Implementing Table Operations Using Structured Query Language (SQL)

The implementation of table operations in relational database management systems is done through use of SQL, or Structured Query Language, the de facto language allowing users to access and manipulate data in RDBM systems.

Remember Operations On Tables
❖ The ability to construct tables from other tables.
❖ The five basic operations on tables are
  - Select
  - Project
  - Union
  - Difference
  - Product
❖ Join is a powerful operation created from product/project/select
❖ Table operations allow the data to be exhibited to users in whatever form they want

Join Example
Table A

<table>
<thead>
<tr>
<th>StudentID</th>
<th>Advisor</th>
<th>FirstName</th>
<th>LastName</th>
<th>Major</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jordan</td>
<td>Michael</td>
<td>1</td>
<td>Dickey</td>
<td>Martin</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>

Table B

<table>
<thead>
<tr>
<th>StudentID</th>
<th>Advisor</th>
<th>FirstName</th>
<th>LastName</th>
<th>Major</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jennings</td>
<td>Waylan</td>
<td>2</td>
<td>Whicker</td>
<td>Grace</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
</tbody>
</table>

Natural Join of Table A and Table B

<table>
<thead>
<tr>
<th>StudentID</th>
<th>Advisor</th>
<th>FirstName</th>
<th>LastName</th>
<th>Major</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jordan</td>
<td>Michael</td>
<td>1</td>
<td>Dickey</td>
<td>Martin</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>Jennings</td>
<td>Waylan</td>
<td>2</td>
<td>Whicker</td>
<td>Grace</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
</tbody>
</table>

Join Example

<table>
<thead>
<tr>
<th>StudentID</th>
<th>Advisor</th>
<th>FirstName</th>
<th>LastName</th>
<th>Major</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jordan</td>
<td>Michael</td>
<td>1</td>
<td>Dickey</td>
<td>Martin</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>Jennings</td>
<td>Waylan</td>
<td>2</td>
<td>Whicker</td>
<td>Grace</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
</tbody>
</table>

Using Multiple Operations

Show Only certain columns and rows from the join of Table A with Table B

<table>
<thead>
<tr>
<th>StudentID</th>
<th>Advisor</th>
<th>FirstName</th>
<th>LastName</th>
<th>Major</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jordan</td>
<td>Michael</td>
<td>1</td>
<td>Dickey</td>
<td>Martin</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>Jennings</td>
<td>Waylan</td>
<td>2</td>
<td>Whicker</td>
<td>Grace</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
</tbody>
</table>

This table doesn’t exist by itself. It is a view of certain rows and columns from other tables.
Implementing Table Operations With SQL

❖ Let's see how various table operations are actually done using a database language
❖ SQL stands for Structured Query Language.
❖ SQL is the de facto query standard for accessing and manipulating data in relational databases
❖ In Access you can also use a graphical query interface, called the QBE (Query By Example), that generates SQL for you

SQL: Structured Query Language

❖ There are many uses for SQL in database structures.
   ❖ SQL can be used to define, or construct, a database
   ❖ SQL can be used do basic management of the database
     √ check into table content
     √ add to table content
     √ delete table content
     √ etc.
   ❖ SQL can be used to query the database
     √ create virtual tables or “views” from existing table(s)
     √ A view may be selected attributes from various tables
❖ We will focus on the basic SQL commands that allow us to do simple database management and to create virtual tables (views) of the contents of the database

Queries: Create Tables From Tables

❖ CONCEPT: The operations on databases: Select, Project, Union, Difference, and Product create tables from tables. These actions are done with a Query
❖ How are queries implemented?
   ❖ Database systems come with a “query language” … SQL is the most common one and is the standard for Relational databases
   ❖ The most common clauses used in SQL for queries are shown below:

```
SELECT <fields of desired table>  what columns will be retrieved
FROM <list of tables>  which table contains the column data
WHERE <T/F predicate>  criteria for returning rows
```

SQL Syntax

❖ SQL is not case sensitive.
❖ SQL statements combine several table operations together to display or modify the data
   ❖ But note the difference between Select and the table operation Selection
   ❖ The table operation SELECTION brings back rows based on some criteria
   ❖ Select clause in SQL is actually the Projection table operation
A Simple ERD and Database Schema

