

CoffeeScript

a better way to write javascript

高見龍





You can get this slide on

www.eddie.com.tw/slides

Who am I ?

I'm a Flash guy.



高見龍

a.k.a Eddie or Aquarianboy

- Live and work in Taipei, Taiwan.
- Serving in my own little tiny company.
- Flash / AS3 / Ruby / Rails / Python programming for living.
- A little bit Objective-C for personal interests.
- Technical Education and Consultant.
- PTT Flash BM (since 2007/4).
- Adobe Certificated Flash Developer (Since 2006/7).
- Linux Professional Institute Certification (Since 2005/3).



or just google me with keyword "高見龍"



JavaScript
becomes
hugely
popular.

9th on TIOBE, Nov 2011



JavaScript
seems
easy..



But it's
not easy
to write
good
JavaScript
code.

Today, I won't tell you..

You should give up the way you did

I am going to tell you..

Maybe you have a better way to do with this

CoffeeScript

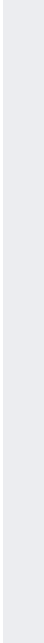
We're not
talking
about this
kind of
coffee.



WTF?

CoffeeScript is JavaScript

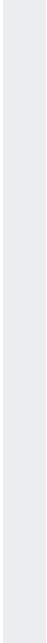
just written with different syntax



..exposes the good parts of
JavaScript in a simple way.

Syntax borrowed from
Python and Ruby.

I love Python & Ruby



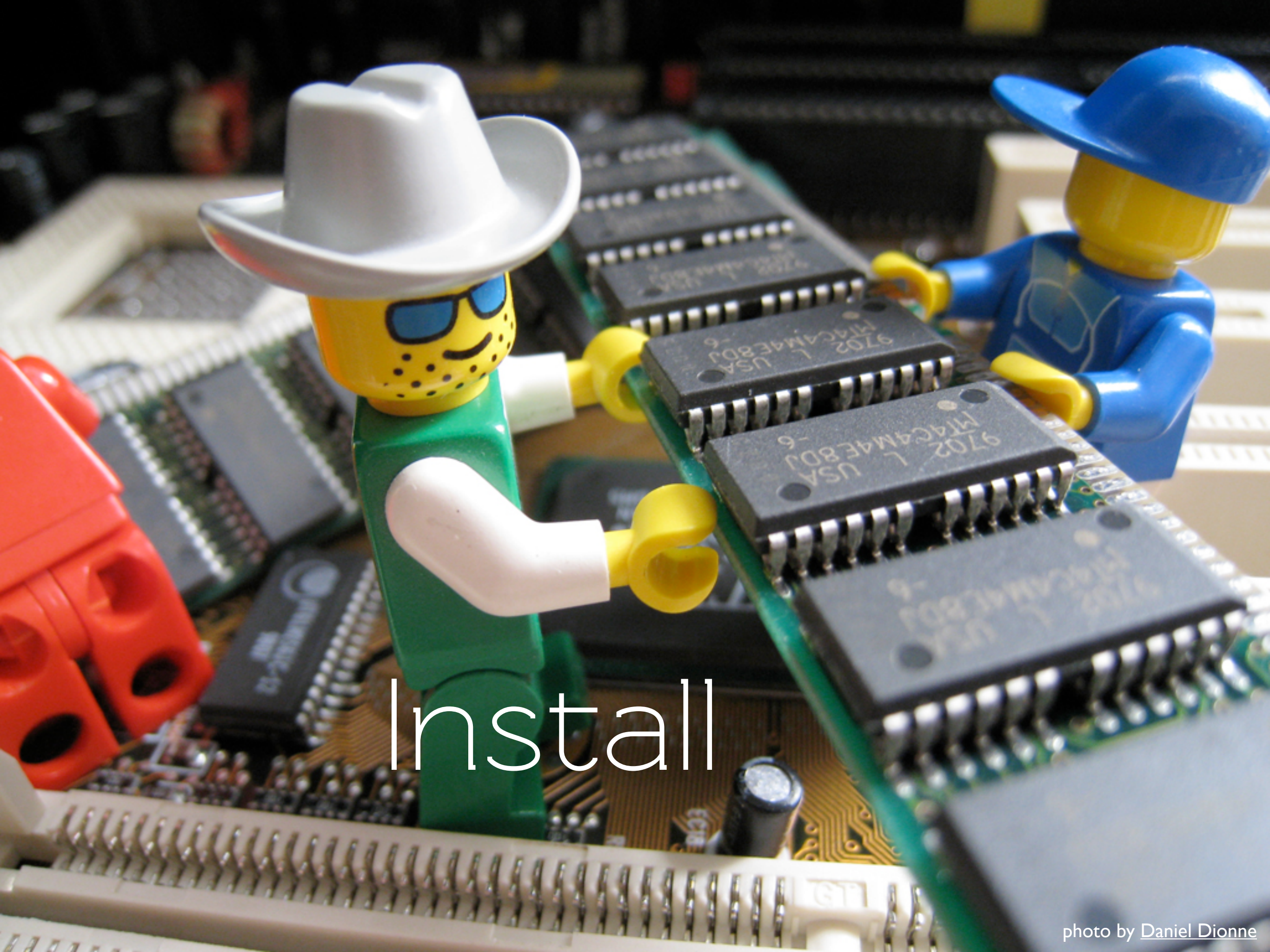
..and will be compiled into
JavaScript code.

