



RUBY ON RAILS

5 DAY BootCAMP





RUBY 2 DAY



THE CORE LANGUAGE

HELLO!

WELCOME!

Renée De Voursney

✉ reneee@nird.us

🐦 [@gigglegirl4e](https://twitter.com/gigglegirl4e)

𝑺𝒐𝒖𝒍 [reneedv](https://www.reneedv.com)

nird.us



NIRD

INTRODUCTIONS

- ❖ Name
- ❖ Coding Experience
- ❖ What would you like to learn?



CURRICULUM

RUBY 2 DAY

- ❖ Ruby
- ❖ Core-Syntax
- ❖ Standard Libraries
- ❖ Common Patterns and Best Practices

RAILS 3 DAY

- ❖ RoR Toolset
- ❖ Rails Walk-Through
- ❖ Gems and Deployment
- ❖ Full Apps and Ruby Inside Rails

SCHEDULE

Times	Day 1	Day 2
9am - 10:30	Intro and Rspec	Regular Expressions
Break		
10:45 - 12	Strings, Variables, Classes, Objects	Files and Rake
Lunch		
1pm - 2:30	Numbers, Ranges, Symbols, Conditions	Blocks
Break		
2:45 - 4:30	Modules, Inheritance, Collections	MetaProgramming and Method Missing

LEARNING GOALS

- ❖ Basic Ruby
- ❖ Common Patterns and Practices
- ❖ Where to go for help

LEARNING PATH

- ❖ Experimentation
- ❖ Test First Teaching
- ❖ Ask Questions
- ❖ Resources for finding answers

STRUCTURE

- ❖ Explanation (me talking)
- ❖ Demonstration (me typing)
- ❖ Experimentation (in-class coding)
- ❖ Questions



RUBY VS. RAILS

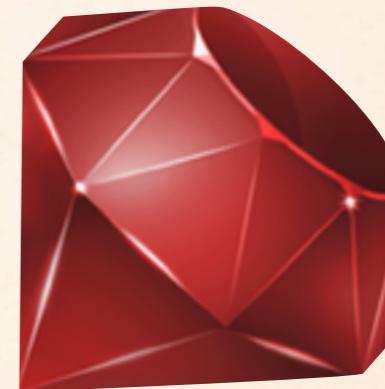
- ❖ Ruby is the Language -
 - ❖ developed 1993
 - ❖ Yukihiko Matsumoto “Matz”

- ❖ Rails is the Framework -
 - ❖ developed 2004
 - ❖ David Heinemeier Hansson “DHH”



THE RUBY WAY

- ❖ Written in C
- ❖ Influenced by SmallTalk & Pearl
- ❖ Ruby is for People
- ❖ MINSWAN
- ❖ OSS (Open Source Software)
- ❖ RubyGems

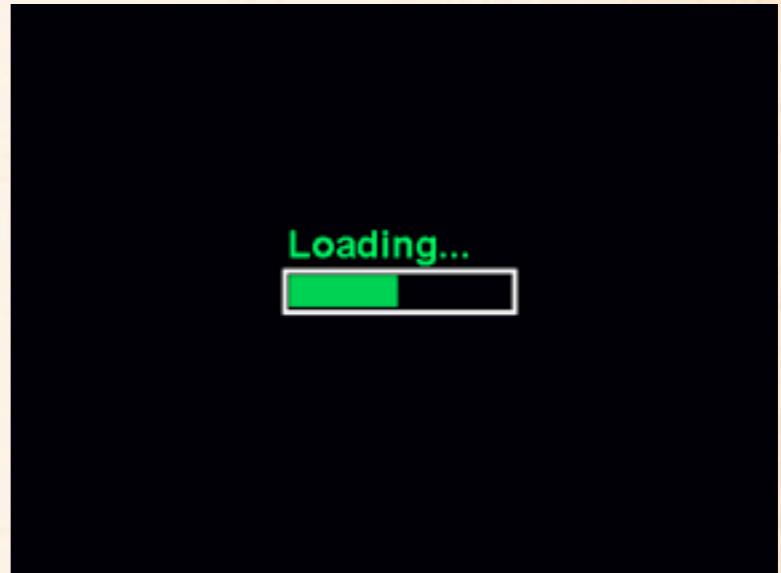


THE RUBY LANGUAGE

- ❖ Dynamically Typed
- ❖ Interpreted (Standard Lib)
- ❖ Runtime Modifications
- ❖ Object Oriented
- ❖ Blocks and Lambdas
- ❖ Regular Expressions

