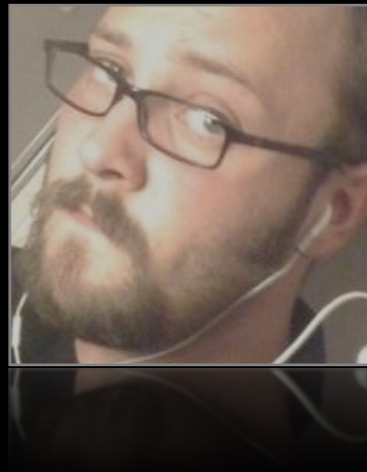


Cocoa for Sys Admins



Zack Smith
@acidprime



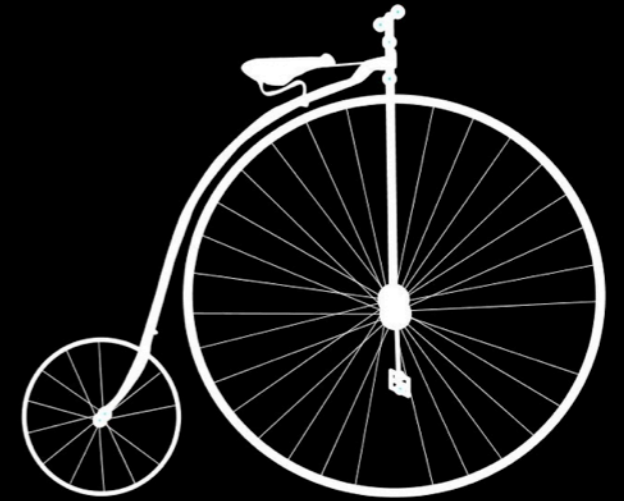
The background of the entire image is a dense, vertical stream of green characters and numbers, reminiscent of the 'Matrix' digital rain effect. The characters are white and green, falling from the top of the frame. A black rectangular box with a thin green border is centered in the middle of the image, containing white text.

Wake up sys admins
follow the blue cubes...

Cocoa for Sys Admins

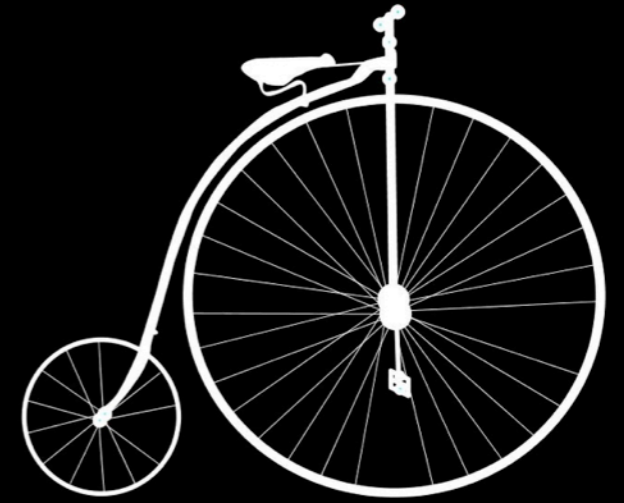
Cocoa for Sys Admins

- **Why** would you do this ?



Cocoa for Sys Admins

- **Why** would you do this ?

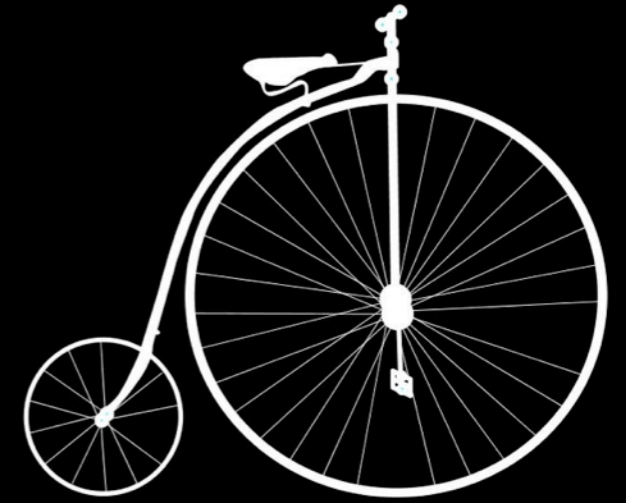


- **How** you can do it.



Cocoa for Sys Admins

- **Why** would you do this ?



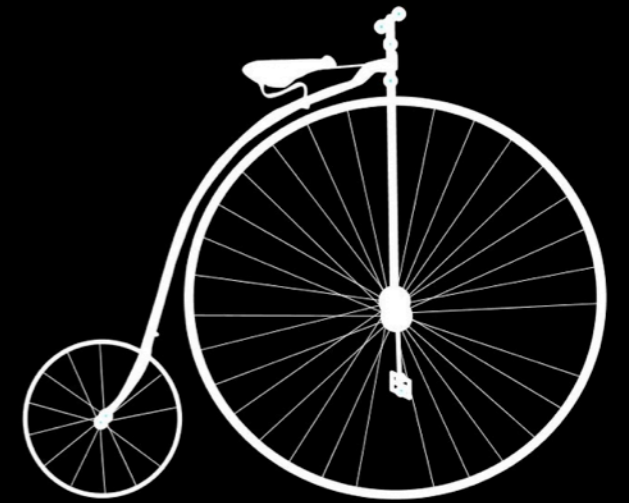
- **How** you can do it.



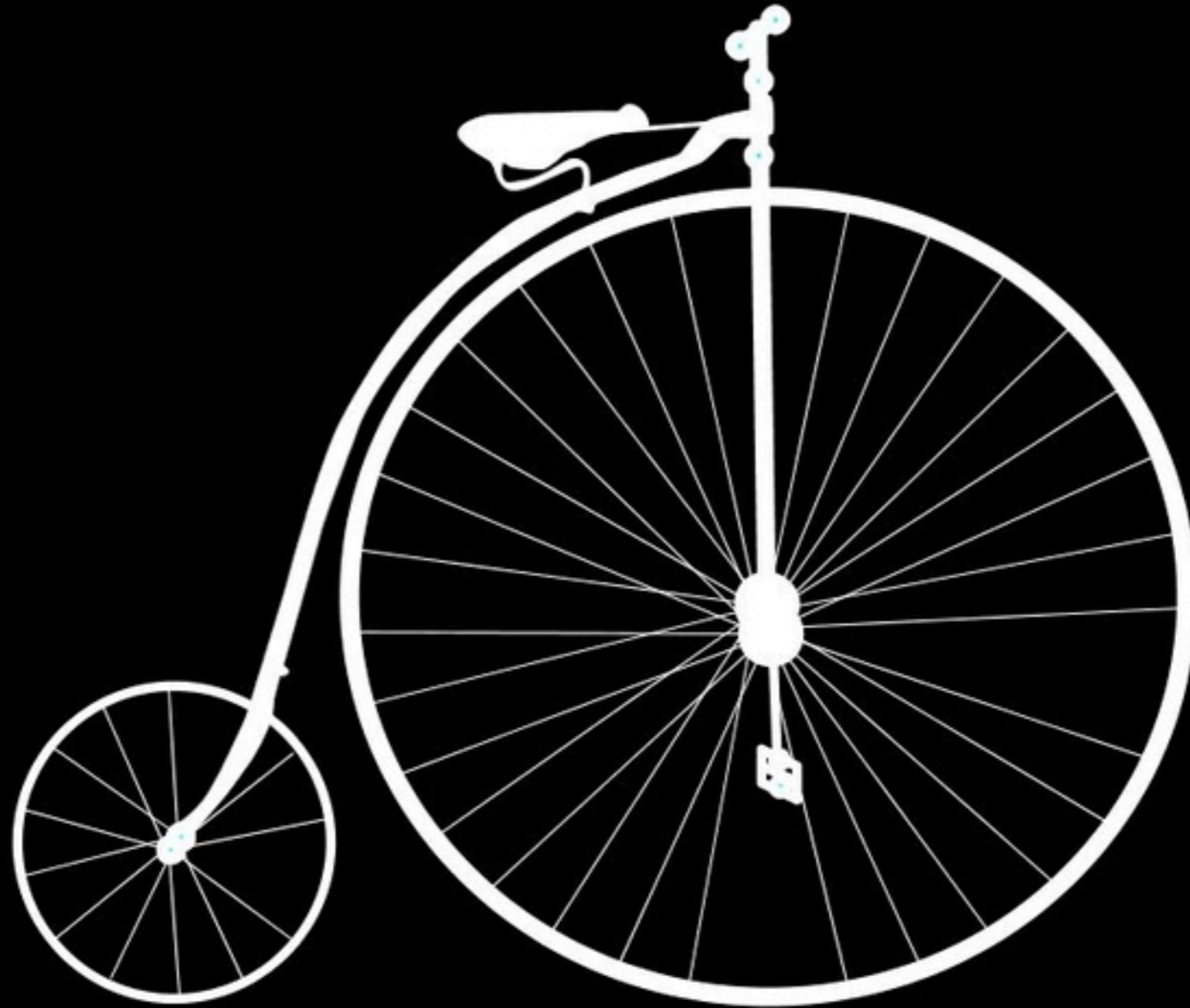
- **What** is possible?



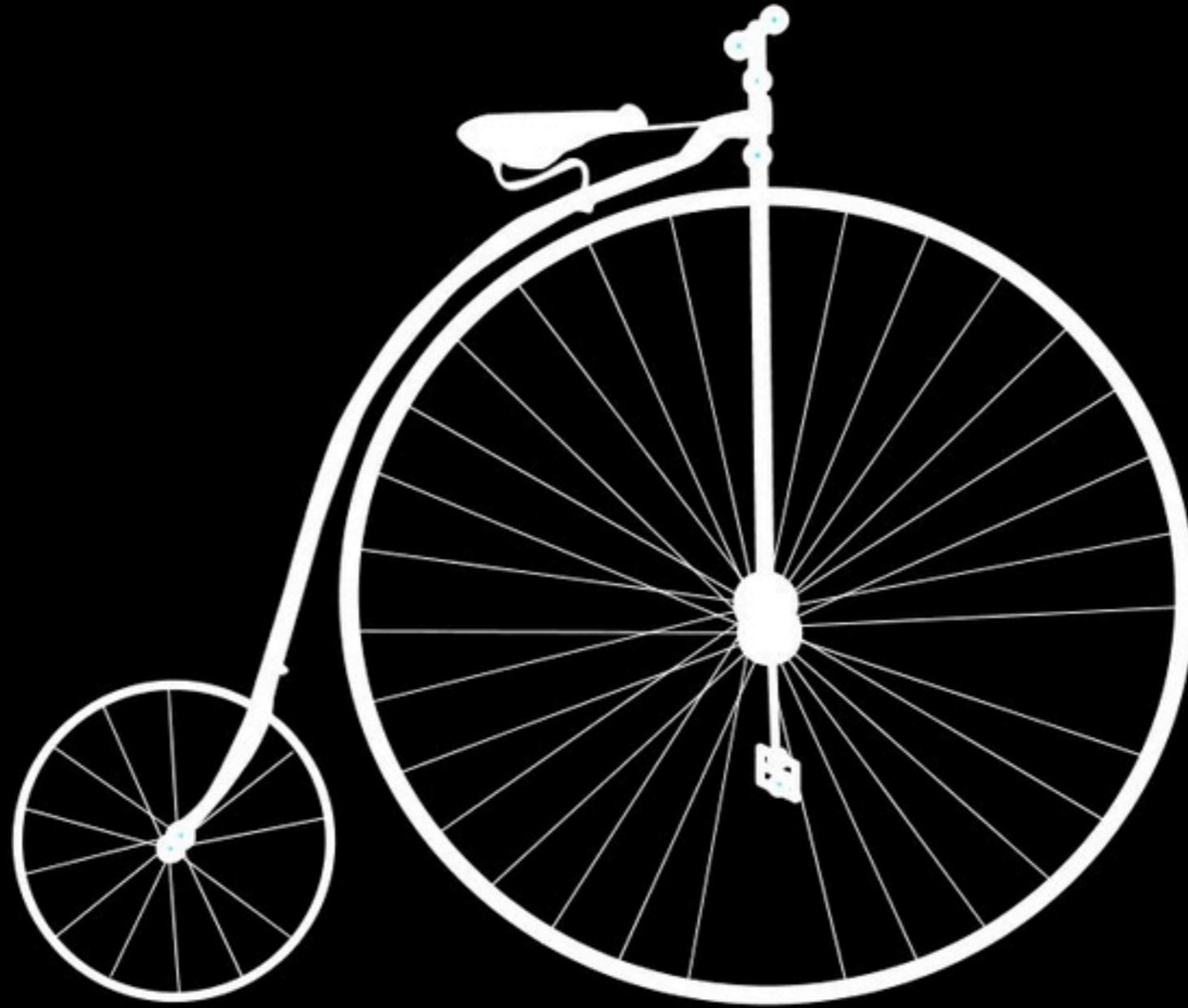
- Why would you do this ?



`#!/bin/bash`



`#!/bin/bash`



10-10

“Developer” Applications



“Developer” Applications



“Developer” Applications



WWSAD

What would a sys admin do?

#devops



#devops



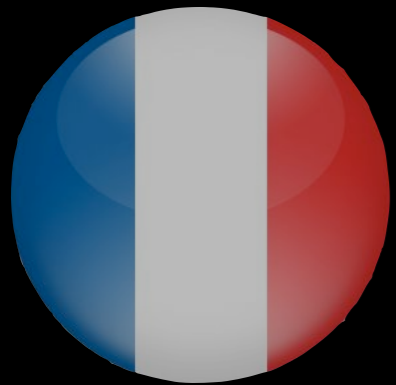
How



Learn the Melody of a language

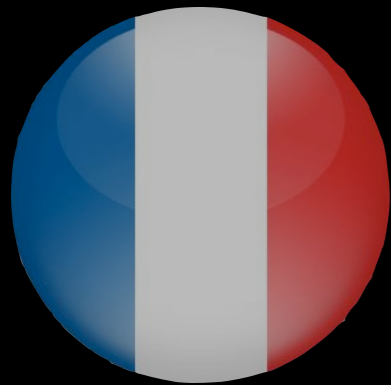
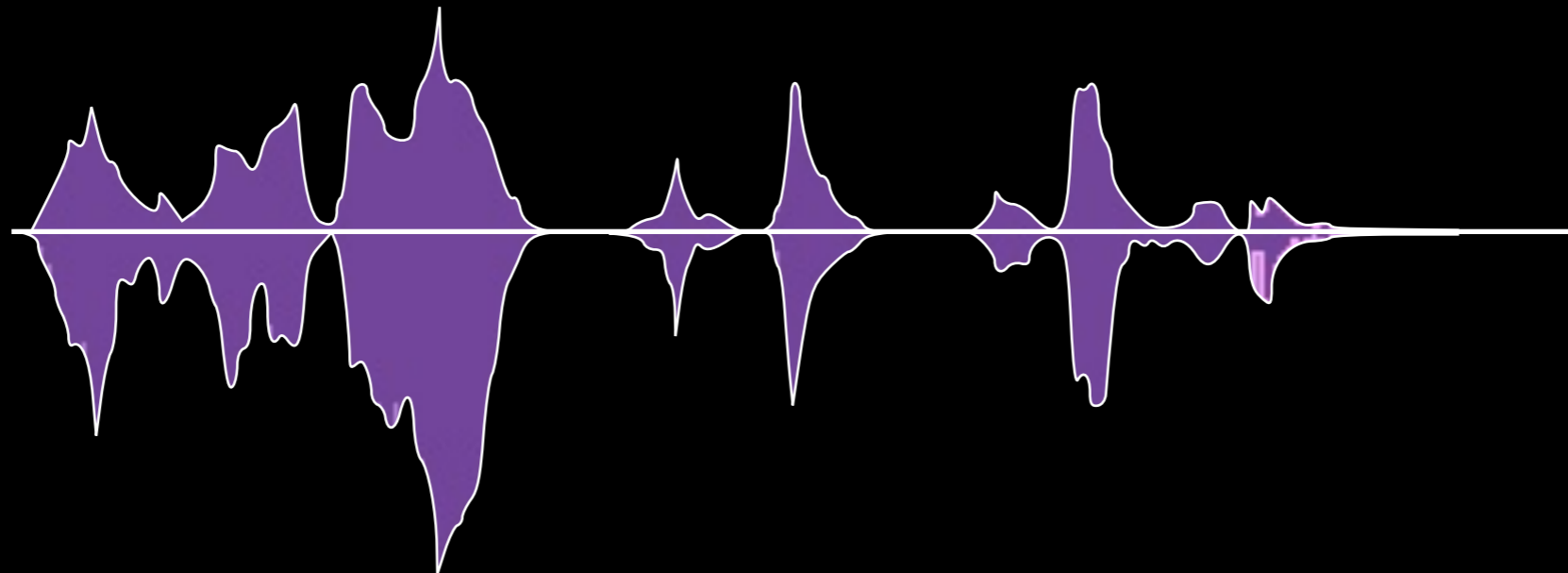


