



Happy Programming with CoffeeScript



# 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.
- Mac OS user, Objective-C for personal interests.
- Technical Education and Consultant.
- PTT Flash BM (since 2007/4).
- Ruby Certified Programmer (Since 2012/1).
- Adobe Certified Flash Developer (Since 2006/7).
- Linux Professional Institute Certification (Since 2005/3).







高見龍

@eddiekao

*Flash/AS3 Developer / iOS & Rails Rookie*

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

Taipei, Taiwan · <http://www.eddie.com.tw/>



# Intro

簡介



Popular



Popular

Powerful

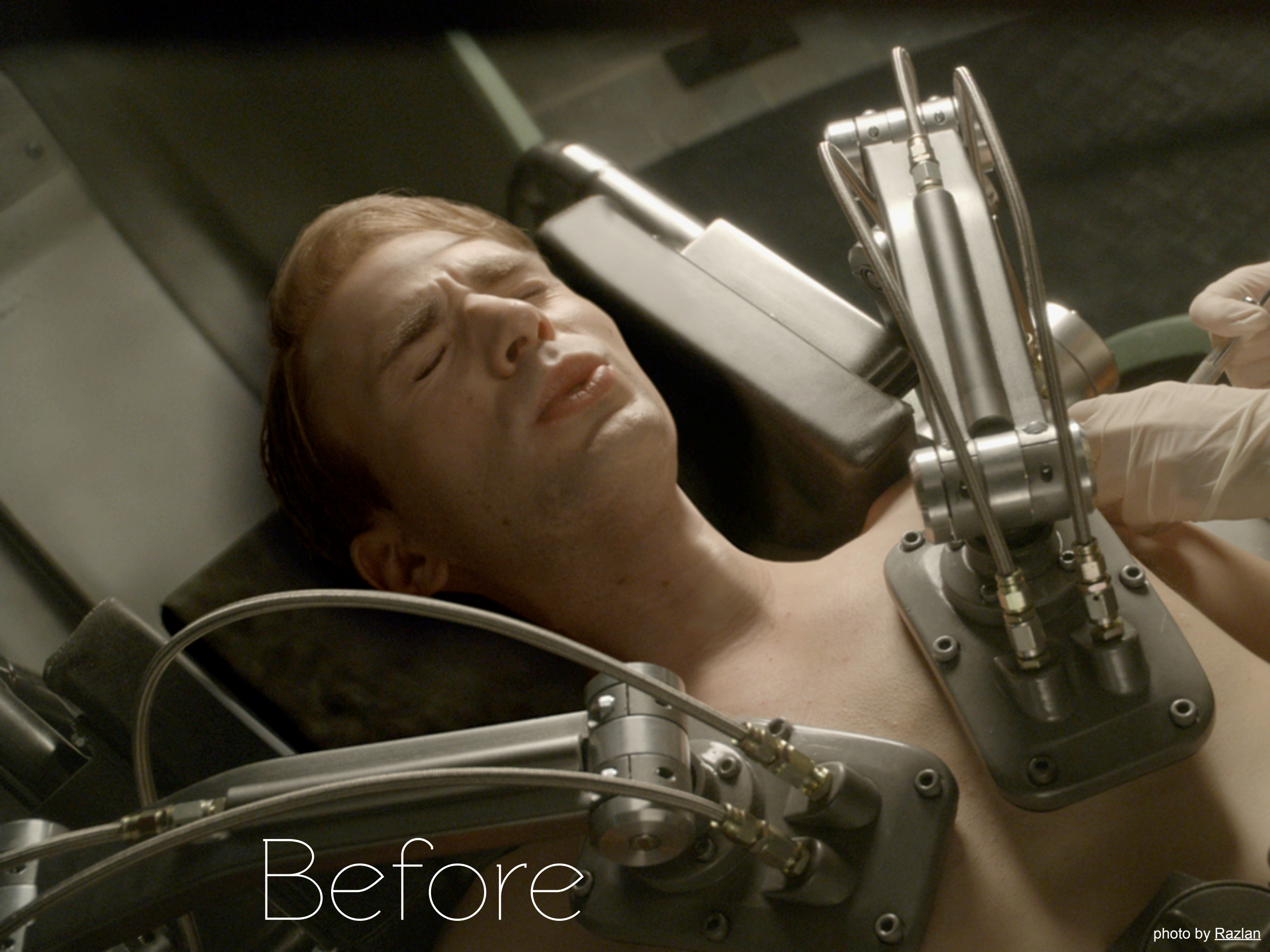


Popular

Powerful

Complicated





Before





Now





seems  
easy





Today, I am going to tell you..