INSTALL CHECK

- ❖ Command Prompt
- ❖ Git
- ❖ Ruby
- ❖ Rspec
- ❖ Editor (Syntax Highlighting)



COMMAND LINE

- ❖ Command Prompt (CLI)
 - ❖ Mac: Terminal, iTerm, etc...
 - ❖ Unix: bash, zsh, csh, etc....
 - ❖ Windows: Command Prompt w/ Ruby on Rails

EDITOR

- ❖ Sublime Text 2 / TextMate
- ❖ RubyMine
- ❖ KomodoEdit
- ❖ Eclipse
- ❖ Vi / Vim / Emacs

SUBLIME TEXT 2

❖ Command line alias

- ❖ Win: doskey subl="C:\Program Files\Sublime Text 2\sublime_text.exe" \$*
- ❖ Mac: ln -s "/Applications/Sublime Text 2.app/Contents/SharedSupport/bin/subl" ~/bin/subl
- ❖ Linux: sudo ln -s /opt/SublimeText2/sublime_text /usr/bin/subl

❖ How to Open a folder

GIT

- ❖ which git
 - ❖ ex: /usr/local/bin/git
- ❖ git --version
 - ❖ ex: git version 1.7.10.2

RUBY

- ❖ Ruby -v
 - ❖ 1.9.2, 1.9.3, 2.0.0 (1.8.7)
- ❖ gem list
- ❖ Bundler, Rake, rubygems, etc...

RVM

RUBY VERSION MANAGER

- ❖ rvm list
 - ❖ rvm use
 - ❖ rvm gemset list
-
- ❖ rvm.io



RUBY VERSIONS

- ❖ MRI, YARV, Standard Library, C-Ruby
- ❖ JRuby - Java
- ❖ Rubinius - Pure Ruby
- ❖ MagLev - SmallTalk
- ❖ MacRuby - Objective-C
- ❖ IronRuby - C# (for .Net)

RUBY INTERPRETER

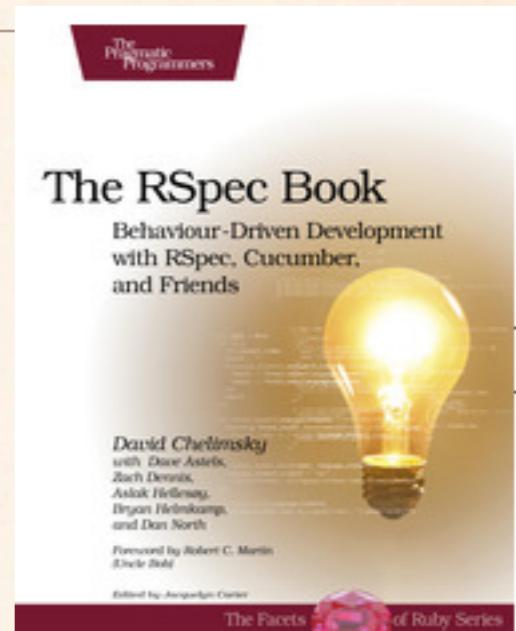
- ❖ irb
- ❖ ex:
- ❖ I.9.3-p194 :001 > I+2
- ❖ => 3
- ❖ I.9.3-p194 :002 > exit

RSPEC

- ❖ gem install rspec
 - ❖ ex: Fetching: rspec-core-2.11.1.gem (100%)
- ❖ gem list
 - ❖ ex:
 - ❖ diff-lcs (1.1.3), rspec (2.11.0), rspec-core (2.11.1), rspec-expectations (2.11.3), rspec-mocks (2.11.3)
- ❖ rspec -v
 - ❖ ex:
 - ❖ 2.11.1

RSPEC INTRO

- ❖ Code that Tests code!
- ❖ Test Driven Development
- ❖ Red-Green-Refactor
- ❖ Test First Teaching
- ❖ Ruby Domain Specific Language (DSL) for testing



RSPEC SPECIFICS

- ❖ Describe (What) / Context (When)
- ❖ It (Does)
- ❖ Should
- ❖ Matchers
 - ❖ <http://rubydoc.info/gems/rspec-expectations/2.4.0/RSpec/Matchers>

RSPEC DEMO!

