 | JACKSON | Vay | 73.7 | 186.3 | 7/4/1991 | NJ |
| 15 | BRAIN | Gattshall | 69.5 | 187 | 11/21/1986 | ND |
| 16 | KENDRICK | Burde | 73.2 | 202.9 | 4/12/1986 | FL |
| 17 | ALI | Dicapua | 70.8 | 183.1 | 3/16/1991 | ME |
| 18 | JACKSON | Aitcheson | 71.3 | 203.9 | 1/22/1989 | NH |
| 19 | MOISES | Nore | 72.1 | 175.3 | 11/23/1991 | ME |
| 20 | CLIFF | Shely | 72.4 | 196.3 | 4/15/1992 | OK |
| 21 | ALVARO | Billiott | 73.1 | 208.5 | 8/3/1985 | DE |
| 22 | JEFFRY | Pacitto | 72.6 | 184.9 | 1/16/1989 | NH |
| 23 | BRYON | Younie | 75.8 | 191.4 | 12/19/1990 | ME |
| 24 | JACKSON | Sherick | 70.9 | 186.9 | 9/25/1991 | FL |
| 25 | MOISES | Kopperman | 75.6 | 191.8 | 8/12/1988 | HI |

College Database

Sections

| course_id | section_name | instructor_id |
|-----------|--------------|---------------|
| 10 | AA | 10 |
| 10 | AB | 10 |
| 30 | AA | 20 |
| 30 | WB | 40 |
| 30 | AB | 20 |
| 40 | AA | 10 |

Instructors

| instructor_id | firstname | lastname | dob |
|---------------|-----------|----------|----------|
| 10 | DARIUS | Bokman | 7/8/1957 |
| 20 | CAREY | Mccament | 1/3/1984 |
| 30 | FIDEL | Blondell | 4/5/1976 |
| 40 | ELWOOD | Farb | 9/1/1986 |

Registrations

| course_id | section_name | student_id |
|-----------|--------------|------------|
| 10 | AA | 11 |
| 10 | AA | 1 |
| 10 | AA | 14 |
| 10 | AA | 18 |
| 10 | AA | 4 |
| 10 | AA | 15 |
| 10 | AB | 10 |
| 10 | AB | 17 |
| 10 | AB | 5 |
| 30 | WB | 13 |
| 30 | WB | 24 |
| 30 | WB | 22 |
| 30 | WB | 14 |
| 30 | WB | 19 |
| 30 | WB | 21 |
| 30 | WB | 17 |
| 30 | WB | 13 |

Courses

| course_id | course_name | credits |
|-----------|--|---------|
| 10 | African Paintings As A Feminist Genre | 1 |
| 20 | Life Of Russian Self-Actualization: A Process Approach | 3 |
| 30 | Race, Conflict, And Conflict In Polytheistic Music: Discovery, Development, and Interpretation | 4 |
| 40 | The Populist Dimension Of Southern Middle Eastern Literature | 2 |
| 50 | Masterpieces Of Middle Class Chinese Literature | 1 |

Schema for college Database

Table name

Column names

| | | | | | | | |
|-----------------|------------|------------|----------|--------|--------|-----|-------|
| Students | student_id | firstname, | lastname | height | weight | dob | state |
|-----------------|------------|------------|----------|--------|--------|-----|-------|

| | | | | |
|--------------------|---------------|-----------|----------|-----|
| Instructors | instructor_id | firstname | lastname | dob |
|--------------------|---------------|-----------|----------|-----|

| | | | |
|----------------|-----------|-------------|---------|
| Courses | course_id | course_name | credits |
|----------------|-----------|-------------|---------|

| | | | |
|-----------------|-----------|--------------|---------------|
| Sections | course_id | section_name | instructor_id |
|-----------------|-----------|--------------|---------------|

| | | | |
|----------------------|-----------|--------------|------------|
| Registrations | course_id | section_name | student_id |
|----------------------|-----------|--------------|------------|

Schema for SBA Database

Tables too big to show in entirety

Table name Column names

| | | | | | | | | | | |
|-----------------------|-----------|-------------------|------------------|---------------|-------------------|--------------------|--------------|------------|-------------|---------|
| Coaches | coach_id | coach_first_name | coach_last_name | | | | | | | |
| Venues | venue_id | venue_name | street_address | city | state | zip | capacity | year_built | | |
| Players | player_id | player_first_name | player_last_name | birth_date | height | weight | team_id | home_state | birth_state | quality |
| Teams | team_id | team_name | coach_id | home_venue_id | captain_id | | | | | |
| Games | game_id | first_team_id | second_team_id | venue_id | first_team_points | second_team_points | time_started | time_ended | attendance | |
| Participations | game_id | player_id | minutes | total_time | points | assists | fouls | | | |

| | | | | | | | |
|----------------------|---------------|--------------|---------------|--------|--------|-----|-------|
| Students | student_id | firstname | lastname | height | weight | dob | state |
| Instructors | instructor_id | firstname | lastname | dob | | | |
| Courses | course_id | course_name | credits | | | | |
| Sections | course_id | section_name | instructor_id | | | | |
| Registrations | course_id | section_name | student_id | | | | |

Query 46 (College): Get the first letter of the first name and the entire last name for each student.

```
SELECT SUBSTRING(firstname, 1, 1), lastname
FROM students
```

SUBSTRING(firstname, 1, 1)