Happy

快樂

We're not  
talking  
about this  
kind of  
coffee.





Not this either.





I have no idea why engineers like using food as their project name.



- 040-literal
- 060-signif
- 070-postf
- 100-everyt
- 110-lexica
- 120-switch
- 200-string
- 205-chain
- 210-comp
- 220-half-a
- 220-splats
- 230-existe
- 260-classe
- 280-patter
- 281-fancy
- 300-self-c
- underscore

Jeremy  
Ashkenas

2009/12



and I started to use  
CoffeeScript since 2011/5

and I started to use  
CoffeeScript since 2011/5

because of Rails



**Jeremy Ashkenas**

@jashkenas

Following



Excited to learn how many people have 5+ years experience with CoffeeScript on GitHub. Clearly there's much for me to learn from them...

← Reply

↻ Retweeted

★ Favorite

50+

RETWEETS

24

FAVORITES



7:34 AM - 13 Jan 12 via Twitter for Mac · Embed this Tweet



yet another new language !?



# CoffeeScript

[TABLE OF CONTENTS](#)[TRY COFFEES](#)

**CoffeeScript is a little language that compiles into JavaScript.** Underneath braces and semicolons, JavaScript has always had a gorgeous object model and CoffeeScript is an attempt to expose the good parts of JavaScript in a simple

The golden rule of CoffeeScript is: *"It's just JavaScript"*. The code compiles on equivalent JS, and there is no interpretation at runtime. You can use any existing code seamlessly from CoffeeScript (and vice-versa). The compiled output is readable, passes through [JavaScript Lint](#) without warnings, will work in every JavaScript environment, and is guaranteed to run as fast or faster than the equivalent handwritten JavaScript.

**Latest Version:** [1.3.3](#)



"It's just JavaScript"

just written in different syntax



borrowed from  
Python, Ruby and Haskell.

and would be compiled into  
JavaScript code



JSLint Compatible

So, it's still Javascript



"hmm.."

"It's just JavaScript"



~~"It's just JavaScript"~~

"I think it's just CoffeeScript"



whatever..

WTH can you do with  
CoffeeScript?

anything which JavaScript can do.

front-end, back-end, mobile.. etc.



but with less line of code..  
~ 30% off

CoffeeScript compiler is written in  
CoffeeScript

A close-up photograph of a young man's face, focusing on his eyes, nose, and skin. His skin appears to have some issues, with visible redness, small bumps, and what looks like mild acne or irritation, particularly around the eyes and on the forehead. He has light brown hair and is looking directly at the camera with a neutral expression. The background is dark and out of focus.

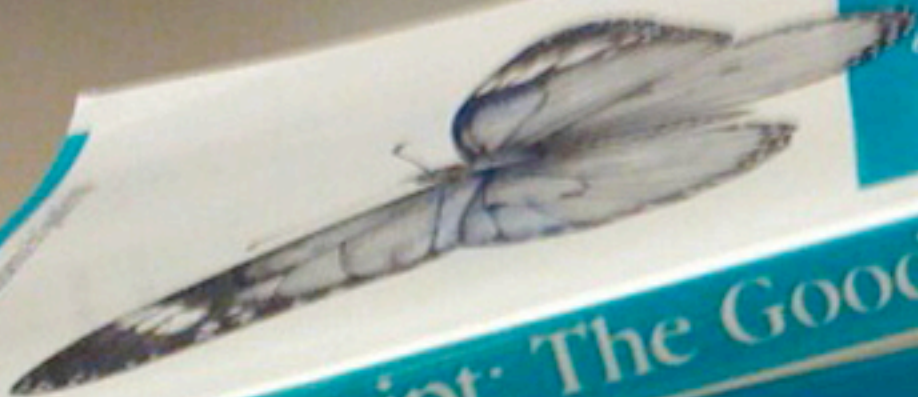
Is  
JavaScript  
that  
bad?



