

Advanced Ruby 3 Day

The Magical World of Ruby

By: Kerri Miller and Renée Hendrickson



Kerri Miller
Software Developer

✉ kerri@nird.us
📞 206.351.3106
🐦 @kerrizor

Contact me about:

- Teaching
- Ruby
- Poker

Visit us at nird.us

Renée Hendrickson
resident & Chief Adventure Officer

✉ renée@nird.us
📞 707.266.6473
🐦 @nirdllc
🌐 reneedv

Contact me about:

- RoR Projects
- Strategy
- BI

Visit us at nird.us

Welcome!

- ▶ Name
- ▶ Experience
- ▶ Class Goals





Objects in Ruby

A quick tour of Ruby objects, classes, and modules

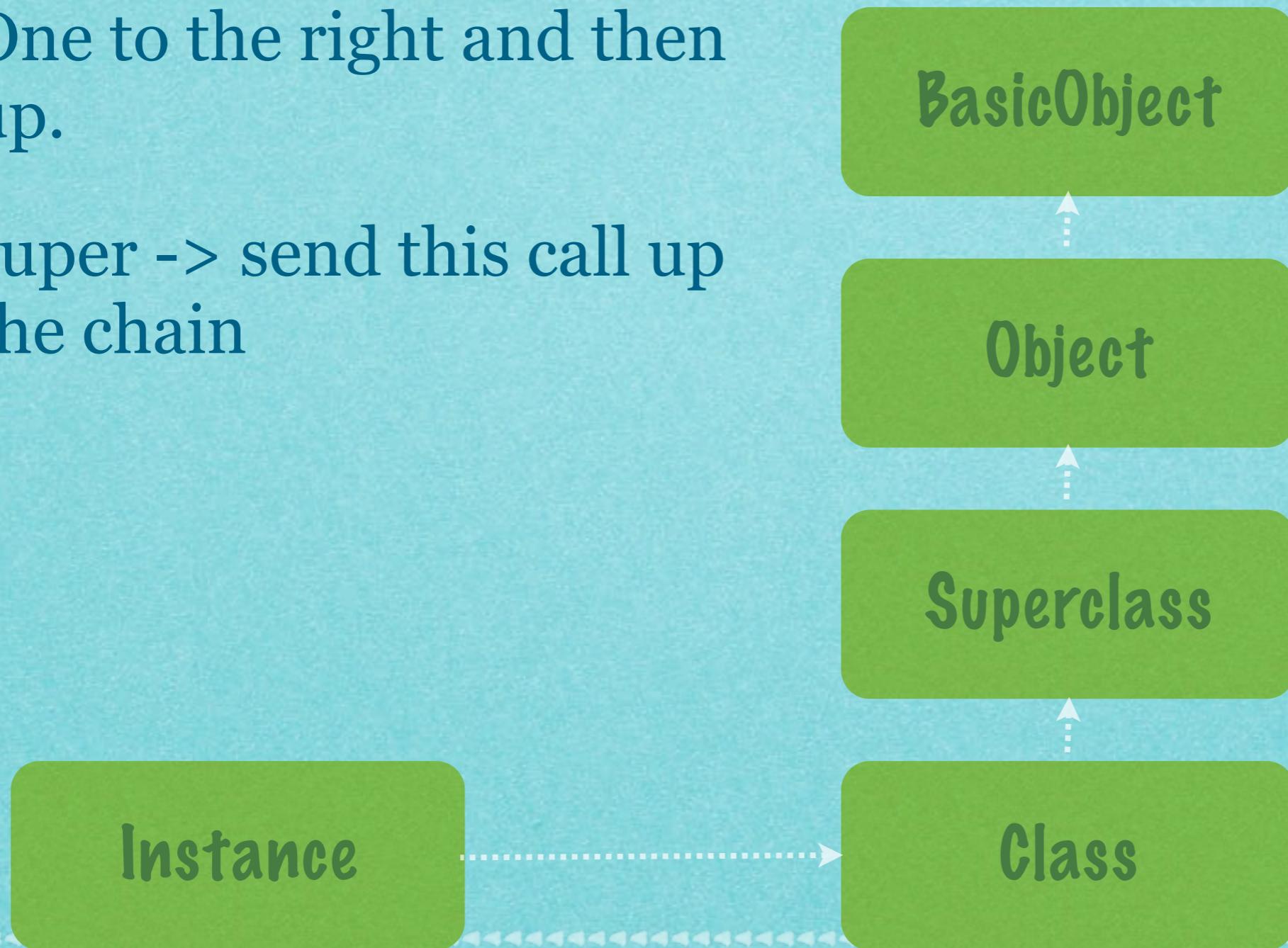
- Everything is an object
- metaclasses and metaobjects
- Classes aren't that special!