Jag förstår lite svenska



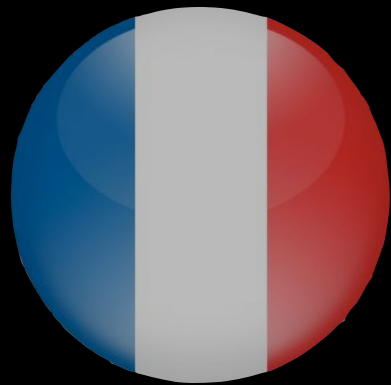
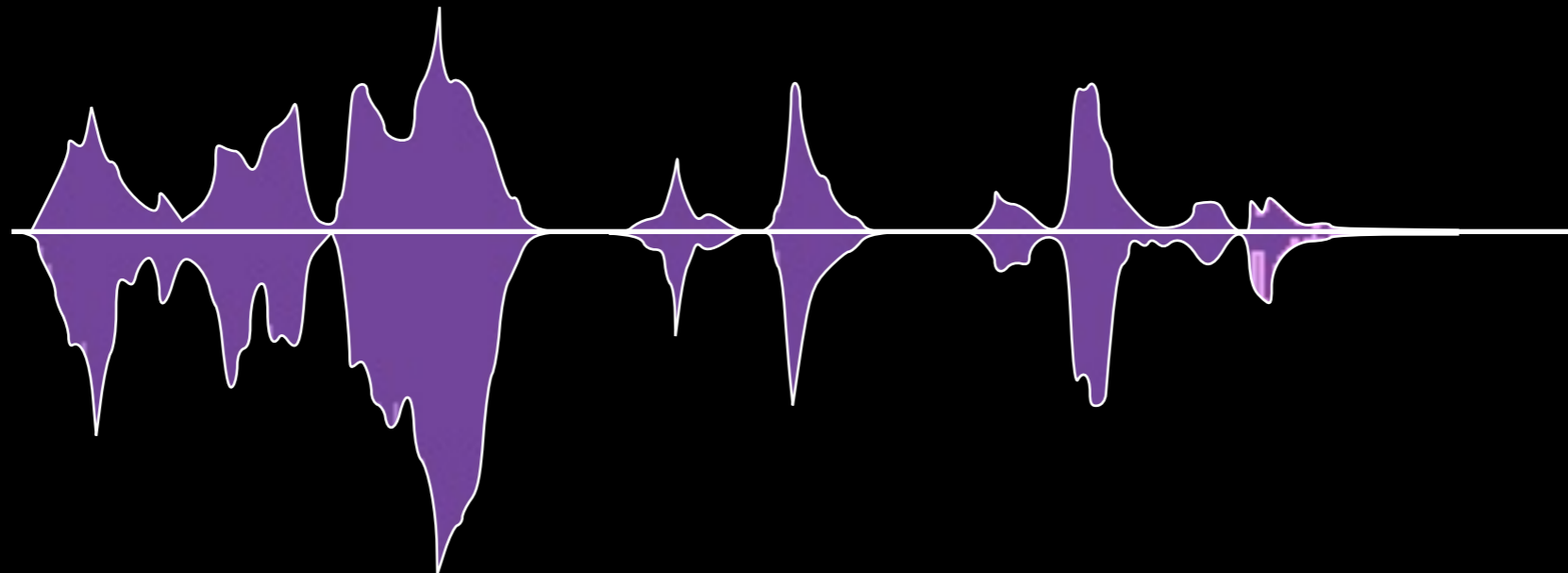
Je comprends un peu le français

Learn the Melody of a language



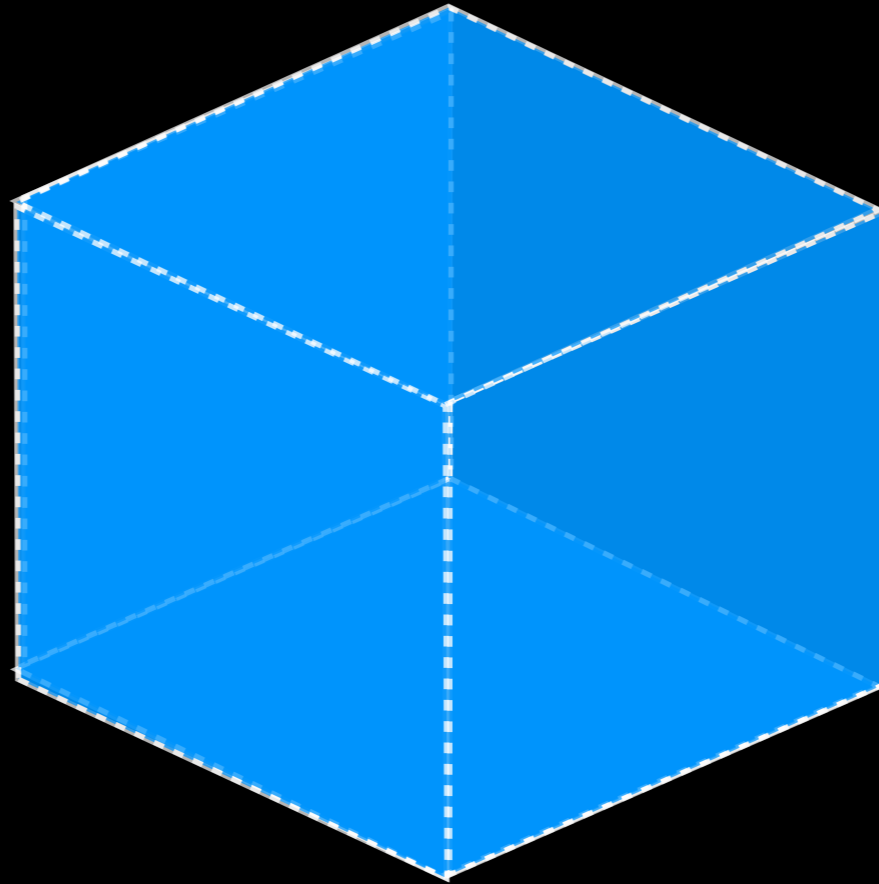
Je comprends un peu le français

Learn the Melody of a language



Je comprends un peu le français

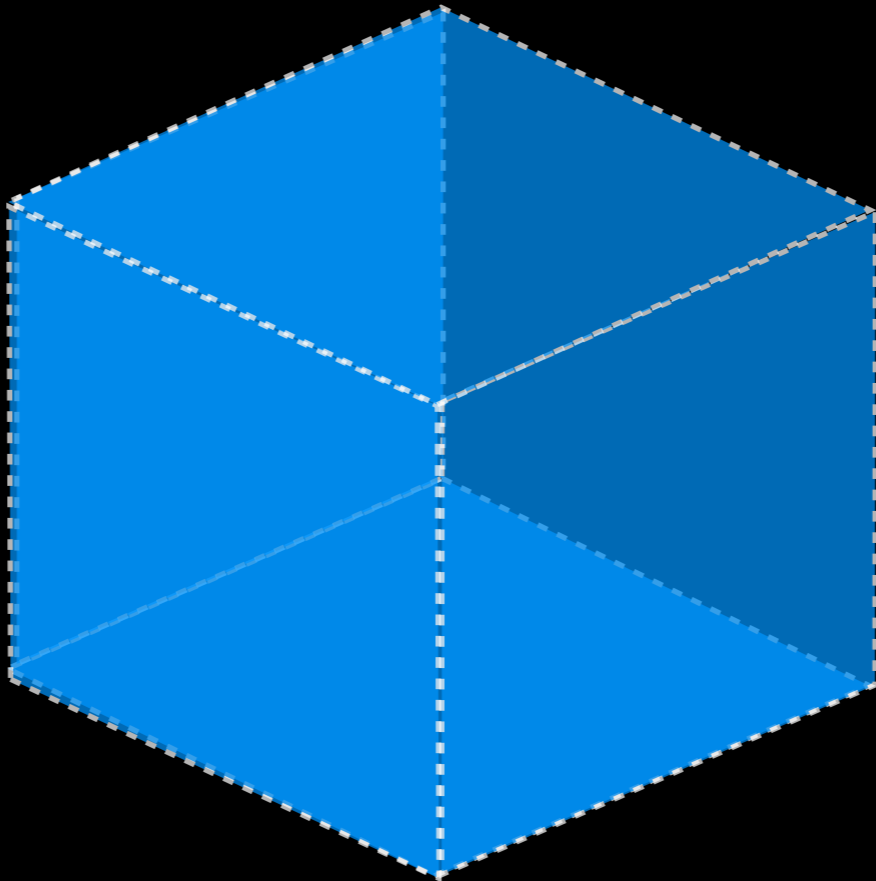




```
...  
C  
return [self alloc];  
}  
...  
+ (instancetype)copy {  
C  
self = [super copy];  
if (self == nil) {  
return nil;  
}  
else {  
self = [super copy];  
}  
return self;  
}  
...  
C  
[self autorelease];  
[autorelease release]; _autorelease = nil;  
if (self == nil) {  
return nil;  
}  
}
```

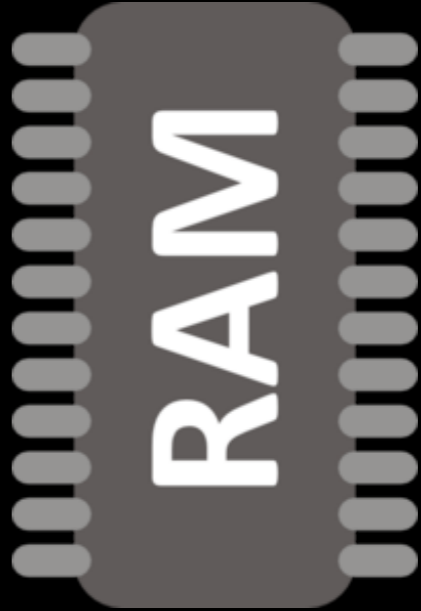
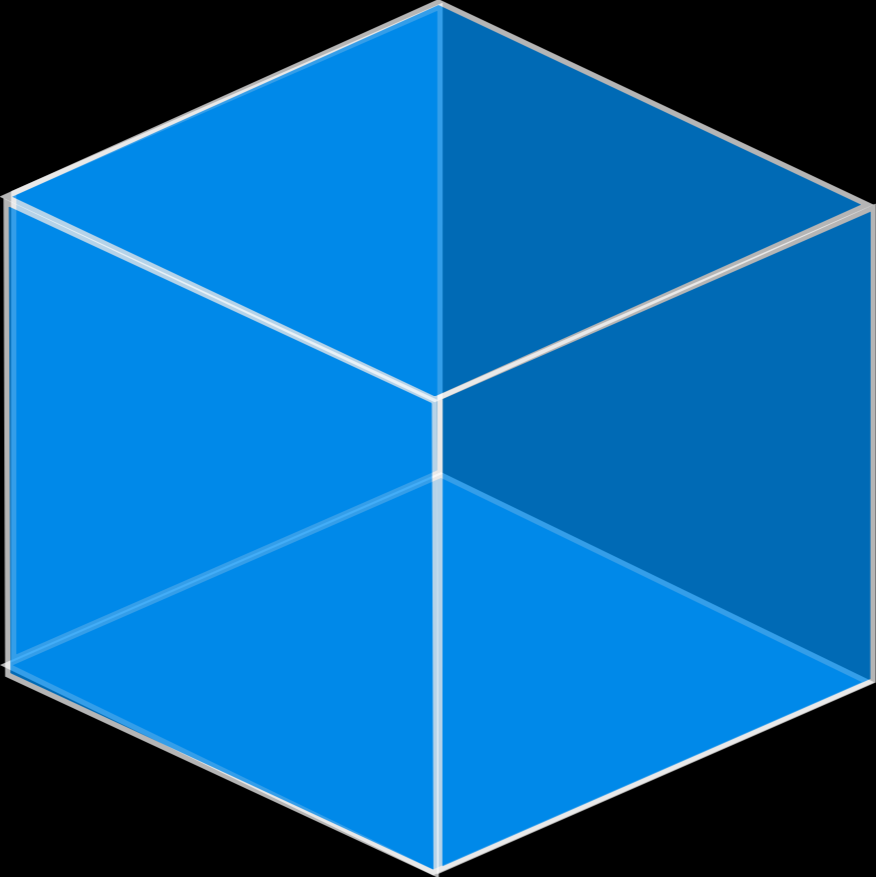
```
...  
C  
return [self alloc];  
}  
...  
+ (instancetype)copy {  
C  
self = [super copy];  
if (self == nil) {  
return nil;  
}  
else {  
self = [super copy];  
}  
return self;  
}  
...  
C  
[self autorelease];  
[autorelease release]; _autorelease = nil;  
if (self == nil) {  
return nil;  
}  
}
```

Class *obj = [[Class alloc] init];

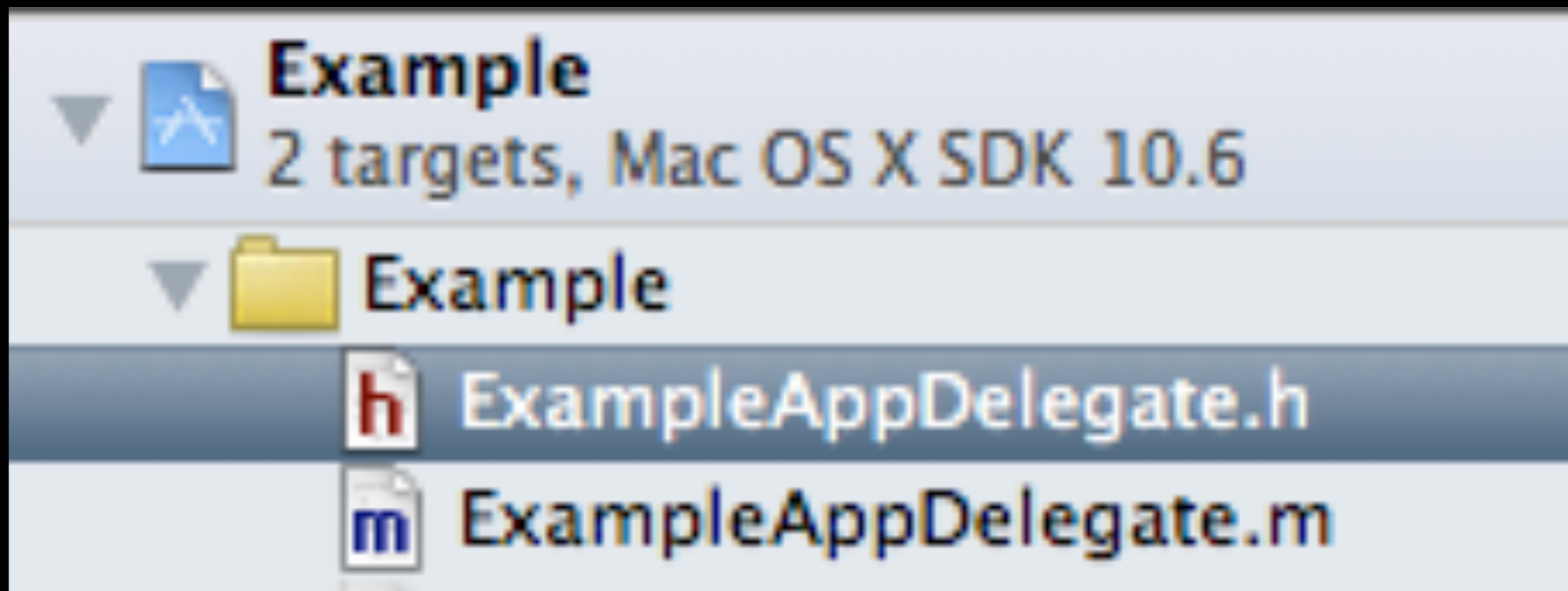
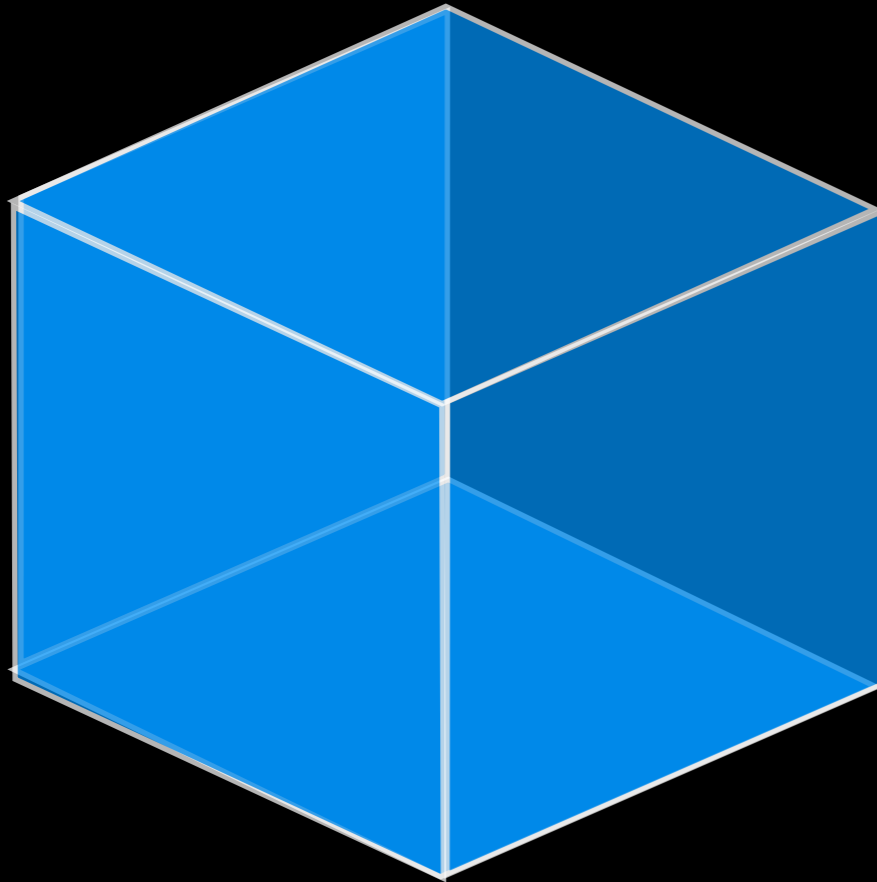


```
...  
C  
return [self toString()]  
}  
}  
  
- (id) NSString *toString  
C  
self = [super toString];  
if (self == nil) {  
    return nil;  
}  
else {  
    NSString *str = [NSString stringWithFormat:@"%s", self];  
    return str;  
}  
}  
- (id) NSString *toString  
C  
[self toString];  
[NSString stringWithFormat:@"%s", self];  
if (self == nil) {  
    return nil;  
}  
}
```