..that doesn't mean you can
have no knowledge about
JavaScript.

CoffeeScript is not used
to replace JavaScript.



Let's get our feet wet!



Install

Requirements

You need to install some software first..

Node.js

- > `git clone git://github.com/joyent/node.git`
- > `cd node`
- > `./configure`
- > `make`
- > `sudo make install`

Requirements

You need to install some software first..

NPM, the “node package manager”

```
> curl http://npmjs.org/install.sh | sh
```

Install CoffeeScript

CoffeeScript

```
> npm install coffee-script  
> coffee -v  
CoffeeScript version 1.1.3
```



How to use

Usage

Compile *.coffee into *.js

Compile

```
> coffee --watch --compile app.coffee
```



Syntax

}

}

}

}

No { }

indentations rule!

whitespace matters!

() is not necessary.

No trailing semicolon.

Return is not necessary.

everything is an expression

No { }, (), and ;

```
if(age > 20){  
    vote();  
}
```

// javascript

```
if age > 20  
    vote()
```

coffeescript

Variable & Function

Variable

You don't have to declare it before using it.

```
lang = ["php", "python", "perl", "ruby"]  
name = "Eddie"
```

coffeescript

```
var lang, name;  
lang = ["php", "python", "perl", "ruby"];  
name = "Eddie";
```

// javascript

Function

```
say_hello = (guest1, guest2 = "Nayumi") ->  
  "Hello #{guest1} and #{guest2}"
```

```
say_hello "Eddie"
```

coffeescript

```
var say_hello;  
say_hello = function(guest1, guest2) {  
  if (guest2 == null) {  
    guest2 = "Nayumi";  
  }  
  return "Hello " + guest1 + " and " + guest2;  
};  
say_hello("Eddie");
```

// javascript

Array

Array

```
heroes = [  
  'Spider Man',  
  'Captain America',  
  'X-men',  
  'Iron Man'  
]
```

```
students = [1..10]  
teachers = [1..10]
```

coffeescript

```
var heroes, students, teachers;  
heroes = ['Spider Man', 'Captain America', 'X-men', 'Iron Man'];  
  
students = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];  
teachers = [1, 2, 3, 4, 5, 6, 7, 8, 9];
```

// javascript

Array

```
heroes[0..2]
```

```
heroes[1..2] = ["Batman", "ThunderCat"]
```

coffeescript

```
heroes.slice(0, 3);
```

```
var _ref;
```

```
[].splice.apply(heroes, [1, 2].concat(_ref = ["Batman", "ThunderCat"])), _ref;
```

// javascript

Object

Object

```
eddie = { name: "Eddie Kao", age: 18, speciality: "eat" }
```

coffeescript

```
var eddie;  
eddie = {  
  name: "Eddie Kao",  
  age: 18,  
  speciality: "eat"  
};
```

// javascript

Object

```
eddie =  
  name: "Eddie Kao"  
  age: 18  
  lovers:  
    nayumi:  
      name: "Nayumi Hung"  
      age: 18  
    mary:  
      name: "Mary Bloody"  
      age: 20
```

coffeescript

```
var eddie;  
eddie = {  
  name: "Eddie Kao",  
  age: 18,  
  lovers: {  
    nayumi: {  
      name: "Nayumi Hung",  
      age: 18  
    },  
    mary: {  
      name: "Mary Bloody",  
      age: 20  
    }  
  }  
};
```

