CS 327E Lecture 6

Shirley Cohen

February 10, 2016

Agenda

- Announcements
- Readings for today
- Reading Quiz
- Concept Questions
- Homework for next time

Announcements

- Reminder: Midterm #1 is next Wednesday
- Short review on Monday

Homework for Today

- Chapter 8 from the Learning SQL book
- Exercises at the end of Chapter 8

Quiz Question 1

```
mysql> select * from employee;
+--------+----------+-----------+-----------------+
| emp_id | fname    | lname     | superior_emp_id |
+--------+----------+-----------+-----------------+
|      1 | Michael  | Smith     |            NULL |
|      2 | Susan    | Barker    |               1 |
|      3 | Robert   | Tyler     |               1 |
|      4 | Susan    | Hawthorne |               3 |
|      5 | John     | Gooding   |               4 |
+--------+----------+-----------+-----------------+
```

What **value** does the following query return?

```
SELECT COUNT(*) FROM employee;
```

A. 1 B. 3 C. 5 D. 4

Quiz Question 2

```
mysql> select * from employee;
+--------+----------+-----------+-----------------+
| emp_id | fname    | lname     | superior_emp_id |
+--------+----------+-----------+-----------------+
|      1 | Michael  | Smith     |            NULL |
|      2 | Susan    | Barker    |               1 |
|      3 | Robert   | Tyler     |               1 |
|      4 | Susan    | Hawthorne |               3 |
|      5 | John     | Gooding   |               4 |
+--------+----------+-----------+-----------------+
```

What **value** does the following query?

```
SELECT COUNT(superior_emp_id)
FROM employee;
```

A. 1 B. 2 C. 3 D. 4

Quiz Question 3

```
mysql> select * from employee;
+--------+----------+-----------+-----------------+
| emp_id | fname    | lname     | superior_emp_id |
+--------+----------+-----------+-----------------+
|      1 | Michael  | Smith     |            NULL |
|      2 | Susan    | Barker    |               1 |
|      3 | Robert   | Tyler     |               1 |
|      4 | Susan    | Hawthorne |               3 |
|      5 | John     | Gooding   |               4 |
+--------+----------+-----------+-----------------+
```

How many **groups** does the following query return?

```
SELECT superior_emp_id, COUNT(*)
FROM employee
GROUP BY superior_emp_id;
```

A. 0 B. 3 C. 4 D. 5

Quiz Question 4

```
mysql> select * from employee;
+--------+----------+-----------+-----------------+
| emp_id | fname    | lname     | superior_emp_id |
+--------+----------+-----------+-----------------+
|      1 | Michael  | Smith     |            NULL |
|      2 | Susan    | Barker    |               1 |
|      3 | Robert   | Tyler     |               1 |
|      4 | Susan    | Hawthorne |               3 |
|      5 | John     | Gooding   |               4 |
+--------+----------+-----------+-----------------+
```

What many **groups** does the following query return?

```
SELECT superior_emp_id, COUNT(*)
FROM employee
WHERE COUNT(*) > 1
GROUP BY superior_emp_id;
```

A. 1 B. 3 C. 4 D. N/A. The query is syntactically incorrect

Quiz Question 5

Which of the following statements regarding `NULL` values is true?

A. All aggregate functions omit rows containing `NULL` values.
B. All aggregate functions replace `NULL` values with `0` before the performing the appropriate computation.
C. Attempting to execute a query with an aggregate function and `NULL` values results in a syntax error.
D. None of the above.

Concept Question 1

Recall our retail store database. It keeps product details in the `SKU_Data` table that is shown below. How can we find out the number of different departments that have a product in this table?

A.
```
SELECT COUNT(*)
FROM Department
```

B.
```
SELECT COUNT(*)
FROM SKU_Data
WHERE Department IS NOT NULL
```

C.
```
SELECT COUNT(Department)
FROM SKU_Data
```

D.
```
SELECT COUNT(DISTINCT Department)
FROM SKU_Data
```

E. None of the above

SKU_Data (SKU, SKU_Description, Department)

SELECT * FROM SKU_Data

SKU	SKU_Description	Department
100100	Std. Scuba Tank, Yellow	Water Sports
100200	Std. Scuba Tank, Magenta	Water Sports
101100	Dive Mask, Small Clear	Water Sports
101200	Dive Mask, Med Clear	NULL
201000	Half-dome Tent	Camping
202000	Half-dome Tent Vestibule	Camping
301000	Light Fly Climbing Harness	Climbing
302000	Locking carabiner, Oval	NULL

Concept Question 2

We have the same `SKU_Data` table as before. Now we want to generate a more user-friendly report that shows the name of each department along with the number of products that it sells.

