Implementing Table Operations Using Structured Query Language (SQL)

FIT 100

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.

FIT 100 Using Multiple Operations

StudentID	tblStudent.LN	tblStudent.F	AdvisorID	tblAdvisor.LNam	tblAdvisor.FNan
2	Jordan	Michael	1	Dickey	Martin
1	Crowley	Caro	2	Whiteaker	Grace
12	Jennings	Waylan	2	Whiteaker	Grace

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

tblStudent.LN	tblStudent.F	tblAdvisor.LNam	tblAdvisor.FNar
Jordan	Michael	Dickey	Martin
Crowley	Caro	Whiteaker	Grace

- This table doesn't exist by itself. It is a view of certain rows and columns from other tables.

FIT 100 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

FIT 100 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

FIT 100 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:

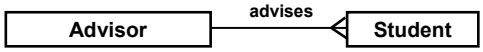
```
SELECT <fields of desired table>     'what columns will be retrieved
FROM <list of tables>                'which table contains the column data
        INNER JOIN <keys>            'key constraints (joins) on tables
WHERE <T/F predicate>;               'non key criteria for returning rows
```


FIT 100 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
      - Select returns certain columns

FIT 100 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	Student
AdvisorID	SID
FName	FName
LName	LName
Department	MajorID
HireDate	AdvisorID
PK AdvisorID	PK SID

FIT 100 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

FIT 100 Queries

- Partial Listing of Table Contents

```
SELECT <attributes>
FROM <table name(s)>
WHERE <T/F predicates>;
```

  - Examples:

```
SELECT FName, LName, Major
FROM Student
WHERE SID = 0023892;

SELECT FName, LName
FROM Student
WHERE Major = "INFO";
```

The WHERE clause reduces output of rows based on some specified criteria. It is one implementation of Selection Operator

FIT 100 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;
```


FIT 100 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;
```


FIT 100 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

FIT 100 Queries Using Joins

- Example of a Join that includes Product, Projection and Selection:

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

Natural Join using the SQL92 standard

FIT 100 Comparison Operators

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

FIT 100 Queries

- Use Relationship Operators for constraints on rows to be returned
  - Examples:

```
SELECT FName, Lname
FROM Advisor
WHERE HireDate >= 1987;
```

```
SELECT FName, LName, Major
FROM Student
WHERE SID < > 0023892;
```


FIT 100 Queries

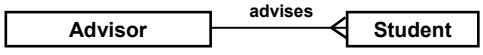
- Use logical operators to combine multiple constraints
- Logical Operators: AND, OR
  - Examples:

```
SELECT FName, Lname
FROM Advisor
WHERE HireDate > 1987 OR
      HireDate < 1962;
```

```
SELECT FName, LName
FROM Student
WHERE AdvisorID = 44232 AND
      Major = "INFO";
```

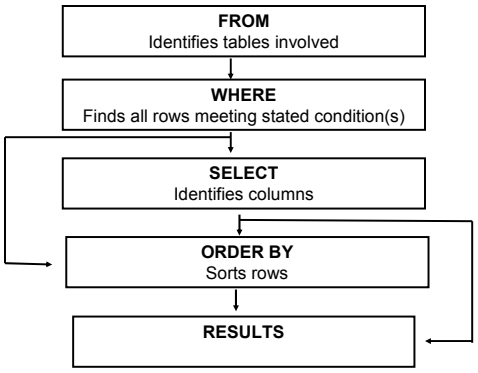
FIT 100 Simple Join Queries

- Return the advisor and student name for student id 0001234
- Return the advisor name for student "Johnson"



Advisor	Student
AdvisorID	SID
FName	FName
LName	LName
Department	MajorID
HireDate	AdvisorID
PK AdvisorID	PK SID

Order of Execution of SQL Statements Covered in Class



FIT 100 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 the SQL statements so you can explain what the SQL is doing in one of the queries for Project 3, Part II
- Assignment 3 is due at the end of lab Th/F. You will practice SQL with this assignment