

# CS 327E Class 3

February 11, 2019

1) A join is used to concatenate rows from two tables that are related via referential integrity. For example, joining  $T$  and  $U$  on  $T.b$  and  $U.b$  produces  $V$  when projecting all attributes from  $T$  and  $U$ .

$T\{a: \text{int}, b: \text{string}\}$

$U\{b: \text{string}, c: \text{string}, d: \text{date}\}$

$V\{a: \text{int}, b: \text{string}, b: \text{string}, c: \text{string}, d: \text{date}\}$

- |                             |
|-----------------------------|
| <p>A. True<br/>B. False</p> |
|-----------------------------|

2) A join is also used to union rows from two tables that share the same schema. For example, joining  $T$  and  $U$  produces  $V$ .

$T$

<u>id</u>	street	city
46	San Jacinto	Austin
82	Memorial Dr.	Houston
79	Main St.	Fort Davis

$U$

<u>id</u>	street	city
1	Chestnut	Philadelphia
2	South St.	Philadelphia
3	Market	Philadelphia

$V$

<u>id</u>	street	city
46	San Jacinto	Austin
82	Memorial Dr.	Houston
79	Main St.	Fort Davis
1	Chestnut	Philadelphia
2	South St.	Philadelphia
3	Market	Philadelphia

A. True  
B. False

3) The fields in a join condition must be of compatible type. For example, a field of type `String` cannot be joined to a field of type `Integer`.

- A. True
- B. False

4) A `SELECT` statement can contain at most one join.

- A. True
- B. False

5) Which is not a valid join type?

- A. Self join
- B. Outer join
- C. Partial join
- D. Inner join

# Syntax of Join Queries

SELECT <list of desired fields>

**FROM** <single table T1>

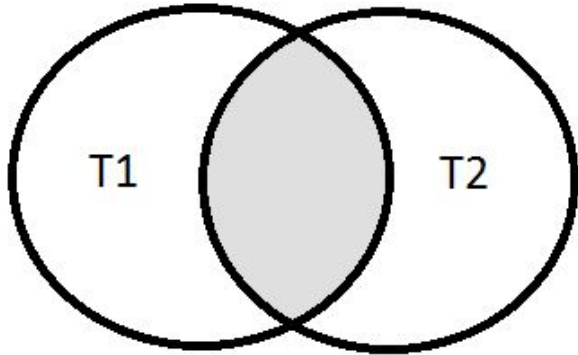
**JOIN** <single table T2> **ON** <T1.c1 = T2.c1>

WHERE <boolean conditions>

ORDER BY <list of fields to sort on>

# Inner Join

```
SELECT *  
FROM T1  
[INNER] JOIN T2 ON T1.c1 = T2.c1
```





# Inner Join

```
SELECT *  
FROM T1  
[INNER] JOIN T2 ON T1.c1 = T2.c1  
[INNER] JOIN T3 ON T2.c2 = T3.c2
```

# Inner Join

**Employee**

<u>empid</u>	emp_name	emp_dep
2	Mike	1
23	Dave	2
3	Sarah	
5	Jim	4
6	Sunil	1
37	Morgan	4

**Department**

<u>depid</u>	dep_name
1	Sales
2	Product
3	Research
4	Engineering
5	HR

```
SELECT emp_name, dep_name  
FROM Employee JOIN Department ON emp_dep = depid;
```

**Result Table**

emp_name	dep_name
Mike	Sales
Dave	Product
Jim	Engineering
Sunil	Sales
Morgan	Engineering

# First Question

*What are first and last names and grades of students who take CS329E with Prof. Mitra?*

Current\_Student(sid, fname, lname, dob, cno, cname, credits, grade)

New\_Student(sid, fname, lname, dob)

Class(tid, instructor, dept, cno, cname, credits)

# iClicker Question

*What are first and last names and grades of students who take CS329E with Prof. Mitra?*

How many records are in the answer?