Function for getting parts
of a textual field

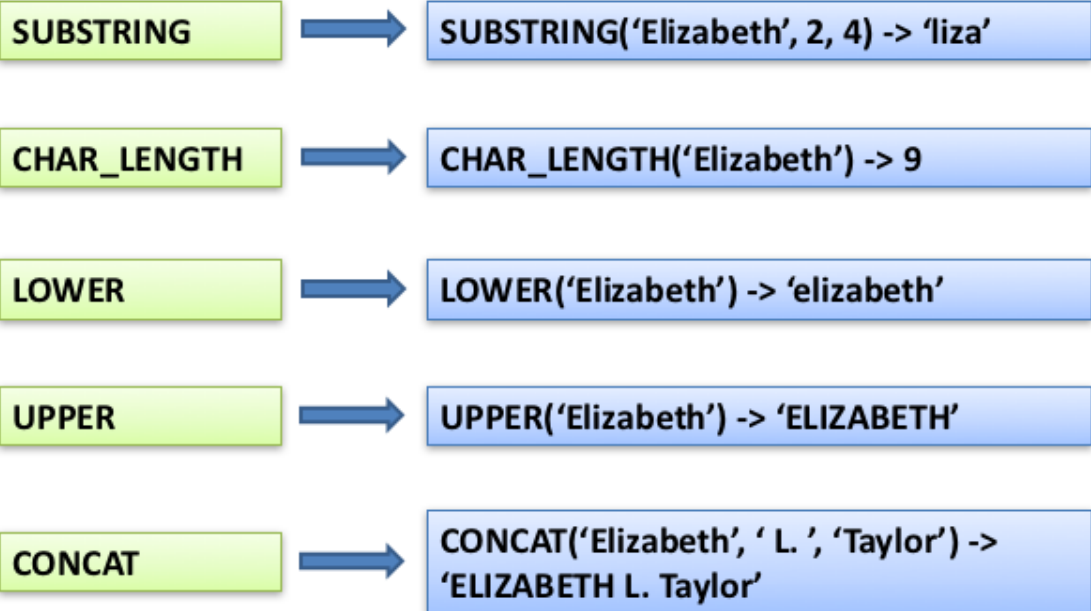
Field from which
substring will be obtained

Starting position and number of characters to
get from the field

SUBSTRING – aggregate function?

No Does not aggregate from multiple rows

**SUBSTRING
CHAR_LENGTH
CONCAT
LOWER
UPPER**



Query 46 (College, continued): Get the first letter of the first name and the entire last name for each student.

```
SELECT SUBSTRING(firstname, 1, 1),  
       lastname  
FROM students
```

Use AS to change column title
if needed.

| SUBSTRING(firstname, 1, 1) | lastname |
|----------------------------|-------------|
| D | Brands |
| B | Kunsch |
| K | Kafton |
| A | Lachat |
| A | Borio |
| J | Samzy |
| D | Talahytewa |
| C | Casher |
| M | Jacot |
| M | Titze |
| R | Sitko |
| A | Aufderheide |
| C | Dense |
| J | Vay |
| B | Gettshall |
| K | Burde |
| A | Dicapua |
| J | Aitcheson |
| M | Nore |
| C | Shely |
| A | Billiott |
| J | Pacitto |
| B | Younie |
| J | Sherick |
| M | Kopperman |

| Students | student_id | firstname | lastname | height | weight | dob | state |
|----------|------------|-----------|----------|--------|--------|-----|-------|
|----------|------------|-----------|----------|--------|--------|-----|-------|

| Instructors | instructor_id | firstname | lastname | dob |
|-------------|---------------|-----------|----------|-----|
|-------------|---------------|-----------|----------|-----|

| Courses | course_id | course_name | credits |
|---------|-----------|-------------|---------|
|---------|-----------|-------------|---------|

| Sections | course_id | section_name | instructor_id |
|----------|-----------|--------------|---------------|
|----------|-----------|--------------|---------------|

| Registrations | course_id | section_name | student_id |
|---------------|-----------|--------------|------------|
|---------------|-----------|--------------|------------|

Query 47 (College): List each alphabet and the number of students whose last name begins with the alphabet.

```
SELECT SUBSTRING(lastname, 1, 1) as Letter,
       COUNT(*) AS "# Students"
FROM students
GROUP BY Letter
ORDER BY Letter
```

| Letter | # Students |
|--------|------------|
| A | 2 |
| B | 4 |
| C | 1 |
| D | 2 |
| G | 1 |
| J | 1 |
| K | 3 |
| L | 1 |
| N | 1 |
| P | 1 |
| S | 4 |
| T | 2 |
| V | 1 |
| Y | 1 |

| | | | | | | | |
|----------------------|---------------|--------------|---------------|--------|--------|-----|-------|
| Students | student_id | firstname | lastname | height | weight | dob | state |
| Instructors | instructor_id | firstname | lastname | dob | | | |
| Courses | course_id | course_name | credits | | | | |
| Sections | course_id | section_name | instructor_id | | | | |
| Registrations | course_id | section_name | student_id | | | | |

Query 48 (College): List the first name, last name and height of students whose last name has more than 8 characters .

```
SELECT firstname, lastname, height
FROM students
WHERE CHAR_LENGTH(lastname) > 8
```

| firstname | lastname | height |
|-----------|-------------|--------|
| DANE | Talahytewa | 69.9 |
| ALI | Aufderheide | 71.9 |
| BRAIN | Gattshall | 69.5 |
| JACKSON | Aitcheson | 71.3 |
| MOISES | Kopperman | 75.6 |

| | | | | | | | | | | |
|-----------------------|-----------|-------------------|------------------|---------------|-------------------|--------------------|--------------|------------|-------------|---------|
| Coaches | coach_id | coach_first_name | coach_last_name | | | | | | | |
| Venues | venue_id | venue_name | street_address | city | state | zip | capacity | year_built | | |
| Players | player_id | player_first_name | player_last_name | birth_date | height | weight | team_id | home_state | birth_state | quality |
| Teams | team_id | team_name | coach_id | home_venue_id | captain_id | | | | | |
| Games | game_id | first_team_id | second_team_id | venue_id | first_team_points | second_team_points | time_started | time_ended | attendance | |
| Participations | game_id | player_id | minutes | total_time | points | assists | fouls | | | |

Query 49 : (SBA, Your turn) List all details of players whose first name contains “D” or whose last name contains “d”.

```
SELECT * FROM players
WHERE player_first_name LIKE '%D%'
OR player_last_name LIKE '%d%'
```

| | | | | | | | |
|----------------------|---------------|--------------|---------------|--------|--------|-----|-------|
| Students | student_id | firstname | lastname | height | weight | dob | state |
| Instructors | instructor_id | firstname | lastname | dob | | | |
| Courses | course_id | course_name | credits | | | | |
| Sections | course_id | section_name | instructor_id | | | | |
| Registrations | course_id | section_name | student_id | | | | |

More precision than we might care for ...