Ruby Methods

- ▶ Basic Object Class
- ▶ Object Class
- ▶ Kernel Module
- ▶ The Revenge of method_missing()
- ▶ <http://bit.ly/1voGdIF>

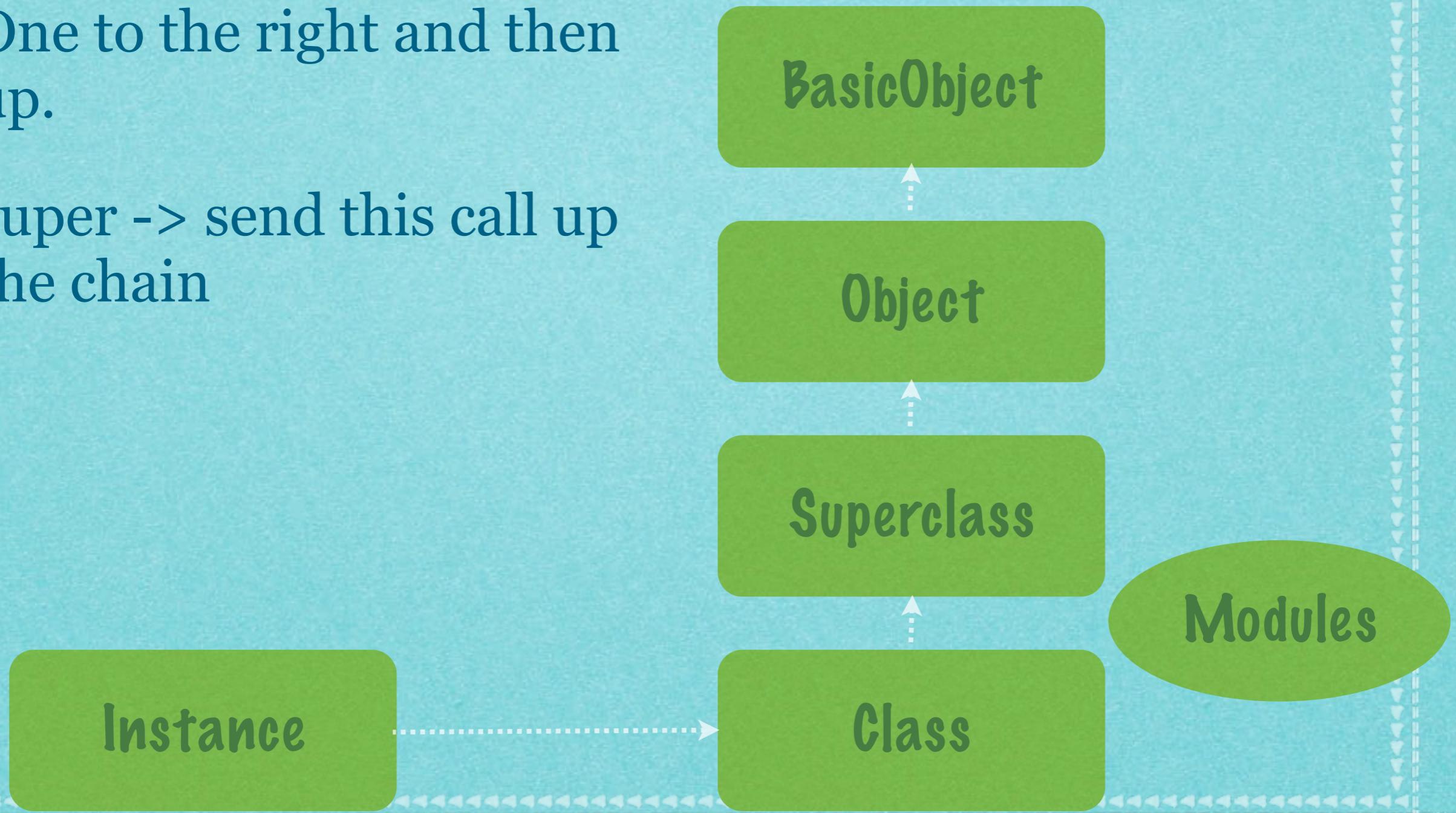
Ruby Call Chain

- ▶ One to the right and then up.
- ▶ super -> send this call up the chain



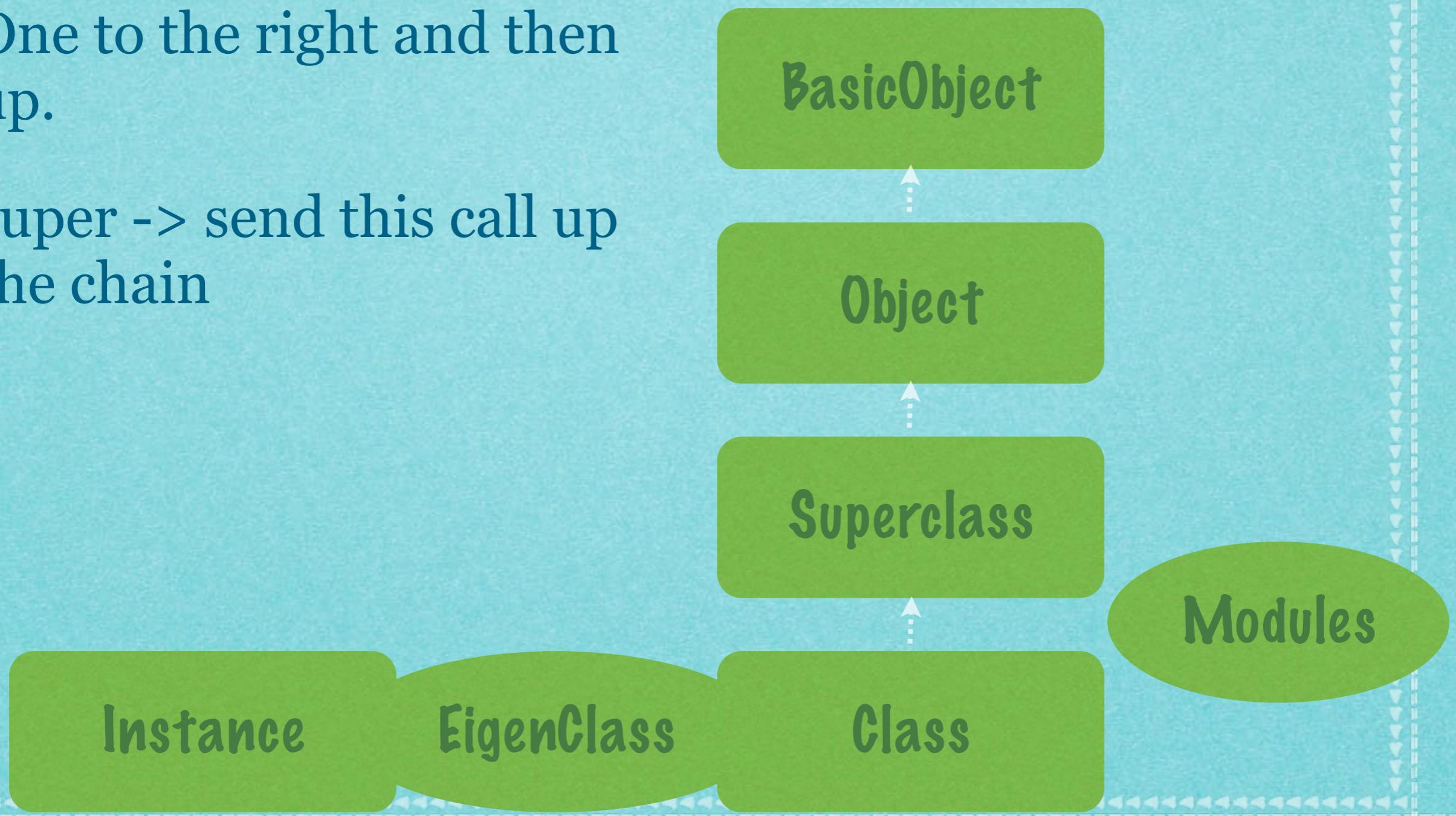
Ruby Call Chain

- ▶ One to the right and then up.
- ▶ super -> send this call up the chain



Ruby Call Chain

- ▶ One to the right and then up.
- ▶ super -> send this call up the chain



The Changing Face of Self

WHAT MATTERS MOST
IS HOW YOU SEE YOURSELF.



method call chain

call chain self experiment

The object model

