Thomas Watson

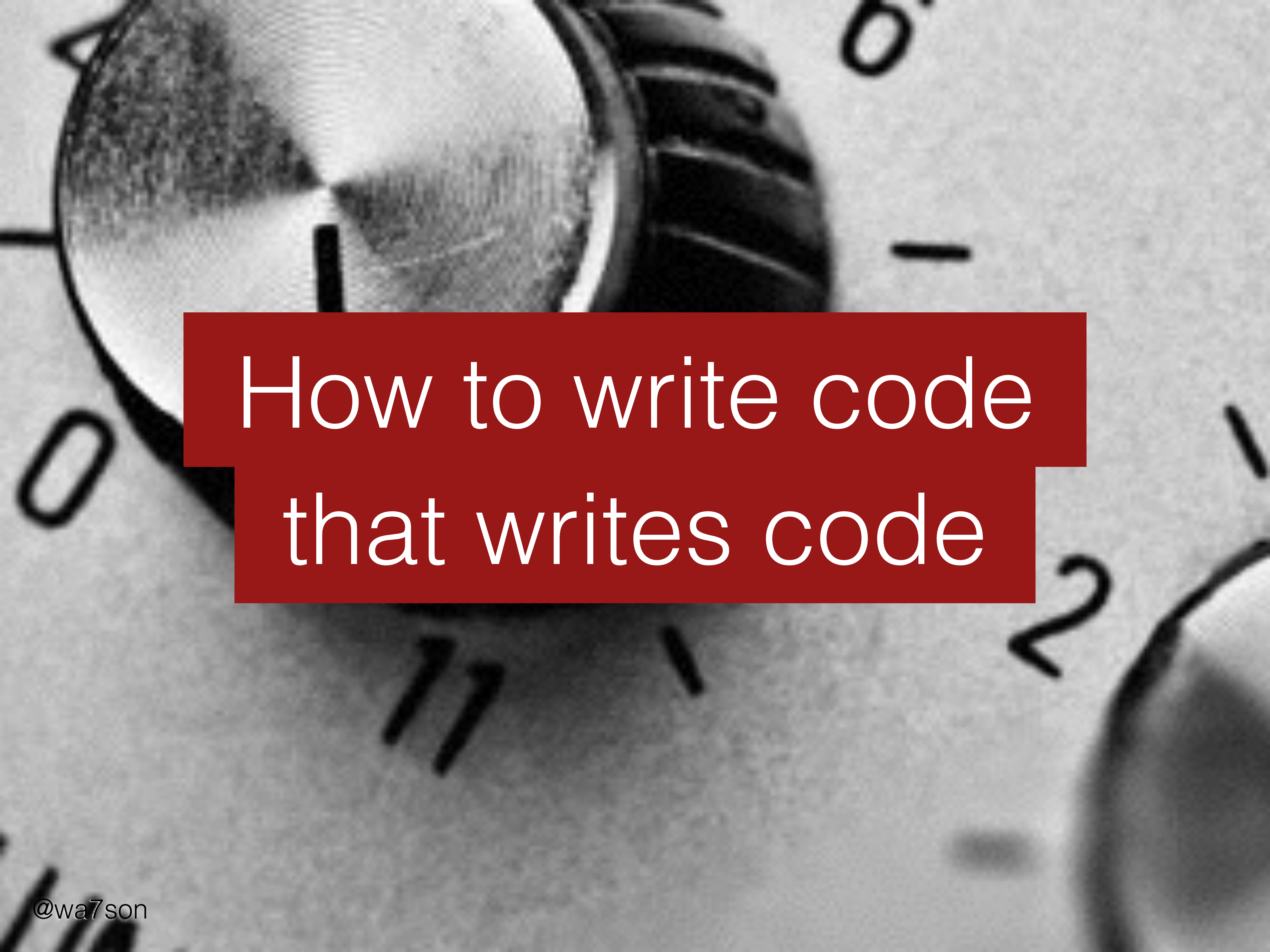
@wa7son

github.com/watson

@wa7son

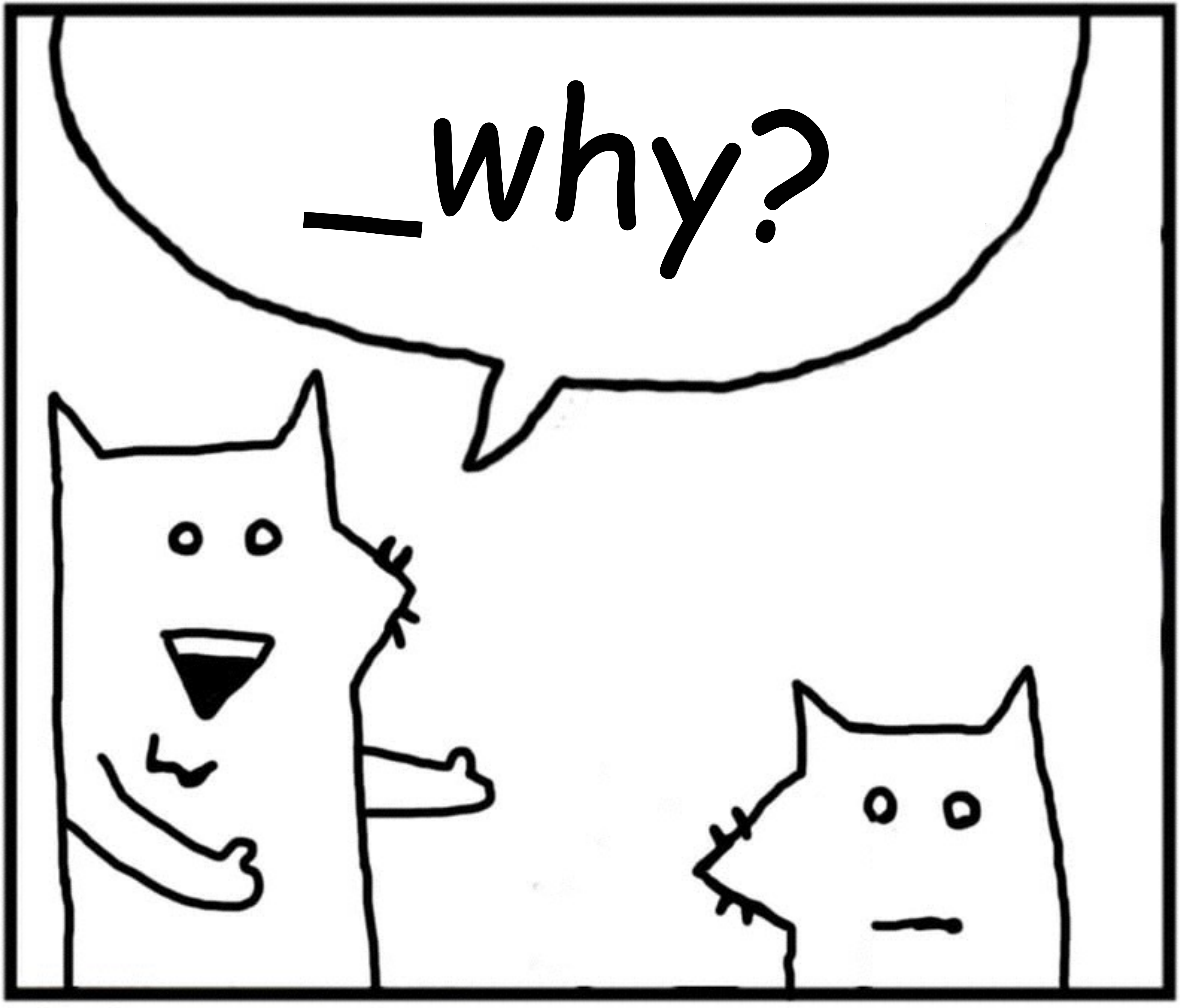
# It Goes To Eleven!