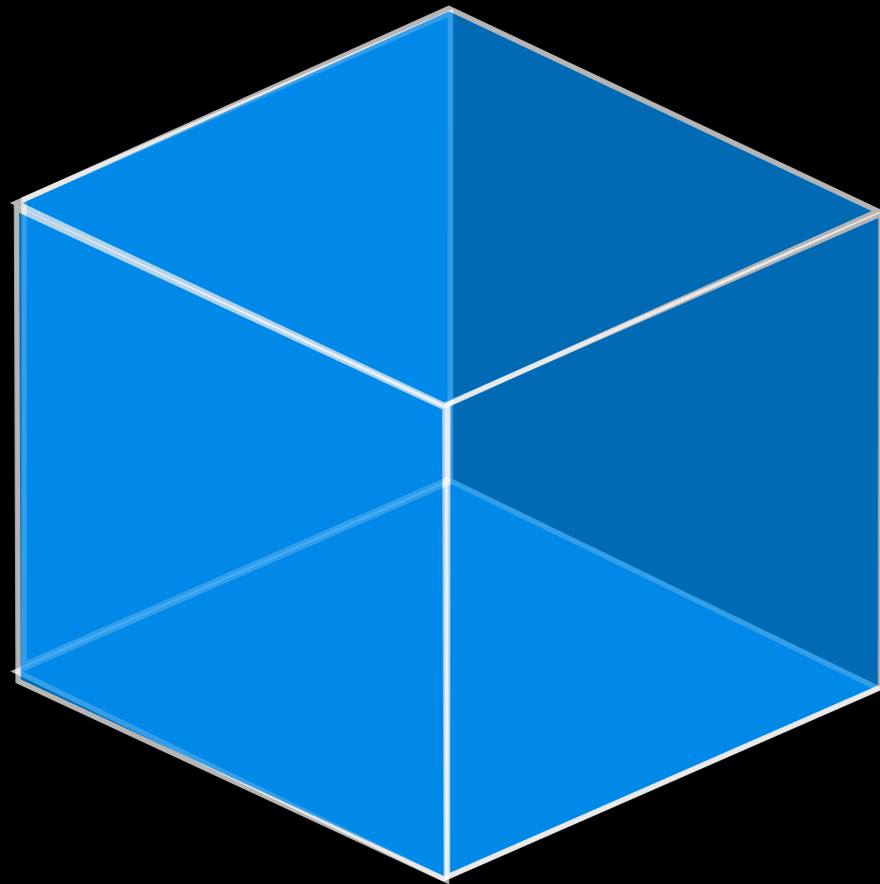
```
...  
C  
return [self toString()]  
}  
}  
  
- (id) NSString *toString  
C  
self = [super toString];  
if (self == nil) {  
    return nil;  
}  
else {  
    NSString *str = [NSString stringWithFormat:@"%s", self];  
    return str;  
}  
}  
- (id) NSString *toString  
C  
[self toString];  
[NSString stringWithFormat:@"%s", self];  
if (self == nil) {  
    return nil;  
}  
}
```



AppDelegate



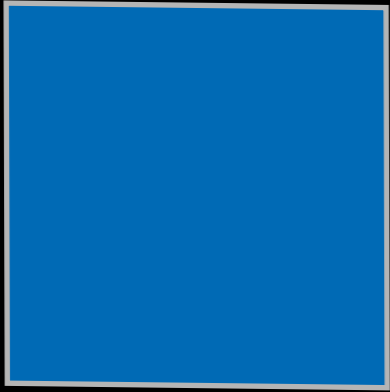
Methods



```
def __init__(self, x, y, z):  
    self.x = x  
    self.y = y  
    self.z = z  
  
def __str__(self):  
    return f'({self.x}, {self.y}, {self.z})'  
  
def __repr__(self):  
    return f'Cube({self.x}, {self.y}, {self.z})'  
  
def __add__(self, other):  
    return Cube(self.x + other.x, self.y + other.y, self.z + other.z)  
  
def __sub__(self, other):  
    return Cube(self.x - other.x, self.y - other.y, self.z - other.z)  
  
def __mul__(self, other):  
    return Cube(self.x * other.x, self.y * other.y, self.z * other.z)  
  
def __div__(self, other):  
    return Cube(self.x / other.x, self.y / other.y, self.z / other.z)
```

```
def __init__(self, x, y, z):  
    self.x = x  
    self.y = y  
    self.z = z  
  
def __str__(self):  
    return f'({self.x}, {self.y}, {self.z})'  
  
def __repr__(self):  
    return f'Cube({self.x}, {self.y}, {self.z})'  
  
def __add__(self, other):  
    return Cube(self.x + other.x, self.y + other.y, self.z + other.z)  
  
def __sub__(self, other):  
    return Cube(self.x - other.x, self.y - other.y, self.z - other.z)  
  
def __mul__(self, other):  
    return Cube(self.x * other.x, self.y * other.y, self.z * other.z)  
  
def __div__(self, other):  
    return Cube(self.x / other.x, self.y / other.y, self.z / other.z)
```

Methods



```
. (defn h
  []
  (return [self to the n]))
}

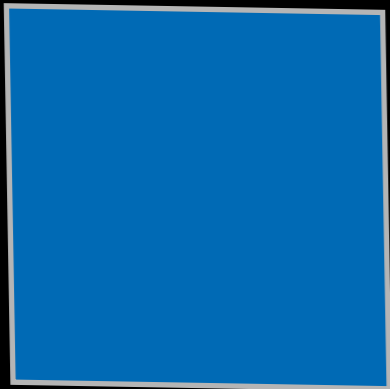
. (defn h-to-string "h"
  []
  (let [self (js/HTMLDocument)]
    (if (not (is-a? self HTMLDocument))
      (return "h")
      (let [h (self.querySelector "h")]
        (return (if (not (nil? h))
                  (js/get h "innerHTML")
                  "h"))))
    (return self))
  )
}

. (defn h-to-string
  [url]
  (let [self (js/HTMLDocument)]
    (if (not (is-a? self HTMLDocument))
      (return "h")
      (let [h (self.querySelector "h")]
        (return (if (not (nil? h))
                  (js/get h "innerHTML")
                  "h"))))
    (return self))
  )
}
```

```
. (defn m
  []
  (return [self to the n]))
}

. (defn m-to-string "m"
  []
  (let [self (js/HTMLDocument)]
    (if (not (is-a? self HTMLDocument))
      (return "m")
      (let [m (self.querySelector "m")]
        (return (if (not (nil? m))
                  (js/get m "innerHTML")
                  "m"))))
    (return self))
  )
}

. (defn m-to-string
  [url]
  (let [self (js/HTMLDocument)]
    (if (not (is-a? self HTMLDocument))
      (return "m")
      (let [m (self.querySelector "m")]
        (return (if (not (nil? m))
                  (js/get m "innerHTML")
                  "m"))))
    (return self))
  )
}
```



Methods



```
@interface AppDelegate : NSObject
{
@private
    UIWindow *window;
}
```

```
-(IBAction)buttonClicked:(id)sender;
```



```
@implementation AppDelegate
```

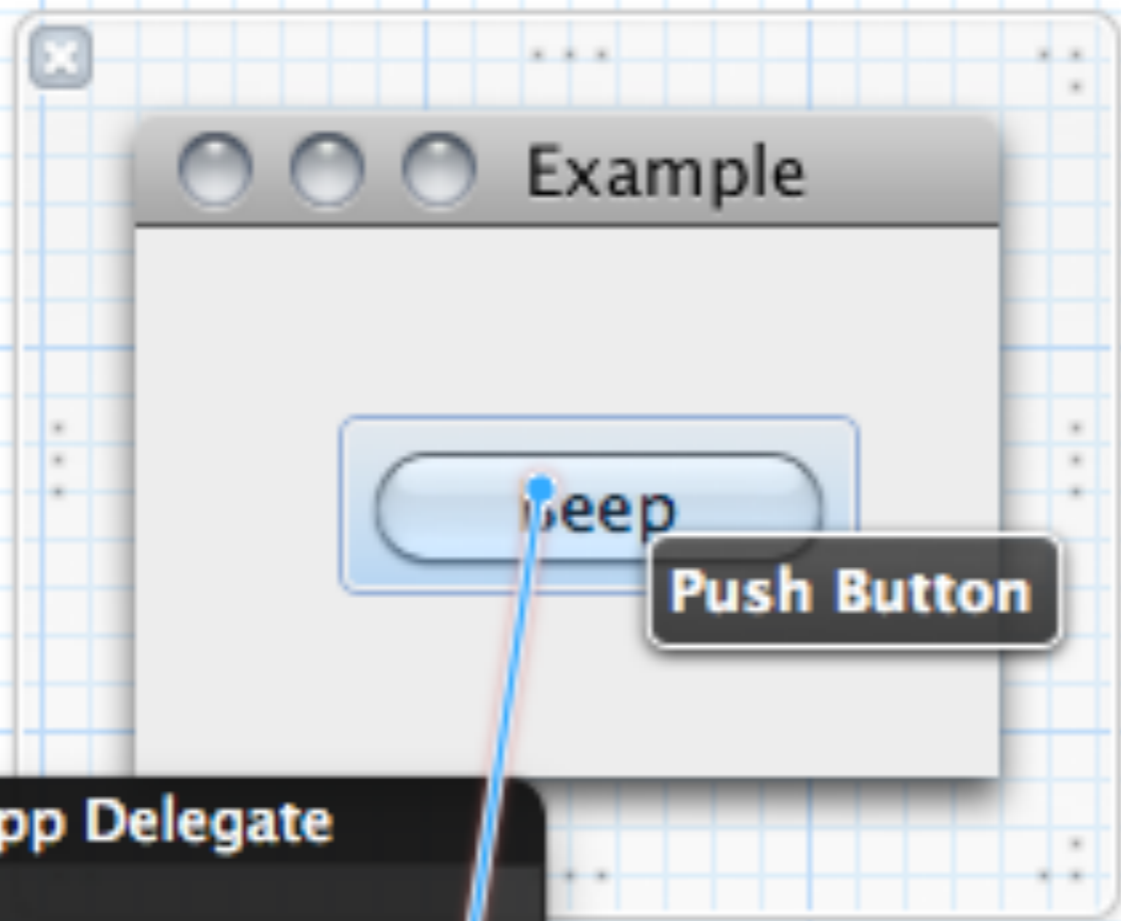
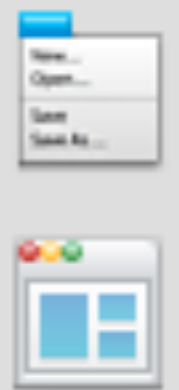
```
@synthesize window;
```

```
-(IBAction)buttonClicked:(id)sender;
```

```
{
```

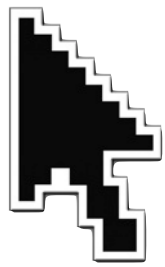
```
    NSBeep();
```

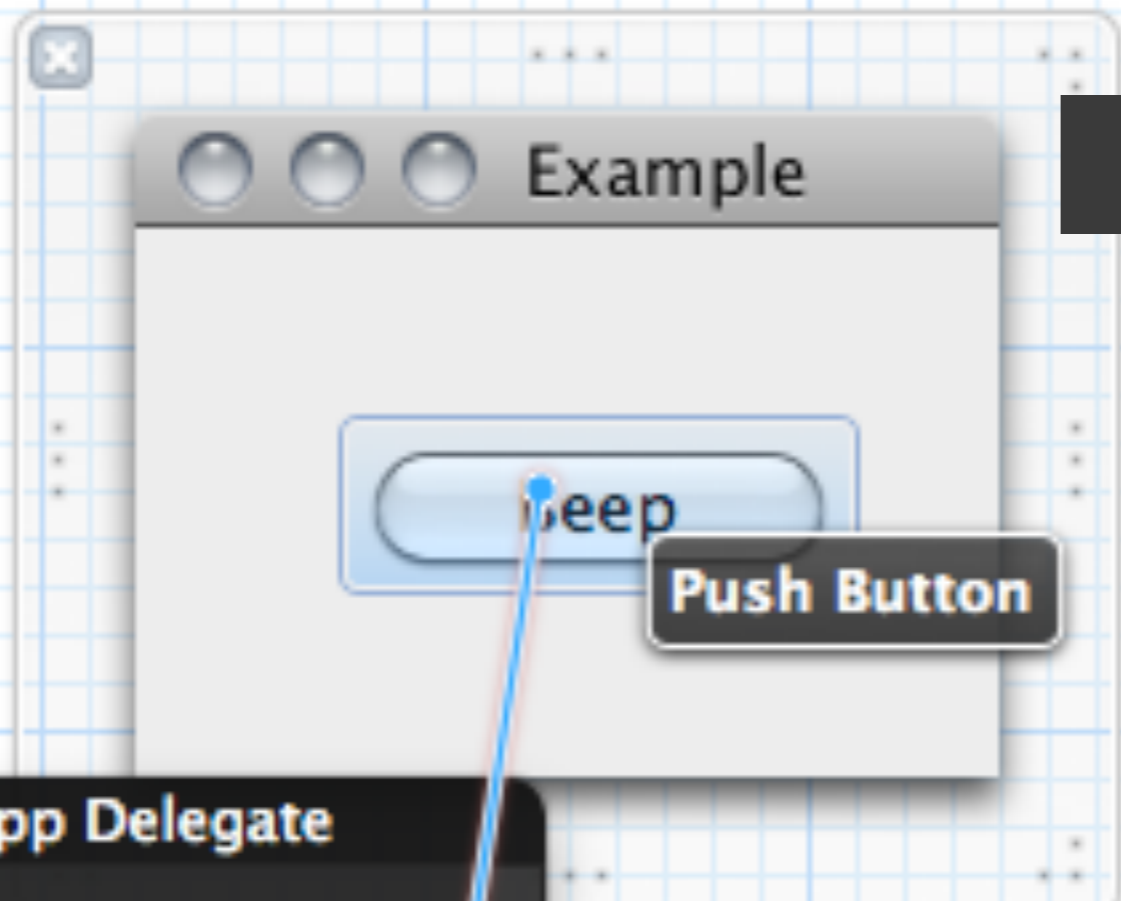
```
}
```



Example App Delegate

- ▼ **Outlets**
 - window — * Windo...
- ▼ **Referencing Outlets**
 - delegate — * File's ...
 - New Referencing Outlet
- ▼ **Received Actions**
 - buttonClicked:





Push Button

Example App Delegate

- ▼ **Outlets**
 - window — * Windo...
- ▼ **Referencing Outlets**
 - delegate — * File's ...
 - New Referencing Outlet
- ▼ **Received Actions**
 - buttonClicked:



buttonClicked:





