

Set Named Parameters in a Curried Proc (#6253)

Currently:

- Curried Procs are not much used
- Problem - `a_curried_proc.call('foo') #=> proc? value? no way to know`

In Ruby 2.0, it gets worse:

```
my_proc = proc { |a, name: 'Matz'| puts "#{name} said #{a}" }.curry
my_proc.call('Hello') #=> "Matz said Hello"
my_proc.call('Hi', name: 'Josh') #=> "Josh said Hi"
my_proc.call(name: 'Rodin').call("I'm thinking...") #=> not possible
```

But it could be so much better:

- Method object is a common pattern
- Making named parameters into properties on curried procs means easy method objects in Ruby!
- Example:

```
def greet(name, greeting: "Hello")
  puts "#{greeting}, #{name}!"
end

greet_method = method(:greet).to_proc.curry
greet_method.greeting = "こんにちは"

greet_method.call("Ruby") #=> "こんにちは, Ruby!"
```