LinkedIn

LinkedIn

lin.co



JavaScript: The Good Parts

JavaScript  
The  
Definitive  
Guide

Flanagan

Crockford

O'REILLY

FOURTH EDITION  
COVERS JAVASCRIPT 1.5

the good parts

CoffeeScript =  
English-like grammar +  
nice Syntactic Sugar +  
Goodies



# Syntax

語法



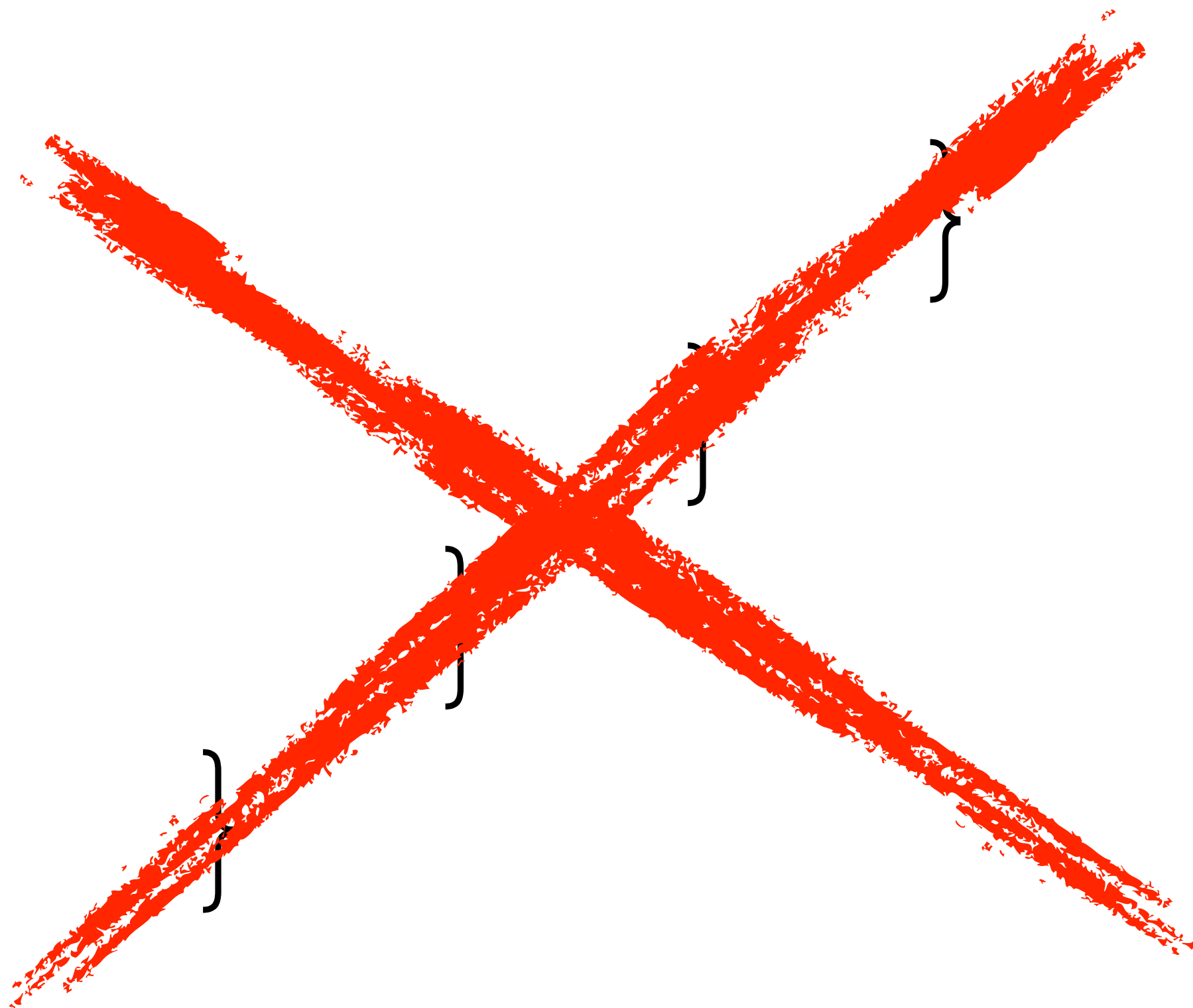
indentations rule,  
whitespace matters!