POWELL'S TECHNICAL BOOKS

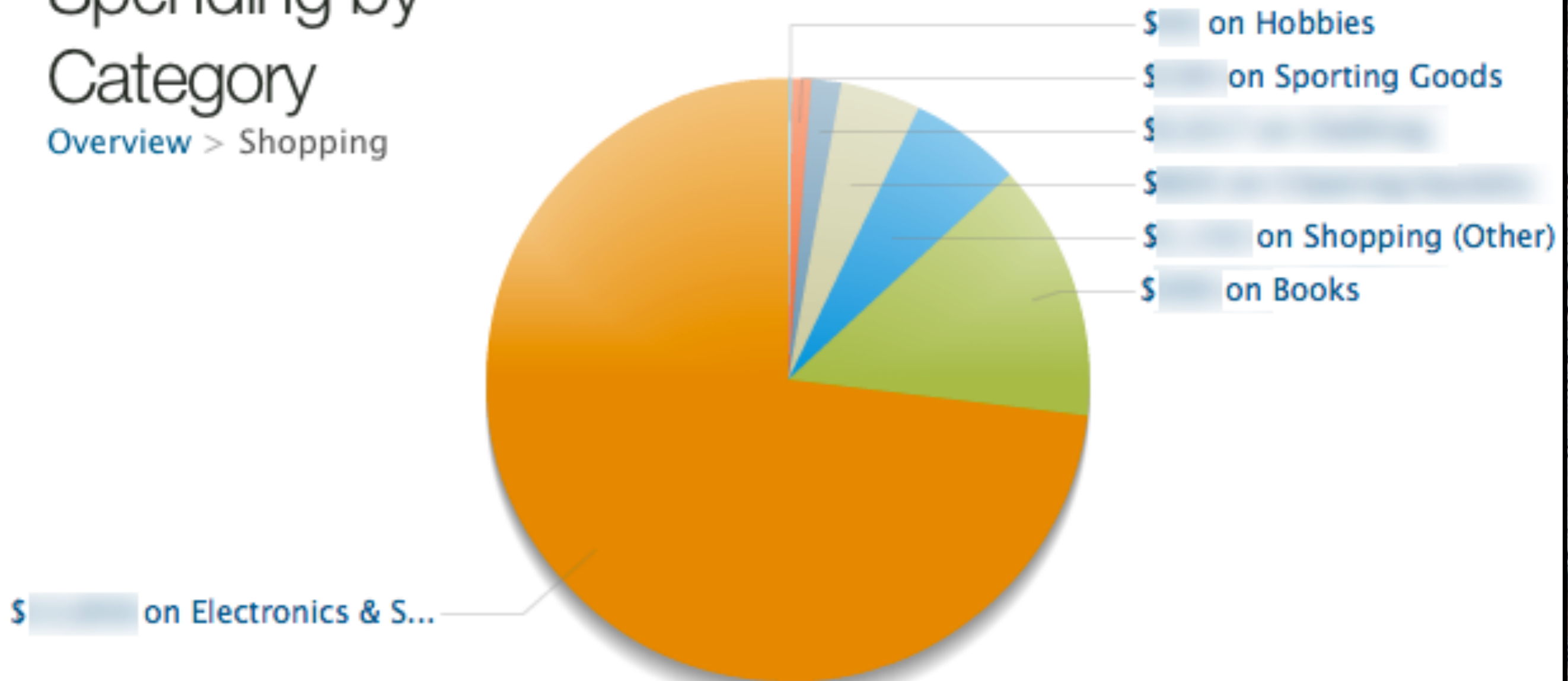
PEARL DISTRICT
NW COUCH ST

OPEN

ATM

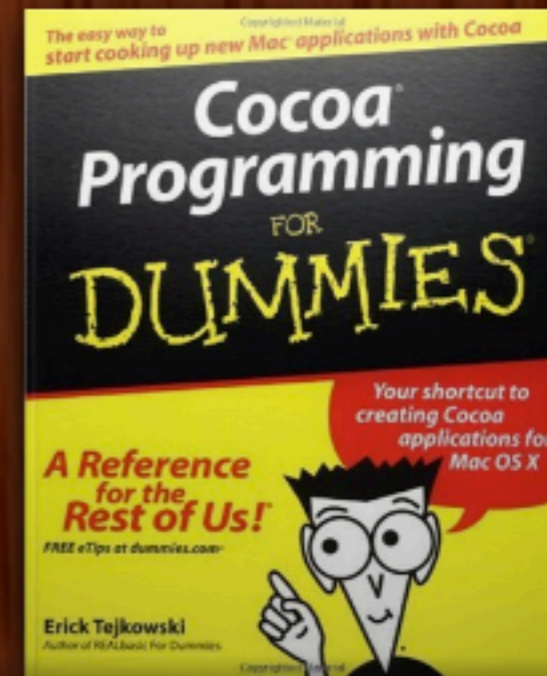
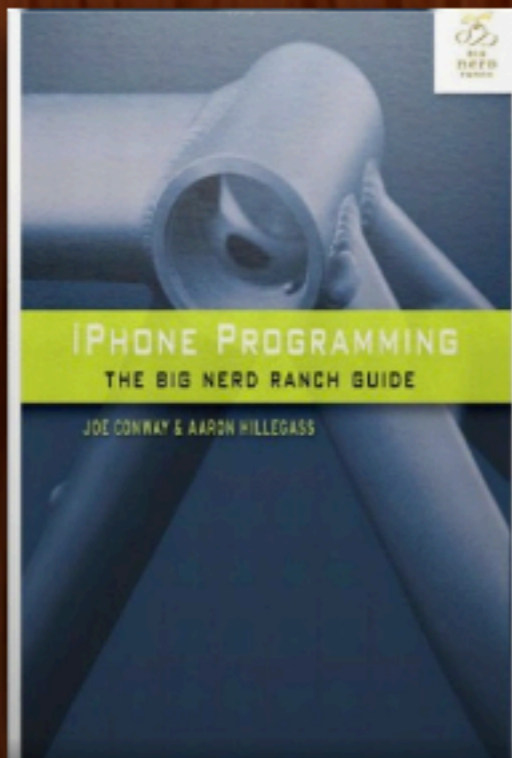
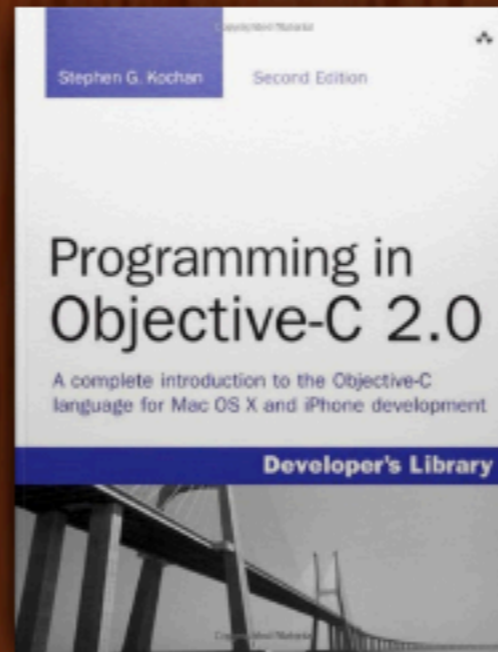
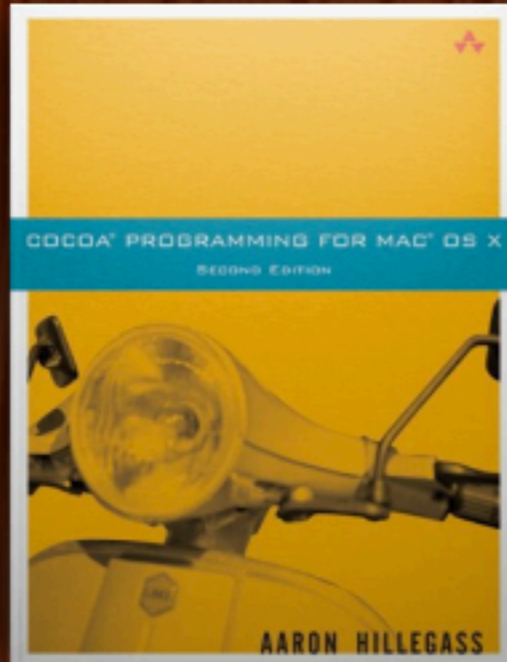
Spending by Category

Overview > Shopping





source: chestofbooks.com



COCOA® PROGRAMMING FOR MAC® OS X

SECOND EDITION



AARON HILLEGASS

Stephen G. Kochan

Second Edition

Programming in Objective-C 2.0

A complete introduction to the Objective-C language for Mac OS X and iPhone development

Developer's Library



Developer's Library
Recommended title

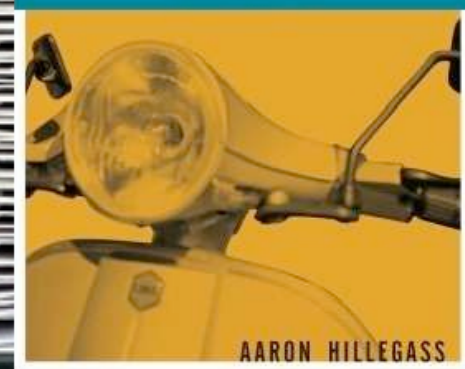
Learning Cocoa with Objective-C



O'REILLY

Jonny Davies. Distributed by Apple Computer, Inc.

COCOA® PROGRAMMING FOR MAC® OS X
SECOND EDITION



AARON HILLEGASS

Stephen G. Kochan

Second Edition

Programming in Objective-C 2.0

A complete introduction to the Objective-C language for Mac OS X and iPhone development

Developer's Library



Developer's Library
Recommended title

Learning Cocoa with Objective-C



O'REILLY
Jonny Davies. Distributed by Apple Computer, Inc.

Rentals

TV Shows

Podcasts

Music Videos

iTunes U



Coding Headstarts



Foundational Sessions - Ap...



Foundational Sessions - De...



Foundational Sessions - Inf...



Foundational Sessions - ...

STANFORD

WINTER 2010

iPhone Application Development

iPhone Application Develo...

STANFORD UNIVERSITY

iPhone Application Development WINTER 2010

iPhone Application Develo...

RWTH

iPhone Programming

iPhone Application Progra...

STANFORD

SPRING 2009

iPhone Application Programming

iPhone Application Progra...



iPhone Getting Started Vid...



iPhone Tech-Talk-World-Tour

STANFORD UNIVERSITY

Programming Methodology

Programming Methodology...

Podcasts

Hot Cocoa

2010



10/17/08

17:24

● Hot Cocoa 1: Introduction to Xcode

An introduction and tour around Xcode and Interface Builder.

10/24/08

12:49

● Hot Cocoa 2: What is Cocoa?

We take a look at the history of Apple, and give you an introduction to Cocoa and Objective-C. [more](#)

11/15/08

11:01

Hot Cocoa 5: Core Data - Part II

We continue our look at Core Data and learn how to access the database programmatically. [more](#)

11/23/08

23:24

① Hot Cocoa 6: Intro to iPhone Program...

We take our first look at iPhone programming and see how it differs from programming for OS X. [more](#)

12/5/08

29:19

① Hot Cocoa 7: Twitter App I - Getting the...

In this first episode in a series of episodes looking at building a Twitter client we take a look at getting and processin... [more](#)

12/20/08

20:49

● Hot Cocoa 9: Deploying Your iPhone App

In this episode we take a look at how you get your iPhone app running on an actual iPhone. [more](#)

12/26/08

19:20

● Hot Cocoa 10: Twitter App III - Updatin...

In this week's episode: We continue work on our Twitter App



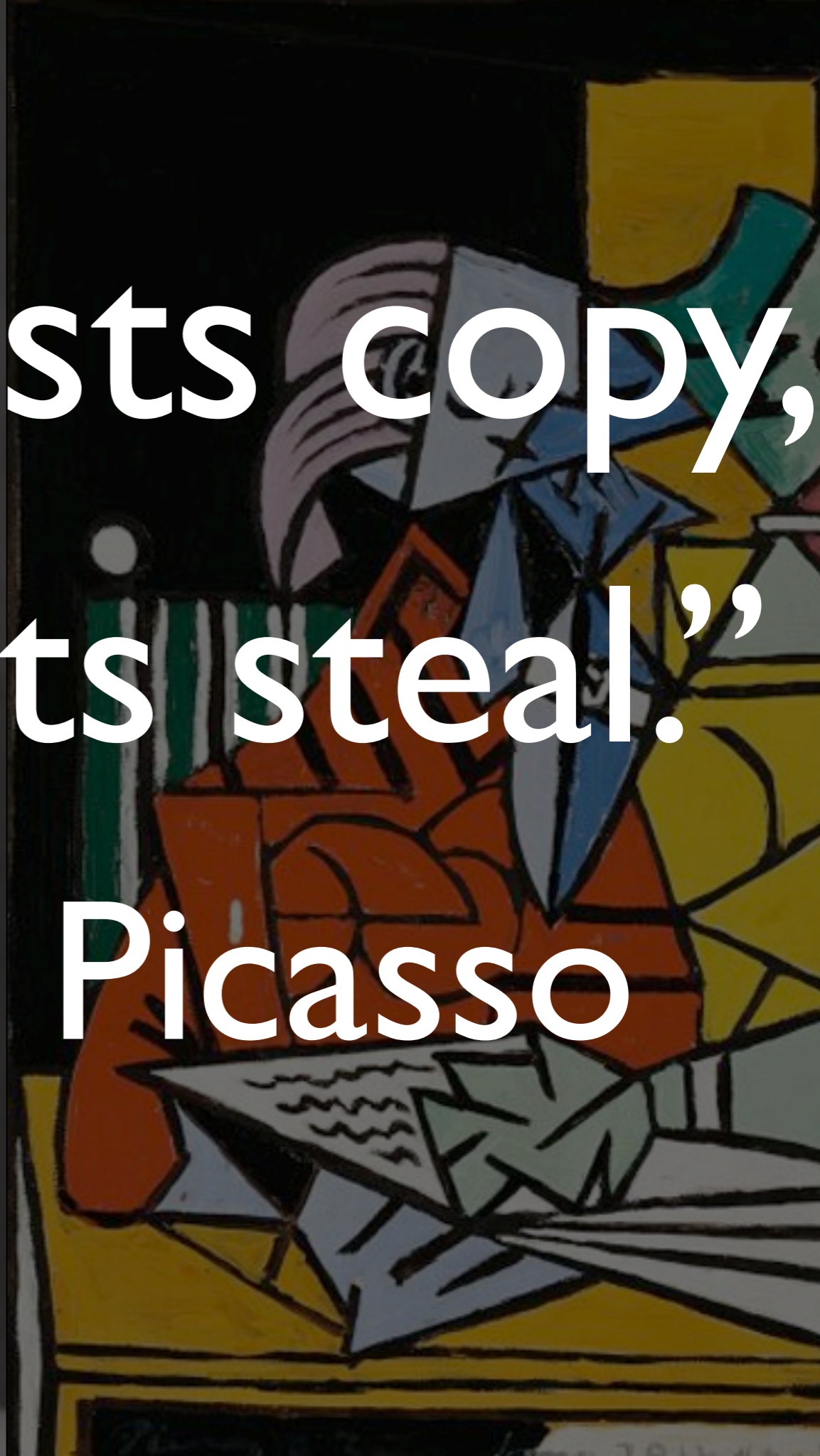
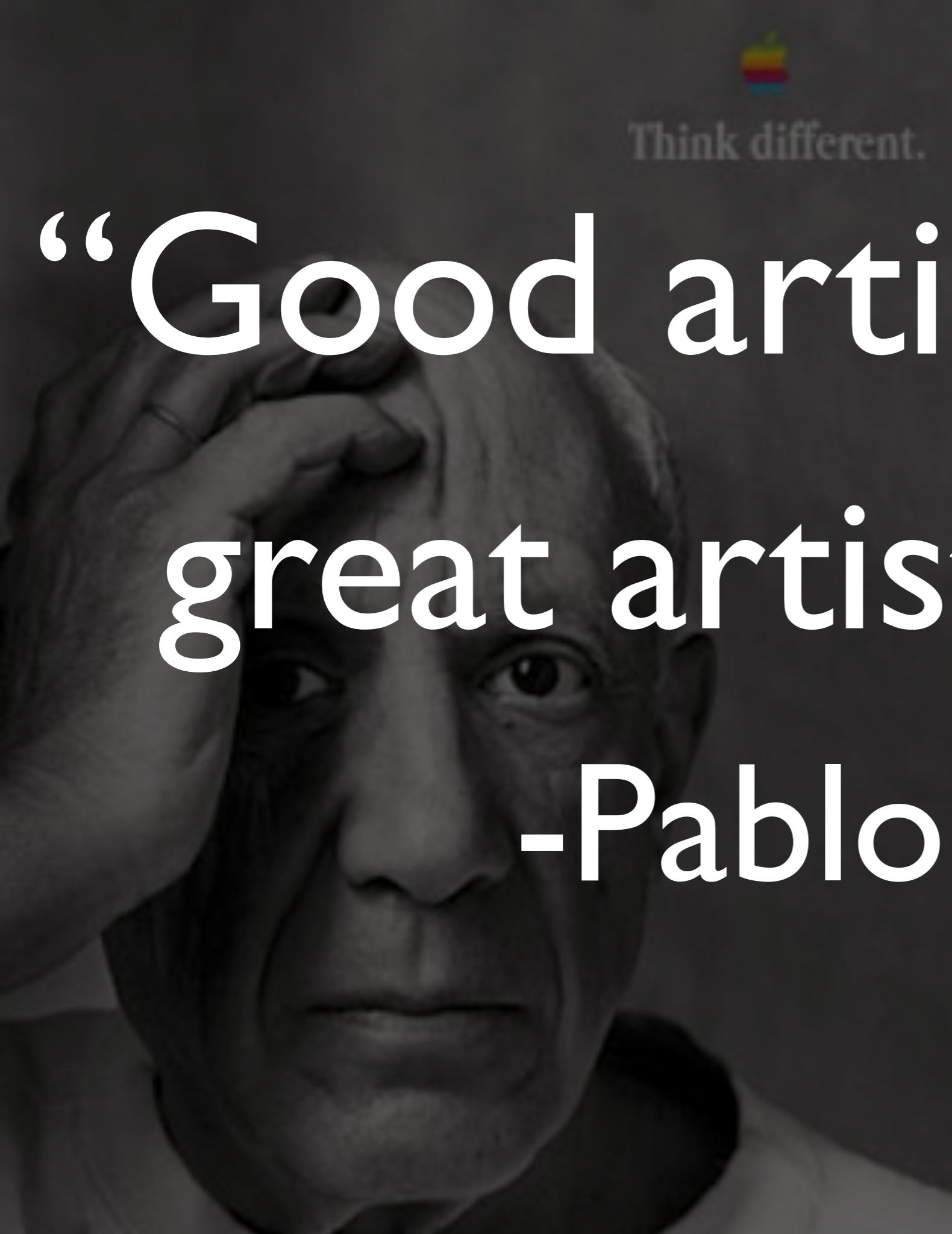
Podcasts



Think different.

“Good artists copy,
great artists steal.”