Query 50 (College): List the first name, last name and the height to weight ratio for each student. List the rows in descending order of the ratio.

```
SELECT firstname, lastname, height/weight
      AS hw_ratio
FROM students
ORDER BY hw_ratio DESC
```

| firstname | lastname | hw_ratio |
|-----------|-------------|-------------|
| MARCEL | Jacot | 0.416893733 |
| ALI | Borio | 0.411406423 |
| MOISES | Nore | 0.411294923 |
| CAREY | Dense | 0.407632743 |
| KENDRICK | Kafton | 0.402762431 |
| BRAIN | Kunsch | 0.400755124 |
| BRYON | Younie | 0.396029258 |
| JACKSON | Vay | 0.395598497 |
| MOISES | Kopperman | 0.394160584 |
| JEFFRY | Pacitto | 0.392644673 |
| JACKSON | Samay | 0.391428571 |
| ALI | Dicapua | 0.386673949 |
| DARIUS | Brands | 0.385822785 |
| ARMAND | Lachat | 0.380600949 |
| JACKSON | Sherick | 0.379347245 |
| ALI | Aufderheide | 0.372153209 |
| MARCEL | Titze | 0.37194474 |
| BRAIN | Gattshall | 0.371657754 |
| RAPHAEL | Sitko | 0.371181263 |
| CLIFF | Shely | 0.36882323 |
| CLIFF | Casher | 0.366385721 |
| KENDRICK | Burde | 0.360768852 |
| ALVARO | Billiott | 0.35059952 |
| DANE | Talahytewa | 0.350551655 |
| JACKSON | Aitcheson | 0.349681216 |

| | | | | | | | |
|----------------------|---------------|--------------|---------------|--------|--------|-----|-------|
| Students | student_id | firstname | lastname | height | weight | dob | state |
| Instructors | instructor_id | firstname | lastname | dob | | | |
| Courses | course_id | course_name | credits | | | | |
| Sections | course_id | section_name | instructor_id | | | | |
| Registrations | course_id | section_name | student_id | | | | |

Query 50 (College, continued): List the first name, last name and the height to weight ratio for each student. List the rows in descending order of the ratio.

```
SELECT firstname, lastname,
       ROUND(height/weight, 3) AS hw_ratio
FROM students
ORDER BY hw_ratio DESC
```

| firstname | lastname | hw_ratio |
|-----------|-------------|----------|
| MARCEL | Jacot | 0.417 |
| ALI | Borio | 0.411 |
| MOISES | Nore | 0.411 |
| CAREY | Dense | 0.408 |
| KENDRICK | Kafton | 0.403 |
| BRAIN | Kunsch | 0.401 |
| BRYON | Younie | 0.396 |
| JACKSON | Vay | 0.396 |
| MOISES | Kopperman | 0.394 |
| JEFFRY | Pacitto | 0.393 |
| JACKSON | Samay | 0.391 |
| ALI | Dicapua | 0.387 |
| DARIUS | Brands | 0.386 |
| ARMAND | Lachat | 0.381 |
| JACKSON | Sherick | 0.379 |
| MARCEL | Titze | 0.372 |
| BRAIN | Gattshall | 0.372 |
| ALI | Aufderheide | 0.372 |
| RAPHAEL | Sitko | 0.371 |
| CLIFF | Shely | 0.369 |
| CLIFF | Casher | 0.366 |
| KENDRICK | Burde | 0.361 |
| DANE | Talahytewa | 0.351 |
| ALVARO | Billiott | 0.351 |
| JACKSON | Aitcheson | 0.35 |

| | | | | | | | | | | | |
|-----------------------|-----------|-------------------|------------------|---------------|-------------------|--------------------|--------------|------------|-------------|---------|--|
| Coaches | coach_id | coach_first_name | coach_last_name | | | | | | | | |
| Venues | venue_id | venue_name | street_address | city | state | zip | capacity | year_built | | | |
| Players | player_id | player_first_name | player_last_name | birth_date | height | weight | team_id | home_state | birth_state | quality | |
| Teams | team_id | team_name | coach_id | home_venue_id | captain_id | | | | | | |
| Games | game_id | first_team_id | second_team_id | venue_id | first_team_points | second_team_points | time_started | time_ended | attendance | | |
| Participations | game_id | player_id | minutes | total_time | points | assists | fouls | | | | |

Query 51 : (SBA, Your turn) List the coach id and the total number of characters in the name (first and last name combined).

```
SELECT coach_id, CHAR_LENGTH(coach_first_name) +
        CHAR_LENGTH(coach_last_name) AS "Name length"
FROM coaches
```


| | | | | | | | | | | |
|-----------------------|-----------|-------------------|------------------|---------------|-------------------|--------------------|--------------|------------|-------------|---------|
| Coaches | coach_id | coach_first_name | coach_last_name | | | | | | | |
| Venues | venue_id | venue_name | street_address | city | state | zip | capacity | year_built | | |
| Players | player_id | player_first_name | player_last_name | birth_date | height | weight | team_id | home_state | birth_state | quality |
| Teams | team_id | team_name | coach_id | home_venue_id | captain_id | | | | | |
| Games | game_id | first_team_id | second_team_id | venue_id | first_team_points | second_team_points | time_started | time_ended | attendance | |
| Participations | game_id | player_id | minutes | total_time | points | assists | fouls | | | |

