

Completing the SCCHC Database, Part I



Linking forms and constructing queries for the
Seattle Central Community Health Clinic

© Copyright 2000-2001, University of Washington

Review of SCCHC Operation

- ❖ At the SCCHC there are various operations, some performed for each client, others performed at larger intervals
 - Enter client data
 - Start the process for a client to visit with health professional
 - Health professional fills out consultation chart

- ❖ To be covered on Friday (part II)
 - Order tests
 - Label specimens
 - Compare specimens to manifest
 - Record results from test outcome

© Copyright 2000-2001, University of Washington

How Does the Client Provide Information?

- ❖ A form has been created called frmClient that will be used to provide personal information by first-time patients
- ❖ After the form is filled out, the information from it is entered into the database and is stored in the tblClient table
- ❖ One could imagine that this form might be filled out by the Client themselves and returned to the desk, or a volunteer in charge of new patients might enter the information into the form directly after asking questions
- ❖ Once the information is entered, another form can be used to view and verify the information just before setting up a Visit with a Health Professional

© Copyright 2000-2001, University of Washington

What Is A Visit?

- ❖ When the receptionist is done verifying data from the frmReception form, an output of data from the tblClient table, the client is set up for a face-to-face visit / appointment with a health professional
- ❖ What does a visit mean in within the context of our database?
- ❖ Remember a Visit in the SCCHC database means to establish the relationship between the Client and the HealthPro ... we represent a single one of these relationships by a row in the Visit table, so...
 - Setting up an appointment/visit must involve creating a row in the Visit table with the initial data of a client and a health professional

© Copyright 2000-2001, University of Washington

FIT 100

Construct a Reception Form To Bring a Client and Health Professional Together

- ❖ Once the data is verified on the frmReception Form and changes stored in the Client table, the receptionist must choose a medical professional.

SCCHC
Seattle Central Community Health Clinic
1429 Yessler Way
Seattle, Washington
08100

Date: 12/4/2000

Make Appointment

ClientID: [] Address: 1234 17th Ave SW
Mr, Mrs, Ms: Ms City: Seattle
First Name: Holly State (eg. WA): WA
Middle Name: Catherine Zip Code: 98112
LName: Eggerton Home Phone: (206) 123-4567
Jr, Sr, III: Work Phone: (206) 709-1234
Birthday: 7/28/1971

University of Washington

FIT 100

Clicking on "Make Appointment..."

- ❖ Clicking on the button with the "Make Appointment" caption brings up a linked form with 3 (minimum) fields in it from the Visit table, not from the Client Table

VisitID	MedProID	ClientID
1	2	2
	Mac The Kraft	
	Joe the Hack	
	Jack "the big sleep" Kevorkian	

Record: 1 of 14

- ❖ Pairing the client and the medical professional creates the relationship we want to represent, so...
- ❖ By entering the health professionals ID and the client's ID, the receptionist sets up the visit between doctor and client AND sets up the row in Visit to record the data for the visit

FIT 100

Getting a Linked Form

- ❖ Follow these steps in the Form Wizard to create a linked forms between frmReception and frmSeeHealthPro:
 - Move all of the fields from tblClient table to form
 - Move the three fields from tblVisit: VisitID, ClientID, HealthProID
 - When asked how you want the data displayed, say linked form
 - Linked forms allow the Receptionist to enter or modify data into two tables: tblClient and tblVisit. The entry in tblVisit is through the linked form named frmSeeHealthPro
 - You could also use what you learned about adding combo boxes in Lab 17 to be able to display the name of the Health Professional in the linked form.

© Copyright 2000-2001, University of Washington

FIT 100

Screen Shot of Form Wizard for frmReception Form with frmSeeHealthPro

Form Wizard

How do you want to view your data?

by tblClient

by tblVisit

Preview: CID, Prefix, FirstName, MiddleName, LastName, Suffix, HomeAddress, City, State, ZipCode, HomePhone, WorkPhone, BirthDate

Selected fields: VisitID, HealthProID, ClientID

Form with subform(s) Linked forms

Buttons: Cancel, < Back, Next >, Finish

Washington



Creating the Initial frmConsult Form

- ❖ You will create the consult form in part I, and add more to it in part II:
 - The form will require information from all three tables
 - What is necessary?

