RubyMotion

Ruby iOS and OS X Gloriousness

What?

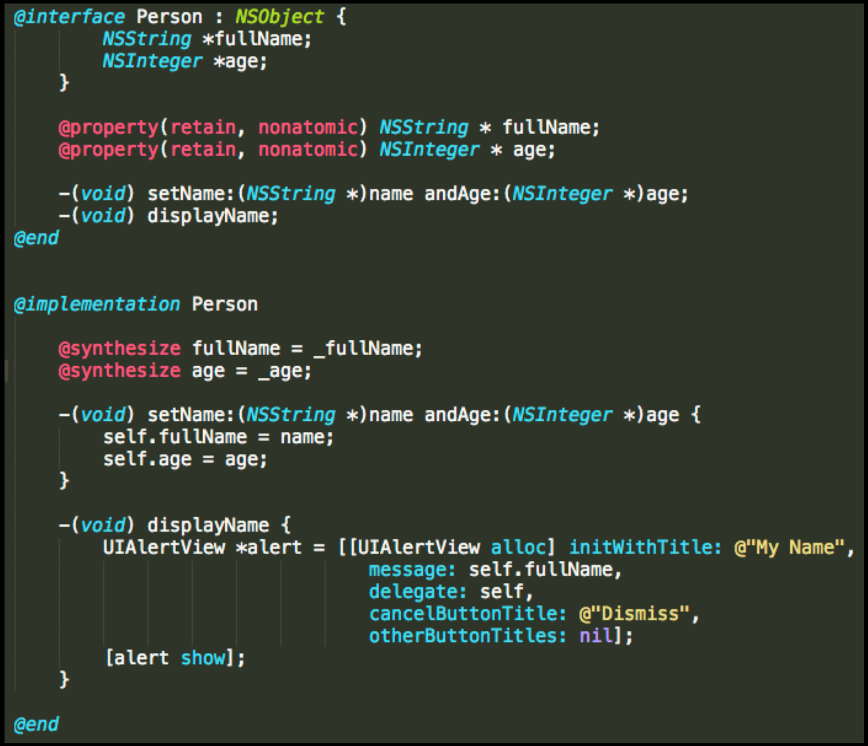
- Use Ruby (*like) language to develop apps for iOS and OS X.
- Compiled to optimized machine code. (Fast)
- Automatic memory management. (As one would expect)
- Fully integrable with:
  - Third party Objective-C libraries
  - RubyMotion gems
  - Some pure Ruby gems. (Usually ones without lots of DSL “magic”)
    - *define_method* now exists and can be used for some dynamic magic
    - *eval* cannot be used as the app source code is compiled.

Why?

- Because Ruby Syntax > Objective-C Syntax.
- Enjoy simple Ruby OOP and dynamic typing to create lightweight complex systems.
- Ruby Wrappers. DSLs, Active-Record like data management. Excellent encapsulation.
- Interactivity, Testing, Debugging.
- No need to use XCode.
- Finally a nice GUI exists for Ruby . (no more ‘Shoes’)

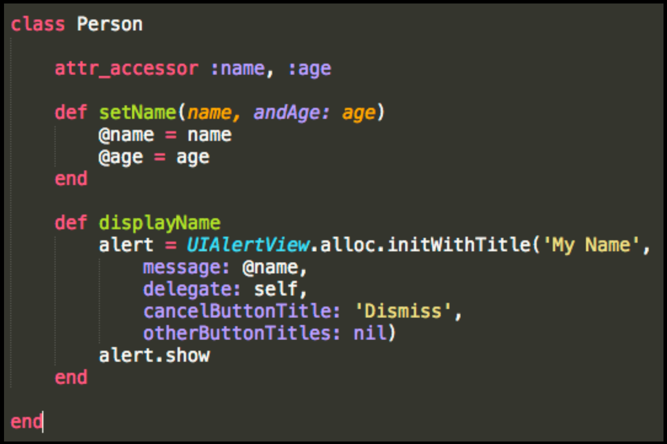
Syntax Comparison

Objective-C



```
@interface Person : NSObject {
        NSString *fullName;
        NSInteger *age;
    }

    @property(retain, nonatomic) NSString * fullName;
    @property(retain, nonatomic) NSInteger * age;

    -(void) setName:(NSString *)name andAge:(NSInteger *)age;
    -(void) displayName;
@end


@implementation Person

    @synthesize fullName = _fullName;
    @synthesize age = _age;

    -(void) setName:(NSString *)name andAge:(NSInteger *)age {
        self.fullName = name;
        self.age = age;
    }

    -(void) displayName {
        UIAlertView *alert = [[UIAlertView alloc] initWithTitle: @"My Name",
                                message: self.fullName,
                                delegate: self,
                                cancelButtonTitle: @"Dismiss",
                                otherButtonTitles: nil];
        [alert show];
    }

@end
```

RubyMotion



```
class Person

    attr_accessor :name, :age

    def setName(name, andAge: age)
        @name = name
        @age = age
    end

    def displayName
        alert = UIAlertView.alloc.initWithTitle('My Name',
            message: @name,
            delegate: self,
            cancelButtonTitle: 'Dismiss',
            otherButtonTitles: nil)
        alert.show
    end

end
```

Show Me Something

The following is a simple RubyMotion app to show what can be done with a small amount of code.

What Else?... Resources.

- RubyMotion
  http://www.rubymotion.com

- RubyMotion Wrappers (Gems)
  http://rubymotion-wrappers.com

- JoyBox (Cocos2D + Box2d physics engine gem)
  http://joybox.io

- TeaCup (Interface builder DSL + stylesheets gem)
  https://github.com/rubymotion/teacup

- MotionModel (DataMapper for CoreData gem)
  https://github.com/sxross/MotionModel

- RubyMotion Samples
  https://github.com/HipByte/RubyMotionSamples