- from instance all the way up to BasicObject
- The changing face of `self` http://ruby-metaprogramming.rubylearning.com/html/when_does_self_change.html
- class vs instance methods
- the scope of `self`
- method call chains
- How methods are called dynamically
 - the magic of MethodMissing
- delegation patterns in Ruby
 - MethodMissing
 - StdLib: Forwardable

search ID: mfln479



Exercise

*Delegate to a TaxRateCalculator Object
In 3 Parts!*



Inheritance

Privacy

9

```
## Methods
- structure
- method dispatch
- public, protected, private
- breaking the rules with .send
- Inheritance
```

Exercise!!

- ▶ Build out the Zombie class
 - ▶ How many monsters are zombies?
 - ▶ How many are fast zombies and slow zombies?
- ▶ Build a WereWolf class -
 - ▶ I should be able to call a method to see how many humans any werewolf has converted.
 - ▶ What are the names of humans bitten by werewolves?

Exercise:

- Monster Academy
- build a small menagerie of creatures that inherit from 2 or 3 base classes



Modules

Shared Dry Behavior

11

Modules & Mixins

- What are modules?
 - Modules as namespaces and reusable libraries
 - Require VS Load VS Include VS Extend
 - Mixing methods into classes using included
 - Mixins: Ruby's path to Multiple Inheritance

EYE SPY OVER 2,000 COMBINATIONS!

HOW MANY WILL YOU TRY?