@wa7son



# How to write code that writes code

@wa7son


_why?

# performance

@wa7son

*f(seed, data)*

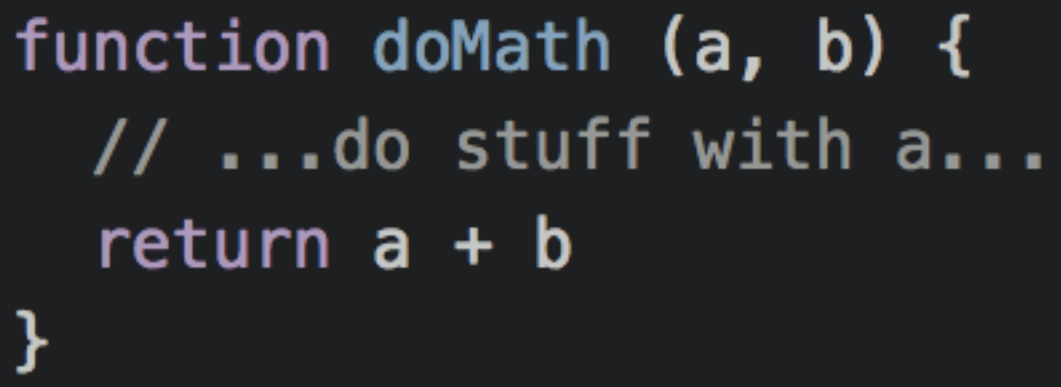
@wa7son

$$f_2 = f_1(seed)$$

$$f_2(data)$$

