Lecture-12
DC

classmate
Date 15/02/18
Page

Join

- Cross
- Inner
- Outer

Table:

Loan

Branch_name	loan_no	amount
Downtown	L-170	3000
Redwood	L-230	4000
Perryridge	L-260	1700

Borrower

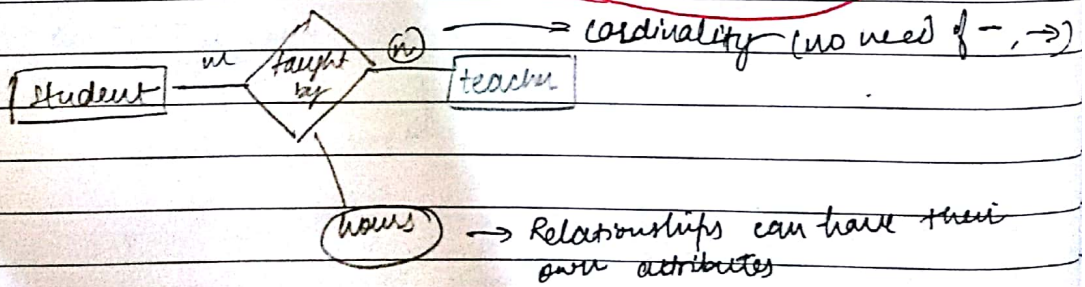
cust_name	loan no
Jones	L-170
Smith	L-230
Hayes	L-155

Join - To work with multiple tables as though they are single entity, we use joins. Tables are joint on columns that have the same data type and data width.

ER D

Entity - Noun
Relationship - Verb
E-R - Sentence
ERD - Story

Left → Right Top ↓ Bottom



→ cardinality (no need of -, →)

→ Relationships can have their own attributes + primary key of the joining entities.

Inner Join

These are also known as equi-join since the where statement generally compares two columns from two tables with the equivalence operator "=". It can be used in situations where selecting only those rows that have values in common in the columns specified. Inner join returns all rows from both tables

classmate

where there is a match.

Annotations: if checking; also data type & data width should be same; cannot put; here if loan-no is not primary / foreign key.

SQL> select * from loan inner join borrower; on loan.loan_no = borrower.loan_no

Output

Downtown	L-170	3000	Jones	L-170
Redwood	L-230	4000	Smith	L-230

can remove this repeating data by using natural join

SQL> select * from loan, borrower where loan.loan_no = borrower.loan_no;

↳ same as natural join

Output

Downtown	L-170	3000	Jones
Redwood	L-230	4000	Smith

⇒ Cross join → no condition checking.
All combinations are output (like matrix multiplicⁿ)

all other joins are subset of cross join.

SQL> select * from loan, borrower;

Output

Downtown	L-170	3000	Jones	L-170
Downtown	L-170	3000	Smith	L-230

⇒ Theta-join : A sort of inner join but θ is an operator which can be anything (not only =)