

## 61A Lecture 34

Friday, November 21

## Announcements

- Project 4 due Friday 11/21 @ 11:59pm
- Please submit project 4 in two different ways:
  - `python3 ok --submit` and set your group on `ok.cs61a.org`
  - Copy files to class account and **submit proj4**
- A correction to `tests/q20.py` was released yesterday afternoon
- Homework 9 (6 pts) due Wednesday 11/26 @ 11:59pm
  - Homework Party Monday evening, location TBD
- Guest in live lecture, TA Soumya Basu, on Monday 11/24 (videos by John)
- No lecture on Wednesday 11/26 (turkey)
- No lab next Tuesday 11/25 & Wednesday 11/26

## Local Tables

## Local Tables

A `create table` statement names a table globally

```
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";
```

Parent	Child
abraham	barack
abraham	clinton
delano	herbert
fillmore	abraham
fillmore	delano
fillmore	grover
eisenhower	fillmore

## Local Tables

A `create table` statement names a table globally

A `with` clause of a `select` statement names a table that is local to the statement

```
create table parents as
select "abraham" as parent, "barack" as child union
...
with
best(dog) as (
select "eisenhower" union
select "barack"
)
select parent from parents, best where child=dog;
```

Part of the select statement

Local table only exists for this select

parent
abraham

(Demo)

## Example: Relationships

(A) What are appropriate names for the columns in this result?

(B) How many rows will result?

```
with
siblings
what(first, second) 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 as nephew, second as uncle
from parents, what where parent=first;
```

parent	child	first	second
abraham	barack	abraham	delano

## Recursive Local Tables

## Local Tables can be Declared Recursively

An ancestor is your parent or an ancestor of your parent

```
create table parents as
select "abraham" as parent, "barack" as child union
...
with
ancestors(ancestor, descendent) as (
select parent, child from parents union
select ancestor, child
from ancestors, parents
where parent = descendent
)
select ancestor from ancestors where descendent="herbert";
```

ancestor
delano
fillmore
eisenhower

## Global Names for Recursive Tables

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

```
create table odds as
with
  odds(n) as (
    select 1 union
    select n+2 from odds where n < 15;
  )
select n from odds;
```

n
1
3
5
7
9
11
13
15

Which names above can change without affecting the result?

## 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

```
with
  odds(x) as (
    select 1 union select x+1 from evens
  ),
  evens(x) as (
    select x+1 from odds
  )
select x from odds
```

*Nope!*

No tree recursion: the table being defined can only appear once in a from clause

```
with
  ints(x) as (
    select 1 union
    select x-1 from ints union
    select x+1 from ints
  )
select x from ints;
```

*Nope!*

```
with
  ints(x) as (
    select 1 union
    select a.x + b.x
      from ints as a, ints as b
  )
select x from ints;
```

*Nope!*

## 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  
that the bird chased

The dog chased the cat  
the bird chased

The dog the bird the cat chased chased chased me

Bulldogs bulldogs bulldogs fight fight fight

(Demo)

## Integer Examples

## Input-Output Tables

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

```
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?

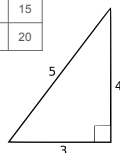
(Demo)

## Example: Pythagorean Triples

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

```
with
  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 i as a, i as b, i as c
where a.n < b.n and a.n*a.n + b.n*b.n = c.n*c.n;
```

a	b	c
3	4	5
5	12	13
6	8	10
8	15	17
9	12	15
12	16	20



## Example: Fibonacci Sequence

Computing the next Fibonacci number requires both the previous and current numbers

```
create table fibs as
with
  fib(previous, current) as (
    select 0, 1 union
    select current, previous+current from fib
    where current <= 14.15926535
  )
select previous as n from fib;
```

n
0
1
1
2
3
5
8
13

## A Very Interesting Number

---

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

(Demo)

---