Declarative Languages

Database Management Systems

A table is a collection of records, which are rows that have a value for each column.

A column has a name and a type.

A table has columns and rows.

Declarative Programming

In **declarative languages** such as SQL & Prolog:
- A "program" is a description of the desired result
- The interpreter figures out how to generate the result

In **imperative languages** such as Python & Scheme:
- A "program" is a description of computational processes
- The interpreter carries out execution/evaluation rules

Create table cities as

```sql
create table cities as
select 38 as latitude, 122 as longitude, "Berkeley" as name union
select 45, 93, "Minneapolis";
```

Select "west coast" as region, name from cities where longitude >= 115 union

Select "other", name from cities where longitude < 115;

Structured Query Language (SQL)

The Structured Query Language (SQL) is perhaps the most widely used programming language.

SQL is a declarative programming language.

SQL Overview

The SQL language is an ANSI and ISO standard, but DBMS’s implement custom variants.

- A **select** statement creates a new table, either from scratch or by projecting a table.
- A **create table** statement gives a global name to a table.
- Lots of other statements exist: analyze, delete, explain, insert, replace, update, etc.
- Most of the important action is in the **select** statement.

Getting Started with SQL

Install sqlite (version 3.8.3 or later): [sqlite.org/download.html]

Use sqlite online: [http://sqlite.codeplex.com/wikipage?title=sqldf]

Today's theme:
Selecting Value Literals

A select statement always includes a comma-separated list of column descriptions. A column description is an expression, optionally followed by as and a column name:

\[ \text{select \ expression \ as \ [name], \ expression \ as \ [name];} \]

Selecting literals creates a one-row table:

\[ \text{select \ "fillmore", \ "delano" \ as \ parent, \ "abraham", \ "barack" \ as \ child; \union} \]

The union of two select statements is a table containing the rows of both of their results:

\[ \text{select \ "delano" \ as \ parent, \ "herbert" \ as \ child; \union} \]

\[ \text{select \ "abraham", \ "barack" \ union} \]

\[ \text{select \ "fillmore", \ "delano" \ union} \]

\[ \text{select \ "fillmore", \ "grover" \ union} \]

\[ \text{select \ "eisenhower", \ "fillmore" \ union} \]

Select Statements Project Existing Tables

A select statement can specify an input table using a from clause:

\[ \text{select \ [expression] \ from \ [table] \ where \ [condition];} \]

An ordering over the remaining rows can be declared using an order by clause:

\[ \text{select \ [columns] \ from \ [table] \ where \ [condition] \ order by \ [order \ clause];} \]

Column descriptions determine how each input row is projected to a result row:

\[ \text{select \ [columns] \ from \ [table] \ where \ [condition] \ order by \ [order \ clause];} \]

A subset of the rows of the input table can be selected using a where clause:

\[ \text{select \ [columns] \ from \ [table] \ where \ [condition];} \]

Arithmetic in Select Expressions

Arithmetic expressions can combine row values and constants:

\[ \text{create table \ lift \ as \ select \ \text{lift} \ as \ chair, \ 2 \ \times \ \text{single} \ as \ couple \ union \ select \ 102 \ \text{as \ lift};} \]

\[ \text{select \ chair, \ \text{single} \ \times \ 2 \ \text{as \ total \ from \ lift;}} \]

Discussion Question

Given the table \text{lits} that describes how to use powers of 2 to form various integers:

\[ \text{create table \ lits \ as \ select \ "zero" \ as \ word, \ 0 \ as \ num, \ 0 \ \text{as \ two}, \ 0 \ \text{as \ four}, \ 0 \ \text{as \ eight}, \ 0 \ \text{as \ sixteen}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine} \ union \ select \ "one" \ as \ word, \ 1 \ as \ num, \ 0 \ \text{as \ two}, \ 0 \ \text{as \ four}, \ 0 \ \text{as \ eight}, \ 0 \ \text{as \ sixteen}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine} \ union \ select \ "two" \ as \ word, \ 0 \ as \ num, \ 0 \ \text{as \ two}, \ 1 \ \text{as \ four}, \ 0 \ \text{as \ eight}, \ 0 \ \text{as \ sixteen}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine} \ union \ select \ "three" \ as \ word, \ 0 \ as \ num, \ 0 \ \text{as \ two}, \ 0 \ \text{as \ four}, \ 1 \ \text{as \ eight}, \ 0 \ \text{as \ sixteen}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine} \ union \ select \ "four" \ as \ word, \ 0 \ as \ num, \ 0 \ \text{as \ two}, \ 0 \ \text{as \ four}, \ 0 \ \text{as \ eight}, \ 1 \ \text{as \ sixteen}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine} \ union \ select \ "five" \ as \ word, \ 0 \ as \ num, \ 0 \ \text{as \ two}, \ 0 \ \text{as \ four}, \ 0 \ \text{as \ eight}, \ 0 \ \text{as \ sixteen}, \ 1 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine} \ union \ select \ "six" \ as \ word, \ 0 \ as \ num, \ 0 \ \text{as \ two}, \ 0 \ \text{as \ four}, \ 0 \ \text{as \ eight}, \ 0 \ \text{as \ sixteen}, \ 0 \ \text{as \ thirty-two}, \ 1 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine} \ union \ select \ "seven" \ as \ word, \ 0 \ as \ num, \ 0 \ \text{as \ two}, \ 0 \ \text{as \ four}, \ 0 \ \text{as \ eight}, \ 0 \ \text{as \ sixteen}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 1 \ \text{as \ twentynine}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine} \ union \ select \ "eight" \ as \ word, \ 0 \ as \ num, \ 0 \ \text{as \ two}, \ 0 \ \text{as \ four}, \ 0 \ \text{as \ eight}, \ 0 \ \text{as \ sixteen}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine}, \ 1 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine} \ union \ select \ "nine" \ as \ word, \ 0 \ as \ num, \ 0 \ \text{as \ two}, \ 0 \ \text{as \ four}, \ 0 \ \text{as \ eight}, \ 0 \ \text{as \ sixteen}, \ 0 \ \text{as \ thirty-two}, \ 0 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine}, \ 0 \ \text{as \ thirty-two}, \ 1 \ \text{as \ sixty-four}, \ 0 \ \text{as \ twentynine} \union} \]

(A) Write a select statement for a two-column table of the word and value for each integer:

<table>
<thead>
<tr>
<th>word</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>zero</td>
<td>0</td>
</tr>
<tr>
<td>one</td>
<td>1</td>
</tr>
<tr>
<td>two</td>
<td>2</td>
</tr>
<tr>
<td>three</td>
<td>3</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

(B) Write a select statement for the word names of the powers of two:

<table>
<thead>
<tr>
<th>word</th>
</tr>
</thead>
<tbody>
<tr>
<td>one</td>
</tr>
<tr>
<td>two</td>
</tr>
<tr>
<td>four</td>
</tr>
<tr>
<td>eight</td>
</tr>
</tbody>
</table>