CSC 337

Homer Simpson
Nancy Cartwr...
Dan Castellan...

Bart Simpson

Marge Simpson
Julie Kavner

Lisa Simpson
Yeardley Smith

Multi-table SQL Queries (Joins)

Rick Mercer, Marty Stepp

Related tables and keys for Database simpsons

id	name	email
123	Bart	bart@fox.com
456	Milhouse	milhouse@fox.com
888	Lisa	lisa@fox.com
404	Ralph	ralph@fox.com

students

id	name
1234	Krabappel
5678	Hoover
9012	Obourn

teachers

id	name	teacher_id
10001	Computer Science 142	1234
10002	Computer Science 143	5678
10003	Computer Science 154	9012
10004	Informatics 100	1234

courses

student_id	course_id	grade
123	10001	B-
123	10002	C
456	10001	B+
888	10002	A+
888	10003	A+
404	10004	D+

grades

- **Primary key**: a column guaranteed to be unique for each record
  - e.g. Lisa's ID 888
- **Foreign key**: a column in table A storing a primary key value from table B
  - (e.g. records in `grades` with `student_id` of 888 are Lisa's grades)
- **Normalizing** splitting tables to improve structure / redundancy (linked by unique IDs)

SQL Joins

- A SQL join is a Structured Query Language (**SQL**) instruction to combine data from two or more sets of data (two or more tables)

- There are several types of joins
  - We'll use JOIN to selects records that have matching values in both tables
  - Can match the primary key of one table with foreign key (s) in another table

General Form

SELECT column-names FROM table-name1
JOIN table-name2
ON column-name1 = column-name2
WHERE condition

- **JOIN** combines records from 2 or more tables that satisfy certain conditions
  - Creates a temporary table
- The **ON** clause specifies which records from each table are matched
- The rows are often linked by their key columns (id)
- The WHERE clause is optional

Easier to qualify table names

SELECT table1.col2, table2.col2, table1.col3 FROM table1name
JOIN table2name
ON table1. col1 = table2. col1
WHERE condition

student_id	course_id	grade
123	10001	B-
123	10002	C
404	10002	B
404	10004	D+
456	10001	B+
456	10002	D-
888	10002	A+
888	10003	A+

grades

- Example

```
SELECT students.name, grades.course_id, grades.grade FROM students
JOIN grades
ON students.id = grades.student_id
WHERE students.name = 'Bart';
```

name	course_id	grade
Bart	10001	B-
Bart	10002	C

id	name	email	password
123	Bart	bart@fox.com	bartman
456	Milhouse	milhouse@fox.com	fallout
888	Lisa	lisa@fox.com	vegan
404	Ralph	ralph@fox.com	catfood

students

Filtered JOIN with ON and WHERE

student_id	course_id	grade
123	10001	B-
123	10002	C
404	10002	B
404	10004	D+
456	10001	B+
456	10002	D-
888	10002	A+
888	10003	A+

grades

```
SELECT students.name, grades.course_id, grades.grade FROM students
JOIN grades
ON students.id = grades.student_id
WHERE students.name = 'Bart';

+------+-----------+-------+
| name | course_id | grade |
+------+-----------+-------+
| Bart |     10001 | B-    |
| Bart |     10002 | C     |
+------+-----------+-------+
```

id	name	email	password
123	Bart	bart@fox.com	bartman
456	Milhouse	milhouse@fox.com	fallout
888	Lisa	lisa@fox.com	vegan
404	Ralph	ralph@fox.com	catfood

students

- FROM / JOIN glue tables together, WHERE filters results
- What goes in the ON clause, and what goes in WHERE?
  - ON directly links columns of the joined tables
  - WHERE sets additional constraints such as particular values (123, 'Bart')

Join Data from Two Tables

```
select * from grades;
+------------+-----------+-------+
| student_id | course_id | grade |
+------------+-----------+-------+
|        123 |     10001 | B-    |
|        123 |     10002 | C     |
|        456 |     10001 | B+    |
|        888 |     10002 | A+    |
|        888 |     10003 | A+    |
|        404 |     10004 | D+    |
|        404 |     10002 | B     |
|        456 |     10002 | D-    |
+------------+-----------+-------+
```