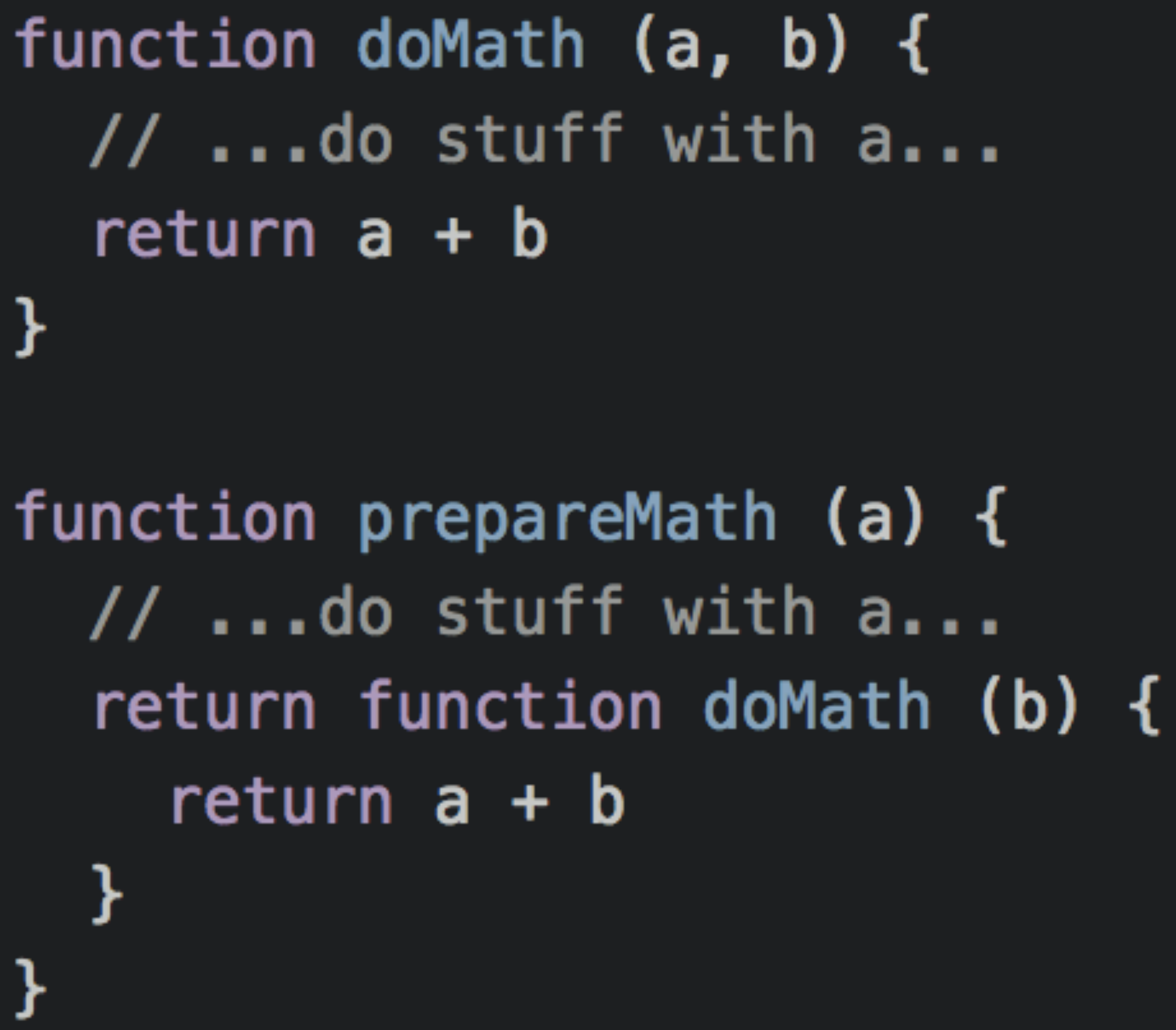
@wa7son

# An oversimplified example



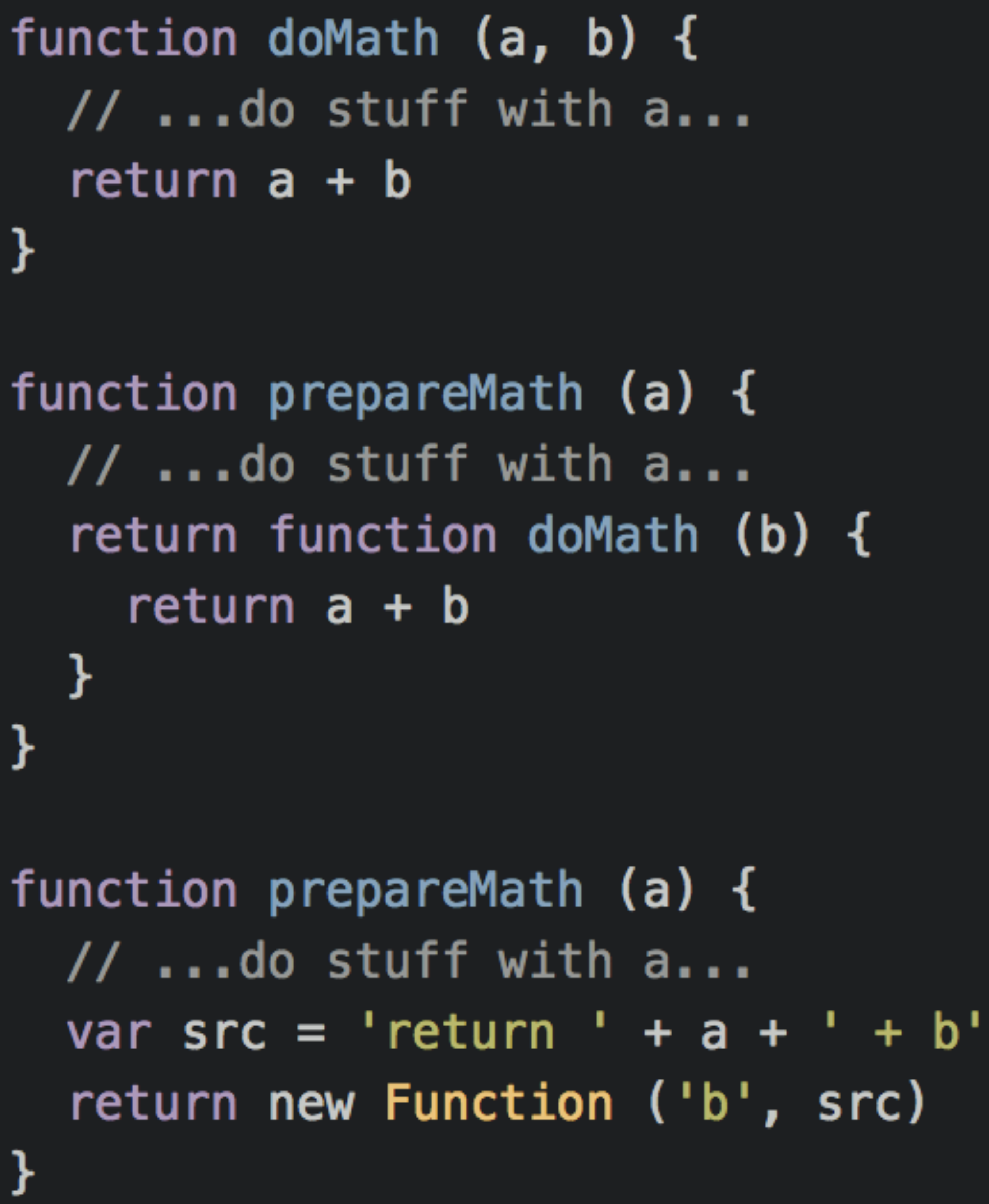
```
function doMath (a, b) {
  // ...do stuff with a...
  return a + b
}
```