-Pablo Picasso





stackoverflow

stackoverflow



stackoverflow

search: "file check test cocoa"



stackoverflow

search: "file check test cocoa"



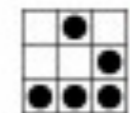
Cocoa has a class that provides support for dealing with filesystem objects: [NSFileManager](#). You can use a standard file manager (provided by `+defaultManager`) in conjunction with the `-fileExistsAtPath:` method, which returns a `BOOL` value for whether the file exists or not:

```
if([[NSFileManager defaultManager] fileExistsAtPath:@"/data"]) { /* ... */ }
```

Keep in mind this doesn't tell you anything about whether the file is accessible (permissions-wise), is a directory, etc. There are other `NSFileManager` functions for those.

[link](#) | [improve this answer](#)

answered Oct 30 '09 at 18:40



Tim

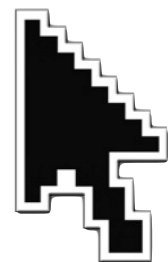
12.9k ● 4 ● 31 ● 58

Concepts are hard to Google for sometimes...

The Google logo is centered on the page, featuring its characteristic multi-colored letters: 'G' in blue, 'o' in red, 'o' in yellow, 'g' in blue, 'l' in green, and 'e' in red.A long, empty search input field with a thin blue border, positioned horizontally below the logo.

Google Search

I'm Feeling Lucky



Concepts are hard to Google for sometimes...

Google

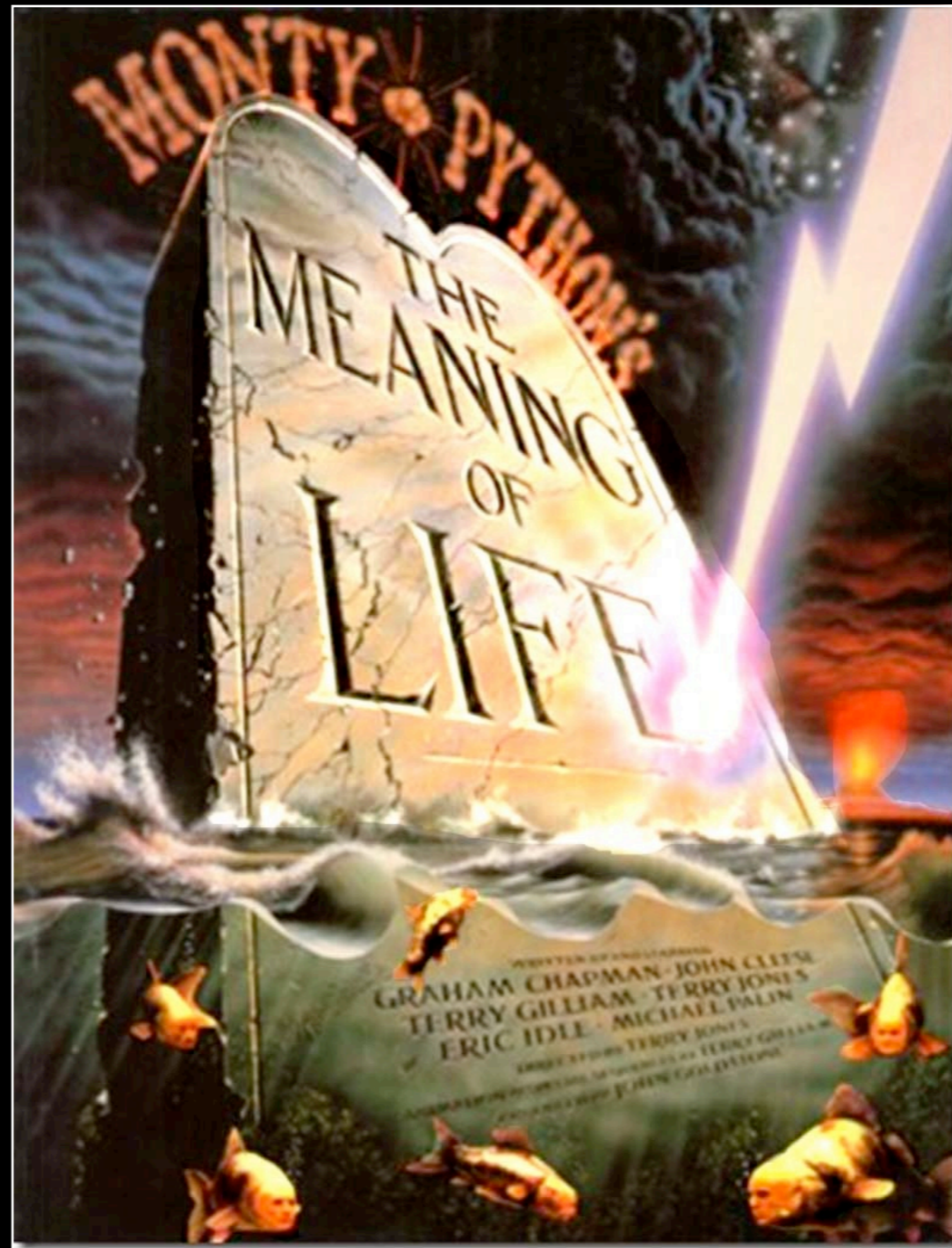
The meaning of life

Google Search

I'm Feeling Lucky



Concepts are hard to
Google for sometimes...