- A. 1
- B. 2
- C. 3

# Second Question

*Who are the students who take both CS327E and CS329E?*

Current\_Student(sid, fname, lname, dob, cno, cname, credits, grade)

New\_Student(sid, fname, lname, dob)

Class(tid, instructor, dept, cno, cname, credits)

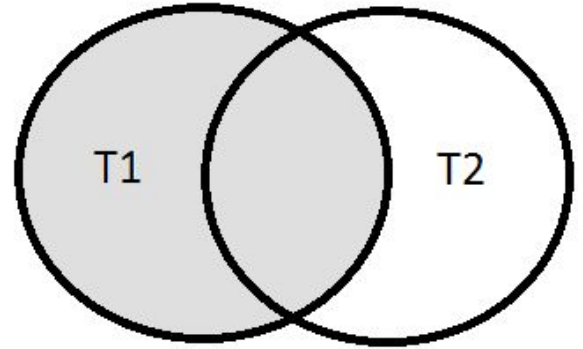
# Second Question

*Who are the students who take both CS327E and CS329E?*

```
SELECT sid
FROM Current_Student
JOIN Current_Student on sid = sid
WHERE cno = 'CS327E'
AND cno = 'CS329E'
```

# Left Outer Join

```
SELECT *  
FROM T1 LEFT [OUTER] JOIN T2  
ON T1.c1 = T2.c1
```



# Left Outer Join

Employee

<u>empid</u>	emp_name	emp_dep
2	Mike	1
23	Dave	2
3	Sarah	
5	Jim	4
6	Sunil	1
37	Morgan	4

Department

<u>depid</u>	dep_name
1	Sales
2	Product
3	Research
4	Engineering
5	HR

```
SELECT emp_name, dep_name
FROM Employee LEFT JOIN Department ON emp_dep = depid
ORDER BY emp_name;
```

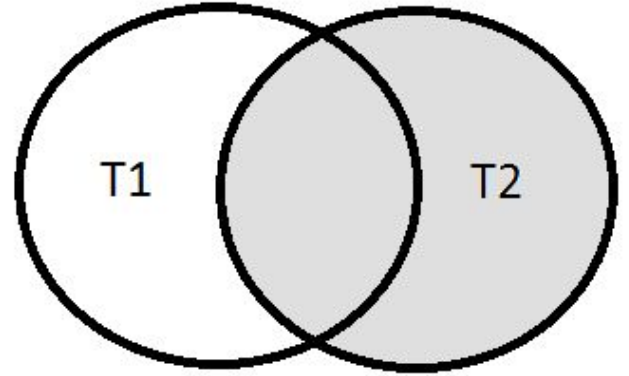
Result Table

emp_name	dep_name
Dave	Product
Jim	Engineering
Mike	Sales
Morgan	Engineering
Sarah	
Sunil	Sales



# Right Outer Join

```
SELECT *  
FROM T1 RIGHT [OUTER] JOIN T2  
ON T1.c1 = T2.c1
```



# Right Outer Join

Employee

empid	emp_name	emp_dep
2	Mike	1
23	Dave	2
3	Sarah	
5	Jim	4
6	Sunil	1
37	Morgan	4

Department

depid	dep_name
1	Sales
2	Product
3	Research
4	Engineering
5	HR

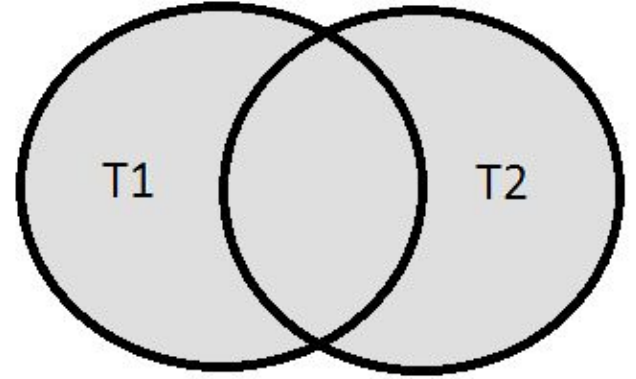
```
SELECT emp_name, dep_name
FROM Employee RIGHT JOIN Department ON emp_dep = depid
ORDER BY dep_name, emp_name;
```

Result Table

emp_name	dep_name
Jim	Engineering
Morgan	Engineering
	HR
Dave	Product
	Research
Mike	Sales
Sunil	Sales

# Full Outer Join

```
SELECT *  
FROM T1 FULL [OUTER] JOIN T2  
ON T1.c1 = T2.c1
```



# Full Outer Join

Employee

empid	emp_name	emp_dep
2	Mike	1
23	Dave	2
3	Sarah	
5	Jim	4
6	Sunil	1
37	Morgan	4

Department

depid	dep_name
1	Sales
2	Product
3	Research
4	Engineering
5	HR

```
SELECT emp_name, dep_name
FROM Employee FULL JOIN Department ON emp_dep = depid
ORDER BY dep_name, emp_name;
```

Result Table

emp_name	dep_name
Jim	Engineering
Morgan	Engineering
	HR
Dave	Product
	Research
Mike	Sales
Sunil	Sales
Sarah	

# Third Question

*Which instructors have no students in their class?*

Current\_Student(sid, fname, lname, dob, cno, cname, credits, grade)

New\_Student(sid, fname, lname, dob)

Class(tid, instructor, dept, cno, cname, credits)

# iClicker Question

*Which instructors have no students in their class?*

What type of join does this query require?