# An oversimplified example



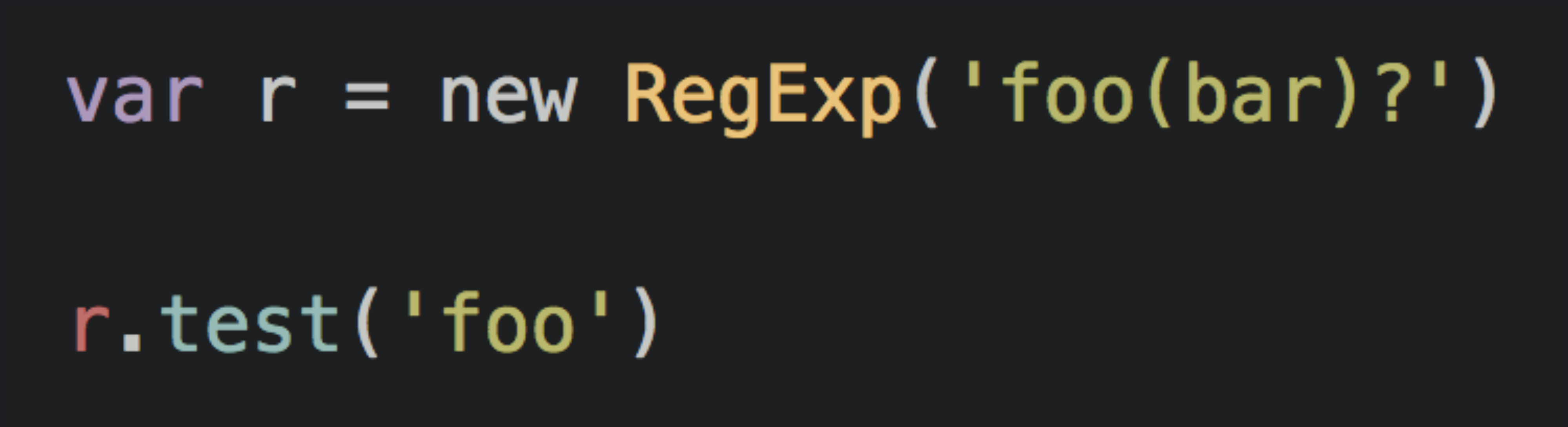
```
function doMath (a, b) {
  // ...do stuff with a...
  return a + b
}

function prepareMath (a) {
  // ...do stuff with a...
  return function doMath (b) {
    return a + b
  }
}
```

# An oversimplified example

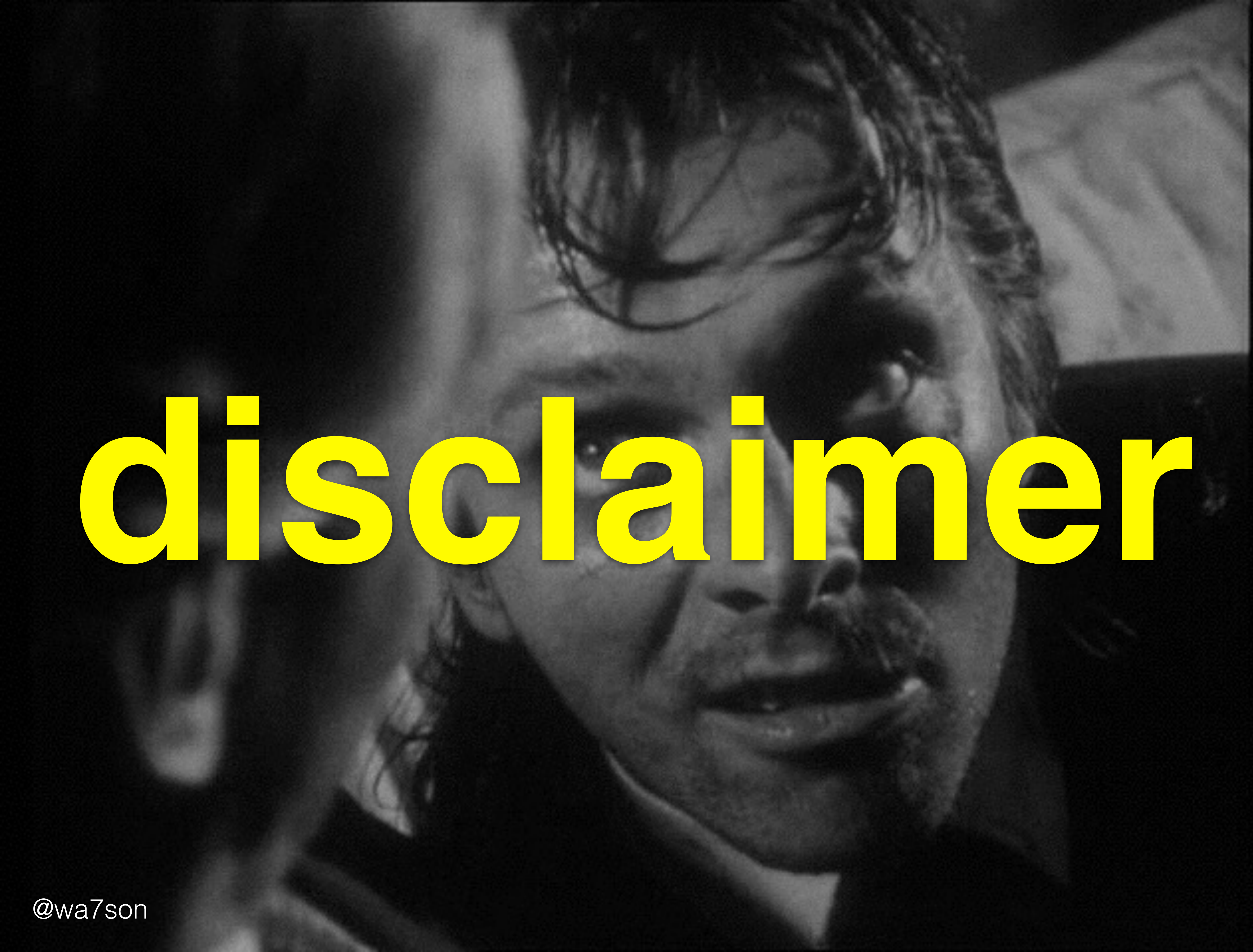


```
function doMath (a, b) {
  // ...do stuff with a...
  return a + b
}

function prepareMath (a) {
  // ...do stuff with a...
  return function doMath (b) {
    return a + b
  }
}

function prepareMath (a) {
  // ...do stuff with a...
  var src = 'return ' + a + ' + b'
  return new Function ('b', src)
}
```