A.
```
SELECT Department, COUNT(*)
FROM SKU_Data
GROUP BY Department
```

B.
```
SELECT Department, COUNT(*)
FROM SKU_Data
```

C.
```
SELECT Department,
COUNT(Department)
FROM SKU_Data
```

D.
```
SELECT Department,
COUNT(Department)
FROM SKU_Data
GROUP BY Department
```

E. None of the above

SKU_Data (SKU, SKU_Description, Department)

SELECT * FROM SKU_Data

SKU	SKU_Description	Department
100100	Std. Scuba Tank, Yellow	Water Sports
100200	Std. Scuba Tank, Magenta	Water Sports
101100	Dive Mask, Small Clear	Water Sports
101200	Dive Mask, Med Clear	NULL
201000	Half-dome Tent	Camping
202000	Half-dome Tent Vestibule	Camping
301000	Light Fly Climbing Harness	Climbing
302000	Locking carabiner, Oval	NULL

Concept Question 3

We want to extend the previous report to include the `SKU_Description` field. That is, we would like to display the `SKU_Description` alongside the department name while still grouping by department. Can this be done with a select-from-group-by query?

A.
```
SELECT Department,
SKU_Description, COUNT(*)
FROM SKU_Data
GROUP BY Department
```

B.
```
SELECT Department,
COUNT(Department),
SKU_Description
FROM SKU_Data
GROUP BY Department
```

C.
```
SELECT Department,
SKU_Description, COUNT(*)
FROM SKU_Data
GROUP BY Department,
SKU_Description
```

D. None of the above

SKU_Data (SKU, SKU_Description, Department)

SELECT * FROM SKU_Data

SKU	SKU_Description	Department
100100	Std. Scuba Tank, Yellow	Water Sports
100200	Std. Scuba Tank, Magenta	Water Sports
101100	Dive Mask, Small Clear	Water Sports
101200	Dive Mask, Med Clear	NULL
201000	Half-dome Tent	Camping
202000	Half-dome Tent Vestibule	Camping
301000	Light Fly Climbing Harness	Climbing
302000	Locking carabiner, Oval	NULL

Concept Question 4

What's wrong with this query?

```
SELECT bp.product_id, MAX(b.date_reported), b.bug_id
FROM BugsProducts bp JOIN Bugs b USING (bug_id)
GROUP BY bp.product_id
```

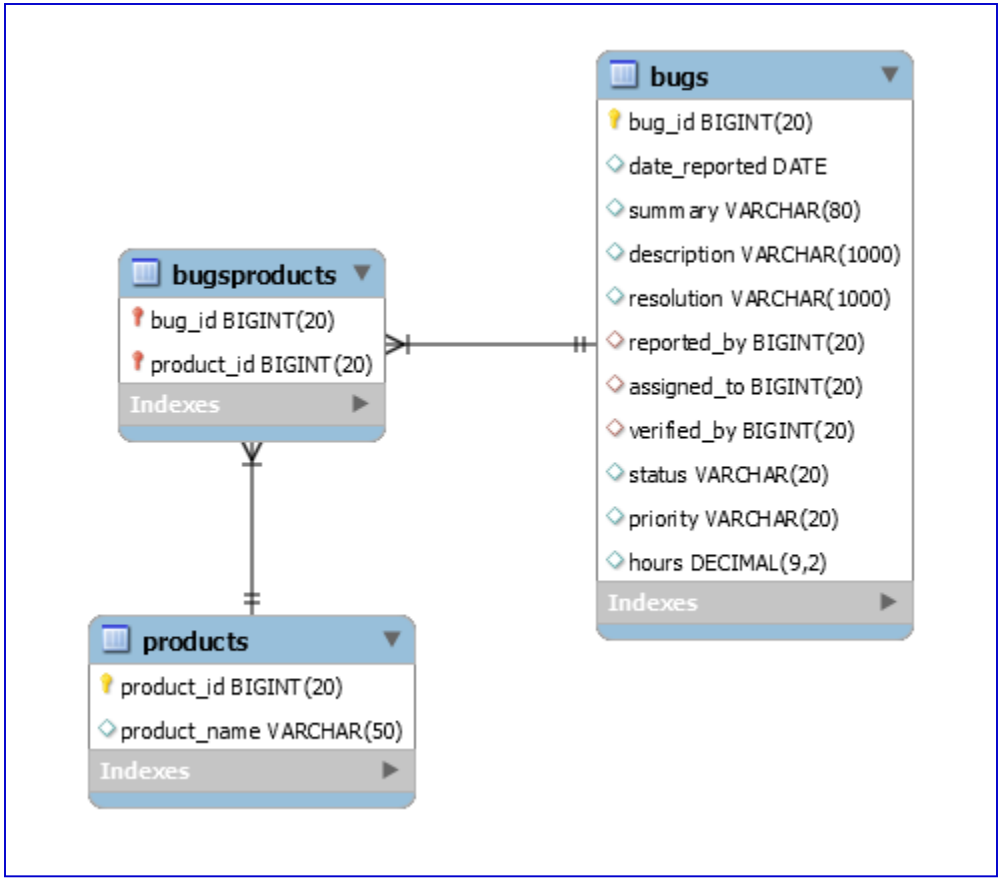
A. MAX(b.date_reported)