Query 52 : (SBA, Your turn) List each alphabet and the average height and weight of players whose last names start with each alphabet – round the averages to three decimal places.

```
SELECT SUBSTR(player_last_name, 1,1) AS letter,
       ROUND (AVG (height) , 3) , ROUND (AVG (weight) , 3)
FROM players
GROUP BY letter
```

Query 53 (College): For each section, list the course name, section_name and instructor_name.

| Sections | | | Courses | | | Instructors | | | |
|-----------|--------------|---------------|-----------|-----------------------|---------|---------------|-----------|----------|---------------|
| course_id | section_name | instructor_id | course_id | course_name | credits | instructor_id | firstname | lastname | dob |
| 10 | AA | 10 | 10 | African Paintings As | 1 | 10 | DARIUS | Bokman | 7/8/1957 0:00 |
| 10 | AB | 10 | 10 | African Paintings As | 1 | 10 | DARIUS | Bokman | 7/8/1957 0:00 |
| 30 | AA | 20 | 30 | Race, Conflict, And C | 4 | 20 | CAREY | Mccament | 1/3/1984 0:00 |
| 30 | AB | 20 | 30 | Race, Conflict, And C | 4 | 20 | CAREY | Mccament | 1/3/1984 0:00 |
| 30 | WB | 40 | 30 | Race, Conflict, And C | 4 | 40 | ELWOOD | Farb | 9/1/1986 0:00 |
| 40 | AA | 10 | 40 | The Populist Dimens | 2 | 10 | DARIUS | Bokman | 7/8/1957 0:00 |

```

SELECT course_name, section_name, firstname, lastname
FROM sections s
JOIN courses c ON s.course_id = c.course_id
JOIN instructors i ON s.instructor_id = i.instructor_id

```

Query 53 (College, continued): For each section, list the course name, section_name and instructor_name.

```
SELECT course_name, section_name, firstname, lastname
FROM sections s
JOIN courses c ON s.course_id = c.course_id
JOIN instructors i ON s.instructor_id = i.instructor_id
```

| course_name | section_name | firstname | lastname |
|------------------------|--------------|-----------|----------|
| African Paintings As / | AA | DARIUS | Bokman |
| African Paintings As / | AB | DARIUS | Bokman |
| Race, Conflict, And Cr | AA | CAREY | Mccament |
| Race, Conflict, And Cr | AB | CAREY | Mccament |
| Race, Conflict, And Cr | WB | ELWOOD | Farb |
| The Populist Dimensi | AA | DARIUS | Bokman |

Query 53 (College, another approach): For each section, list the course name, section_name and instructor_name.

| Sections | | | Courses | | | Instructors | | | |
|-----------|--------------|---------------|-----------|-----------------------|---------|---------------|-----------|----------|---------------|
| course_id | section_name | instructor_id | course_id | course_name | credits | instructor_id | firstname | lastname | dob |
| 10 | AA | 10 | 10 | African Paintings As | 1 | 10 | DARIUS | Bokman | 7/8/1957 0:00 |
| 10 | AB | 10 | 10 | African Paintings As | 1 | 10 | DARIUS | Bokman | 7/8/1957 0:00 |
| 30 | AA | 20 | 30 | Race, Conflict, And C | 4 | 20 | CAREY | Mccament | 1/3/1984 0:00 |
| 30 | AB | 20 | 30 | Race, Conflict, And C | 4 | 20 | CAREY | Mccament | 1/3/1984 0:00 |
| 30 | WB | 40 | 30 | Race, Conflict, And C | 4 | 40 | ELWOOD | Farb | 9/1/1986 0:00 |
| 40 | AA | 10 | 40 | The Populist Dimens | 2 | 10 | DARIUS | Bokman | 7/8/1957 0:00 |

```

SELECT course_name, section_name, firstname, lastname
FROM instructors i
JOIN sections s ON s.instructor_id = i.instructor_id
JOIN courses c ON s.course_id = c.course_id

```

Query 54 (College): For each section, list the course name, section_name and instructor_name. **For instructors who are not teaching any section, include just the names.**

```
SELECT course_name, section_name, firstname, lastname
FROM instructors i
LEFT JOIN sections s ON s.instructor_id = i.instructor_id
JOIN courses c ON s.course_id = c.course_id
```

| course_name | section_name | firstname | lastname |
|------------------------|--------------|-----------|----------|
| African Paintings As / | AA | DARIUS | Bokman |
| African Paintings As / | AB | DARIUS | Bokman |
| Race, Conflict, And C | AA | CAREY | Mccament |
| Race, Conflict, And C | AB | CAREY | Mccament |
| Race, Conflict, And C | WB | ELWOOD | Farb |
| The Populist Dimensi | AA | DARIUS | Bokman |

Why no FIDEL Blondell? Because of the LEFT JOIN FIDEL should he not have been included with NULL for all the other columns???

Query 54 (College, continued): For each section, list the course name, section_name and instructor_name. For instructors who are not teaching any section, include just the names.

```
SELECT course_name, section_name, firstname, lastname
FROM instructors i
LEFT JOIN sections s ON s.instructor_id = i.instructor_id
JOIN courses c ON s.course_id = c.course_id
```

First instructors LEFT JOIN sections

To the result of first join, JOIN courses

| Instructors | | | | Sections | | | Courses | | |
|---------------|-----------|----------|---------------|-----------|--------------|---------------|-----------|-----------------------|---------|
| instructor_id | firstname | lastname | dob | course_id | section_name | instructor_id | course_id | course_name | credits |
| 10 | DARIUS | Bokman | 7/8/1957 0:00 | 10 | AA | 10 | 10 | African Paintings As. | 1 |
| 10 | DARIUS | Bokman | 7/8/1957 0:00 | 10 | AB | 10 | 10 | African Paintings As. | 1 |
| 20 | CAREY | Mccament | 1/3/1984 0:00 | 30 | AA | 20 | 30 | Race, Conflict, And C | 4 |
| 20 | CAREY | Mccament | 1/3/1984 0:00 | 30 | AB | 20 | 30 | Race, Conflict, And C | 4 |
| 40 | ELWOOD | Farb | 9/1/1986 0:00 | 30 | WB | 40 | 30 | Race, Conflict, And C | 4 |
| 10 | DARIUS | Bokman | 7/8/1957 0:00 | 40 | AA | 10 | 40 | The Populist Dimens | 2 |
| 30 | FIDEL | Blondell | 4/5/1976 0:00 | NULL | NULL | NULL | | | |

No match for NULL course_id in courses table – hence row gets eliminated from result. Makes sense ...

