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
select 38 as latitude, 122 as longitude, "Berkeley" as name union
select 42, 71, "Cambridge" 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
for database management systems (DBMS) are important, heavily used, and interesting!

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

<table>
<thead>
<tr>
<th>Latitude</th>
<th>Longitude</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>122</td>
<td>Berkeley</td>
</tr>
<tr>
<td>42</td>
<td>71</td>
<td>Cambridge</td>
</tr>
<tr>
<td>45</td>
<td>93</td>
<td>Minneapolis</td>
</tr>
</tbody>
</table>

Declarative Languages

Database Management Systems

Database management systems (DBMS) are important, heavily used, and interesting!

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 row has a value for each column

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): [http://sqlite.org/download.html](http://sqlite.org/download.html)

Use sqlite online: [http://kripken.github.io/sql.js/GUI/](http://kripken.github.io/sql.js/GUI/)

Today’s theme:
**Discussion Question**

Given the table that describes how to sum powers of 2 to form various integers:

<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>four</td>
<td>4</td>
</tr>
<tr>
<td>eight</td>
<td>8</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

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

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

**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:

```
select [expression] as [name], [expression] as [name];
```

Selecting literals creates a one-row table:

- `select "delano" as parent, "herbert" as child` union
- `select "abraham", "fillmore" union
- `select "fillmore", "delano" union
- `select "fillmore", "grover" union
- select "eisenhower", "fillmore";

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

**Arithmetic in Select Expressions**

Arithmetic expressions can combine row values and constants:

```
create table lift as
select 101 as chair, 2 as single, 2 as couple union
select 102, 4, 3 as union
select 103, 6, 11;
```

```
select chair, single + 2 * couple as total from lift;
```

**Naming Tables**

SQL is often used as an interactive language.

The result of a select statement is displayed to the user, but not stored.

A `create table` statement gives the result a name:

```
create table [name] as [select statement];
```

**Selecting parents as**

- `select "delano" as parent, "herbert" as child union
- select "abraham", "fillmore" union
- select "fillmore", "delano" union
- select "fillmore", "grover" union
- select "eisenhower", "fillmore";

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

**Creating a table containing the rows of both of their results**

- `create table parents as
  select "delano" as parent, "herbert" as child
union
  select "abraham", "fillmore" union
union
  select "fillmore", "delano" union
union
  select "fillmore", "grover" union
union
  select "eisenhower", "fillmore";
```

Parents:

<table>
<thead>
<tr>
<th>Parent</th>
<th>Child</th>
</tr>
</thead>
<tbody>
<tr>
<td>abraham</td>
<td>barack</td>
</tr>
<tr>
<td>delano</td>
<td>herbert</td>
</tr>
</tbody>
</table>

**Select Statements Project Existing Tables**

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

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

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

```
select [columns] from [table] where [condition] order by [order];
```

- `select child from parents where parent = "abraham";
- select parent from parents where parent > child;

**Arithmetic in Select Expressions**

- `select chair, single + 2 * couple as total from lift;
```

**Creating a table containing the rows of both of their results**

- `create table lift as
  select 101 as chair, 2 as single, 2 as couple union
  select 102, 4, 3 as union
  select 103, 6, 11;
```

```
select chair, single + 2 * couple as total from lift;
```

**Creating a table containing the rows of both of their results**

- `create table lift as
  select 101 as chair, 2 as single, 2 as couple union
  select 102, 4, 3 as union
  select 103, 6, 11;
```

```
select chair, single + 2 * couple as total from lift;
```