// javascript

Loop

Loop

```
alert i for i in [1..10]
```

```
alert i for i in [1..10] when i  
% 2 == 0
```

```
alert i for i in [1..10] by 2
```

```
alert i * 2 for i in [1..10]
```

```
# coffeescript
```

```
var i, _step;  
for (i = 1; i <= 10; i++) {  
    alert(i);  
}  
for (i = 1; i <= 10; i++) {  
    if (i % 2 === 0) {  
        alert(i);  
    }  
}  
for (i = 1, _step = 2; i <= 10; i +=  
_step) {  
    alert(i);  
}  
for (i = 1; i <= 10; i++) {  
    alert(i * 2);  
}  
// javascript
```

Modifier

Modifier

You can put "if", "unless", "while", "until" behind

```
if (age > 20) {  
  vote();  
}
```

// javascript

```
vote() if age > 20
```

coffeescript

Syntactic Sugar

Syntactic Sugar

write more readable code by using syntactic sugar.

coffeescript

// javascript

is	===
isnt	!==
true, on, yes	true
false, off, no	false
not	!
and	&&
or	
unless	if not
until	while not
of	in

Syntactic Sugar

```
alert "I can't see anything" if light is off
```

```
alert "It's impossible!" if eddie isnt handsome
```

```
if girl is not single  
    alert "Don't Touch! Be Careful!"
```

coffeescript

```
if (light === false) {  
    alert("I can't see anything");  
}  
if (eddie !== handsome) {  
    alert("It's impossible!");  
}  
if (girl === !single) {  
    alert("Don't Touch! Be Careful!");  
}
```

// javascript

Syntactic Sugar

```
if (Answer === true) {  
    alert("I'll marry you!");  
}
```

// javascript

alert "I'll marry you!" if Answer is yes

Syntactic Sugar

```
age = 18</pre
```

```
# coffeescript
```

```
if (typeof age !== "undefined" && age !== null) {  
    age;  
} else {  
    age = 18;  
};
```

```
// javascript
```

Raw JavaScript

If you still prefer the original way

Raw JavaScript

```
say_hello = `function(name){  
  return "Hello, " + name  
}`
```

coffeescript

```
var say_hello;  
say_hello = function(name){  
  return "Hello, " + name  
};
```

// javascript

oop

OOP - new

```
class Animal
  constructor: (name, age) ->
    this.name = name
    this.age = age
```

```
animal = new Animal("eddie", 18)
alert animal
```

coffeescript

```
var Animal, animal;
Animal = (function() {
  function Animal(name, age) {
    this.name = name;
    this.age = age;
  }
  return Animal;
})();
animal = new Animal("eddie", 18);
alert(animal);
```

// javascript

OOP - method

```
class Animal
  constructor: (@name, @age) ->

  say_hello: (something) ->
    console.log "Hello, #{something}"
```

```
animal = new Animal("eddie", 18)
animal.say_hello("CoffeeScript")
```

coffeescript

```
var Animal, animal;
Animal = (function() {
  function Animal(name, age) {
    this.name = name;
    this.age = age;
  }
  Animal.prototype.say_hello = function(something) {
    return console.log("Hello, " + something);
  };
  return Animal;
})();
animal = new Animal("eddie", 18);
animal.say_hello("CoffeeScript");
```

// javascript

OOP - inheritance

```
class Animal
  constructor: (@name, @age) ->
  say_hello: (something) ->
    alert "Hello, #{something}"
```

```
class Human extends Animal
  walk: ->
    alert "I can walk with my foots!"
```

```
eddie = new Human("eddie", 18)
eddie.say_hello "CoffeeScript"
eddie.walk()
```

coffeescript

TL; DR

// javascript

References

Websites:

<http://jashkenas.github.com/coffee-script/>

<http://blog.eddie.com.tw/category/coffeescript/>

Book:

<http://pragprog.com/book/tbcoffee/coffeescript>



LinkedIn

LinkedIn

lin.co

JavaScript: The Good Parts

JavaScript
The Definitive Guide

Flanagan

O'REILLY

Crockford

O'REILLY

the good parts
I love

photo by [Steve Ganz](#)

I Love..

Coding Style

I love Python & Ruby, of course :)

I Love..

Indentation!

I Love..

Anonymous function

No global function and variable by default

I Love..

String Interpolation

sorry, but string building really sucks :)

I Love..

List Comprehension

I Love..

Syntactic Sugar

I Love..

English-like grammar

alert "of course it is!" if PHPConf is awesome

I Love..

