

Ruby trunk - Bug #15646

method_defined? inside method_added behaves differently for `initialize`

03/08/2019 08:56 PM - alanwu (Alan Wu)

Status: Open	
Priority: Normal	
Assignee:	
Target version:	
ruby -v: ruby 2.7.0dev (2019-03-08 trunk 67194) [x86_64-darwin18]	Backport: 2.4: UNKNOWN, 2.5: UNKNOWN, 2.6: UNKNOWN
Description	
Repro:	
<pre>class MethodAddedSpecialCase def self.method_added(name) puts "#{name} added, method_defined?: #{method_defined?(name)}" end def hello end def initialize end def world end end</pre>	
Output:	
<pre>hello added, method_defined?: true initialize added, method_defined?: false world added, method_defined?: true</pre>	

History

#1 - 03/08/2019 08:57 PM - alanwu (Alan Wu)

- Description updated

#2 - 03/08/2019 09:05 PM - alanwu (Alan Wu)

Ah, I didn't know initialize is always private:

```
irb(main):001:0> class A
irb(main):002:1>   def initialize
irb(main):003:2>   end
irb(main):004:1> end
=> :initialize
irb(main):005:0> A.new.initialize
Traceback (most recent call last):
  4: from /opt/rubies/2.6.1/bin/irb:23:in `<main>'
  3: from /opt/rubies/2.6.1/bin/irb:23:in `load'
  2: from /opt/rubies/2.6.1/lib/ruby/gems/2.6.0/gems/irb-1.0.0/exe/irb:11:in `<top (required)>'
  1: from (irb):5
NoMethodError (private method `initialize' called for #<A:0x00007faedb0bf428>)
```

I think this fact should be documented at least.

#3 - 03/08/2019 11:45 PM - nobu (Nobuyoshi Nakada)

I think it is documented somewhere, but can't remember where it is now.

Anyway, these methods are always defined as private, unless singleton methods.
initialize, initialize_copy, initialize_clone, initialize_dup, respond_to_missing?

#4 - 03/10/2019 12:44 AM - alanwu (Alan Wu)

This behavior is from [Feature [#6539](#)]. Do you remember why these methods are made to be always private, instead of setting them to be private in Kernel only?

I don't think this rule make sense for BasicObject classes since they don't have #dup and #clone and #respond_to?:

```
class A < BasicObject
  public

  def initialize_dup
    end
end
```

```
A.new.initialize_dup # private method `initialize_dup' called
```

#5 - 03/10/2019 06:25 AM - nobu (Nobuyoshi Nakada)

initialize isn't involved in [Feature [#6539](#)].

It has been always private since more earlier days, 1.1 at least.