- ❖ make a new rspec file
- ❖ run rspec
- ❖ walk through rspec_spec.rb

RSPEC FORMATTING

- ◊ alias rspec="rspec --color --format nested"

IN-CLASS EXERCISE

- ❖ Run `exercises/rspec_spec.rb`
- ❖ Fix the broken in-class test
- ❖ Add the 3 pending in-class tests (don't forget the `do/end` and `should!`)

BREAK!

❖ 10-15min

RSPEC REVIEW

- ❖ Code that Tests Code!!
- ❖ Describe a feature, set an expectation.
- ❖ **Red-Green-Refactor**

RUBY

- ❖ Everything is an Object!
- ❖ Type Doesn't matter! (`respond_to?`)
- ❖ Speaking Ruby : Interpreter Needed!
- ❖ MRI, YARV, JRuby

RUBY GEMS

- ❖ Packaged Ruby Code
- ❖ Open Source
- ❖ iPhone user: “There’s an app for that!”
- ❖ Ruby programer: “There’s a **gem** for that!”

IRB

- ❖ Interactive Ruby!
- ❖ Our Ruby Interpreter!
- ❖ We speak ruby, it translates!
- ❖ Understand Ruby code, evaluates it, gives us the result
(inspect)

RUBY

- ❖ Everything evaluates to something
- ❖ `nil` is NULL or nothing or Not Set
- ❖ Every Object `respond_to?` `inspect`
 - ❖ `=> true`

OBJECTS

- ❖ Ruby: everything is an object
 - ❖ Everything is an instance of a class
- ❖ Ruby knows about certain classes of things: Standard Library

STRING

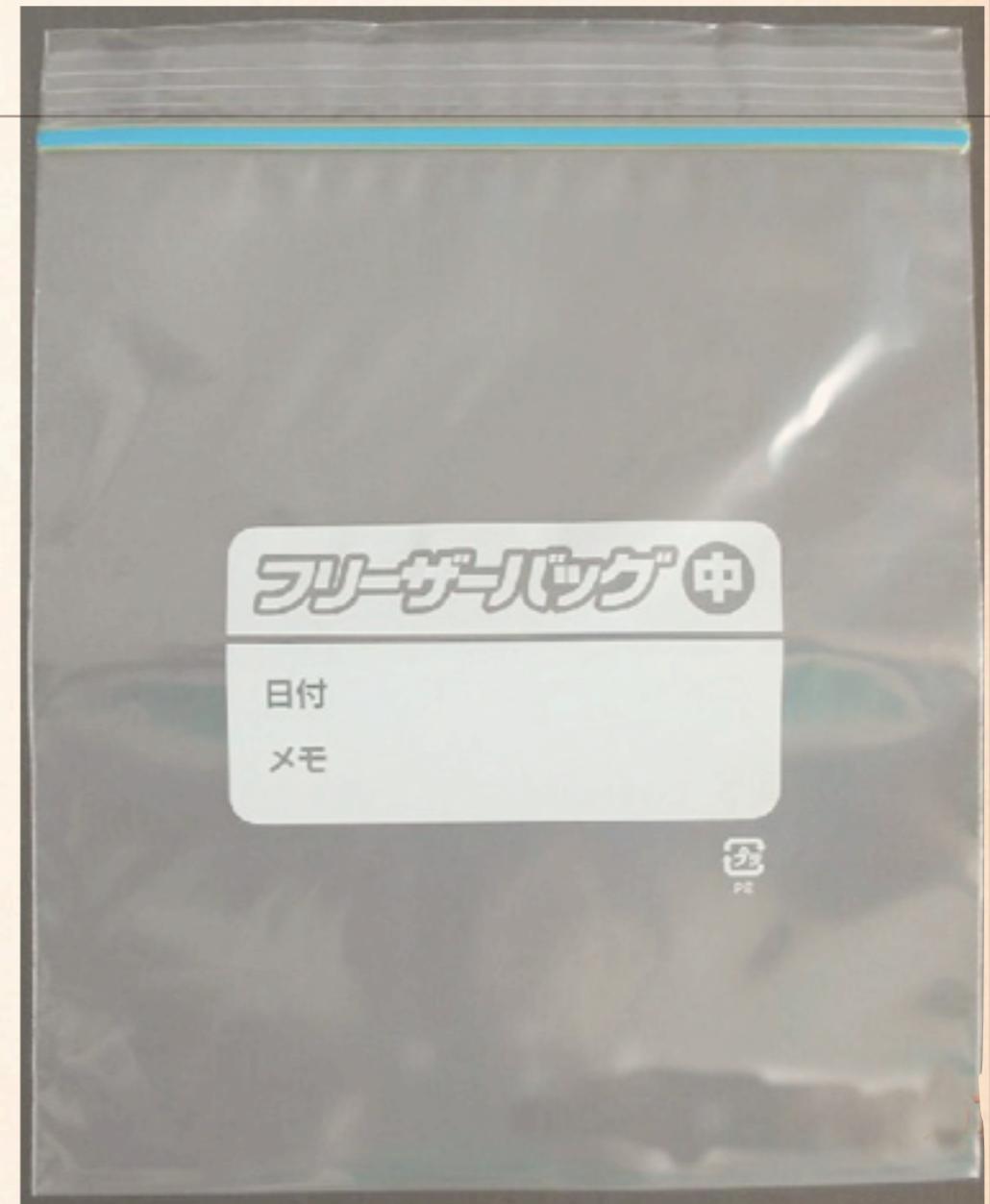
- ❖ String class : part of the standard library
- ❖ How Ruby understands text
- ❖ Instances
- ❖ `String.new` or `"` or `""` or `Q%{}` or `q%{}`