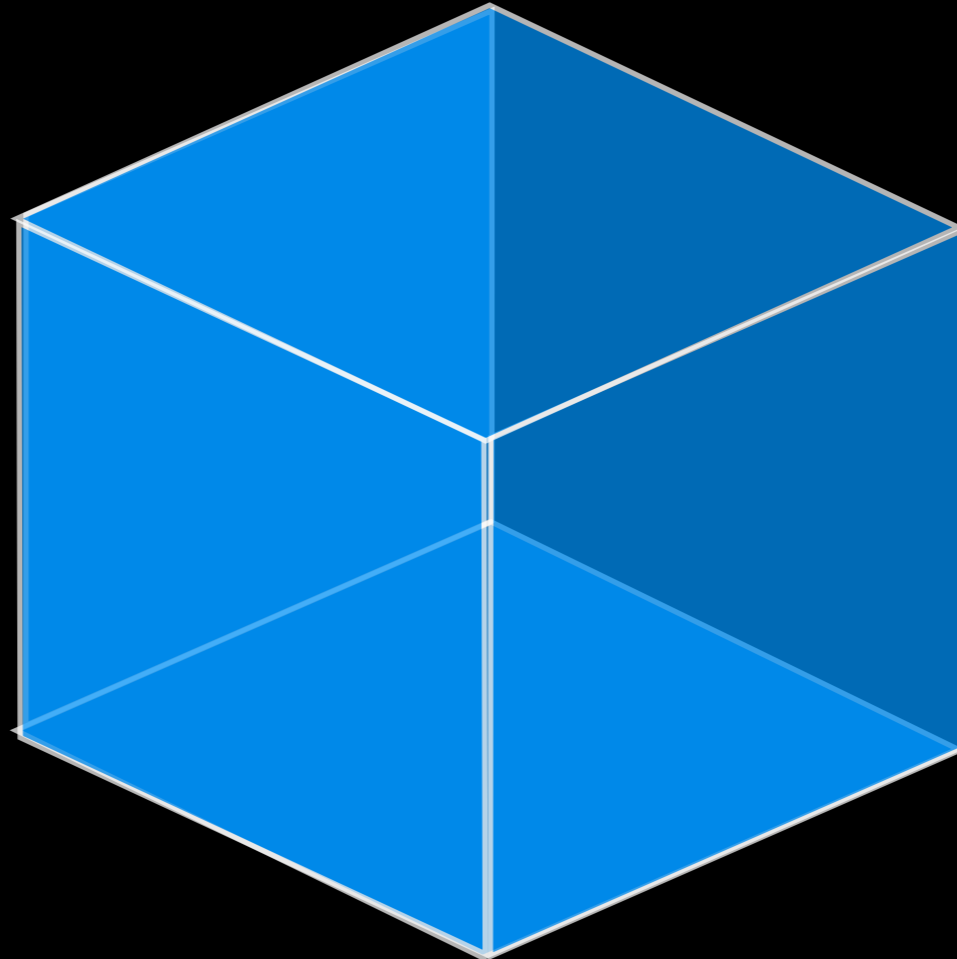
Variable Scoping

bash :
export

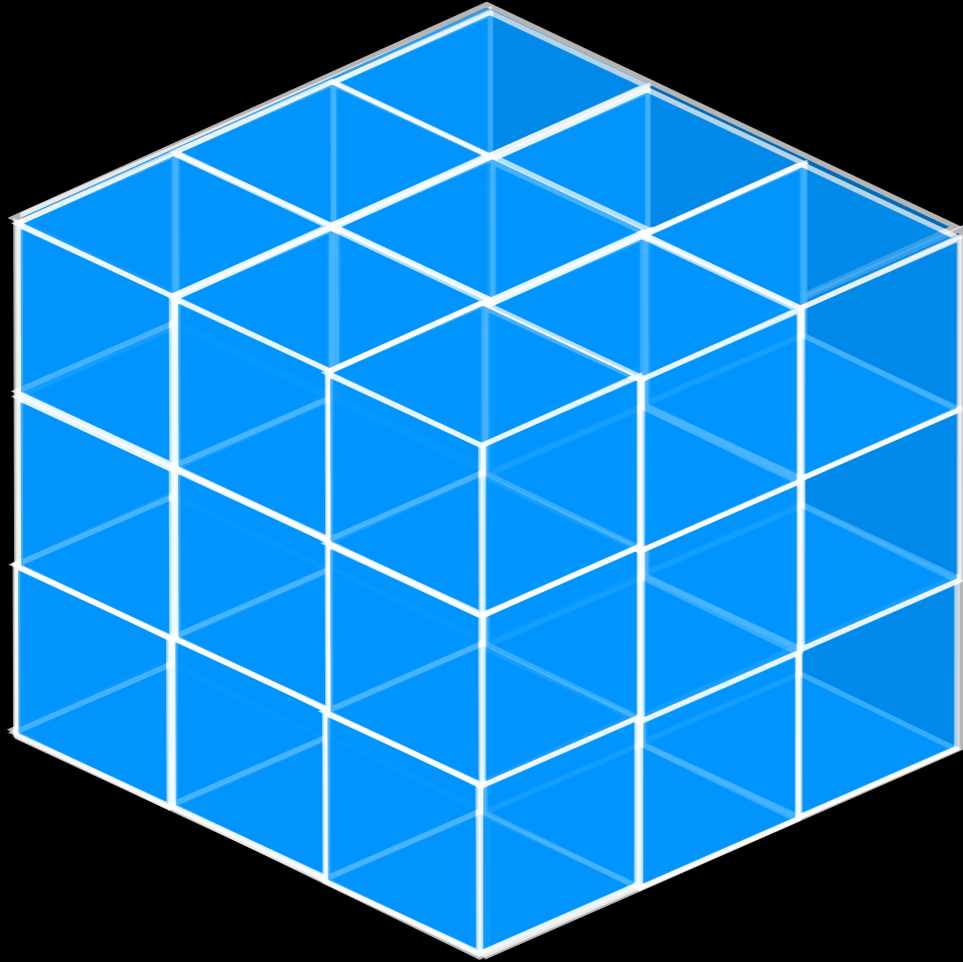
python :
global

perl : my

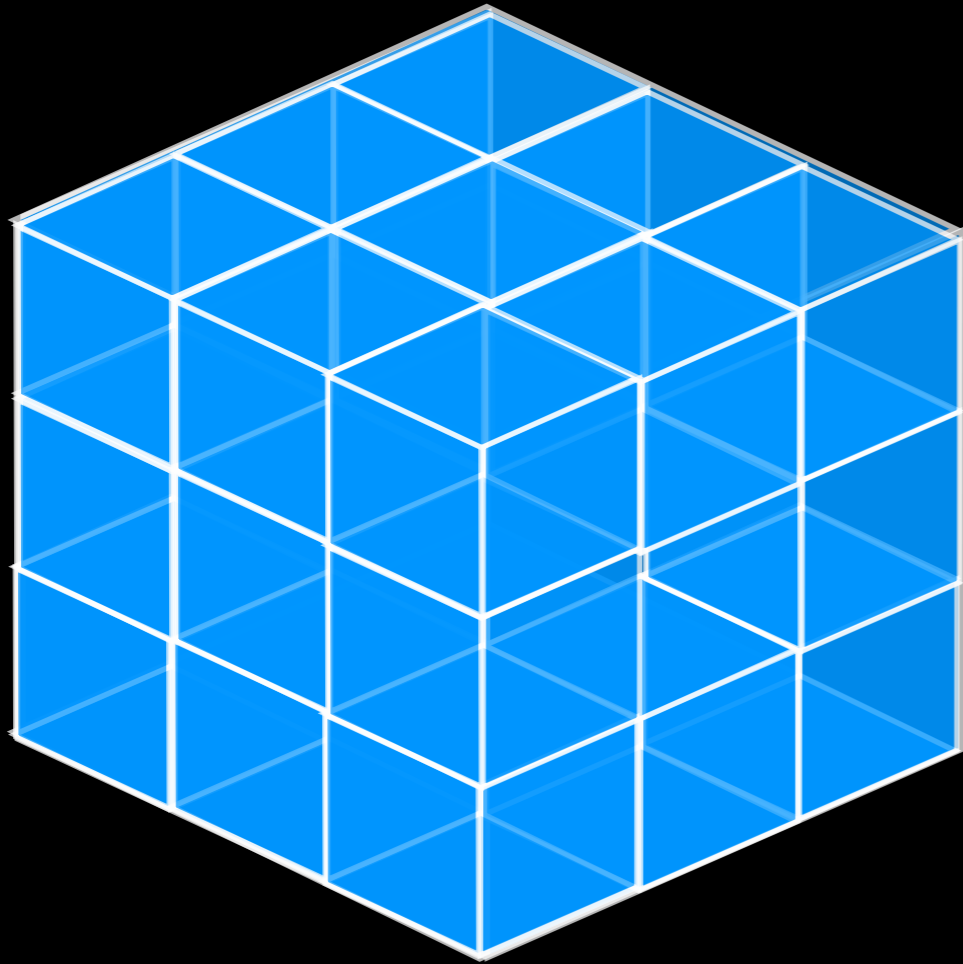
AppleScript :
property



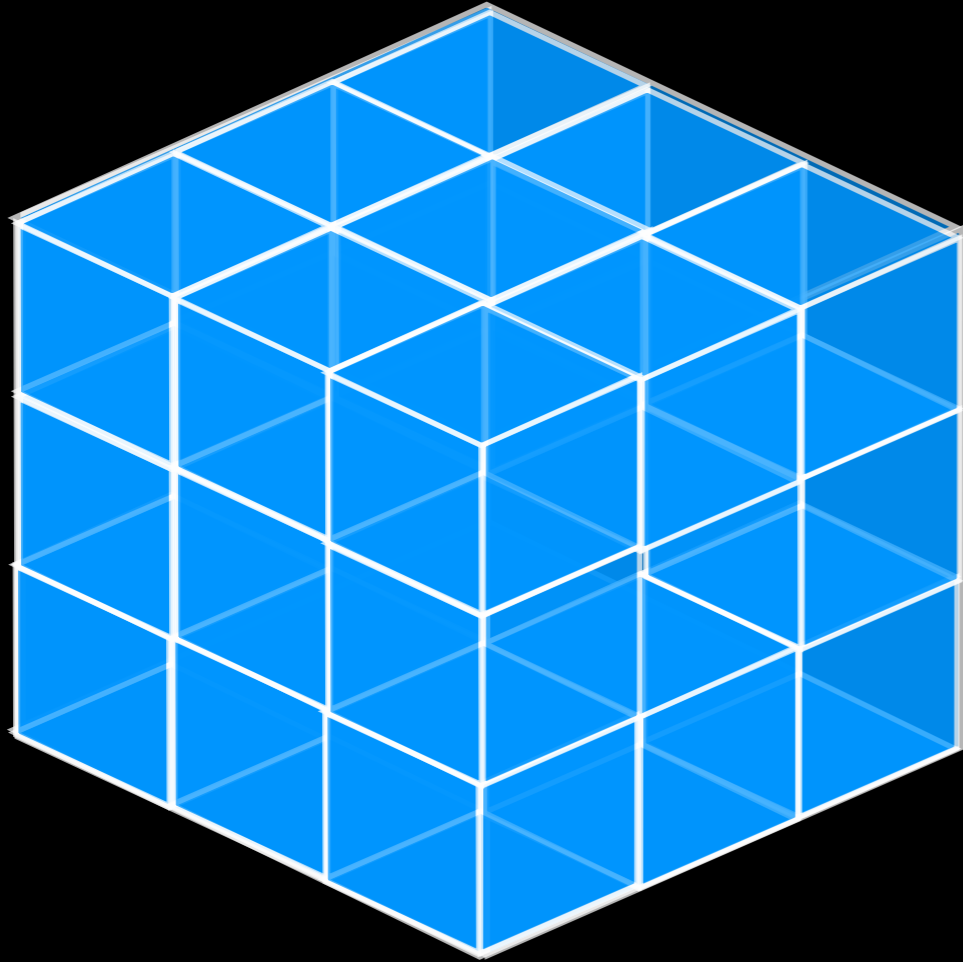
Variable Scoping



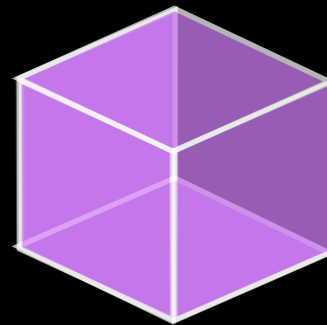
Variable Scoping



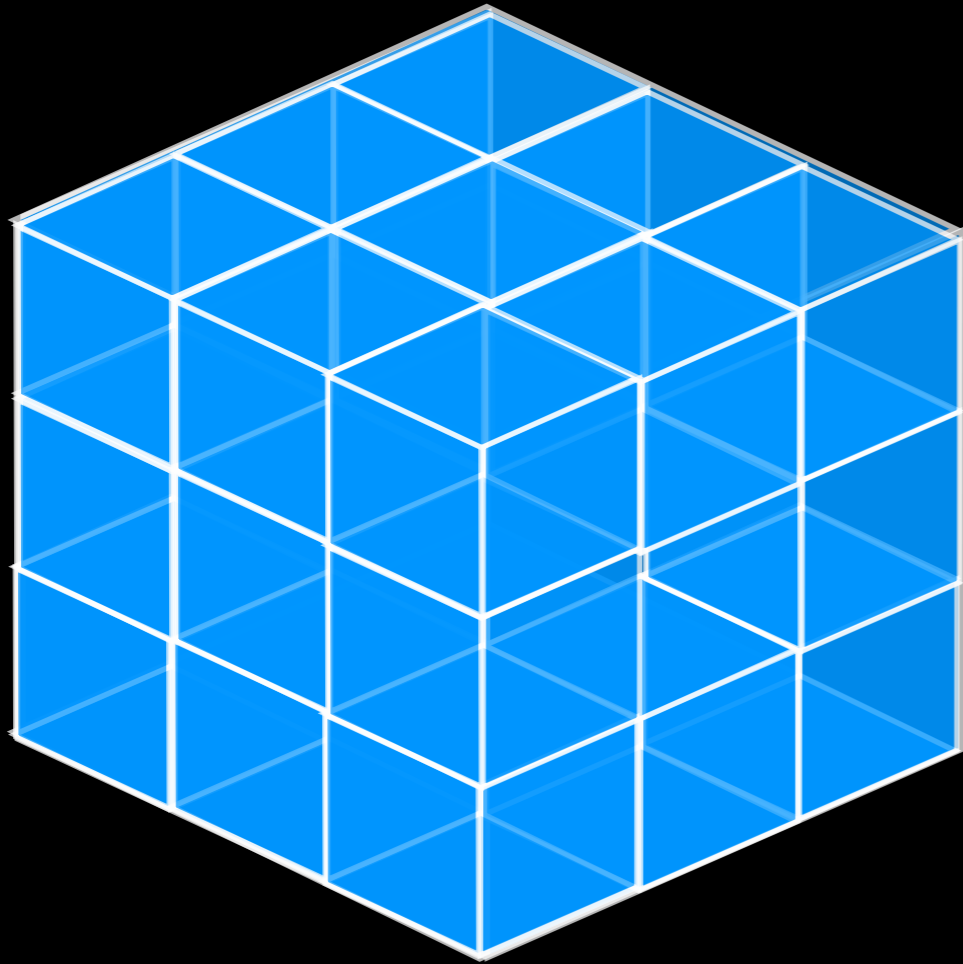
Variable Scoping



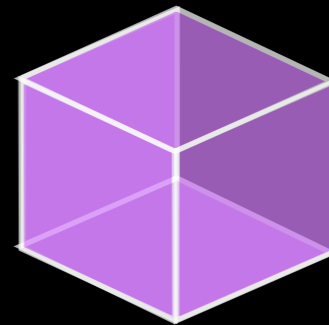
instanceVar



Variable Scoping



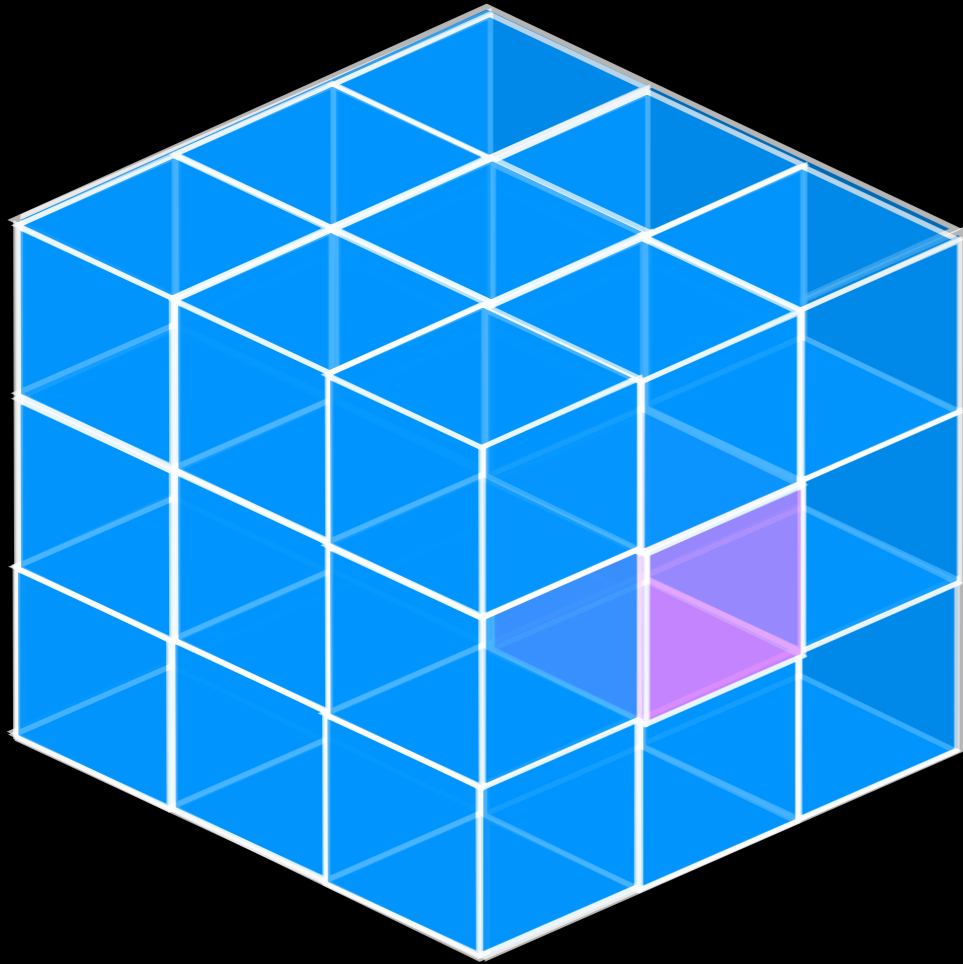
instanceVar



```
@interface AppDelegate : NSObject
{
    @private
    NSString *instanceVar;
}
```



Variable Scoping



instanceVar

```
@interface AppDelegate : NSObject
{
    @private
    NSString *instanceVar;
}
```

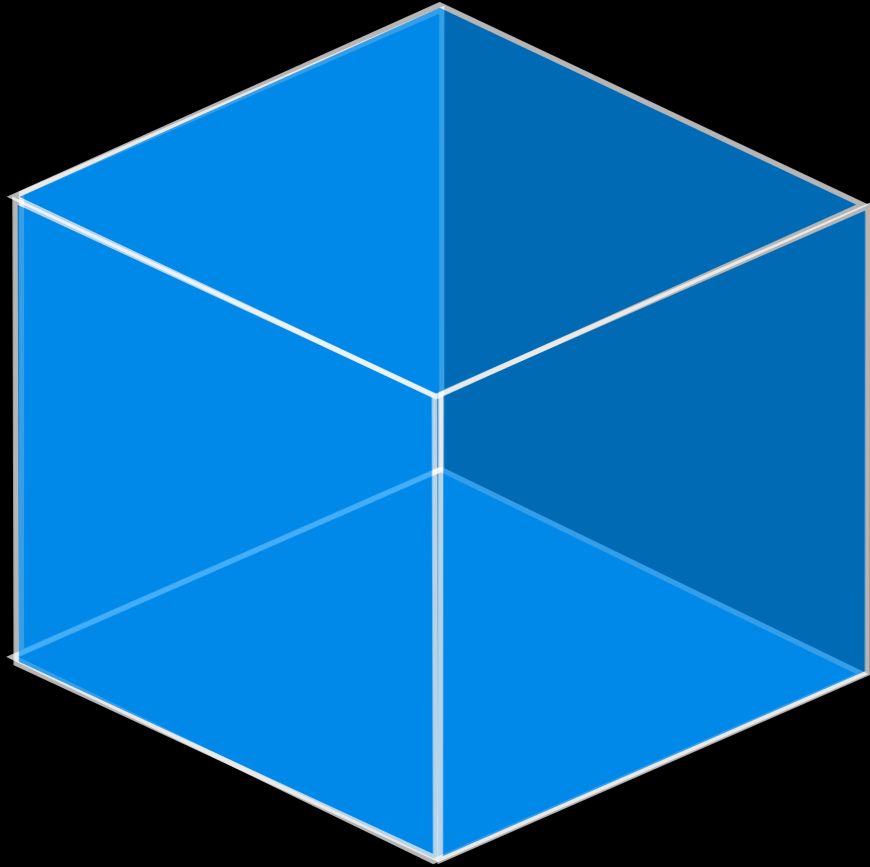


NSNotification

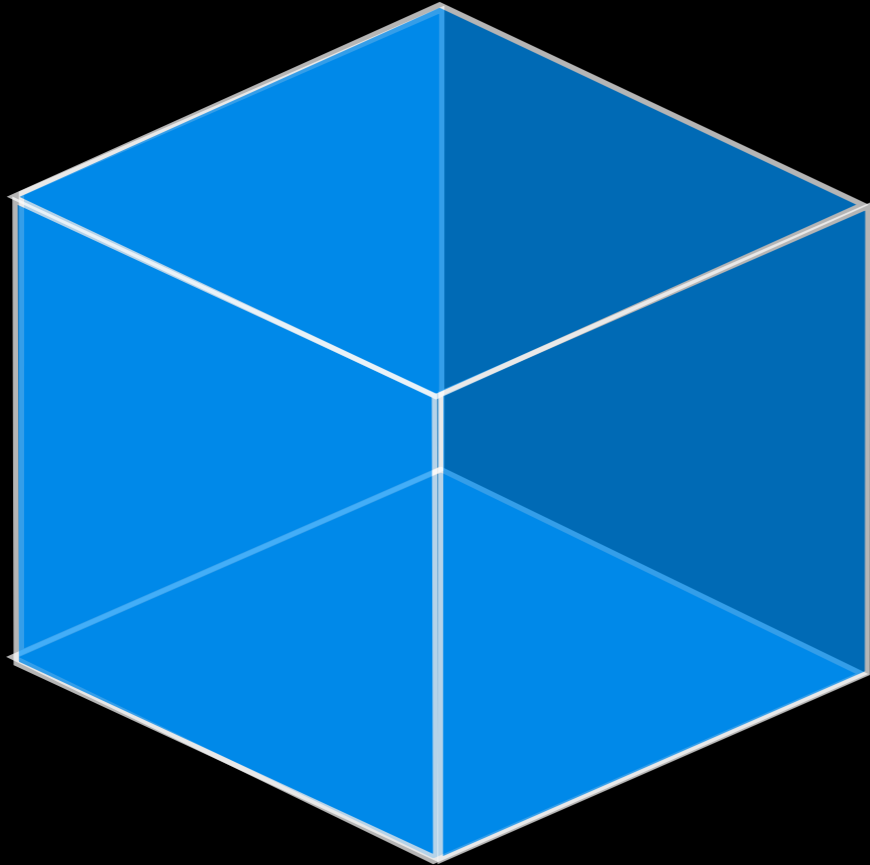
NSNotification



NSNotification



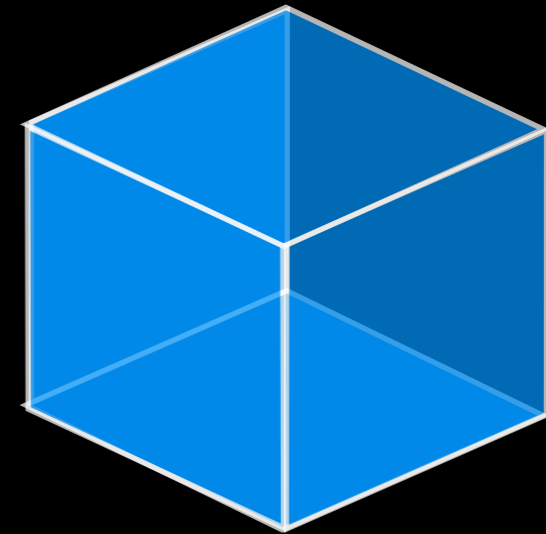
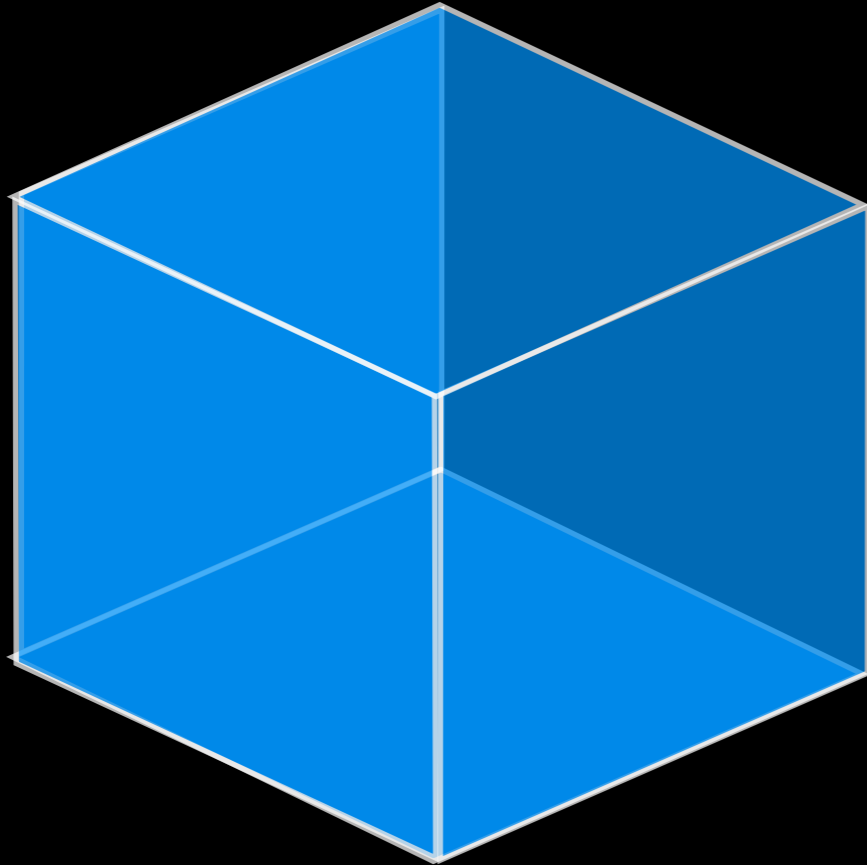
NSNotification



Observer



NSNotification

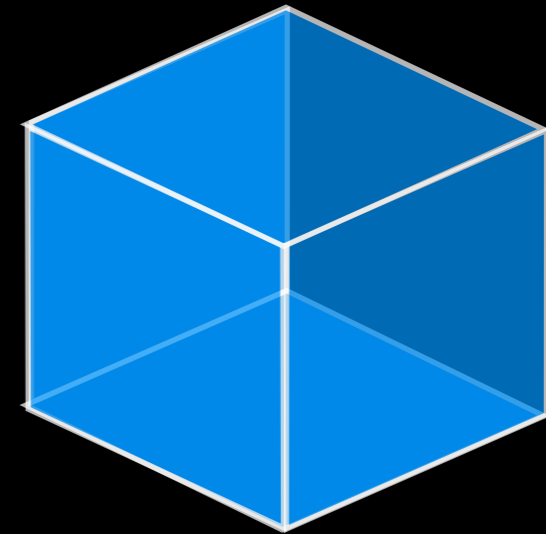
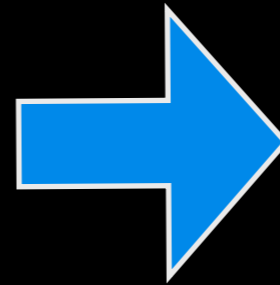
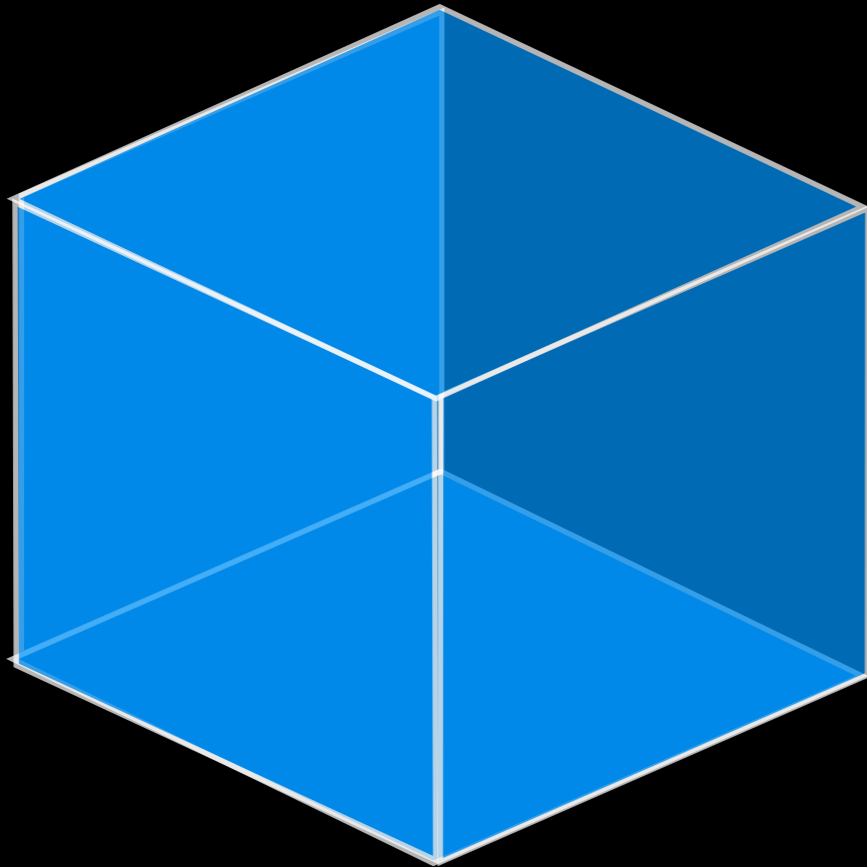