# Regular Expressions

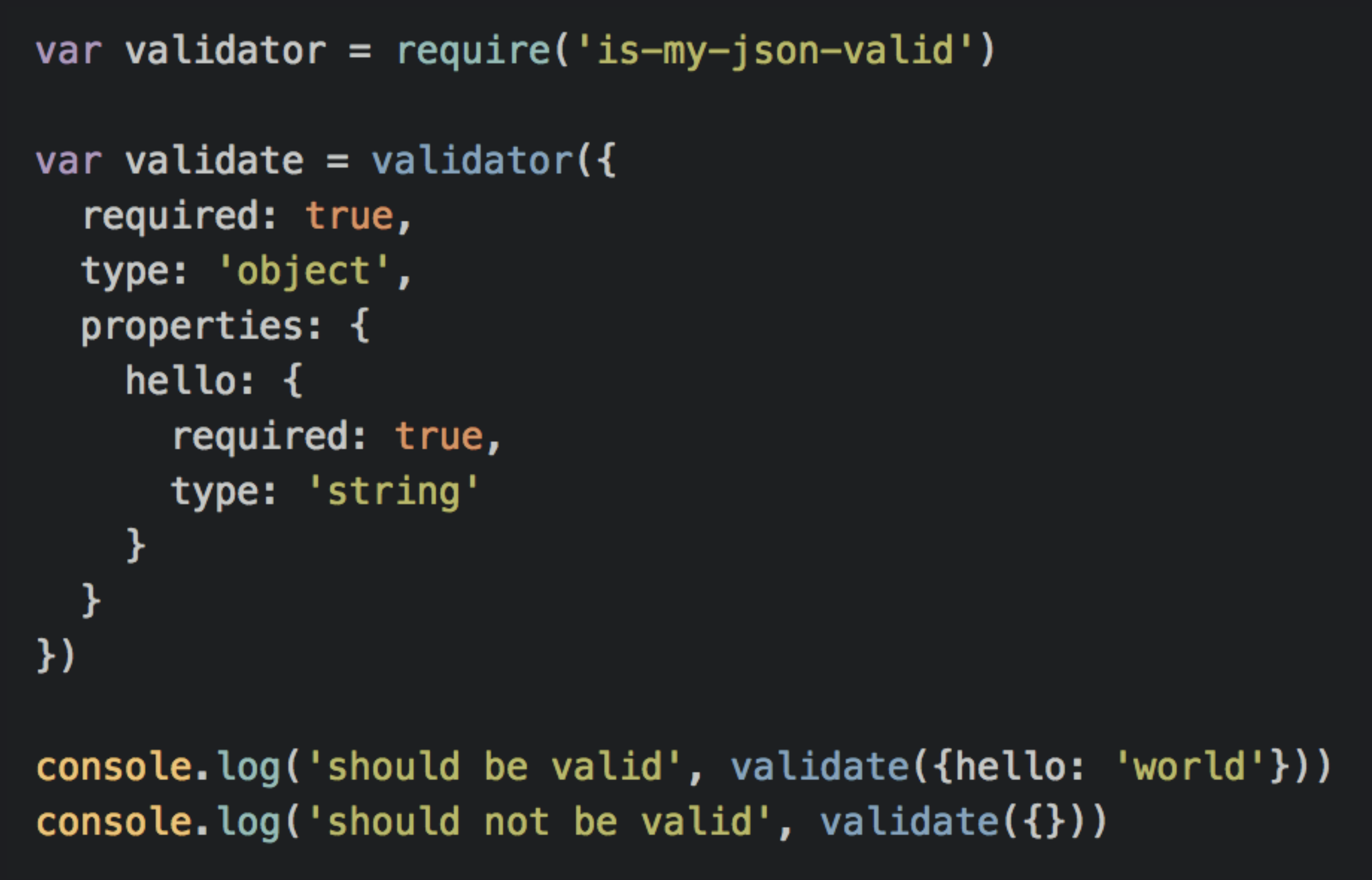


```
var r = new RegExp('foo(bar)?')

r.test('foo')
```