© Copyright 2000-2001, University of Washington



Test Request Information

- ❖ There are two aspects to a test request:
 - The information that a test has been requested
 - The outcome of the test
- ❖ The two aspects will be treated separately
 - For each <test>, there is a <test>R checkbox field for the request
 - For each <test>, there is a <test>O text field for the outcome
- ❖ On the frmConsult form it should make sense that the health professional needs to work with the request checkboxes, but does not need to see the outcome fields since it is assumed that there are no outcomes before a test is even requested

© Copyright 2000-2001, University of Washington



Screen Shot of Form Wizard For Consult

Form Wizard

How do you want to view your data?

by tblClient
by tblHealthPro
by tblHealthPro

VisitID, DateOfVisit, tblHealthPro_FirstName, tblHealthPro_LastName, Degree, Prefix, tblClient_FirstName, MiddleName, tblClient_LastName, Suffix, DrugTestRequired, HIVTestRequired, AidsTestRequired, HepB1TestRequired, ExceptionalTestRequired, RecentMedicalHistory, PresentingSymptoms, Notes, TrackingNumber

Single form Linked forms

Cancel < Back Next > Finish

Washington



Summary of Form Creation

- ❖ Controls a series of critical operations for the clinic database
 - Appointment scheduling .. Setting up the relationship in Visit
 - Consultation between Health Professional and Client .. Establishes the form the Health Professional will be filling out during their visit with the client
- ❖ To be covered on Friday...
 - Ordering tests
 - Developing the tracking number
 - Creating Test Manifests

© Copyright 2000-2001, University of Washington

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.

© Copyright 2000-2001, University of Washington

Remember Operations On Tables

- ❖ A use of tables is to construct other tables from them.
- ❖ The Dean's list of students shown last week was a table constructed from other tables using several table operations

Dean's View of Database

Student_ID	Nick_Name	Major	GPA	Street_Address	City	State	Country	PostalCode
1021253	Chela	INFO	3.89	14 Mountain Ave	Victoria	BC	Canada	V6N4T4
1021343	J.T.	SPAN	3.85	1715 65 th Ave	Seattle	WA	USA	98125
...

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

© Copyright 2000-2001, University of Washington

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 will use a graphical query interface, called the QBE (Query By Example), that generates SQL for you
- ❖ But let's see what is happening when SQL clauses are implemented against data stores

© Copyright 2000-2001, University of Washington

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>
FROM <list of tables>
WHERE <T/F predicate>;
```

© Copyright 2000-2001, University of Washington



SQL: Structured Query Language

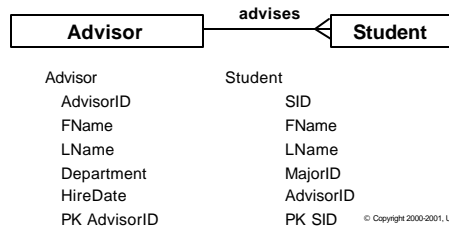
- ❖ There are many uses for SQL in database structures.
 - SQL can be used to define, or construct, a database
 - SQL can be used do basic management of the database
 - ≡ check into table content
 - ≡ add to table content
 - ≡ delete table content
 - ≡ etc.
 - SQL can be used to query the database
 - ≡ create virtual tables or "views" from existing table(s)
- ❖ 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

© Copyright 2000-2001, University of Washington



A Simple ERD

- ❖ 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



© Copyright 2000-2001, University of Washington



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

© Copyright 2000-2001, University of Washington



Queries

- ❖ Partial Listing of Table Contents


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

A predicate is a logical expression that follows the Where clause

 - Examples:


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

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

```
SELECT Student.FName, Student.LName, Advisor.LName
FROM Student, Advisor
WHERE Student.AdvisorID= Advisor.AdvisorID
```

© Copyright 2000-2001, University of Washington

FIT **100** Queries

❖ Using Relationship Operators

□ Examples:

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

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

© Copyright 2000-2001, University of Washington

FIT **100** Queries

❖ Logical Operators: AND, OR, and NOT

□ Examples:

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

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

© Copyright 2000-2001, University of Washington

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 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 4, Part II

© Copyright 2000-2001, University of Washington