

University of Central Florida

CGS 2545

Database Concepts

DEPARTMENT OF ELECTRICAL ENGINEERING & COMPUTER SCIENCE
COMPUTER SCIENCE DIVISION

Joins

- There are benefits of taking a Cartesian product of two relations, which gives all the possible tuples that are paired together.
- However, it might not be feasible in certain cases to take a Cartesian product where the result is huge relations with thousands of tuples having a considerable large number of attributes.

Joins

- **Join**
 - is a combination of a Cartesian product followed by a selection process.
 - join operation pairs two tuples from different relations, if and only if a given join condition is satisfied

Joins

- Theta (θ) Join
 - Theta join combines tuples from different relations provided they satisfy the theta condition.
 - The join condition is denoted by the symbol θ .
 - Notation

$R1 \bowtie_{\theta} R2$

- $R1$ and $R2$ are relations having attributes $(A1, A2, \dots, An)$ and $(B1, B2, \dots, Bn)$ such that the attributes don't have anything in common, that is $R1 \cap R2 = \Phi$.
- Theta join can use all kinds of comparison operators

Joins

Student		
SID	Name	Std
101	Alex	10
102	Maria	11


Subjects	
Class	Subject
10	Math
10	English
11	Music
11	Sports

Joins

```
STUDENT ⋈Student.Std = Subject.Class SUBJECT
```

Student_detail				
SID	Name	Std	Class	Subject
101	Alex	10	10	Math
101	Alex	10	10	English
102	Maria	11	11	Music
102	Maria	11	11	Sports

Joins

- Equijoin
 - When Theta join uses only **equality** comparison operator, it is said to be equijoin. The previous example corresponds to equijoin.
- Natural Join ()
 - Natural join does not use any comparison operator.
 - It does not concatenate the way a Cartesian product does.
 - Can perform a Natural Join only if there is at least one common attribute that exists between two relations.
 - In addition, the attributes must have the same name and domain.
 - Natural join acts on those matching attributes where the values of attributes in both the relations are same.

Joins

Courses		
CID	Course	Dept
CS01	Database	CS
ME01	Mechanics	ME
EE01	Electronics	EE

HoD	
Dept	Head
CS	Alex
ME	Maya
EE	Mira

Joins

Courses \bowtie HoD			
Dept	CID	Course	Head
CS	CS01	Database	Alex
ME	ME01	Mechanics	Maya
EE	EE01	Electronics	Mira

Joins

- Inner Joins
 - Theta Join, Equijoin, and Natural Join are called inner joins.
 - An inner join includes only those tuples with matching attributes and the rest are discarded in the resulting relation.
- Outer Joins
 - use outer joins to include all the tuples from the participating relations in the resulting relation.
 - There are three kinds of outer joins
 - left outer join
 - right outer join
 - full outer join

Joins

- Left Outer Join ($R \bowtie S$)
 - All the tuples from the Left relation, R, are included in the resulting relation.
 - If there are tuples in R without any matching tuple in the Right relation S, then the S-attributes of the resulting relation are made NULL.

Joins

Left	
A	B
100	Database
101	Mechanics
102	Electronics

Right	
A	B
100	Alex
102	Maya
104	Mira

Joins

Courses ⋈ HoD			
A	B	C	D
100	Database	100	Alex
101	Mechanics	---	---
102	Electronics	102	Maya

Joins

- Right Outer Join ($R \bowtie S$)
 - All the tuples from the Right relation, S, are included in the resulting relation.
 - If there are tuples in S without any matching tuple in R, then the R-attributes of resulting relation are made NULL.

Joins

Left	
A	B
100	Database
101	Mechanics
102	Electronics

Right	
A	B
100	Alex
102	Maya
104	Mira

Joins

Courses ⋈ HoD			
A	B	C	D
100	Database	100	Alex
102	Electronics	102	Maya
---	---	104	Mira

Joins

- Full Outer Join ($R \bowtie S$)
 - All the tuples from both participating relations are included in the resulting relation.
 - If there are no matching tuples for both relations, their respective unmatched attributes are made NULL.

Joins

Left	
A	B
100	Database
101	Mechanics
102	Electronics

Right	
A	B
100	Alex
102	Maya
104	Mira

Joins

Courses ⋈ HoD			
A	B	C	D
100	Database	100	Alex
101	Mechanics	---	---
102	Electronics	102	Maya
---	---	104	Mira