

Ruby - Feature #18736

self.p for method chain

04/15/2022 09:03 AM - aDAVISk (Akito D. Kawamura)

Status: Open

Priority: Normal

Assignee:

Target version:

Description

You may want to check object with p method at the middle of a method chain.
Here is my recommendation.

```
class Object
  def sp(method=nil, *args, &block)
    if method
      Kernel.p self.public_send(method, *args, &block)
    elsif block_given?
      Kernel.p block.call(self)
    else
      Kernel.p self
    end
    return self
  end
end
```

Example of usage:

1. check itself in the middle of a method-call

```
p [1,2,3].map{|x| x**2}.sp.map{|x| x**2}
```

- output

```
[1,4,9]
[1,16,81]
```

1. check its value in some string format

```
[1,2,3].sp{|x| "my List = #{x}"}
```

- output

```
"my List = [1,2,3]"
```

1. check its sum with an initial value

```
[1,2,3].sp(:sum,-10)
```

- output

```
-4
```

1. check its value with a map operation

```
[1,2,3].sp(:map){|x| x**2}
```

- output

```
[1,4,9]
```

1. check its value connected in string

```
[1,2,3].sp(:sum,""){|x| x.to_s}
```

- output

"123"

Your brush-up comments are welcomed. thx.

Related issues:

Related to Ruby - Feature #14609: Let `Kernel#p` without an argument print th...

Rejected

History

#1 - 04/15/2022 09:29 AM - sawa (Tsuyoshi Sawada)

It just looks too ugly to me. What do you gain from passing the block to sp instead of the method that is originally supposed to handle it? The following conventional way is much more straightforward. (In addition, in your first example, why are you using p ... in one place instead of sp? It is inconsistent.)

```
[1, 4, 9].tap{|x| p x}.map{|x| x**2}.tap{|x| p x}
#>> [1, 4, 9]
#>> [1, 16, 81]
[1,2,3].tap{|x| p "my List = #{x}"}
#>>"my List = [1, 2, 3]"
[1,2,3].sum(-10).tap{|x| p x}
#>> -4
[1,2,3].map{|x| x**2}.tap{|x| p x}
#>>[1, 4, 9]
[1,2,3].sum("", &:to_s).tap{|x| p x}
#>>"123"
```

You may come to think that you rather want a method that would do tap{|x| p x}, but I think that was already proposed various times, and proposing that here would be a duplicate.

#2 - 04/15/2022 09:37 AM - mame (Yusuke Endoh)

- Related to Feature #14609: Let `Kernel#p` without an argument print the receiver added

#3 - 04/15/2022 11:09 AM - aDAVISk (Akito D. Kawamura)

That's exactly I meant. Thanks a lot, I call this topic closed. Sorry for the bad traffic.