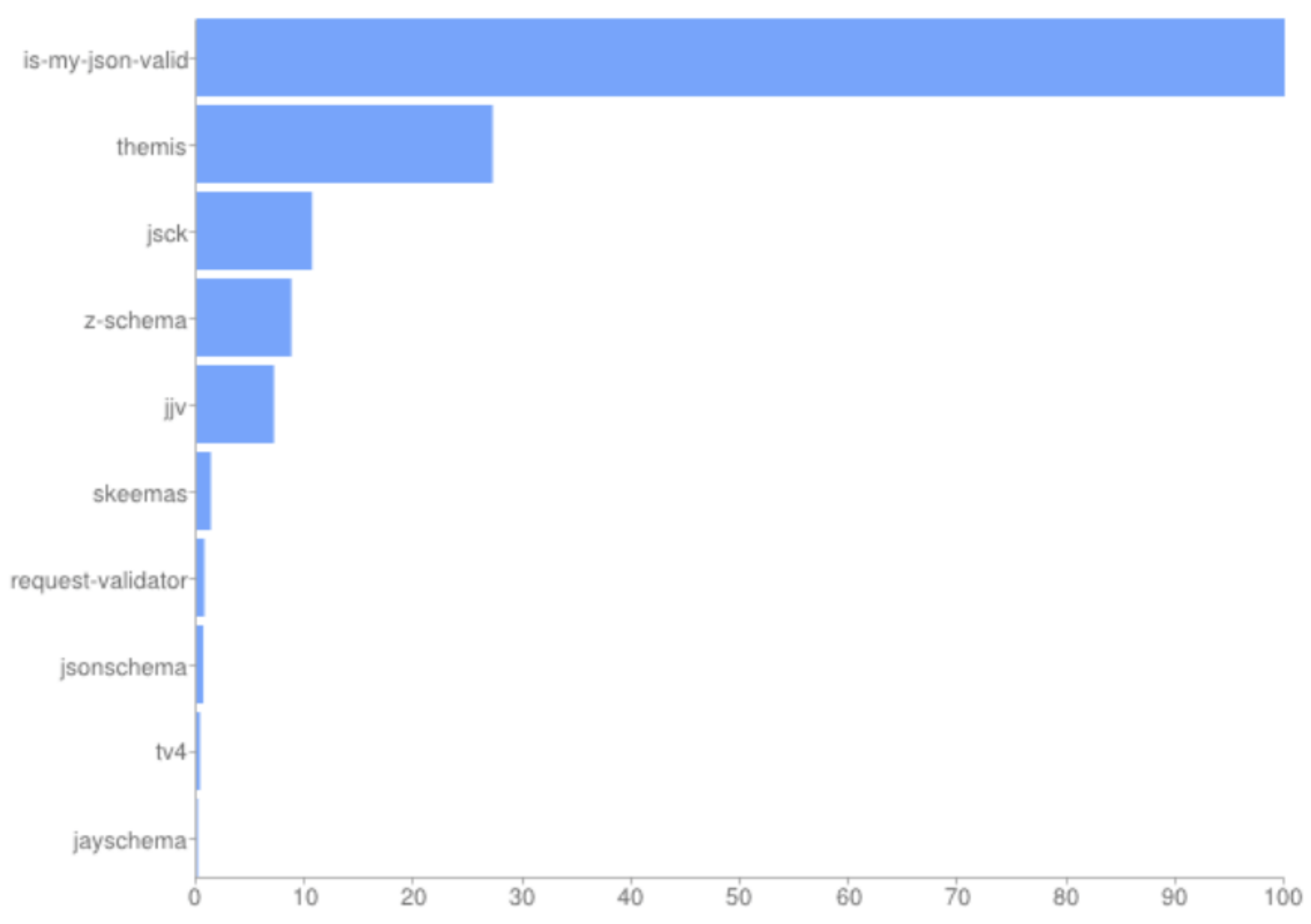


# disclaimer

github.com /
mafintosh /
is-my-json-valid



```
var validator = require('is-my-json-valid')

var validate = validator({
  required: true,
  type: 'object',
  properties: {
    hello: {
      required: true,
      type: 'string'
    }
  }
})

console.log('should be valid', validate({hello: 'world'}))
console.log('should not be valid', validate({}))
```


is-my-json-valid
themis
jsck
z-schema
jjv
skeemas
request-validator
jsonschema
tv4
jayschema
0
10
20
30
40
50
60
70
80
90
100