B. USING (bug_id)

C. GROUP BY bp.product_id

D. b.bug_id

E. None of the above



Concept Question 5

How can we fix this query to include the products that have no bugs?

```
SELECT bp.product_id, MAX(b.date_reported)
FROM BugsProducts bp JOIN Bugs b USING (bug_id)
GROUP BY bp.product_id
```

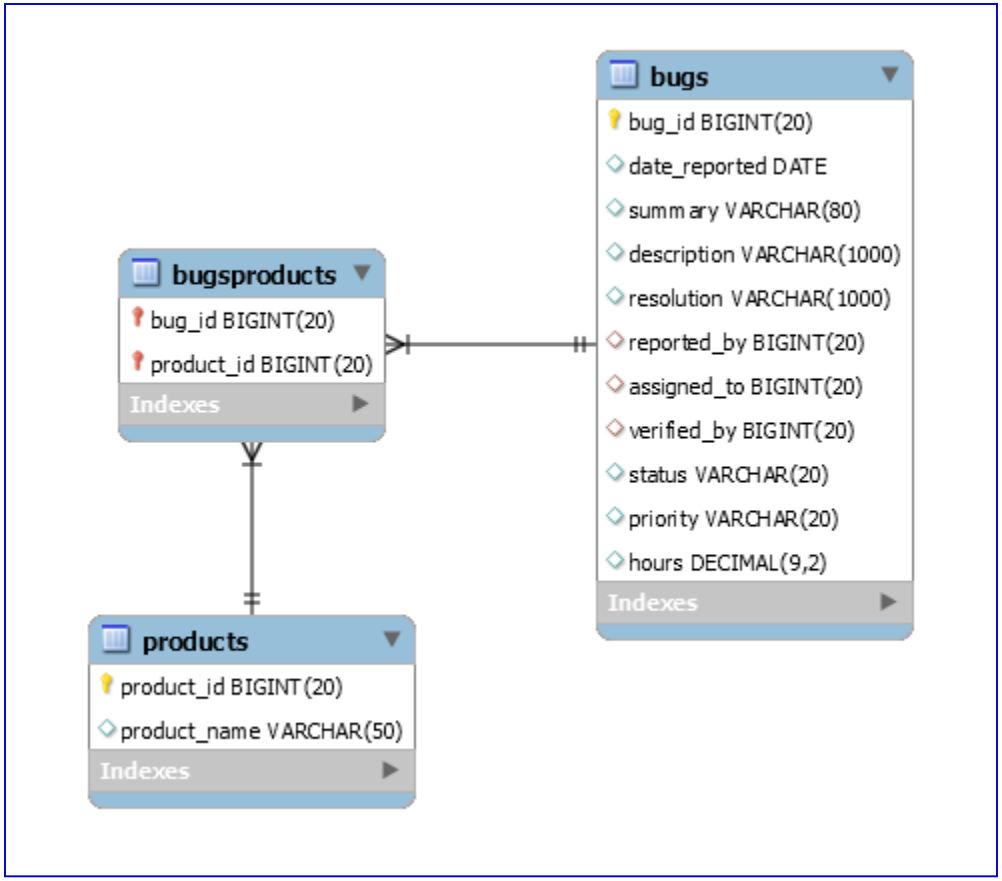
A. Use a LEFT OUTER JOIN

B. Use a RIGHT OUTER JOIN

C. Use a FULL OUTER JOIN

D. All of the above

E. None of the above



Solution: Concept Question 5

How can we fix this query to include the products that have no bugs?

```
SELECT bp.product_id, MAX(b.date_reported)
FROM BugsProducts bp JOIN Bugs b USING (bug_id)
GROUP BY bp.product_id
```