Observer

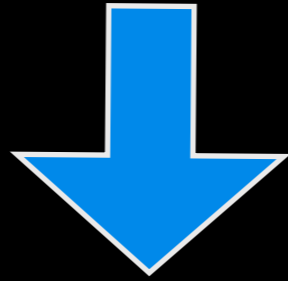
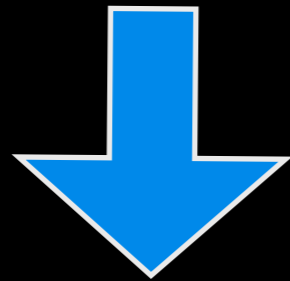
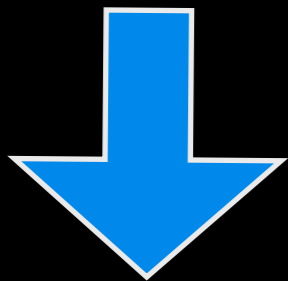
Post Notification



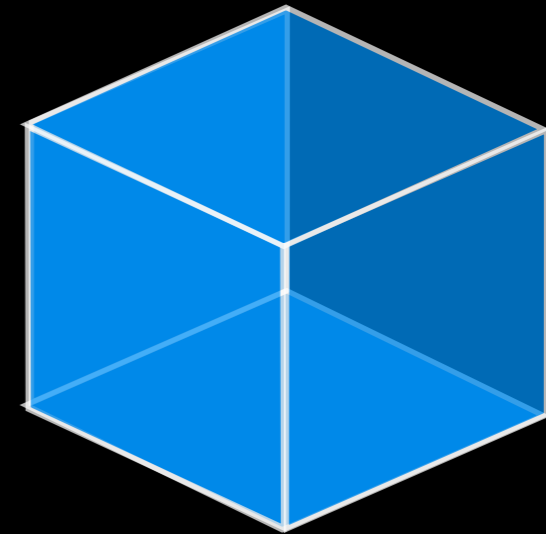
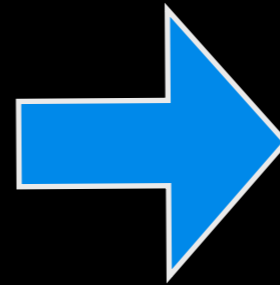
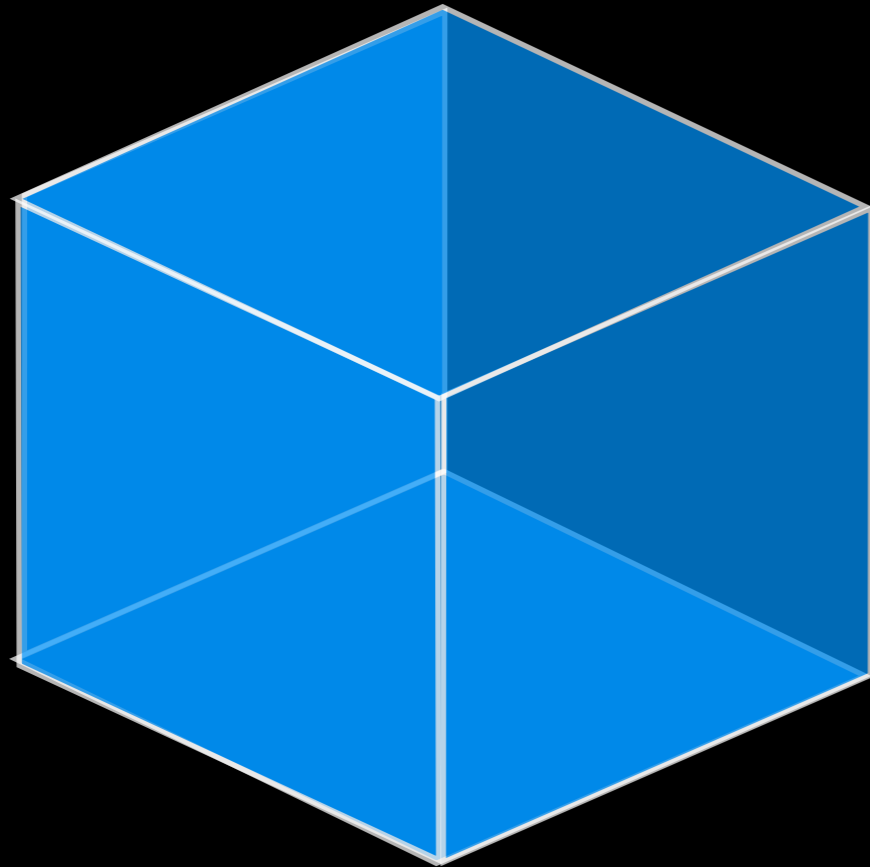
NSNotification



Observer



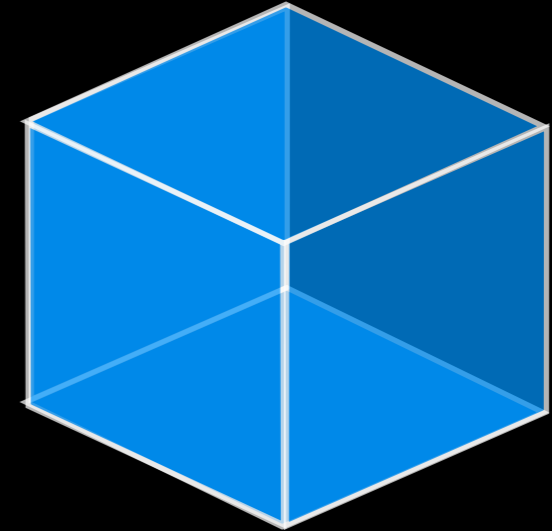
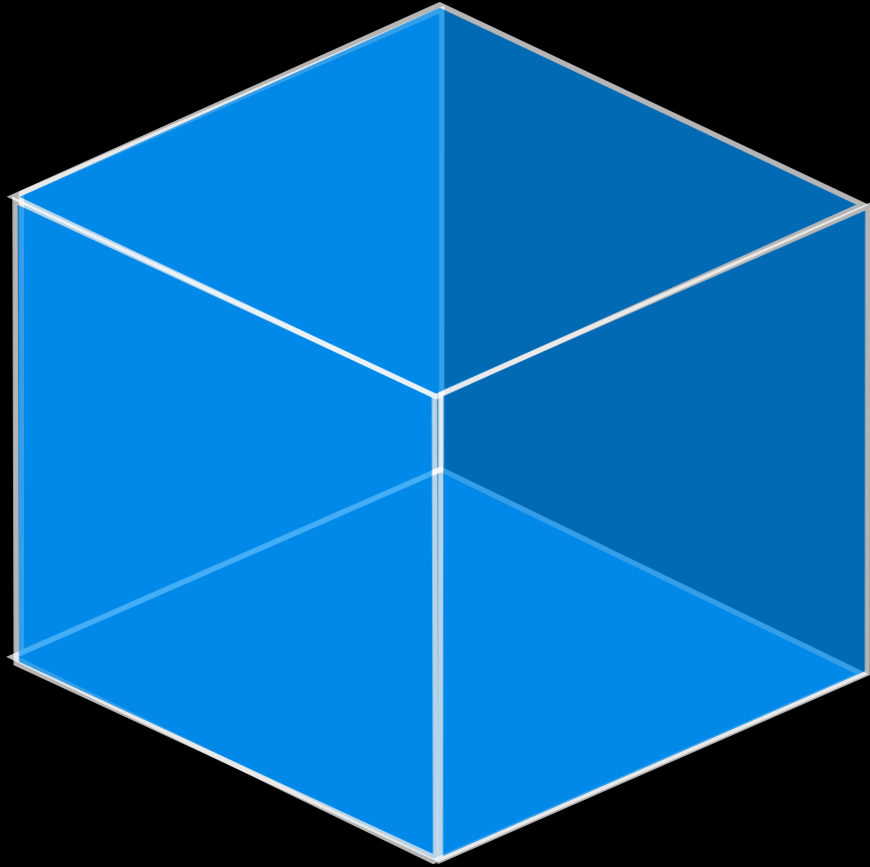
NSNotification



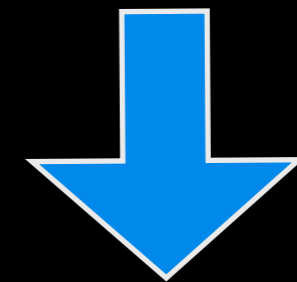
Observer



NSNotification



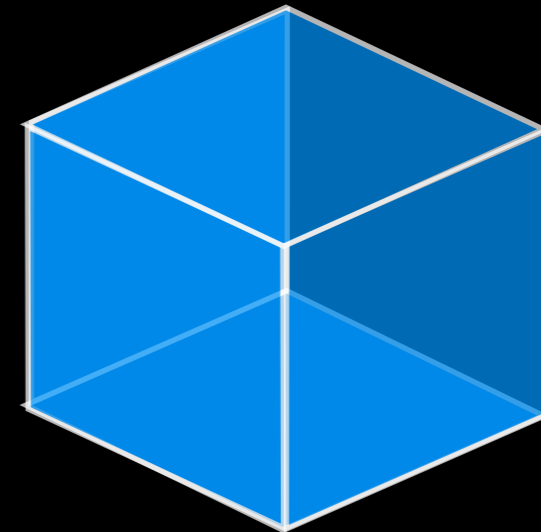
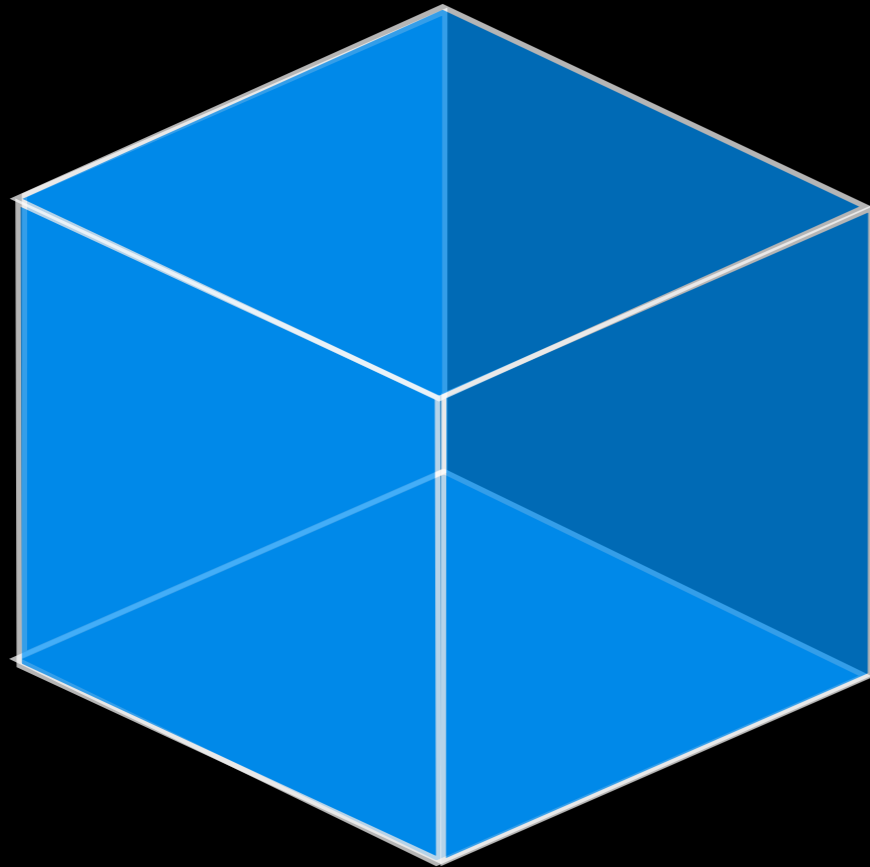
Observer



Method



NSNotification



Observer




Method




What is possible


baz



1.888.857.3318
support@318.com
<http://support.318.com>
My Tool Version is: 1.6
My OS Version is: 10.7.1

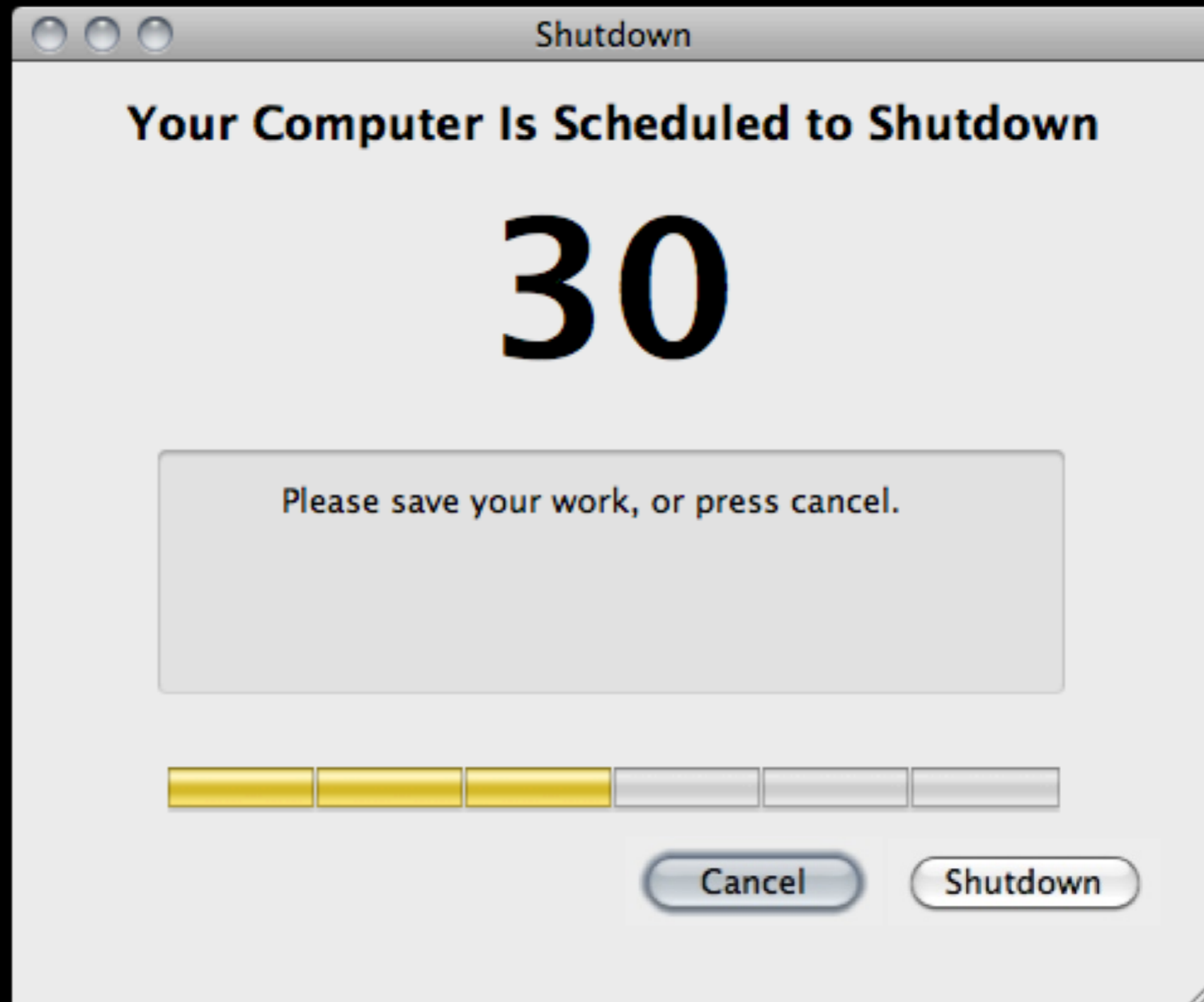


My Katana Device ID is:
767777
My LAN IP Address is:
172.19.131.147
[Remote Support](#)

Disk Space:  6 GB

The 318 support tool has found: 0 Support Issues
Click "Send Support Request" to notify 318 of these issues.