- A. Self join
- B. Outer join
- C. Inner join

# Fourth Question

*Which classes are taught by two teachers?*

*Show the answer as the cno of the class and tid for both teachers.*

```
Current_Student(sid, fname, lname, dob, cno, cname, credits, grade)
```

```
New_Student(sid, fname, lname, dob)
```

```
Class(tid, instructor, dept, cno, cname, credits)
```

# iClicker Question

*Which classes are taught by two teachers?*

*Show the answer as the cno of the class and tid for both teachers.*

How many records does the answer have?

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



# Demo: Creating an ERD

## College ERD v1

Class		
	tid	String
	instructor	String
	dept	String
	cno	String
	cname	String
	credits	Integer

Current_Student		
	sid	String
	fname	String
	lname	String
	dob	String
	cno	String
	cname	String
	credits	Integer
	grade	String

New_Student		
PK	sid	String
	fname	String
	lname	String
	dob	String

# Milestone 3

<http://www.cs.utexas.edu/~scohen/milestones/Milestone3.pdf>

# First Question

*What are first and last names  
and grades of students  
who take CS313E with Prof. Mitra?*

Student(sid, fname, lname, dob)

Class(cno, cname, credits)

Teacher(tid, fname, lname, dept)

Takes(sid, cno, grade)

Teaches(tid, cno)

# iClicker Question

*What are first and last names and grades of students who take CS313E with Prof. Mitra?*

How many inner joins are required for this query?

- A. 2 joins
- B. 3 joins
- C. 4 joins

Student(sid, fname, lname, dob)

Class(cno, cname, credits)

Teacher(tid, fname, lname, dept)

Takes(sid, cno, grade)

Teaches(tid, cno)

# Second Question

*Who are students who take both  
CS327E and CS329E?*

Student(sid, fname, lname, dob)

Class(cno, cname, credits)

Teacher(tid, fname, lname, dept)

Takes(sid, cno, grade)

Teaches(tid, cno)

# Second Question

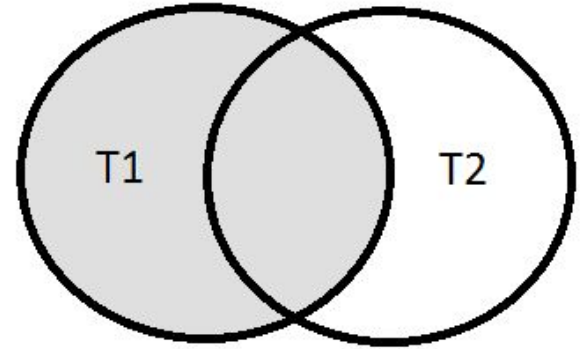
*Who are students who take both  
CS327E and CS329E?*

```
SELECT sid
FROM Takes
INNER JOIN Takes on sid = sid
WHERE cno = 'CS327E'
AND cno = 'CS329E'
```

Student(sid, fname, lname, dob)  
Class(cno, cname, credits)  
Teacher(tid, fname, lname, dept)  
Takes(sid, cno, grade)  
Teaches(tid, cno)

# Left Outer Join

```
SELECT *  
FROM T1 LEFT [OUTER] JOIN T2  
ON T1.c1 = T2.c2;
```





# Left Outer Join

Employee

<u>empid</u>	emp_name	emp_dep
2	Mike	1
23	Dave	2
3	Sarah	
5	Jim	4
6	Sunil	1
37	Morgan	4

