Ruby master - Feature #8515

don't allow irb to dump output for forever and ever

06/12/2013 01:41 AM - crankharder (Josh Sharpe)

Status:	Feedback	
Priority:	Normal	
Assignee:		
Target version:		

Description

You've done this before, I know you have. You were screwing around in IRB and you ran something stupid and IRB just started dumping output. LOTS of output. Output that would scroll by for an hour if you don't go kill it. Output going by so fast that it's useless. So much output that your lappy laptop starts burning your knees. (That much output!)

Then, to compound problems, Crtl-C doesn't work here. Maybe you're in some weird context, or on a low-mem server and everything is just... What, Bam, UNRESPONSIVE.

Ugh, the troubles! Now you gotta gotta bust out the 'kill -9' (if you can even get another terminal open) and lose any progress you may have been working on.... OH MAN!

Wouldn't it be nice if IRB limited its output, much the way psql/less do? That is, at some point if enough output has scrolled by, Stop, and let me have the option to continue or not.

That'd be cool.

<3

History

#1 - 06/12/2013 03:23 AM - Anonymous

On 06/11/2013 09:41 AM, crankharder (Josh Sharpe) wrote:

> Feature #8515: don't allow irb to dump output for forever and ever
> https://bugs.ruby-lang.org/issues/8515

Agreed, it would be nice to have LESS(1) functionality in irb.

Have you tried the pry gem? It does behave like less (including regex searches).

If you want to use irb, there is this snippet for your .irbrc (though it seems incompatible with wirble), which abbreviates long output:

```
# credit to Stian Haklev
class IRB::Context
attr_accessor :max_output_size
```

```
alias initialize_before_max_output_size initialize
def initialize(*args)
  initialize_before_max_output_size(*args)
  @max_output_size = IRB.conf[:MAX_OUTPUT_SIZE] || 500
end
```

```
end
```

```
class IRB::Irb
def output_value
text =
if @context.inspect?
sprintf @context.return_format, @context.last_value.inspect
else
sprintf @context.return_format, @context.last_value
end
max = @context.max_output_size
if text.size < max
puts text
```

```
else
puts text[0..max-1] + "..." + text[-2..-1]
end
end
end
```

#2 - 06/12/2013 09:53 AM - nobu (Nobuyoshi Nakada)

You can run irb with --noecho option, or set `IRB.conf[:ECHO] = false' in .irbrc.

#3 - 08/15/2014 02:34 AM - hsbt (Hiroshi SHIBATA)

- Status changed from Open to Feedback

#4 - 08/15/2014 09:20 PM - avit (Andrew Vit)

pry's pager is very cool, but it seems that it needs to build the whole string output before it gets echoed back to the screen using less. Maybe something better could be done for that within irb by "streaming" the output buffer?