- Advisor and Student tables
  - Each student is allowed a single advisor at any one time
  - An advisor may have zero, one or many students to advise

**Advisor**
- AdvisorID
- FName
- LName
- Department
- HireDate
- PK AdvisorID

**Student**
- SID
- FName
- LName
- MajorID
- AdvisorID

A Simple ERD and Database Schema

- Advisor and Student tables
  - Each student is allowed a single advisor at any one time
  - An advisor may have zero, one or many students to advise

**Advisor**
- AdvisorID
- FName
- LName
- Department
- HireDate
- PK AdvisorID

**Student**
- SID
- FName
- LName
- MajorID
- AdvisorID

Basic Data Management

- Checking the Tables Contents
  - `SELECT <attributes> FROM <table name(s)>`;

  - Examples:
    - `SELECT * FROM Student;`
    - is the same as
    - `SELECT SID, FName, LName, MajorID, AdvisorID FROM Student;`

- This will essentially mimic the table Student and show all current contents in a view of the table

Queries

- Partial Listing of Table Contents
  - `SELECT <attributes> FROM <table name(s)>`;

  - Examples:
    - `SELECT FName, LName, Major FROM Student WHERE SID = 0023892;`
    - `SELECT FName, LName FROM Student WHERE Major = "INFO";`

NULL Means Nothing

- A NULL character means that nothing has been entered. This is different from a space or a zero.

  - `SELECT LName FROM Student WHERE FName IS NULL;`
ORDER BY… Sorting Outputs

❖ Sorting in descending order…
SELECT StudentID, Name
FROM Student
ORDER BY Name DESC;

❖ Sorting in ascending order…
SELECT StudentID, Name
FROM Student
ORDER BY Name ASC;

Preparing for a Join….

❖ Example of a Product and Projection Operation:
SELECT Student.FName, Student.LName, Advisor.LName
FROM Student, Advisor;

❖ What is the result?
Notice that I indicate the table name with the attribute when I have more than one table in the FROM statement. Specifically when I have attributes with the same name in different tables. This is called Table Qualification.

Queries Using Joins

❖ Example of a Join that includes Product, Projection and Selection:
SELECT Student.FName, Student.LName, Advisor.LName
FROM Student, Advisor
WHERE Student.AdvisorID = Advisor.AdvisorID;

SELECT Student.FName, Student.LName, Advisor.LName
FROM Student INNER JOIN Advisor ON
Student.AdvisorID = Advisor.AdvisorID;

Comparison Operators

Equals =
Not equals <>
Greater than >
Less than <
Greater than or equal to >=
Less than or equal to <=
**Queries**

- **Using Relationship Operators for criteria involving non-key attribute and/or criteria involving key attributes**

  - Examples:
    
    ```sql
    SELECT FName, Lname 
    FROM Advisor 
    WHERE HireDate >= 1987;
    
    SELECT FName, LName, Major 
    FROM Student 
    WHERE SID < > 0023892;
    ```

- **Logical Operators: AND, OR**

  - Examples:
    
    ```sql
    SELECT FName, Lname 
    FROM Advisor 
    WHERE HireDate > 1987 OR 
    HireDate < 1962;
    
    SELECT FName, LName 
    FROM Student 
    WHERE AdvisorID = 44232 AND 
    Major = "INFO";
    ```

**Just Scratching the Surface**

- There are many more commands available in SQL as well as different standards for the language
- You have been shown some common clauses
- In Access you will be provided with a graphical user interface known as QBE, Query by Example, to create queries. But you can look at SQL View to see the SQL clauses that are generated
- Practice interpreting them as a way to see what your query is doing and to be able to explain what the SQL is doing in one of the queries for Project 3, Part II

**Order of Execution of SQL Statements Covered in Class**

1. FROM: Identifies tables involved
2. WHERE: Finds all rows meeting stated condition(s)
3. SELECT: Identifies columns
4. ORDER BY: Sorts rows
5. RESULTS