



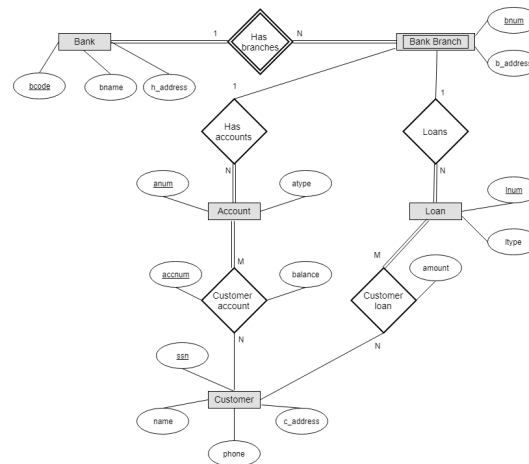
COSC210 Database Management Systems

Lecture 8 - SQL Lecture Two - Query Structure

Dr. Edmund Sadgrove

Summary

- Query Structure
- Multiple Table Queries
- Aggregation
- Creating Views



The SQL Query Structure

- Query Structure
- We can have SQL subqueries and even loops (not covered in this unit).

The 'WHERE' Clause.

- The 'WHERE' clause has all the comparison and logical operators of a regular programming language.

Comparison	Description	Logical	Example
<	less than	BETWEEN	a BETWEEN x AND y
>	greater than	AND	a>=x AND a>=y
<=	less than or equal to	OR	a>=x OR a<=y
>=	greater than or equal to	NOT	a IS NOT NULL
=	equal	IS	a IS DISTINCT
<> or !=	not equal	LIKE	a LIKE '%wordPart%'

One Relation Conditional Queries

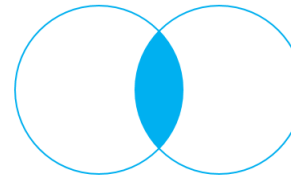
- We can use comparison and logical operators to return tuples based on a defined set of conditions.
- Examples:

One Relation Conditional Queries

- The LIKE clause can return search results when attribute syntax is unknown.
- String wildcards:
 - % : denotes start and end of string.
 - || : concatenates results.
- Remember to use single quotes ('):

Multiple Table Results Using Inner Joins

- An INNER JOIN can be used to join results from multiple tables.
- OUTER JOINS will not produce the same results (more on this later).
 - RIGHT OUTER JOIN
 - LEFT OUTER JOIN
 - FULL OUTER JOIN
- JOIN syntax (inner):



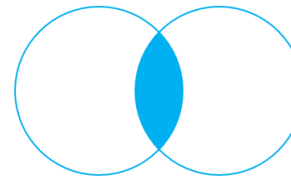
INNER JOIN

Multiple Table Results Using Inner Joins

- Display customers with a student account in Armidale branch.
- Will include tables:
 - customer
 - customer_account
 - account
 - bank_branch

Multiple Table Results Using Inner Joins

- The WHERE clause can be used as an alternative to INNER JOINS.
- This produces the same results, but is more compact.
- But this may be harder to read.
- WHERE syntax (join):



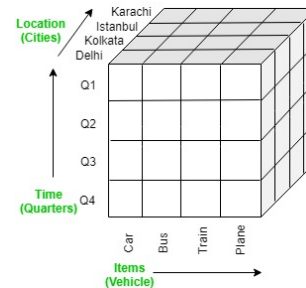
INNER JOIN

Multitable Table Results Using the Where Clause

- Display customers with a student account in Armidale branch.
- Will include tables:
 - customer
 - customer_account
 - account
 - bank_branch
- INNER JOINS and WHERE clauses are often interchangeable.

Aggregate Functions

- Aggregate functions include:
 - MIN()
 - MAX()
 - AVG()
 - COUNT()
 - SUM()
- A *GROUP BY* is required to pivot around an attribute.
- The HAVING clause is used instead of WHERE to compare results.



Aggregate Functions

- Lets display average, total and count of balances per branch.
- Will include tables:
 - bank_branch
 - account
 - customer_account

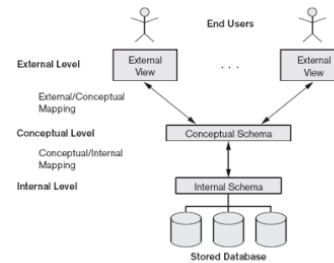
Aggregate Functions

- List the number of accounts and names of customers who have more than one bank account.
- Will include tables:
 - customer
 - customer_account

Creating Views

- PostgreSQL can be used to create different user views.
- Accessing views is similar to accessing tables.
- The `\dv` command can be used instead of `\dt`.

Figure 2.2
The three-schema
architecture.



Creating Views

- A view of customer_details that lists:
 - Customer name
 - Account balance
 - Customer address

Creating Views

- A view of customer_loans that lists:
 - Customer name
 - Loan amount
 - Loan type

Questions?