Query 54 (College, continued): For each section, list the course name, section_name and instructor_name. For instructors who are not teaching any section, include just their name.

| Sections | | | Courses | | | Instructors | | | |
|-----------|--------------|---------------|-----------|-----------------------|---------|---------------|-----------|----------|---------------|
| course_id | section_name | instructor_id | course_id | course_name | credits | instructor_id | firstname | lastname | dob |
| 10 | AA | 10 | 10 | African Paintings As | 1 | 10 | DARIUS | Bokman | 7/8/1957 0:00 |
| 10 | AB | 10 | 10 | African Paintings As | 1 | 10 | DARIUS | Bokman | 7/8/1957 0:00 |
| 30 | AA | 20 | 30 | Race, Conflict, And C | 4 | 20 | CAREY | Mccament | 1/3/1984 0:00 |
| 30 | AB | 20 | 30 | Race, Conflict, And C | 4 | 20 | CAREY | Mccament | 1/3/1984 0:00 |
| 30 | WB | 40 | 30 | Race, Conflict, And C | 4 | 40 | ELWOOD | Farb | 9/1/1986 0:00 |
| 40 | AA | 10 | 40 | The Populist Dimens | 2 | 10 | DARIUS | Bokman | 7/8/1957 0:00 |
| NULL | NULL | NULL | NULL | NULL | NULL | 30 | FIDEL | Blondell | 4/5/1976 0:00 |

```

SELECT course_name, section_name, firstname, lastname
FROM sections s
JOIN courses c ON s.course_id = c.course_id
RIGHT JOIN instructors i ON s.instructor_id = i.instructor_id

```

Changing the JOIN order rescues us

Query 54 (College, another approach) : For each section, list the course name, section_name and instructor_name. For instructors who are not teaching any section, include just their name.

| Instructors | | | | Sections | | | Courses | | |
|---------------|-----------|----------|---------------|-----------|--------------|---------------|-----------|-----------------------|---------|
| instructor_id | firstname | lastname | dob | course_id | section_name | instructor_id | course_id | course_name | credits |
| 10 | DARIUS | Bokman | 7/8/1957 0:00 | 10 | AA | 10 | 10 | African Paintings As | 1 |
| 10 | DARIUS | Bokman | 7/8/1957 0:00 | 10 | AB | 10 | 10 | African Paintings As | 1 |
| 20 | CAREY | Mccament | 1/3/1984 0:00 | 30 | AA | 20 | 30 | Race, Conflict, And C | 4 |
| 20 | CAREY | Mccament | 1/3/1984 0:00 | 30 | AB | 20 | 30 | Race, Conflict, And C | 4 |
| 40 | ELWOOD | Farb | 9/1/1986 0:00 | 30 | WB | 40 | 30 | Race, Conflict, And C | 4 |
| 10 | DARIUS | Bokman | 7/8/1957 0:00 | 40 | AA | 10 | 40 | The Populist Dimens | 2 |
| 30 | FIDEL | Blondell | 4/5/1976 0:00 | NULL | NULL | NULL | NULL | NULL | NULL |

```

SELECT course_name, section_name, firstname, lastname
FROM instructors i
LEFT JOIN
    (courses c JOIN sections s ON s.course_id = c.course_id)
    ON s.instructor_id = i.instructor_id

```


| | | | | | | | | | | |
|-----------------------|-----------|-------------------|------------------|---------------|-------------------|--------------------|--------------|------------|-------------|---------|
| Coaches | coach_id | coach_first_name | coach_last_name | | | | | | | |
| Venues | venue_id | venue_name | street_address | city | state | zip | capacity | year_built | | |
| Players | player_id | player_first_name | player_last_name | birth_date | height | weight | team_id | home_state | birth_state | quality |
| Teams | team_id | team_name | coach_id | home_venue_id | captain_id | | | | | |
| Games | game_id | first_team_id | second_team_id | venue_id | first_team_points | second_team_points | time_started | time_ended | attendance | |
| Participations | game_id | player_id | minutes | total_time | points | assists | fouls | | | |

Query 55 (SBA, your turn) : List team names and the average, maximum and minimum weight of players in each team. List in ascending order of team names. Round the averages to two decimal places.

```

SELECT team_name, ROUND(AVG(weight), 2),
       ROUND(MAX(weight), 2), ROUND(MIN(weight), 2)
FROM teams t
JOIN players p ON p.team_id = t.team_id
GROUP BY team_name
ORDER BY team_name ASC

```

| | | | | | | | | | | |
|-----------------------|-----------|-------------------|------------------|---------------|-------------------|--------------------|--------------|------------|-------------|---------|
| Coaches | coach_id | coach_first_name | coach_last_name | | | | | | | |
| Venues | venue_id | venue_name | street_address | city | state | zip | capacity | year_built | | |
| Players | player_id | player_first_name | player_last_name | birth_date | height | weight | team_id | home_state | birth_state | quality |
| Teams | team_id | team_name | coach_id | home_venue_id | captain_id | | | | | |
| Games | game_id | first_team_id | second_team_id | venue_id | first_team_points | second_team_points | time_started | time_ended | attendance | |
| Participations | game_id | player_id | minutes | total_time | points | assists | fouls | | | |

Query 56 (SBA, your turn) : For each game, list the ids of the teams and the venue.

```
SELECT first_team_id, second_team_id, venue_name
FROM games g
JOIN venues v ON g.venue_id = v.venue_id
```

Can use OUTER JOIN if we want all venues even if no games were played there.