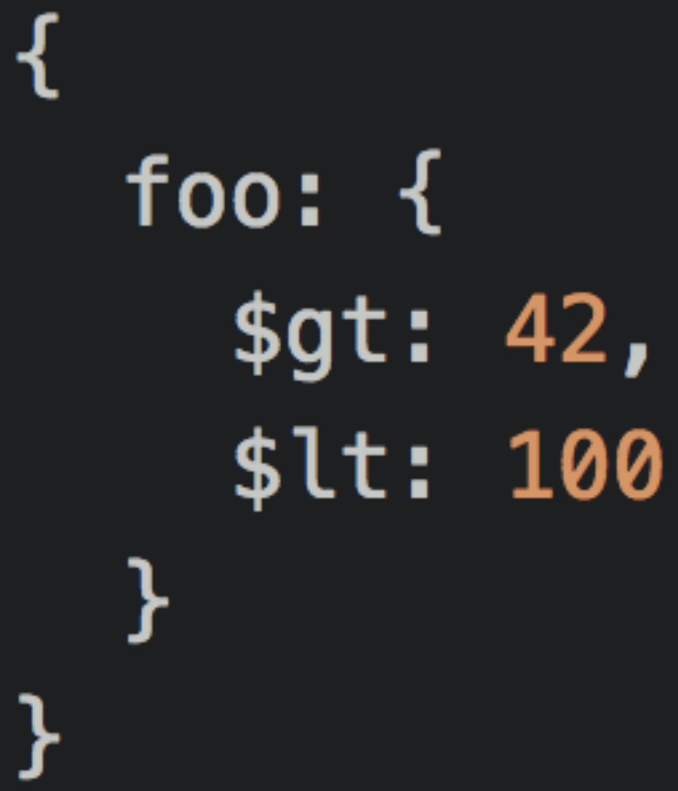
@wa7son

# use cases

@wa7son

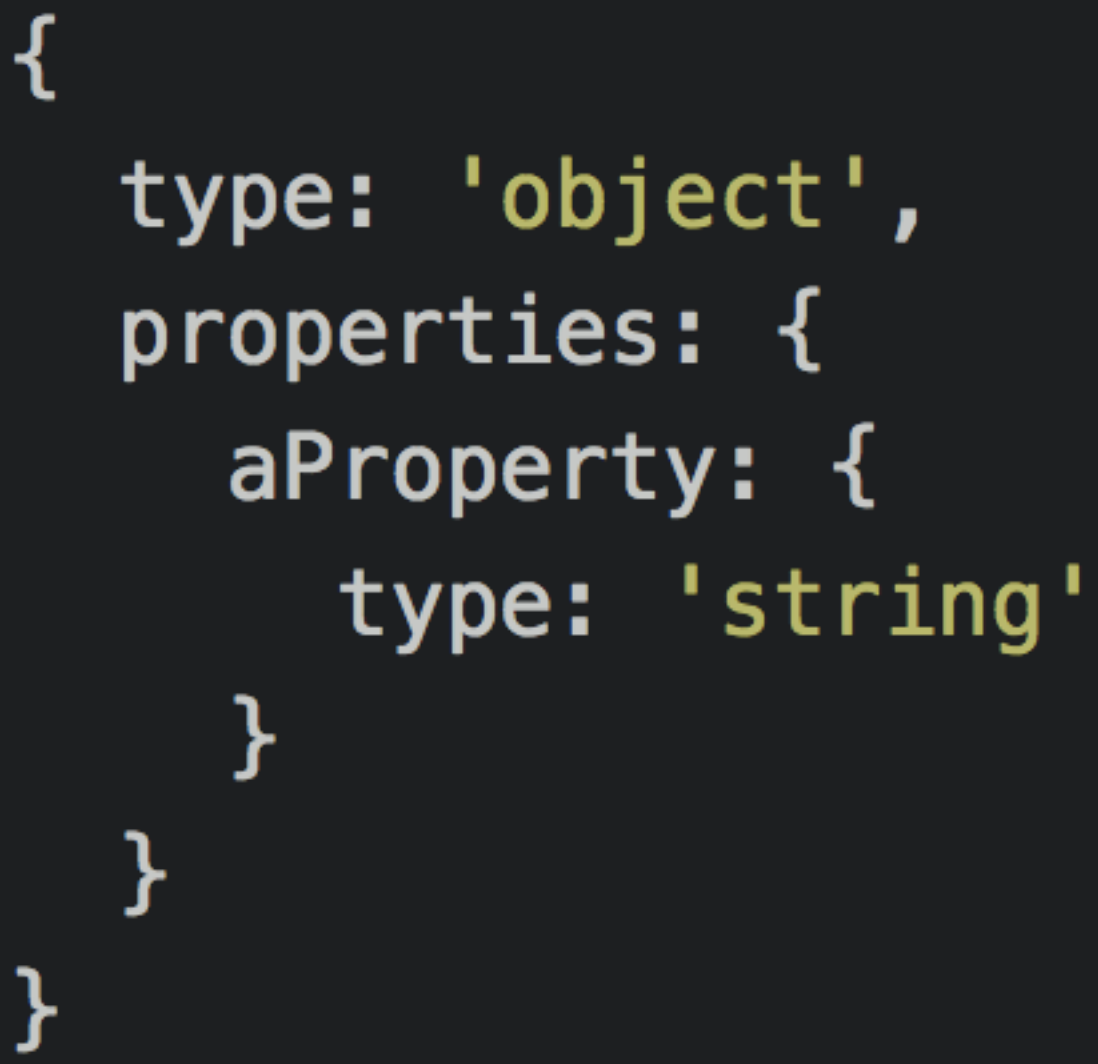
# Let’s write a query language interpreter

# mongodb query language



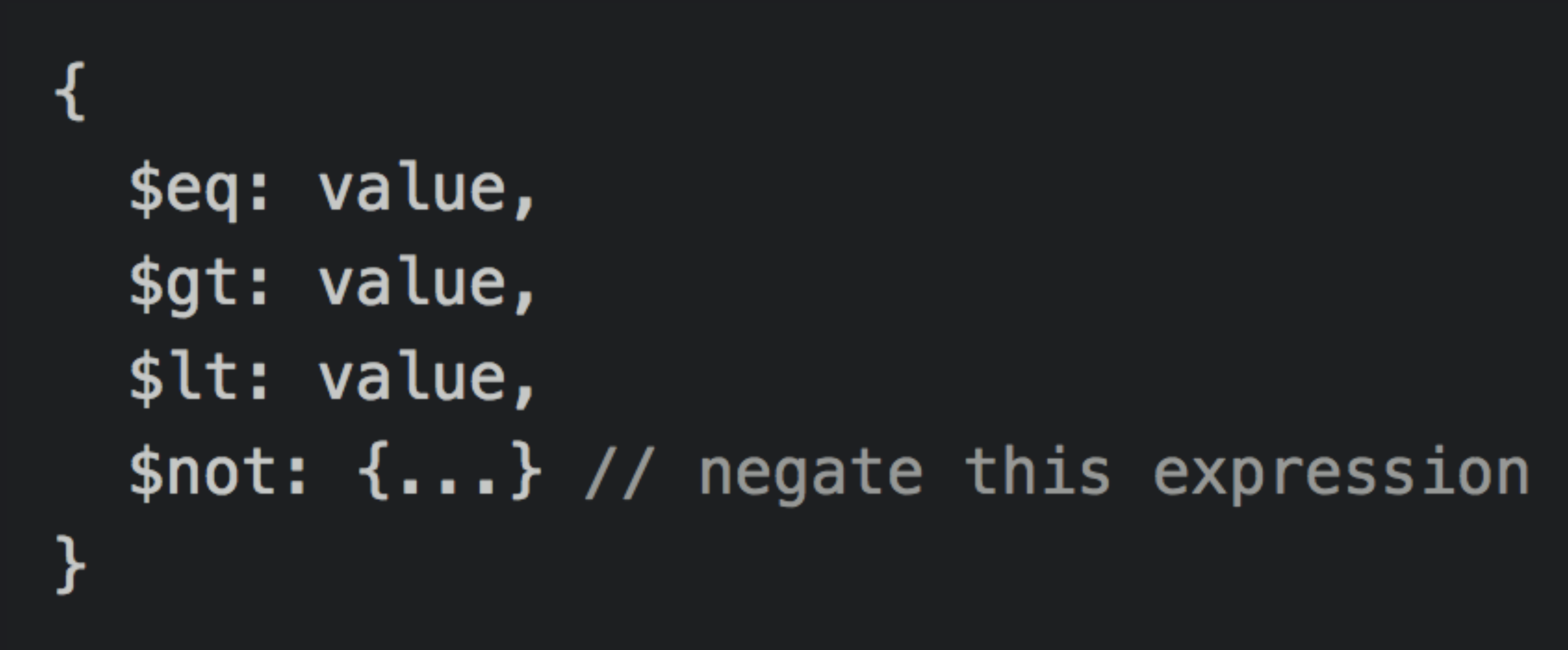
```
{
  foo: {
    $gt: 42,
    $lt: 100
  }
}
```