STRING CLASS

- ❖ <http://www.ruby-doc.org/core-1.9.3/String.html>
- ❖ Methods : Messages we can send, what an instance will respond to
- ❖ Chomp, *, +, include?, upcase

VARIABLES

- ❖ Named location
- ❖ Points to something
- ❖ Contents Can Change



STRING INTERPOLATION

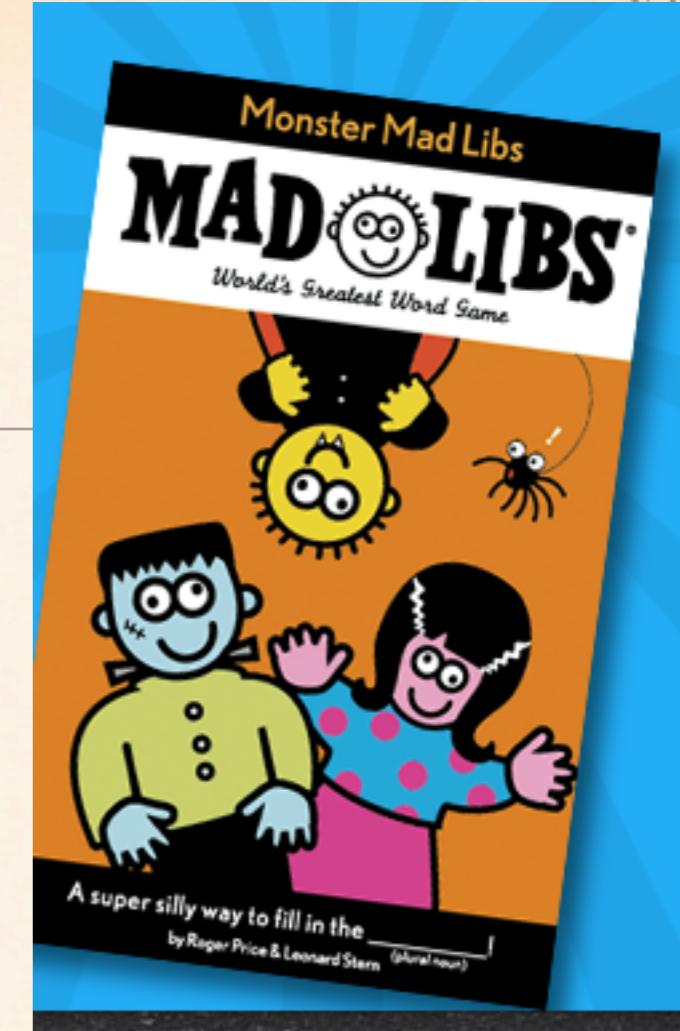
- ❖ my_name = “Renée”
- ❖ “Hello #{my_name}”