SELECT column-names FROM table-name1
JOIN table-name2
ON column-name1 = column-name2
WHERE condition

```
select * from students;
+-----+----------+-------------------+----------+
| id  | name     | email             | password |
+-----+----------+-------------------+----------+
| 123 | Bart     | bart@fox.com      | bartman  |
| 404 | Ralph    | ralph@fox.com     | catfood  |
| 456 | Milhouse | milhouse@fox.com  | fallout  |
| 888 | Lisa     | lisa@fox.com      | vegan    |
+-----+----------+-------------------+----------+
```

- Let's get the student's names and grades where the grades are >= 'B'
  - How many will there be?
  - What is the 1st record stored in a new table?

Answer and result

```
SELECT students.name, grades.grade FROM students
JOIN grades
ON students.id = grades.student_id
WHERE grades.grade > 'B';

+----------+-------+
| name     | grade |
+----------+-------+
| Bart     | B-    |
| Bart     | C     |
| Ralph    | D+    |
| Milhouse | B+    |
| Milhouse | D-    |
+----------+-------+
```

JOIN data in grades and courses

```
select * from grades;
+------------+-----------+-------+
| student_id | course_id | grade |
+------------+-----------+-------+
|        123 |     10001 | B-    |
|        123 |     10002 | C     |
|        456 |     10001 | B+    |
|        888 |     10002 | A+    |
|        888 |     10003 | A+    |
|        404 |     10004 | D+    |
|        404 |     10002 | B     |
|        456 |     10002 | D-    |
+------------+-----------+-------+
```

SELECT column-names FROM table-name1
JOIN table-name2
ON column-name1 = column-name2
WHERE condition

```
select * from courses;
+-------+-----------------------+------------+
| id    | name                  | teacher_id |
+-------+-----------------------+------------+
| 10001 | Computer Science 142  |       1234 |
| 10002 | Computer Science 143  |       5678 |
| 10003 | Computer Science 190M |       9012 |
| 10004 | Informatics 100       |       1234 |
+-------+-----------------------+------------+
```

- All grades for Computer Science 143
  - id is 10001
  - join data from tables `courses` and `grades`
  - show `grades.student_id` `courses.name` and `grades.grade`

Answer and result

```
SELECT courses.name, grades.grade, grades.student_id FROM courses
JOIN grades
ON courses.id = grades.course_id
WHERE courses.id = '10001';
```

```
+------------------------+-------+------------+
| name                   | grade | student_id |
+------------------------+-------+------------+
| Computer Science 142   | B-    |        123 |
| Computer Science 142   | B+    |        456 |
+------------------------+-------+------------+
```

Can we switch the courses and grades tables?

WHERE is Optional

```
select * from grades;
+------------+-----------+-------+
| student_id | course_id | grade |
+------------+-----------+-------+
|        123 |     10001 | B-    |
|        123 |     10002 | C     |
|        456 |     10001 | B+    |
|        888 |     10002 | A+    |
+------------+-----------+-------+
```

```
select * from students;
+-----+----------+------------------+----------+
| id  | name     | email            | password |
+-----+----------+------------------+----------+
| 123 | Bart     | bart@fox.com     | bartman  |
| 404 | Ralph    | ralph@fox.com    | catfood  |
| 456 | Milhouse | milhouse@fox.com | fallout  |
| 888 | Lisa     | lisa@fox.com     | vegan    |
+-----+----------+------------------+----------+
```

- Get all grades for everyone

```
SELECT * FROM grades
JOIN students
ON students.id = grades.student_id;
```

id	name	email	password	student_id	course_id	grade
123	Bart	bart@fox.com	bartman	123	10001	B-
123	Bart	bart@fox.com	bartman	123	10002	C
404	Ralph	ralph@fox.com	catfood	404	10004	D+
404	Ralph	ralph@fox.com	catfood	404	10002	B
456	Milhouse	milhouse@fox.com	fallout	456	10001	B+
456	Milhouse	milhouse@fox.com	fallout	456	10002	D-
888	Lisa	lisa@fox.com	vegan	888	10002	A+
888	Lisa	lisa@fox.com	vegan	888	10003	A+