Comparison & Equality

```
"true" == true // true
```

```
"true" === true // false
```

I Love..

Works with other JS
frameworks well.

Because it's just JavaScript

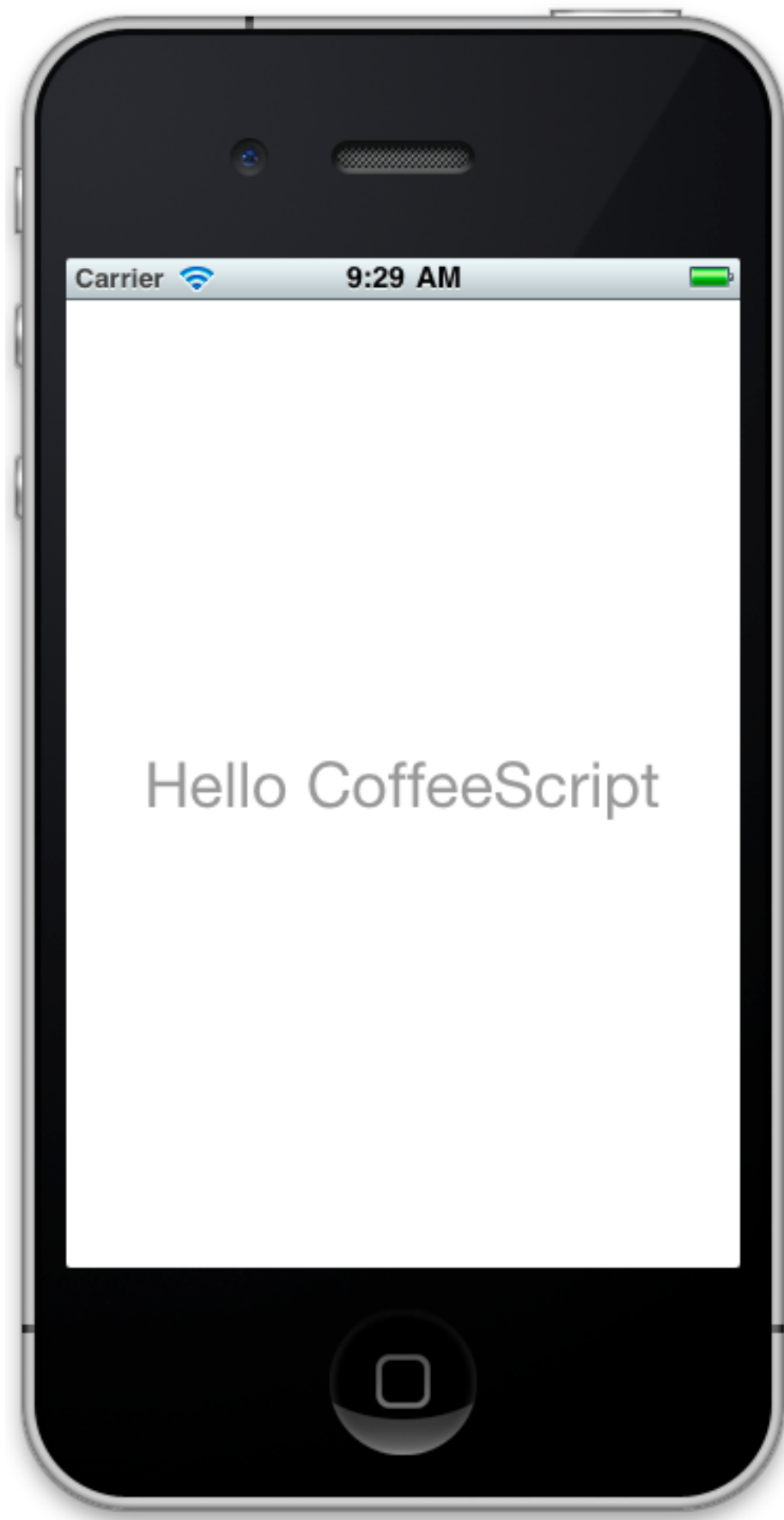
I Love..

Compilation

JSLint Approved

What else?

CoffeeScript compiler is
written in CoffeeScript.