PUTS / GETS

- ❖ put string
- ❖ get string
- ❖ puts “Hello World”
- ❖ user_input = gets
- ❖ user_input = gets.chomp

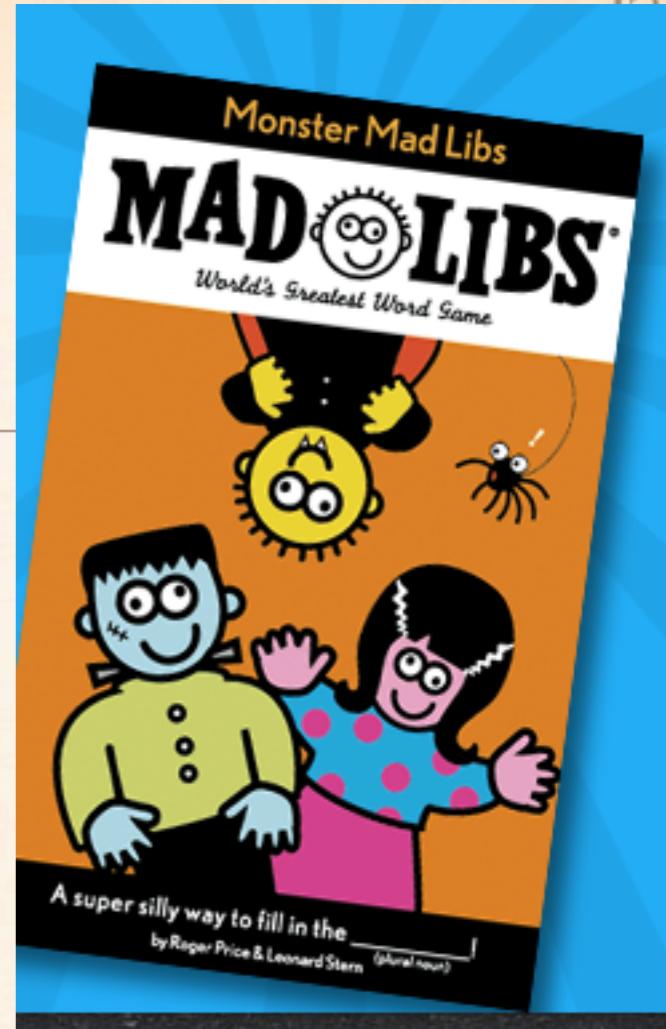
MAD-LIBS!

- ❖ noun = gets.chomp
- ❖ adjective = gets.chomp
- ❖ verb_past_tense = gets.chomp
- ❖ “The #{adjective} #{noun} #{verb_past_tense} past the graveyard”



RUBY SCRIPTS

- ❖ mad_libs.rb
- ❖ ruby mad_libs.rb



MAKING OBJECTS

- ❖ Tell Ruby About Something New!
- ❖ A new class of stuff!

WHAT IS A BOOK

- ❖ All Books have....??
- ❖ TDD

DEFINING NEW CLASSES

- ❖ Initialize
- ❖ attributes / readers and writers
- ❖ Scope
 - ❖ Instance variables
 - ❖ Instance methods

EXERCISE!

- ❖ Simon Says Spec

LUNCH

❖ I Hour

SIMON SAYS REVIEW

❖ rspec simon_says_spec.rb

NUMBERS

- ❖ FixNum
- ❖ I is always I! (unlike strings)
- ❖ Floats (BigDecimal)
- ❖ 4.2
- ❖ 3.141516