Simple Hybrid Script Example



A script running an App



```
@implementation AppDelegate
@synthesize window;
-(IBAction)userClickedShutdown:(id)sender
{
    [NSApp terminate:self];
}
-(IBAction)userClickedCancel:(id)sender
{
    // Bypasses the normal nsapp tear down process
    exit(1);
}
```

Shutdown

Cancel

A script running an App



```
@implementation AppDelegate
@synthesize window;
-(IBAction)userClickedShutdown:(id)sender
{
    [NSApp terminate:self];
}
-(IBAction)userClickedCancel:(id)sender
{
    // Bypasses the normal nsapp tear down process
    exit(1);
}
```

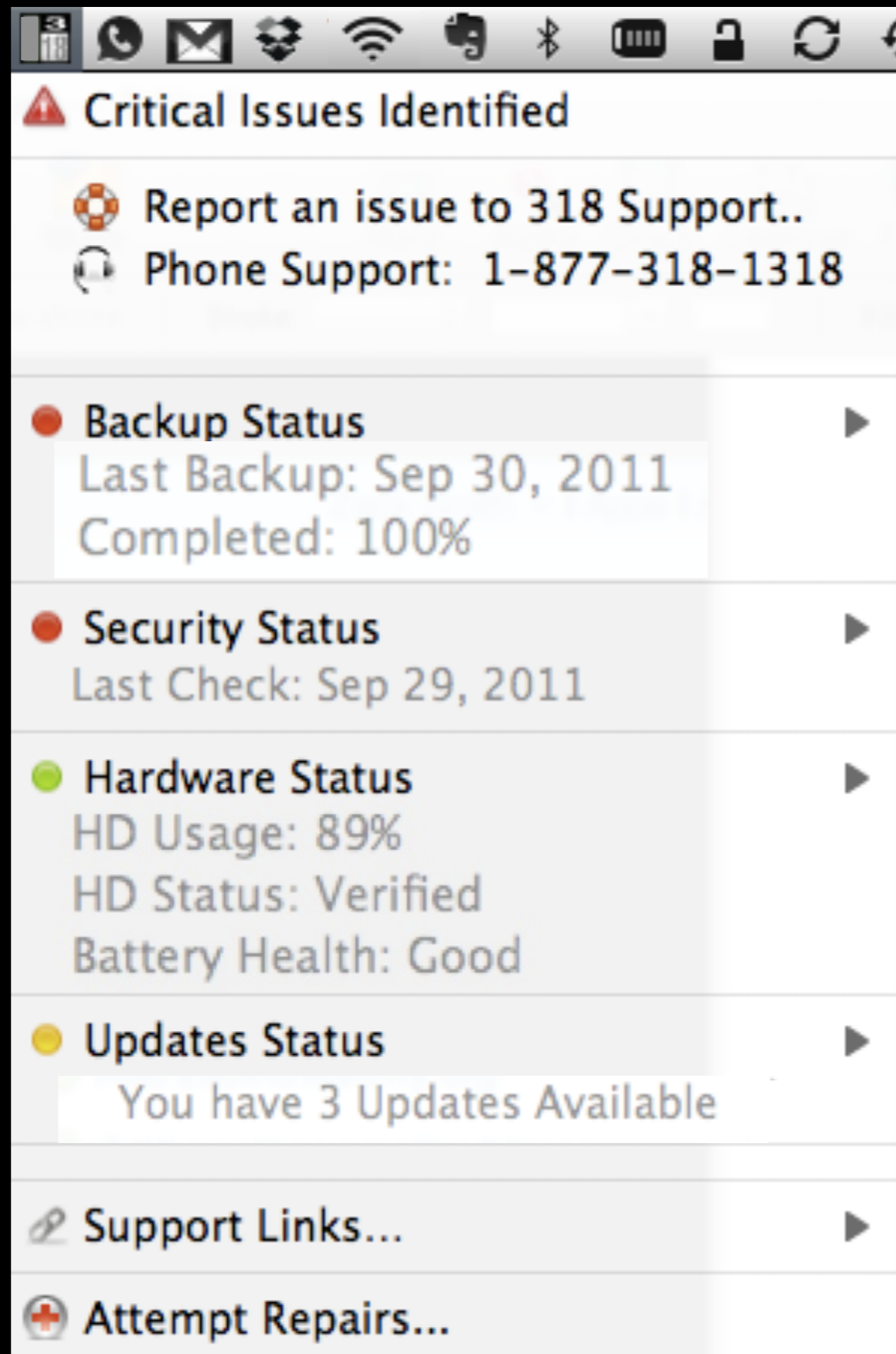
Shutdown

Cancel

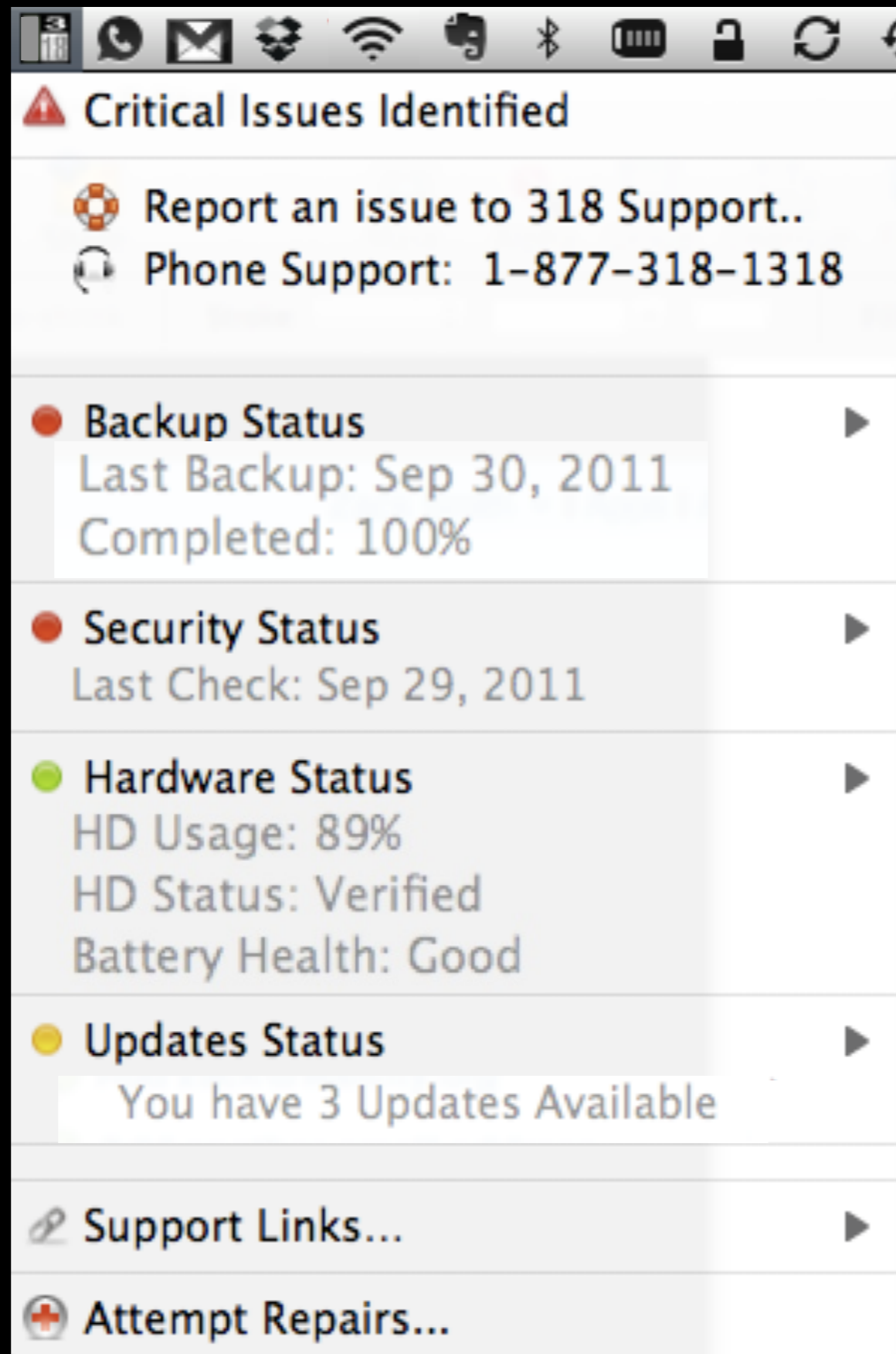


```
#!/bin/bash
export LaunchApp="/Shutdown.app/Contents/MacOS/Shutdown"
if "$LaunchApp" ; then
    echo "User clicked shutdown"
else
    echo "User clicked cancel"
fi
```

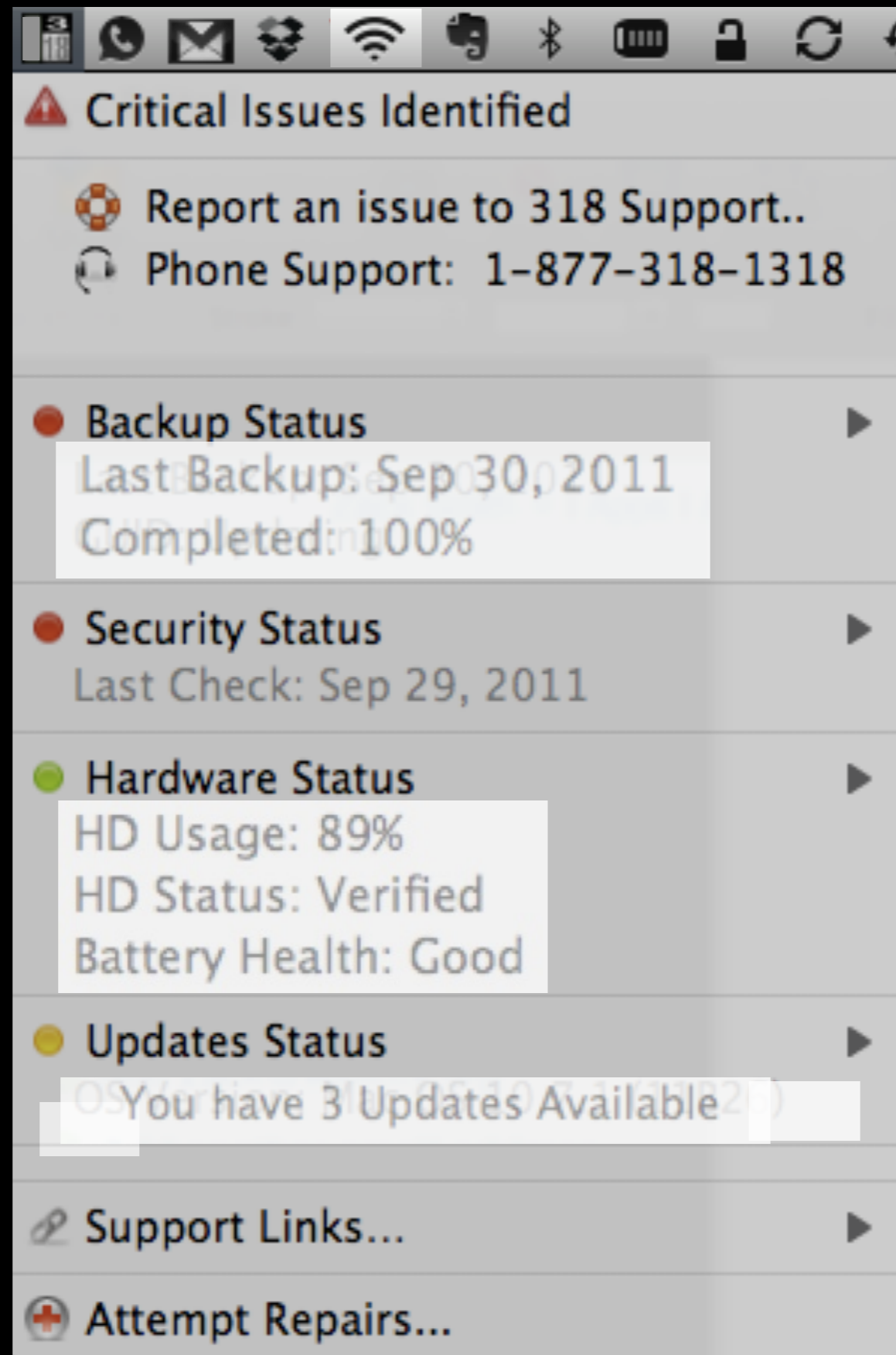
An App running an Script



An App running an Script



An App running an Script



HD Usage: 89%

HD Status: Verified

Battery Health: Good

HD Usage: 89%

HD Status: Verified

Battery Health: Good

```
_task = [[NSTask alloc] init];

NSData *data;
NSPipe *pipe = [NSPipe pipe];
[_task setLaunchPath:@"/usr/sbin/system_profiler"];
[_task setArguments:[NSArray arrayWithObjects:@"-xml",
        @"SPSerialATADataType",
        @"SPPowerDataType",
        nil]];


[_task setStandardOutput: pipe];
[_task setStandardInput:[NSPipe pipe]];
[_task setStandardError: pipe];
[_task launch];
```




Mac Setup Utility

Checking Network Connectivity

This script is checking for network connectivity



Manual Setup

Begin Setup

Choose "Quit" to start stop automated setup and configure manually



Your Mac OS Upgrade is about to begin

Network Install: Your machine will now reboot and display the icons in the picture above.
Once started your system will automatically begin upgrading your Mac.
This process may take from 45 minutes to 1.5 hours and cannot be interrupted.
Please be sure you have ample time before continuing.

When this process is complete your Mac will reboot normally.

Current Server Load for (127.0.0.1)



Cancel

Restart

PleaseWait

Mac AD Utility

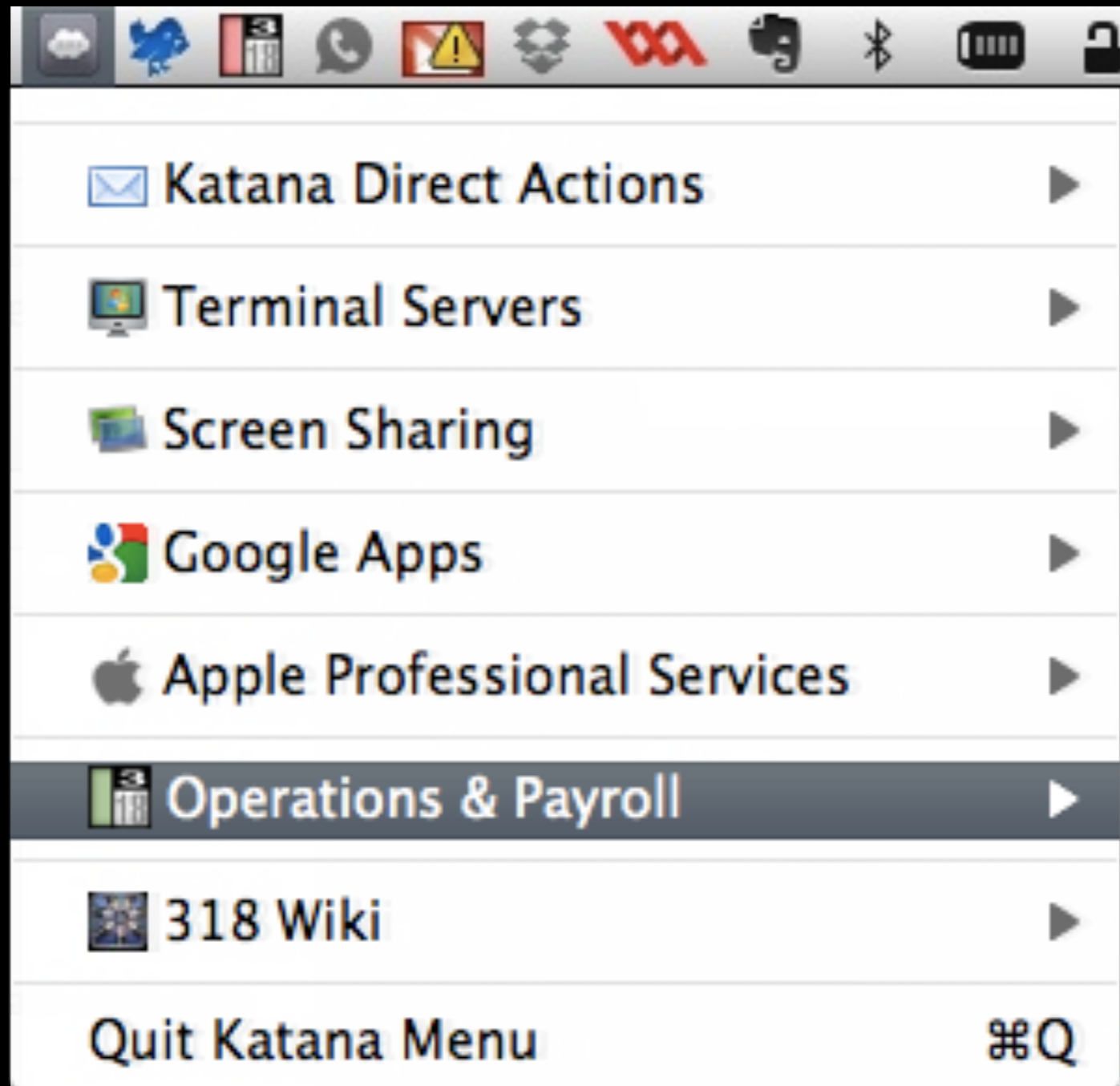


Please Wait...

Please Wait...



Simple Launcher



Simple Launcher