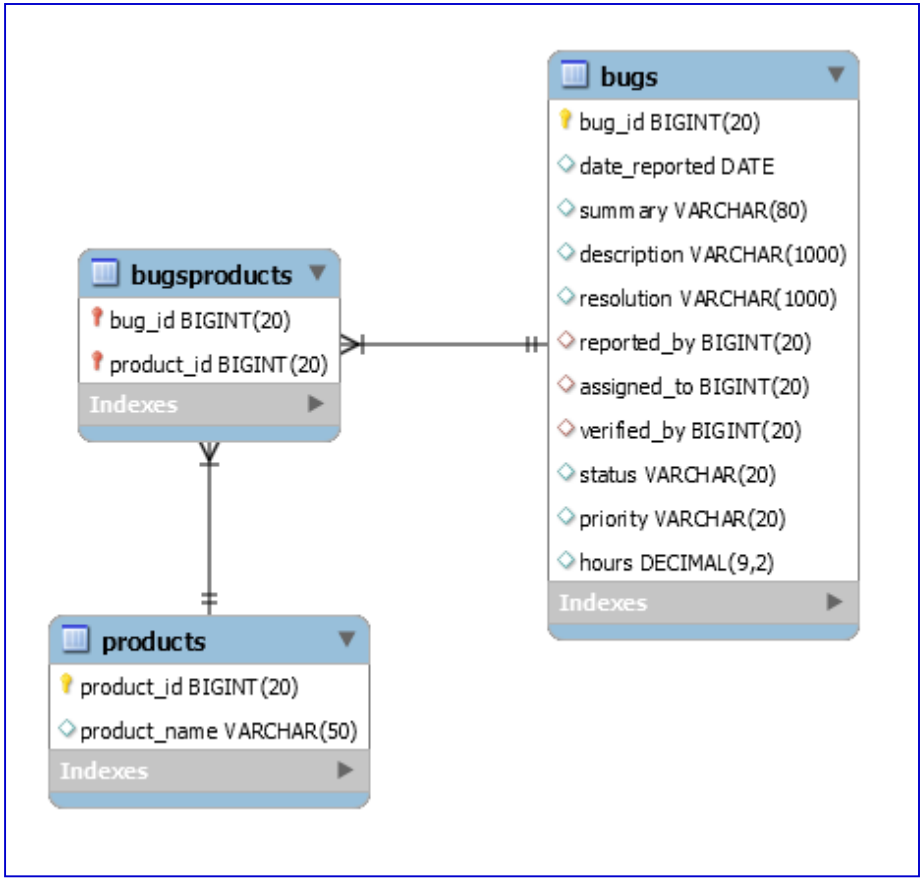
A. Use a LEFT OUTER JOIN

B. Use a RIGHT OUTER JOIN

C. Use a FULL OUTER JOIN

D. All of the above

E. None of the above



Fixed Query:

```
SELECT p.product_id,
MAX(b.date_reported)
FROM Products p LEFT OUTER JOIN
BugsProducts bp USING (product_id)
LEFT OUTER JOIN Bugs b USING (bug_id)
GROUP BY p.product_id
```

Concept Question 6

We have a table of test results. Each test has one or more steps. The table tracks the progress of the testing by providing a completion date for each step in the test. How can we find those tests that are **completed**? Notice that a test is represented as a **set** of records as opposed to a single record.

A.
```
select test_name
from Test_Results
group by test_name
having count(*) =
count(completion_date)
```

B.
```
select test_name
from Test_Results
where completion_date
is not null
group by test_name
```

C.
```
select test_step
from Test_Results
group by test_step
having count(*) =
count(completion_date)
```

D.
```
select test_step
from Test_Results
where completion_date
is not null
group by test_step
```

Table definition:

```
create table Test_Results
(
   test_name CHAR(20) NOT NULL,
   test_step INTEGER NOT NULL,
   completion_date DATE,
   PRIMARY KEY (test_name, test_step)
);
```

Sample dataset:

test_name	test_step	completion_date
Math Skills	1	2016-02-01
Math Skills	2	2016-02-02
Math Skills	3	2016-02-03
Math Skills	4	NULL
Math Skills	5	NULL
Reading Skills	1	2016-02-07
Reading Skills	2	2016-02-08
Reading Skills	3	2016-02-08

Homework for Next Time

- Read chapter 14 from the Learning SQL book
- Exercises at the end of chapter 14