RANGES

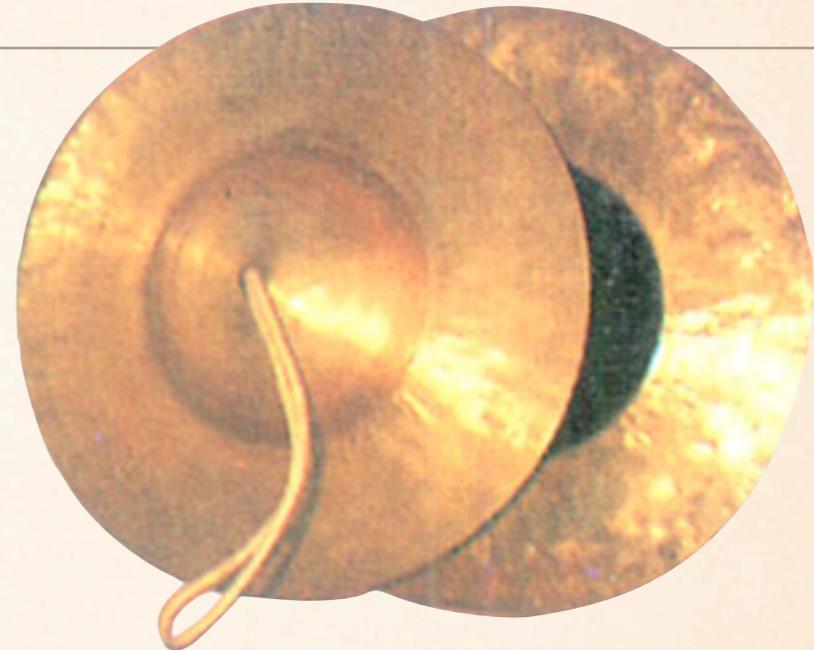
❖ I..5

❖ I...5

❖ a..z



SYMBOLS



- ❖ :symbols_are_fun
- ❖ A static name
- ❖ Stored like numbers, named like strings!

CONTROL

- ❖ Controlling the flow of your application
- ❖ BEWARE: if-else-death, Switch-Statement-Smell (<http://c2.com/cgi/wiki?SwitchStatementsSmell>)

CONDITIONALS

- ❖ if, else
- ❖ unless, else
- ❖ and && (BEWARE: precedence!!)
- ❖ or || (BEWARE: precedence!!)
- ❖ ||=
- ❖ conditional ? then : else

LOOP-DE-LOOP!

- ❖ while
- ❖ until
- ❖ times

EXERCISE

- ❖ Calculator Spec

BREAK

❖ 10 - 15 min

COLLECTIONS



ARRAY

- ❖ Ordered List
- ❖ Referenced by Order(index)



HASH

- ❖ Key, Value Store
- ❖ Referenced by Name(key)

star selections

2005 Lincourt Santa Barbara County (\$25)
Hints of allspice and sweet oak inform the bright red cherry flavors of this graceful red, made with fruit from vineyards in California's Santa Rita Hills.

2005 Raptor Ridge Willamette Valley (\$30)
Most of the grapes in this elegant, multi-vineyard blend come from the Stony Mountain Vineyard, perched on a hillside in Oregon's Yamhill County.

2004 Woodenhead Russian River Valley (\$36)
Co-owner Nikolai Stez (former assistant winemaker at cult producer Williams Selyem) produces stunning Pinot Noirs, like this hauntingly floral bottling.

2005 Scott Paul La Paulée (\$40)
Scott Paul Wright leads two lives, importing high-quality Burgundies and making his own eponymous wines, like this lightly herbal, raspberry-rich cuvée.

ARRAY AND HASH DEMO

- ❖ numbers = [5,4,7,1,3,2,6]
- ❖ numbers[0]
- ❖ numbers.size
- ❖ more_numbers = {"one" => 1, "two" => 2, 3 => "three"}
- ❖ more_numbers["one"]

EXERCISE

- ❖ Calculator Advanced Spec

MODULES

- ❖ Sharing Behavior
- ❖ Namespacing
- ❖ DRY Code
- ❖ Composition



ENUMERABLE

- ❖ Module for Collection Specific Actions
- ❖ Search
- ❖ Sort
- ❖ Iterate (Enumerate)



INJECT

```
total = 0
```

```
[1,2,3].each do |i|
```

```
    total += i
```

```
[1,2,3].inject(:+)
```

```
end
```

```
=> 6
```

```
total
```

```
=> 6
```

DEMO ENUMERABLE

- ❖ 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?

MODULE EXERCISE

- ❖ named_thing.rb
- ❖ Vampire's have names
- ❖ Human's have names

INHERITANCE

- ❖ Parent-Child
- ❖ SubTypes



INHERITANCE EXERCISE

- ❖ monster.rb
- ❖ Monster
 - ❖ Vampire (vampire.rb)
 - ❖ Zombie
 - ❖ Werewolf