Query 57 (SBA) : For each game, list the game id and the names of the participating teams.

| Games table | | | | | Team table joined for first team | | | team table joined for first second team | | |
|-------------|---------------|----------------|----------|-----|-------------------------------------|------------------|-----|--|------------------|-----|
| game id | first team id | second team id | venue id | ... | team id | team name | ... | team id | team name | ... |
| 1 | 3 | 7 | 9 | ... | 3 | Raging Bulls | ... | 7 | Martyrs | ... |
| 2 | 2 | 6 | 8 | ... | 2 | Rockets | ... | 6 | Hilbillies | ... |
| 3 | 1 | 8 | 3 | ... | 1 | Orange Peels | ... | 8 | Patriots | ... |
| 4 | 7 | 9 | 1 | ... | 7 | Martyrs | ... | 9 | Marlins | ... |
| 5 | 2 | 3 | 4 | ... | 2 | Rockets | ... | 3 | Raging Bulls | ... |
| 6 | 4 | 8 | 6 | ... | 4 | Royal Tuskers | ... | 8 | Patriots | ... |
| 7 | 5 | 10 | 2 | ... | 5 | Towering Dunkers | ... | 10 | Electrics | ... |
| 8 | 9 | 10 | 2 | ... | 9 | Marlins | ... | 10 | Electrics | ... |
| 9 | 4 | 10 | 6 | ... | 4 | Royal Tuskers | ... | 10 | Electrics | ... |
| 10 | 4 | 5 | 7 | ... | 4 | Royal Tuskers | ... | 5 | Towering Dunkers | ... |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |

Same table joined more than once in the same query!

Query 57 (SBA, continued) : For each game, list the game id and the names of the participating teams.

Games table

| game id | first team id | second team id | venue id | ... |
|---------|---------------|----------------|----------|-----|
| 1 | 3 | 7 | 9 | ... |
| 2 | 2 | 6 | 8 | ... |
| 3 | 1 | 8 | 3 | ... |
| 4 | 7 | 9 | 1 | ... |
| 5 | 2 | 3 | 4 | ... |
| 6 | 4 | 8 | 6 | ... |
| 7 | 5 | 10 | 2 | ... |
| 8 | 9 | 10 | 2 | ... |
| 9 | 4 | 10 | 6 | ... |
| 10 | 4 | 5 | 7 | ... |
| ... | ... | ... | ... | ... |

Joined for first team

| team id | team name | ... |
|---------|------------------|-----|
| 3 | Raging Bulls | ... |
| 2 | Rockets | ... |
| 1 | Orange Peels | ... |
| 7 | Martyrs | ... |
| 2 | Rockets | ... |
| 4 | Royal Tuskers | ... |
| 5 | Towering Dunkers | ... |
| 9 | Marlins | ... |
| 4 | Royal Tuskers | ... |
| 4 | Royal Tuskers | ... |
| ... | ... | ... |

Joined for second team

| team id | team name | ... |
|---------|------------------|-----|
| 7 | Martyrs | ... |
| 6 | Hilbillies | ... |
| 8 | Patriots | ... |
| 9 | Marlins | ... |
| 3 | Raging Bulls | ... |
| 8 | Patriots | ... |
| 10 | Electrics | ... |
| 10 | Electrics | ... |
| 10 | Electrics | ... |
| 5 | Towering Dunkers | ... |
| ... | ... | ... |



| game id | first team name | second team name |
|---------|------------------|------------------|
| 1 | Raging Bulls | Martyrs |
| 2 | Rockets | Hilbillies |
| 3 | Orange Peels | Patriots |
| 4 | Martyrs | Marlins |
| 5 | Rockets | Raging Bulls |
| 6 | Royal Tuskers | Patriots |
| 7 | Towering Dunkers | Electrics |
| 8 | Marlins | Electrics |
| 9 | Royal Tuskers | Electrics |
| 10 | Royal Tuskers | Towering Dunkers |

Query 57 (SBA, continued): For each game, list the game id and the names of the participating teams.

g

| game_id | first_team_id | second_team_id | venue_id | ... |
|---------|---------------|----------------|----------|-----|
| 1 | 3 | 7 | 9 | ... |
| 2 | 2 | 6 | 8 | ... |
| 3 | 1 | 8 | 3 | ... |
| 4 | 7 | 9 | 1 | ... |
| 5 | 2 | 3 | 4 | ... |
| 6 | 4 | 8 | 6 | ... |
| 7 | 5 | 10 | 2 | ... |
| 8 | 9 | 10 | 2 | ... |
| 9 | 4 | 10 | 6 | ... |
| 10 | 4 | 5 | 7 | ... |
| ... | ... | ... | ... | ... |

t1

| team_id | team_name | ... |
|---------|------------------|-----|
| 3 | Raging Bulls | ... |
| 2 | Rockets | ... |
| 1 | Orange Peels | ... |
| 7 | Martyrs | ... |
| 2 | Rockets | ... |
| 4 | Royal Tuskers | ... |
| 5 | Towering Dunkers | ... |
| 9 | Marlins | ... |
| 4 | Royal Tuskers | ... |
| 4 | Royal Tuskers | ... |
| ... | ... | ... |

t2

| team_id | team_name | ... |
|---------|------------------|-----|
| 7 | Martyrs | ... |
| 6 | Hilbillies | ... |
| 8 | Patriots | ... |
| 9 | Marlins | ... |
| 3 | Raging Bulls | ... |
| 8 | Patriots | ... |
| 10 | Electrics | ... |
| 10 | Electrics | ... |
| 10 | Electrics | ... |
| 5 | Towering Dunkers | ... |
| ... | ... | ... |

