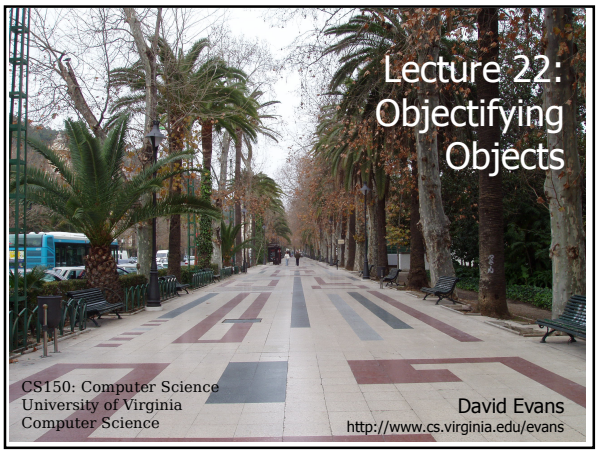
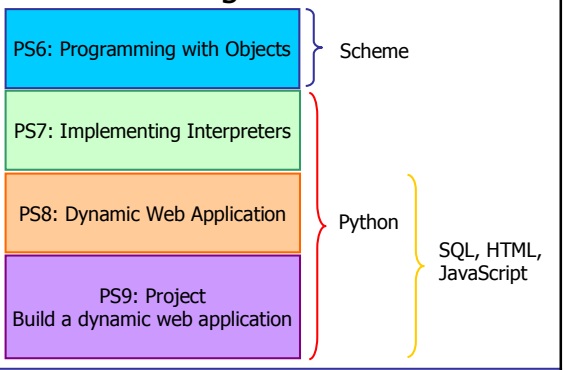


Menu

- PS9 (Preview)
- Objects
- Databases (PS5)

Remaining Problem Sets

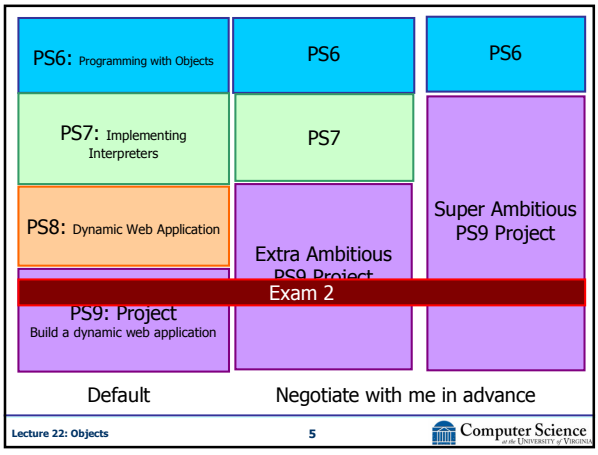


PS9 Assignment

Problem: Make an interesting dynamic web site.

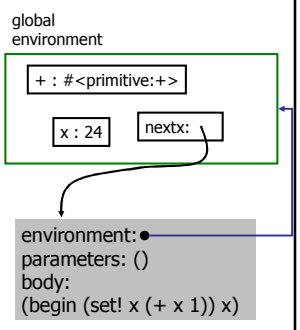
- Teams of 1-52 students
- Can be anything you want that:
  - Involves interesting computation
  - Follows University's use policies (or on external server)
  - Complies with ADA Section 508 (accessible)

Course Forum



from Class 19: nextx

```
(define x 0)
(define (nextx)
  (set! x (+ x 1))
  x)
> (nextx)
1
> (set! x 23)
> (next x)
24
```



A Better Counter

- The place that keeps track of the count should be part of the counter, not part of the global environment
  - Can have more than one counter
  - Counter state is *encapsulated*: can only be modified by counter procedure
- Can we do this?

Application Rule 2:

1. Construct a new environment, whose parent is the environment to which the environment pointer of the applied procedure points.
2. Create a place in that frame for each parameter containing the value of the corresponding operand expression.
3. Evaluate the body in the new environment. Result is the value of the application.

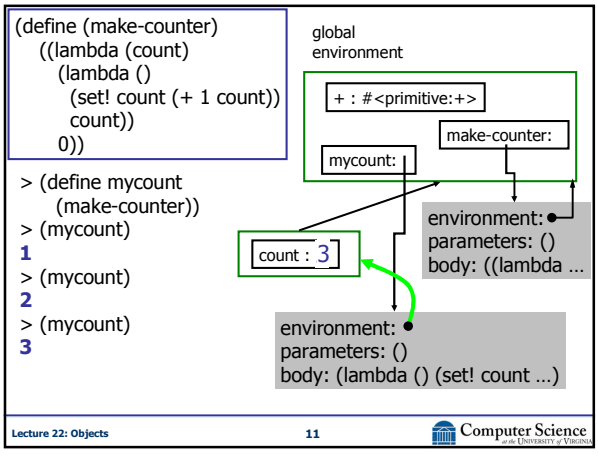
A Better Counter

```
(define (make-counter)
  ((lambda (count)
     (lambda ()
       (set! count (+ 1 count))
       count))
   0))
```


Sweeter Version

```
(define (make-counter)
  (let ((count 0))
    (lambda ()
      (set! count (+ 1 count))
      count)))
```

This is easier to read (syntactic sugar), but means the same thing. The place for count is created because of the application that is meant by the let.