Titanium Mobile

<http://blog.eddie.com.tw/2011/08/03/using-coffeescript-in-titanium-studio/>



Ruby on Rails

Sustainable productivity for web-application development



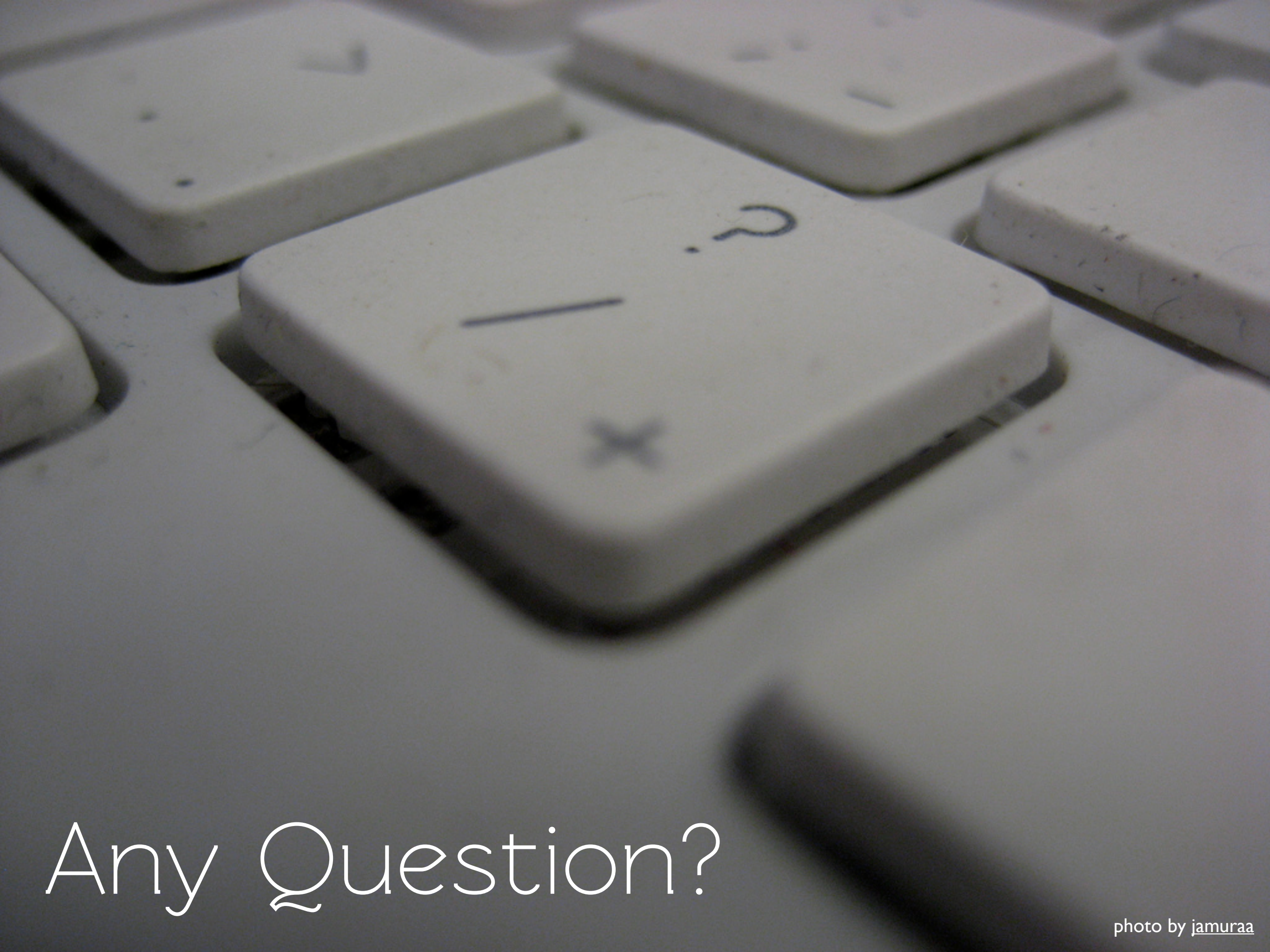
Immature?



Performance?

CoffeeScript Means Giving
Up on JavaScript?

Learn JavaScript, and Use
CoffeeScript.



Any Question?

Contacts

高見龍

- Website <http://www.eddie.com.tw>
- Blog <http://blog.eddie.com.tw>
- Plurk <http://www.plurk.com/aquarianboy>
- Facebook <http://www.facebook.com/eddiekao>
- Google Plus <http://www.eddie.com.tw/+>
- Twitter <https://twitter.com/#!/eddiekao>
- Email eddie@digik.com.tw
- Mobile +886-928-617-687