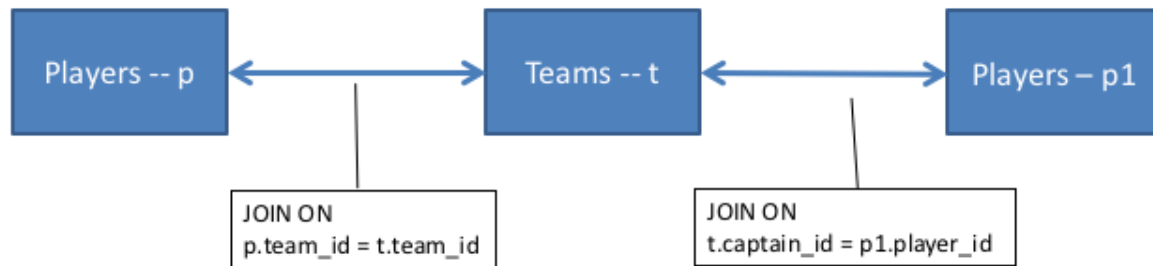
Query 57 (contd.): For each game, list the game id and the names of the participating teams.



```
SELECT g.game_id, t1.team_name, t2.team_name
FROM games g
JOIN teams t1 ON g.first_team_id = t1.team_id
JOIN teams t2 ON g.second_team_id = t2.team_id
```

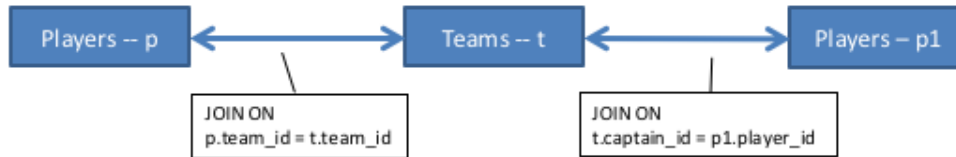
Query 58: Get a list of captains' first and last names (in ascending order of last names) along with the first and last name of the members of their teams.

| Players | player_id | player_first_name | player_last_name | birth_date | height | weight | team_id | home_state | birth_state | quality |
|---------|-----------|-------------------|------------------|---------------|------------|--------|---------|------------|-------------|---------|
| Teams | team_id | team_name | coach_id | home_venue_id | captain_id | | | | | |



Query 58 (contd.): Get a list of captains' first and last names (in ascending order of last names) along with the first and last name of the members of their teams.

| Players | player_id | player_first_name | player_last_name | birth_date | height | weight | team_id | home_state | birth_state | quality |
|---------|-----------|-------------------|------------------|---------------|------------|--------|---------|------------|-------------|---------|
| Teams | team_id | team_name | coach_id | home_venue_id | captain_id | | | | | |

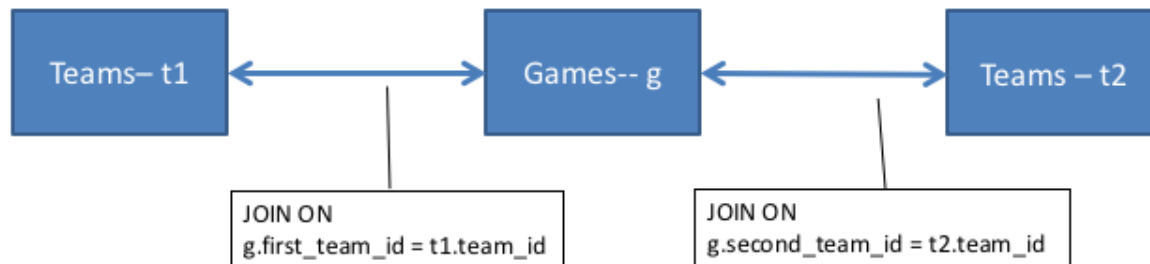


```

SELECT p.player_first_name, p.player_last_name,
       p1.player_first_name AS captain_first_name,
       p1.player_last_name AS captain_last_name
FROM players p
JOIN teams t ON p.team_id = t.team_id
JOIN players p1 on p1.player_id = t.captain_id
ORDER BY captain_first_name, captain_last_name
  
```


Query 59 (SBA): For each game list the names of the two teams that played and the name of the winning team.

| | | | | | | | | | |
|--------------|---------|---------------|----------------|---------------|-------------------|--------------------|--------------|------------|------------|
| Teams | team_id | team_name | coach_id | home_venue_id | captain_id | | | | |
| Games | game_id | first_team_id | second_team_id | venue_id | first_team_points | second_team_points | time_started | time_ended | attendance |



**Select the names of the two participating teams and then, ...
depending on the points scored, select the appropriate team name**

Query 59 (contd.): For each game list the names of the two teams that played and the name of the winning team.



```
SELECT t1.team_name, t2.team_name, ...  
FROM games g  
JOIN teams t1 ON g.first_team_id = t1.team_id  
JOIN teams t2 ON g.second_team_id = t2.team_id
```

Query 59 (contd.): For each game list the names of the two teams that played and the name of the winning team.

```
SELECT t1.team_name, t2.team_name, ...  
FROM games g  
JOIN teams t1 ON g.first_team_id = t1.team_id  
JOIN teams t2 ON g.second_team_id = t2.team_id
```

Depending on points, select the first team name or the second team name

Query 59 (contd.): For each game list the names of the two teams that played and the name of the winning team.

```
SELECT t1.team_name AS first_team, t2.team_name AS second_team,
```

```
CASE
```

```
  WHEN g.first_team_points >
```

```
    g.second_team_points THEN t1.team_name
```

```
  ELSE t2.team_name
```

```
END AS "Winning team"
```

```
FROM games g
```

```
JOIN teams t1 ON g.first_team_id = t1.team_id
```

```
JOIN teams t2 ON g.second_team_id = t2.team_id
```