An Even Better Counter

```
(define (make-counter)
  (let ((count 0))
    (lambda (message)
      (cond ((eq? message 'reset!)
             (set! count 0))
            ((eq? message 'next!)
             (set! count (+ 1 count)))
            ((eq? message 'current) count)
            (else
              (error "Unrecognized message"))))))
```

Using Counter

```
> (define bcounter (make-counter))
> (bcounter 'next)
> (bcounter 'next)
> (bcounter 'next)
> (bcounter 'how-many)
3
> (bcounter 'reset)
> (bcounter 'how-many)
0
```


Objects

An *object* packages:

- **state** ("instance variables")
- **procedures** for manipulating and observing that state ("methods")

Why is this useful?

Problem-Solving Strategies

- PS1-PS4: Functional Programming
  - Focused on **procedures**
  - Break a problem into procedures that can be combined to solve it
- PS5: Imperative Programming
  - Focused on **data**
  - Design data for representing a problem and procedures for updating that data

PS5

How are commercial databases different from what you implemented for PS5?

UVa's Integrated Systems Project to convert all University information systems to use an Oracle database was originally budgeted for **$58.2 Million** (starting in 1999). Actual cost ended up over $100 Million.

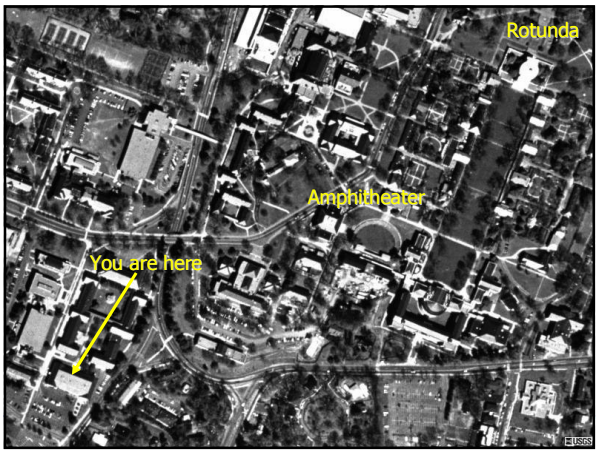
http://www.virginia.edu/isp/

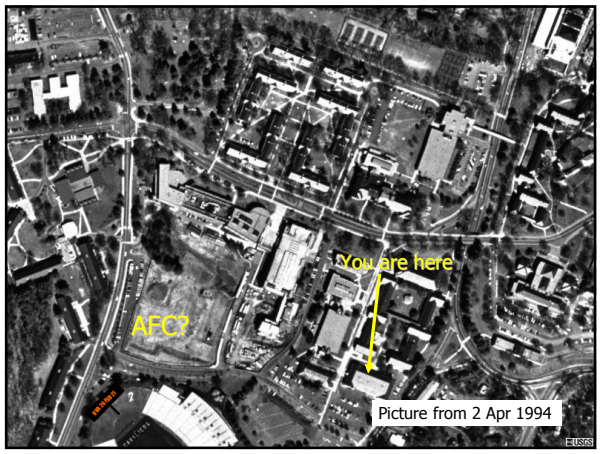
Real Databases

- Atomic Transactions: a transaction may involve many modifications to database tables, but the changes should only happen if the whole transaction happens (e.g., don't charge the credit card unless the order is sent to the shipping dept)
- Security: limit read/write access to tables, entries and fields
- Storage: need to efficiently store data on disk, provide backup mechanisms
- Scale: to support really big data tables, real databases do lots of clever things

How big are big databases?

- Microsoft TerraServer
  - Claimed biggest in 1998
  - Aerial photos of entire US (1 meter resolution)





Big Databases

- Microsoft TerraServer
  - 3.3 Terabytes (claimed biggest in 1998)
  - 1 Terabyte = 2^{40} Bytes ~ 1 Trillion Bytes
- Google Maps (possibly bigger?)
- Winter TopTen:
  - Yahoo! (100TB), Amazon (25TB)
- Wal-Mart
  - 285 Terabytes (2003)
- Stanford Linear Accelerator (BaBar)
  - 500 Terabytes (30 KB per particle collision)

How much work?

- table-select is in $\Theta(n)$ where n is the number of entries in the table
  - Would your table-select work for Wal-Mart?
  - If 1M entry table takes 1s, how long would it take Wal-Mart to select from 285TB ~ 2 Trillion Entries? 2 000 000s ~ 23 days

How do expensive databases perform table-select so much faster?

Problem-Solving Strategies

- PS1-PS4: Functional Programming
  - Focused on **procedures**
  - Break a problem into procedures that can be combined to solve it
- PS5: Imperative Programming
  - Focused on **data**
  - Design data for representing a problem and procedures for updating that data

Problem-Solving Strategies

- PS6: "Object-Oriented Programming"
  - Focused on **objects**: package procedures and state
  - Model a problem by dividing it into objects
  - Lots of problems in real (and imaginary) worlds can be thought of this way

Counter Object

```
(define (make-counter)
  (let ((count 0))          Instance variable
    (lambda (message)
      (cond ((eq? message 'reset!)     Methods
             (set! count 0))
            ((eq? message 'next!)
             (set! count (+ 1 count)))
            ((eq? message 'current) count)
            (else
              (error "Unrecognized message"))))))
```


Defining **ask**

(ask *Object Method*)

```
> (define bcounter (make-counter))
> (ask bcounter 'current)
0
> (ask bcounter 'next)
> (ask bcounter 'current)
1
```

```
(define (ask object message)
  (object message))
```


Charge

- Start PS6 early
  - It is challenging
  - Opportunity for creativity
  - If you want to be assigned a partner, send email before midnight tomorrow
- Start thinking about Project ideas
  - If you want to do an "extra ambitious" project convince me your idea is worthy before March 26 (ps7&8)/April 4 (ps8)