# JSON Schema



```
{
  type: 'object',
  properties: {
    aProperty: {
      type: 'string'
    }
  }
}
```

# Our language

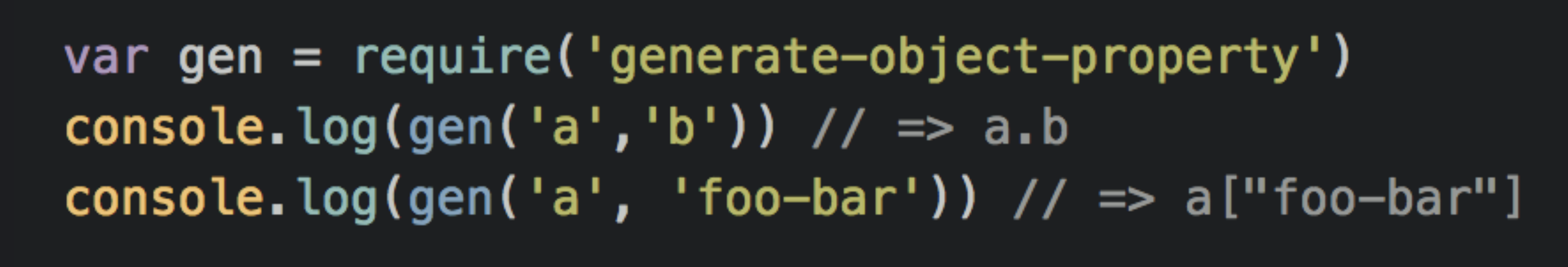


```
{
  $eq: value,
  $gt: value,
  $lt: value,
  $not: {...} // negate this expression
}
```

# coding time

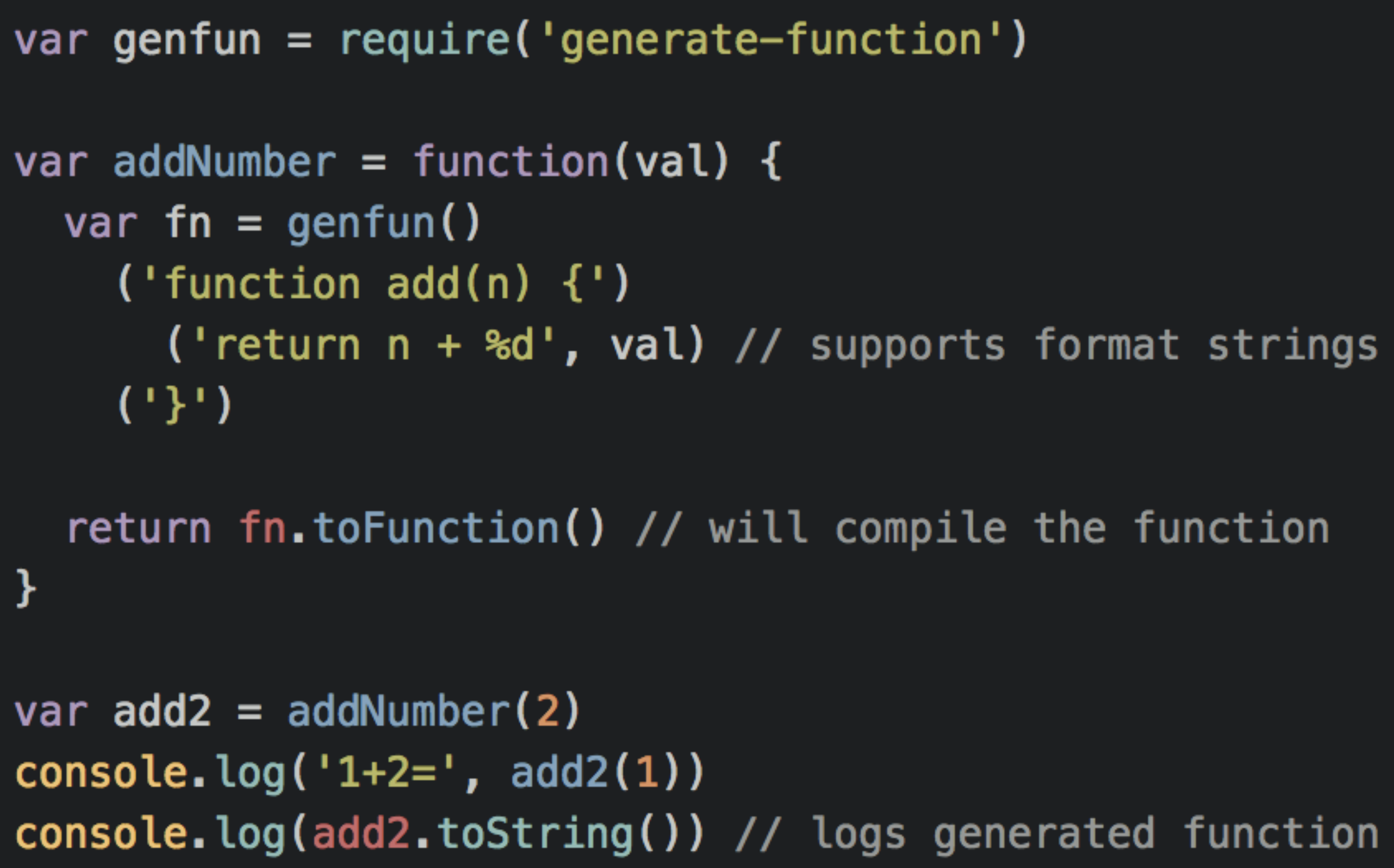
@wa7son

github.com /
mafintosh /
generate-object-property



```
var gen = require('generate-object-property')
console.log(gen('a','b')) // => a.b
console.log(gen('a', 'foo-bar')) // => a["foo-bar"]
```

github.com /
mafintosh /
generate-function



```
var genfun = require('generate-function')

var addNumber = function(val) {
  var fn = genfun()
    ('function add(n) {')
      ('return n + %d', val) // supports format strings
    ('}')

  return fn.toFunction() // will compile the function
}

var add2 = addNumber(2)
console.log('1+2=', add2(1))
console.log(add2.toString()) // logs generated function
```

# Thank you!

@wa7son

github.com/watson

opbeat