Depending on points, select the first team name or the second team name

Query 60 (College): For each registration, list the student first and last name along with the course name, section name and instructor first and last name for the corresponding section.

| | | | | | | | |
|----------------------|---------------|--------------|---------------|--------|--------|-----|-------|
| Students | student_id | firstname | lastname | height | weight | dob | state |
| Instructors | instructor_id | firstname | lastname | dob | | | |
| Courses | course_id | course_name | credits | | | | |
| Sections | course_id | section_name | instructor_id | | | | |
| Registrations | course_id | section_name | student_id | | | | |

3. Then JOIN with instructor table for instructor name

2. JOIN with sections table to get instructor id

1. JOIN with students table for student name

This JOIN involves two columns!

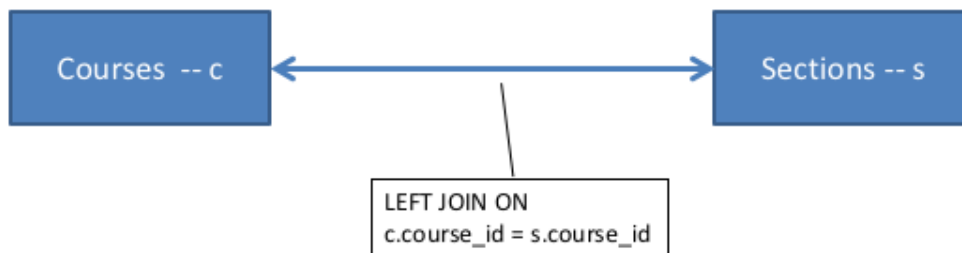
Query 60 (College, continued): For each registration, list the student first and last name along with the course name, section name and instructor first and last name for the corresponding section.

```
SELECT st.firstname, st.lastname, course_name,  
       s.section_name, i.firstname AS "Inst first name",  
       i.lastname AS "Inst last name"  
FROM courses c  
JOIN sections s ON c.course_id = s.course_id  
JOIN registrations r ON r.course_id = s.course_id  
                     AND r.section_name = s.section_name  
JOIN instructors i ON i.instructor_id = s.instructor_id  
JOIN students st ON st.student_id = r.student_id
```

Note how the JOIN is done on two columns ...

Query 61 (College): For each course, list the course name and the number of sections offered. For courses with no sections, show the number of sections as zero.

| | | | | | | | |
|----------------------|---------------|--------------|---------------|--------|--------|-----|-------|
| Students | student_id | firstname, | lastname | height | weight | dob | state |
| Instructors | instructor_id | firstname | lastname | dob | | | |
| Courses | course_id | course_name | credits | | | | |
| Sections | course_id | section_name | instructor_id | | | | |
| Registrations | course_id | section_name | student_id | | | | |



Query 61 (College, continued): For each course, list the course name and the number of sections offered. For courses with no sections, show the number of sections as zero.

courses c LEFT JOIN sections s ON c.course_id = s.course_id

| course_id | course_name | credits | course_id | section_name | instructor_id |
|-----------|----------------------------|---------|-----------|--------------|---------------|
| 10 | African Paintings As A Fer | 1 | 10 | AA | 10 |
| 10 | African Paintings As A Fer | 1 | 10 | AB | 10 |
| 20 | Life Of Russian Self-Actua | 3 | NULL | NULL | NULL |
| 30 | Race, Conflict, And Confli | 4 | 30 | AA | 20 |
| 30 | Race, Conflict, And Confli | 4 | 30 | AB | 20 |
| 30 | Race, Conflict, And Confli | 4 | 30 | WB | 40 |
| 40 | The Populist Dimension C | 2 | 40 | AA | 10 |
| 50 | Masterpieces Of Middle C | 1 | NULL | NULL | NULL |

```
SELECT course_name, COUNT(*)
FROM courses c
LEFT JOIN sections s
ON c.course_id = s.course_id
GROUP BY course_name
```

Two courses without any sections

| course_name | COUNT(*) |
|----------------------|----------|
| African Paintings A | 2 |
| Life Of Russian Seli | 1 |
| Masterpieces Of M | 1 |
| Race, Conflict, And | 3 |
| The Populist Dimer | 1 |



Query 61 (College, continued): For each course, list the course name and the number of sections offered. For courses with no sections, show the number of sections as zero.

```
SELECT course_name, COUNT(*)  
FROM courses c  
LEFT JOIN sections s  
  ON c.course_id = s.course_id  
GROUP BY course_name
```



| course_name | COUNT(*) |
|----------------------|----------|
| African Paintings A | 2 |
| Life Of Russian Self | 1 |
| Masterpieces Of M | 1 |
| Race, Conflict, And | 3 |
| The Populist Dime | 1 |

```
SELECT course_name,  
       COUNT(s.course_id)  
FROM courses c  
LEFT JOIN sections s  
  ON c.course_id = s.course_id  
GROUP BY course_name
```



| course_name | COUNT(*) |
|----------------------|----------|
| African Paintings A | 2 |
| Life Of Russian Self | 0 |
| Masterpieces Of M | 0 |
| Race, Conflict, And | 3 |
| The Populist Dime | 1 |

Query 61 (College, continued): For each course, list the course name and the number of sections offered. For courses with no sections, show the number of sections as zero.

```
SELECT course_name, COUNT(s.course_id)
FROM courses c
LEFT JOIN sections s
      ON c.course_id = s.course_id
GROUP BY course_name
```

We can use any column from the sections table in the COUNT(). All of these will be NULL and hence not counted.

COUNT (*)



Counts number of
rows

COUNT (s.course_id)



Counts number of
non-NULL values for
column mentioned