Two Topics: Queries & Universality

Once database tables are built, the easiest way to perform most subsequent processing is by deriving new tables using database queries. Project 4 illustrates this technique.

Universality is a key property of computers that explains their success and implies future potential.

Universality

- Universality means in broad terms “anything one computer can do, any computer can do”
- Universality says computers are all equivalent
  + A new computer doesn’t do more than the computer it replaces, it just does it faster
  + Though there have been many generations of computers, none have been “more powerful” than the very first computers, in the sense of being able to solve more problems or compute more things
  + Universality is true because the instruction set of one computer -- the computer’s basic operations implemented in hardware -- can be simulated by the instruction set of any other computer, enabling it to do whatever the original machine could do
Processing Information

- Universality is one way in which processing information is different from "processing matter".
- A "matter processing" example...
  - Consider a grinder, say for grinding coffee... it basically only grinds coffee or perhaps a few other similar size "beans"... it doesn't grind tree branches into compost... it doesn't grind steel to sharpen knives... it doesn't mix concrete despite the fact these are essentially equivalent physical motions of a rotating appliance on a spindle.
- A computer can add new capabilities simply by adding new software.

Practical Considerations...

- It often seems like computers are not universal...
  - The computer embedded in the carburetor of a car is not universal, since it only mixes fuel... but this is simply a universal device specialized.
  - A cash machine can print receipts but not term papers... but only because it doesn't have the right printer.
  - My Macintosh can't run my PC programs and my PC can't run my Macintosh programs... software is distributed in "binary" form a specific instruction set encoding for a specific computer, but Macs and PCs can simulate the other machine, albeit at some loss in speed.
Implications Of Universality ...

- Physical machines can have perfectly specialized user interfaces since the application range is limited
- A computer, being a universal device, is “configured” by the user, i.e. the user is involved in making the capability available and matching the software’s behavior to his or her needs
  
  … consider the time spent with computers setting defaults and properties, and in orchestrating a computation by invoking (double clicking on) applications, naming files, moving files, locating files, …
- It gets easier, but it will never go away or be trivial

Most people don’t know how cars work but they can drive and fully use a car. Why learn how computers work …?

Queries

- A query is a statement (in a language like SQL) to define a table by extracting information from existing tables
- An example …

```
SELECT Visit.Date, Visit.Tracking, Visit.HepBO
FROM Visit
WHERE ((Visit.Tracking)<>"" ) AND ((Visit.HepBO)=Yes));
```

- Once the basic tables of a database are set up, the remaining processing is usually derived using queries
The CDC retains its data in the three basic tables, \textit{Clients}, \textit{Doctors}, and \textit{Visit}.

Other tables are needed for information system operation:
- Lab Manifests are derived from the \textit{Visit} table.
- Labels are derived from the Lab Manifest tables.
- A table is needed as input to the letter creation operation, which is built from the \textit{Visit} and the \textit{Clients} tables.
- A table of summary statistics must be created from \textit{Visit} records.

Queries are written in a “query language” … SQL has become the most popular, though others exist.

Writing queries is easily learned (easier than VB6.0!)

Basic components of a query ...

- The \textbf{SELECT} clause specifies the fields in the resulting table.
- The \textbf{FROM} clause specifies which tables are involved in the query, including tables that are JOINed together.
- The \textbf{WHERE} clause imposes constraints on the fields, allowing specific types of information to be extracted.

\begin{verbatim}
SELECT Visit.Date, Visit.Tracking, Visit.HepBO
FROM Visit
WHERE (((Visit.Tracking)<>"") AND ((Visit.HepBR)=Yes));
\end{verbatim}
Using A Wizard For Lab Manifest

- A manifest must be generated for each test from Visit

Restricting The Query

- Constraints are needed because selecting out the fields includes more records than desired

<table>
<thead>
<tr>
<th>HepLabMan Query : Select Query</th>
<th>Date</th>
<th>Tracking</th>
<th>HepBR</th>
<th>HepBO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>6/24/99 CDC4C4E45</td>
<td>☐</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5/28/99 CDC4C4E46</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td></td>
<td>5/20/99</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td></td>
<td>5/28/99 CDC4C4E49</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td></td>
<td>6/28/99</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>

+ Test not selected
+ Test not ordered
+ Test for earlier date
Adding Criteria

- The query can be edited to add restrictions

Resulting Query

- The query to select out all of the tests that were ordered today for Hepatitis B is

```sql
SELECT Visit.Date, Visit.Tracking, Visit.HepB0
FROM Visit
WHERE (((Visit.Date)=Date())
    AND ((Visit.Tracking)<>" ")
    AND ((Visit.HepBR)=Yes));
```

The Report Wizard can help construct the printed manifest
The Lab Manifest Printout

LabManHepB

<table>
<thead>
<tr>
<th>Date</th>
<th>Tracking</th>
<th>HepBO</th>
</tr>
</thead>
<tbody>
<tr>
<td>5/28/99</td>
<td>CDC4C4E46</td>
<td>0</td>
</tr>
<tr>
<td>5/28/99</td>
<td>CDC4C4E49</td>
<td>0</td>
</tr>
</tbody>
</table>

Labels

Hepatitis B Test Sample
CDC4C4B46
Central District Clinic
(206)5551212
5/28/99
An Example … The Drug Test

- Incorporating the new results from the Drug test back into Visit works as follows …
  - There are two tables, Visit and LabManHepB
  - Associating Tracking fields allows tables to be joined

A straightforward query installs DrugO into Visit
Assume this is done

+ Associating Tracking fields allows tables to be joined