Department

<u>depid</u>	dep_name
1	Sales
2	Product
3	Research
4	Engineering
5	HR

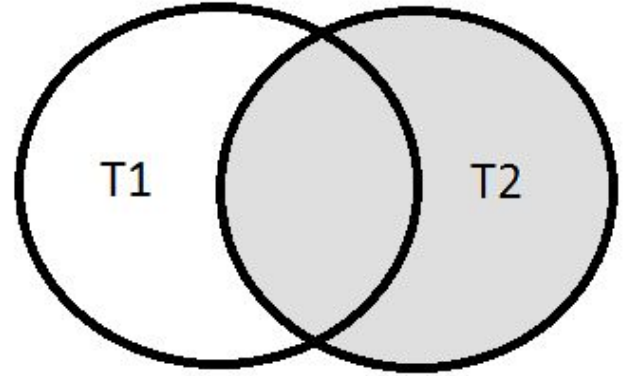
```
SELECT emp_name, dep_name
FROM Employee LEFT JOIN Department ON emp_dep = depid
ORDER BY emp_name;
```

Result Table

emp_name	dep_name
Dave	Product
Jim	Engineering
Mike	Sales
Morgan	Engineering
Sarah	
Sunil	Sales

# Right Outer Join

```
SELECT *  
FROM T1 RIGHT [OUTER] JOIN T2  
ON T1.c1 = T2.c2;
```



# Right Outer Join

Employee

empid	emp_name	emp_dep
2	Mike	1
23	Dave	2
3	Sarah	
5	Jim	4
6	Sunil	1
37	Morgan	4

Department

depid	dep_name
1	Sales
2	Product
3	Research
4	Engineering
5	HR

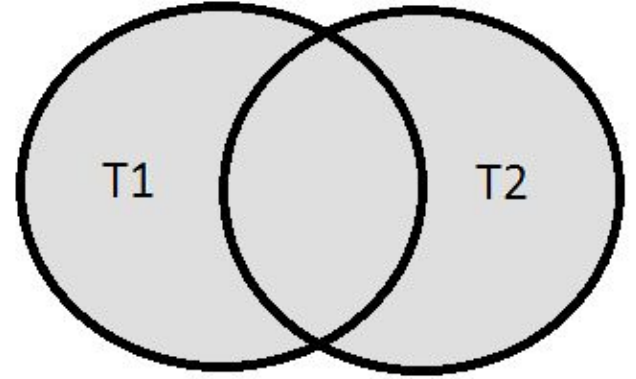
```
SELECT emp_name, dep_name
FROM Employee RIGHT JOIN Department ON emp_dep = depid
ORDER BY dep_name, emp_name;
```

Result Table

emp_name	dep_name
Jim	Engineering
Morgan	Engineering
	HR
Dave	Product
	Research
Mike	Sales
Sunil	Sales

# Full Outer Join

```
SELECT *  
FROM T1 FULL [OUTER] JOIN T2  
ON T1.c1 = T2.c2;
```



# Full Outer Join

Employee

empid	emp_name	emp_dep
2	Mike	1
23	Dave	2
3	Sarah	
5	Jim	4
6	Sunil	1
37	Morgan	4

Department

depid	dep_name
1	Sales
2	Product
3	Research
4	Engineering
5	HR

```
SELECT emp_name, dep_name
FROM Employee FULL JOIN Department ON emp_dep = depid
ORDER BY dep_name, emp_name;
```

Result Table

emp_name	dep_name
Jim	Engineering
Morgan	Engineering
	HR
Dave	Product
	Research
Mike	Sales
Sunil	Sales
Sarah	

# Third Question

*Which students take nothing?*

Student(sid, fname, lname, dob)

Class(cno, cname, credits)

Teacher(tid, fname, lname, dept)

Takes(sid, cno, grade)

Teaches(tid, cno)

# iClicker Question

*Which students take nothing?*

How many joins does this query require?

- A. 1
- B. 2
- C. 0

Student(sid, fname, lname, dob)

Class(cno, cname, credits)

Teacher(tid, fname, lname, dept)

Takes(sid, cno, grade)

Teaches(tid, cno)

# Fourth Question

*Which classes are taught by  
2 teachers?*

Student(sid, fname, lname, dob)

Class(cno, cname, credits)

Teacher(tid, fname, lname, dept)

Takes(sid, cno, grade)

Teaches(tid, cno)



# iClicker Question

*Which classes are taught by  
2 teachers?*

Which type of join does this query  
require?

- A. inner join
- B. left/right outer join
- C. full outer join

Student(sid, fname, lname, dob)

Class(cno, cname, credits)

Teacher(tid, fname, lname, dept)

Takes(sid, cno, grade)

Teaches(tid, cno)