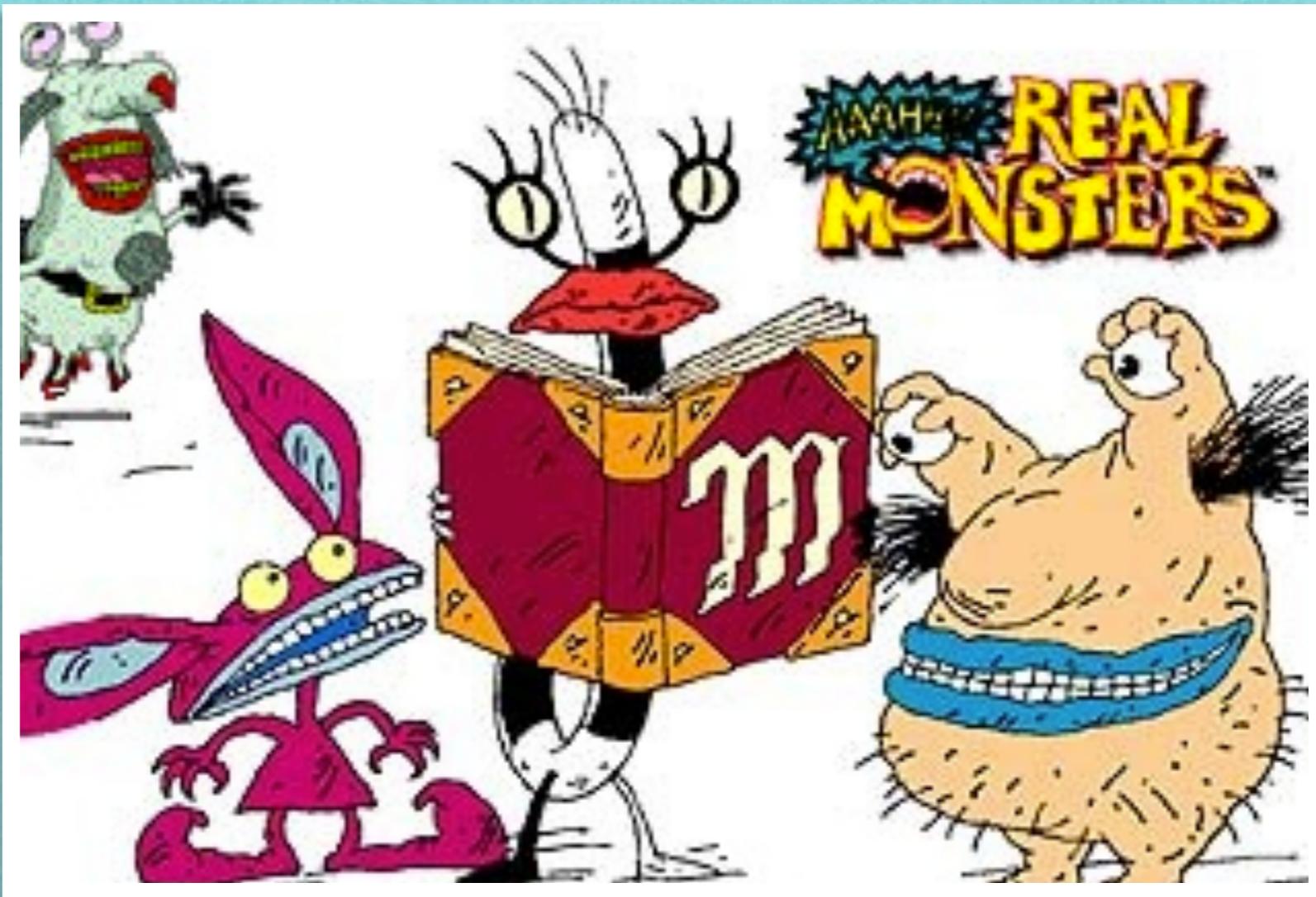


Exercise!!

Refactor Monsters into behavior modules

Exercise:

- extract common functionality from Monster Academy creatures into modules



Enumerable Demo

monsters.rb

standard enumerable demo w/ questions

Enumerable Basics

- ▶ load 'monsters.rb'
- ▶ \$monsters

- ▶ How many monsters are nocturnal?
- ▶ What are the names of the monsters that are nocturnal?
- ▶ How many legs do all our monsters have?
- ▶ What are the 2 most common dangers and vulnerabilities of our monsters?

Ruby.to_SQL

- ▶ \$monsters.find_all{|i| i[:name] == "Vampire"}
 - ▶ Select * from monsters where name = "Vampire";
- ▶ \$monsters.map{|i| i[:name]}
 - ▶ Select name from monsters;
- ▶ \$monsters.find_all{|i| i[:legs] == 2}.map{|i| i[:name]}
 - ▶ Select name from monsters where legs = 2;



Enumerable

Rebuild Enumerable!

Exercise:

- re-implement Enumerable



Pairing and Refactoring

Find a Partner!



Duck Punching

with a side of Monkey Brains...

18

Runtime class extensions

- dynamically adding methods to a class
- "Monkey Patching" to overload a method's functionality
- creating methods on the fly using `define_method`

Runtime evaluation of code

- `eval`, `instance_eval`, and `class_eval`