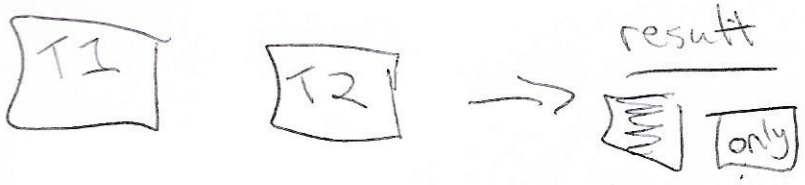


left join: lessening of criteria; show me everything on left table, and only matching records in the right table

right join: same concept, but vise versa
* table 1 right join table 2~~two~~ on cols...

- if the table cant match, ~~colum~~ table columns are NULL

inner join: only gives you columns ~~from~~ that exactly match from both tables

Database Ch. 2b - notes

I. Grouping Rows in SQL Select Stmts

a.) Group By clause (can include > 1 column) → secondar
- pretty straightforward → I've used this clause before
- end clause in a stmt

b.) having clause
- can use aggregate functions w/ hav
- where specifies which rows w
- Having specifies which groups
- Where is always applied b
and group by clause

II. Querying Multiple Tables

Subquery: used to determine a set of
the query (top one) that used (or
- "nested query"

- only columns selected from the top-level query are in the result set

Example:
```
Select Buyer, Department, count(SKU) as number_sold
From SKU_DATA
Where SKU in
    (Select SKU From Order_Item
     Where OrderNumber in
        (Select OrderNumber
         From Retail_Order
         Where OrderMonth = 'January'
         And OrderYear = 2015))
Group By Buyer, Department
Order By number_sold;
```

- when reading subqueries, always start from the bottom up to understand the logic

Querying Multiple Tables w/ Joins

*subquery limitation: selected data can only come from the top-level query

Cross Join: concatenating the row of all tables selected
- Cartesian Product

Inner Join: same, but adds a Where clause requiring a column from T1 = column from T2 (usually an Id)

Example:
```
Select * From Retail_Order, Order_Item
Where Retail_Order.OrderNumber = Order_Item.OrdersNumber;
```

↗ these are implicit joins: the Join keyword is never actually used in the SQL stmt

equijoin: using inner join w/ an 'is equal to' condition
'joining the two tables': process of creating a result table by joining two tables via a Join operation

- Requirement: all columns selected must either be used in an aggregate function or a Group By clause

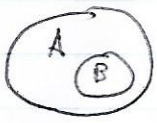
- can join 3 tables together, and by using the AND operator, make a 2nd 'is equal to' condition b/w columns

Comparing Subqueries to joins
- join can do all subqueries can't more
- subqueries are easier to write / understand

SQL 'JOIN ON' Syntax - explicit Joins
- know this syntax already
- Where clause is no longer used
- still requires PK → FK equivalence
  - however, matches can be w/ any tables

Outer Joins
- Right Outer Join = Right Join → same syntax

SQL Set Operators



A subset

~~Set~~ theory: math operations on sets
Set: a group of distinct terms

1.) UNION: all row values in one or both tables

2.) INTERSECT: all row values common to both tables

3.) EXCEPT: complement; area in A that's not B

- all values in T1 but not T2

- Except example: find data in 2014 table that did not also appear in the 2015 table

SQL: a data sublanguage that can be embedded into full programming langs or submitted directly to the DBMS.

* To use set operators, they must have the same or compatible data types and be the same number in each Select component

XML: Extensible Markup Language
ETL: Extract, Transform, Load
OLTP: Online Transaction Processing