}

}

}

}







# String Interpolation

```
greeting = "hi, " + name + ", " + msg
```



```
greeting = "hi, #{name}, #{msg}"
```

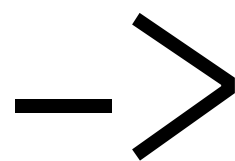
semicolon

# sermon

var



~~SECRET~~



```
say_hello = (msg) ->  
    console.log "hello, JSDC"
```

```
var say_hello;  
say_hello = function(msg) {  
    return console.log("hello, JSDC");  
};
```



default parameter

```
say_something = (msg = "JSDC") ->  
  console.log "say #{msg}"
```

```
var say_something;
```

```
say_something = function(msg) {  
    if (msg == null) {  
        msg = "JSDC";  
    }  
    return console.log("say " + msg);  
};
```

( ) is not always necessary.



```
greeting("eddie", "how are you");
```

greeting "eddie", "how are you"

everything is an expression

return is not necessary.

```
var hello;
```

```
hello = function() {  
    return "hi, JSDC";  
};
```



hello = ->

"hi, JSDC"

Array

```
var a;
```

```
a = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];
```

$$a = [1..10]$$

Loop



```
console.log i for i in [1..10]
```

```
var i, __i;
```

```
for (i = __i = 1; __i <= 10; i = ++__i) {  
    console.log(i);  
}
```

```
console.log "YES" if today == 'JSDC'
```

Sugar

yes, no



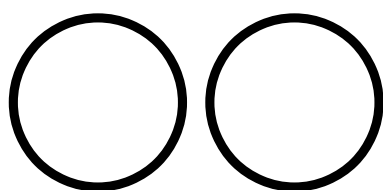
A: “would you marry me?”

B: true

A: “would you marry me?”

B: yes

console.log "go home" if today **isnt** "JSDC"



```
Hello.prototype.greeting = function(name) {  
    console.log("hi, " + name);  
};
```

```
class Hello
```

```
  greeting: (name) ->
```

```
    console.log "hi, #{name}"
```



```
class Hello
```

```
  constructor: (@name) ->
```

```
    greeting: (msg) ->
```

```
      "hi, #{@name}, #{msg}"
```

```
var Hello;  
Hello = (function() {  
    Hello.name = 'Hello';  
    function Hello(name) {  
        this.name = name;  
    }  
    Hello.prototype.greeting = function(msg) {  
        return "hi, " + this.name + ", " + msg;  
    };  
    return Hello;  
})();
```

# Diff?

差別?

# **the Original Way**

Your Brain



JavaScript



Browser

# the CoffeeScript way

Your Brain



CoffeeScript



JavaScript



Browser





lots of languages that compile to JS

<http://goo.gl/nkAkQ>

# Ref

參考

RTFM, it's awesome!

<http://coffeescript.org/>

or my shameless promotion blog

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





Let's get our feet wet!



# Demo

展示

But..

巴特..



CoffeeScript is not so perfect

Readability  $\neq$  Comprehension

People may still write  
CoffeeScript as horribly as they  
wrote JavaScript..

indentations rule,  
whitespace matters!

( ) is not always necessary.

Debugging tool?

You've better have a human  
CoffeeScript compiler in your head!



Then...

然後..

CoffeeScript is not used to replace  
JavaScript.

If you don't know Javascript, and  
you think CoffeeScript can save  
your life..

you would be in trouble.

So..

所以..

Who is using CoffeeScript?

actually, I don't really care!



Who won't need this?

People who don't like CoffeeScript.

People who already know  
javascript(the good parts) very  
well.

Who might need/like this?

someone who don't like ;

someone who don't like { }

someone who don't like the  
PROTOTYPE.

Rails developer



# Anyway

總之

Should I learn CoffeeScript?

Is it worth it?

IMHO

拙見

You should..

learn JavaScript!

learn JavaScript!

learn JavaScript!



learn JavaScript!

.. and use CoffeeScript !

totally worth it!

I can't go back  
anymore!

give it a try

Happy?

快樂?

END

結束

# 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](mailto:eddie@digik.com.tw)
- Mobile +886-928-617-687