Lecture 3

Announcements

Database Management Systems

Database Management System Architecture

Query Planning

The manner in which tables are filtered, sorted, and joined affects execution time.

Select the parents of curly-furred dogs:

```sql
SELECT parent FROM parents, dogs
WHERE child = name AND fur = 'curly';
```

Join all rows of parents to all rows of dogs, filter by `child = name` and `fur = 'curly'`.

Join only rows of parents and dogs where `child = name`, filter by `fur = 'curly'`.

Filter dogs by `fur = 'curly'`, join result with all rows of parents, filter by `child = name`.

Filter dogs by `fur = 'curly'`, join only rows of result and parents where `child = name`.

Local Tables

A `create table` statement names a table globally.

```sql
CREATE TABLE parents
AS
SELECT "abraham" AS parent,
    "barack" AS child
UNION
SELECT "abraham",
    "clinton"
UNION
SELECT "delano",
    "herbert"
UNION
SELECT "fillmore",
    "abraham"
UNION
SELECT "fillmore",
    "delano"
UNION
SELECT "fillmore",
    "grover"
UNION
SELECT "eisenhower",
    "fillmore";
```

A `WITH` clause of a `SELECT` statement names a table that is local to the statement.

```sql
WITH best(dog) AS (
    SELECT "eisenhower"
    UNION
    SELECT "barack"
)
SELECT parent FROM parents, best(dog)
WHERE child = dog;
```

Part of the `SELECT` statement

<table>
<thead>
<tr>
<th>Parent</th>
<th>Child</th>
</tr>
</thead>
<tbody>
<tr>
<td>abraham</td>
<td>barack</td>
</tr>
<tr>
<td>abraham</td>
<td>clinton</td>
</tr>
<tr>
<td>delano</td>
<td>herbert</td>
</tr>
<tr>
<td>fillmore</td>
<td>abraham</td>
</tr>
<tr>
<td>fillmore</td>
<td>delano</td>
</tr>
<tr>
<td>fillmore</td>
<td>grover</td>
</tr>
<tr>
<td>eisenhower</td>
<td>fillmore</td>
</tr>
</tbody>
</table>

Local Tables
Example: Relationships

(A) What are appropriate names for the columns in this result?
(B) How many rows and columns will result?

```sql
WITH
    parent AS (SELECT a.child, b.child
                 FROM parents AS a, parents AS b
                 WHERE a.parent = b.parent AND
                       a.child != b.child)

SELECT child, second
FROM parents, parent
WHERE parent = first;
```

Recursive Local Tables

Local Tables can be Declared Recursively

An ancestor is your parent or an ancestor of your parent

```sql
CREATE TABLE parents AS
    SELECT "abraham" AS parent,
           "barack" AS child
    UNION
    ...  
```

Global Names for Recursive Tables

To create a table with a global name, you need to select the contents of the local table

```sql
CREATE TABLE odds AS
    WITH
        odds(n) AS (SELECT 1
                     UNION
                     SELECT n + 2
                     FROM odds
                     WHERE n < 15)
    SELECT n
    FROM odds;
```

Limits on Recursive Select Statements

Recursive table definitions are only possible within a with clause

No mutual recursion: two or more tables cannot be defined in terms of each other

```sql
WITH
    odds(n) AS (SELECT 1
                 UNION
                 SELECT n + 2
                 FROM odds
                 WHERE n < 15)
    evens(x) AS (SELECT x + 1
                  FROM odds)
    ints(x) AS (SELECT 1
                 UNION
                 SELECT x - 1
                 FROM ints
                 UNION
                 SELECT x + 1
                 FROM ints)

SELECT x
FROM ints;
```

String Examples

Language is Recursive

Noun phrases can contain relative pronouns that introduce relative clauses

The dog chased the cat
  that chased the bird
The dog chased the cat
  the bird chased
The dog the bird the cat chased chased chased me
Bulldogs bulldogs bulldogs fight fight fight

Integer Examples

```sql
WITH
    odds(n) AS (SELECT 1
                 UNION
                 SELECT n + 2
                 FROM odds
                 WHERE n < 15)
    evens(x) AS (SELECT x + 1
                  FROM odds)
    ints(x) AS (SELECT 1
                 UNION
                 SELECT x - 1
                 FROM ints
                 UNION
                 SELECT x + 1
                 FROM ints)

SELECT x
FROM ints;
```
Input-Output Tables

A table containing the inputs to a function can be used to map from output to input.

```sql
create table pairs as
  with
    i(n) as (
      select 1 union
      select n+1 from i where n < 50
    )
  select a.n as x, b.n as y
  from i as a, i as b
  where a.n <= b.n;
```

What integers can I add/multiply together to get 24?

(Demo)

Example: Pythagorean Triples

All triples $a$, $b$, $c$ such that $a^2 + b^2 = c^2$ with $a < b < c$.

```sql
create table pairs as
  with
    i(n) as (
      select 1 union
      select n+1 from i where n < 20
    )
  select a.n as a, b.n as b, c.n as c
  from __________________________________________
  where _______________________________________
  and a.n * a.n + b.n * b.n = c.n * c.n;
```

A Very Interesting Number

The mathematician G. H. Hardy once remarked to the mathematician Srinivasa Ramanujan...