Querying multi-table databases

- When we have larger datasets spread across multiple tables, we need queries that can answer high-level questions such as:
  - What courses has Bart taken and gotten a B- or better?
  - What courses have been taken by both Bart and Lisa?
  - Who are all the teachers Bart has had?
  - How many total students has Ms. Krabappel taught, and what are their names?
- To do this, we'll have to **JOIN** data from several tables
  - Add more JOIN clauses

Multi-way join

```
SELECT students.name, courses.name, grades.grade FROM courses
JOIN grades ON grades.course_id = courses.id
JOIN students ON grades.student_id = students.id
WHERE grades.grade <= 'B-';
```

- More than two tables can be joined
- Data from which tables are joined?
  - _____
- What does the above query represent?
  - _____

What MariaDB would look like

```
MariaDB [simpsons]> SELECT students.name, courses.name, grades.grade FROM courses
    -> JOIN grades ON grades.course_id = courses.id
    -> JOIN students ON grades.student_id = students.id
    -> WHERE grades.grade <= 'B-';
+----------+----------------------+-------+
| name     | name                 | grade |
+----------+----------------------+-------+
| Bart     | Computer Science 142  | B-    |
| Ralph    | Computer Science 143  | B     |
| Milhouse | Computer Science 142  | B+    |
| Lisa     | Computer Science 143  | A+    |
| Lisa     | Computer Science 190M | A+    |
+----------+----------------------+-------+
5 rows in set (0.00 sec)
```

Designing a query:

- Figure out the proper SQL queries in the following way:
  - Which columns do I want in the result set? (`SELECT`)
  - Which table(s) contain the critical data? (`FROM`)
  - How are tables connected (`JOIN`)
  - How are values filtered (`WHERE`)?
- Test on a real data base in MariaDB (`imdb_small`)
- Write the PHP code to run those same queries
  - Fix all of those syntax errors

Our Lab 10 database: imdb

id	first_name	last_name	gender
433259	William	Shatner	M
797926	Britney	Spears	F
831289	Sigourney	Weaver	F
...			

actors

id	name	year	rank
112290	Fight Club	1999	8.5
209658	Meet the Parents	2000	7
210511	Memento	2000	8.7
...			

movies

actor_id	movie_id	role
433259	313398	Capt. James T. Kirk
433259	407323	Sgt. T.J. Hooker
797926	342189	Herself
...		

roles

movie_id	genre
209658	Comedy
313398	Action
313398	Sci-Fi
...	

movies_genres

id	first_name	last_name
24758	David	Fincher
66965	Jay	Roach
72723	William	Shatner
...		

directors

director_id	movie_id
24758	112290
66965	209658
72723	313398
...	

movies_directors

IMDb table relationships / ids
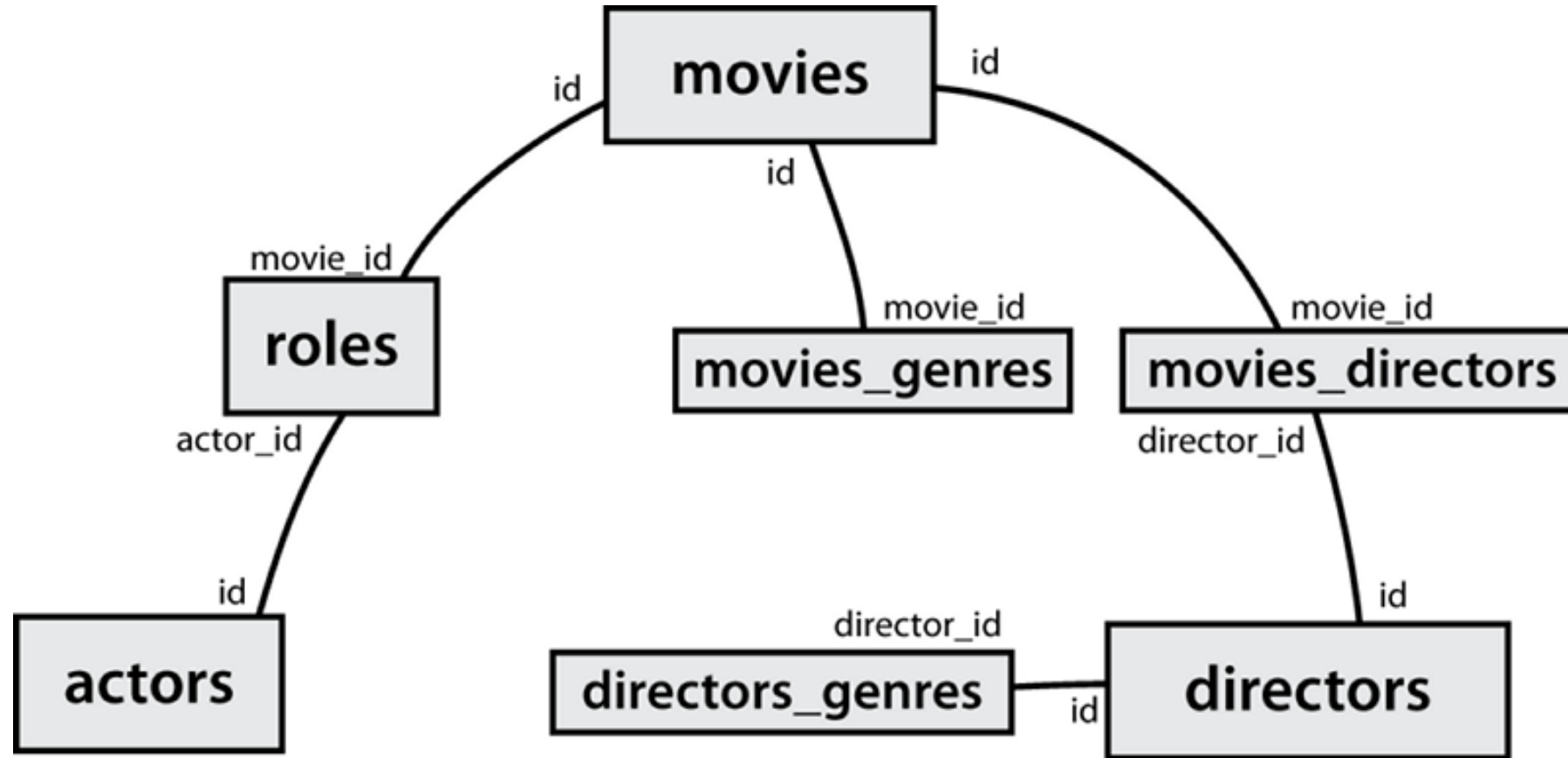
Our other database: imdb_small

id	first_name	last_name	gender
433259	William	Shatner	M
797926	Britney	Spears	F
831289	Sigourney	Weaver	F
...			

actors

id	name	year	rank
112290	Fight Club	1999	8.5
209658	Meet the Parents	2000	7
210511	Memento	2000	8.7
...			

movies

actor_id	movie_id	role
433259	313398	Capt. James T. Kirk
433259	407323	Sgt. T.J. Hooker
797926	342189	Herself
...		

roles

movie_id	genre
209658	Comedy
313398	Action
313398	Sci-Fi
...	

movies_genres

id	first_name	last_name
24758	David	Fincher
66965	Jay	Roach
72723	William	Shatner
...		

directors

director_id	movie_id
24758	112290
66965	209658
72723	313398
...	

movies_directors

Possible queries

- How many people played a part in the movie "Lost in Translation"?
- What are the *names* of all the people who played a part in the movie "Lost in Translation"?
- What are the *names* of all movies Clint Eastwood has directed?
- What are the names of every actor who has appeared in a movie directed by Christopher Nolan?
- Practice with Marty Stepp's CSE 154 Query Tester
  - http://www.martystepp.com/query/?username=cse154&password=cse154
  - It can be a bit flaky