



Ruby Workshop

Scripting and Domain Specific Languages (DSL)

@shritesh
cybereagles.club



Ruby

- Beautiful and Artsy
- Programmer Happiness
- Object Oriented
- Flexible yet powerful
- Easy to get started
- Widely Used
- Probably has a gem for everything

Yukihiro Matsumoto (Matz)

MINASWAN:
Matz is nice and so we are
nice



Installation

- macOS: You're Set
- Windows: <http://rubyinstaller.org/downloads/>
- Linux: Use your package manager
- Advanced Users: <http://rbenv.org> or <https://rvm.io>
- Online: <http://repl.it> also <http://tryruby.org>

REPL \$ irb

- Read
- Eval
- Print
- Loop
- You've got a calculator

```
lang = 'Ruby'  
"##{lang} is Bae"
```

Make string processing great again

```
def add(a, b)  
    a + b  
end
```

```
add 1, 2
```

Look Ma, No Return, No Parens

```
5.times do  
  puts "PHP SUX"  
end
```

So you thought Java was Object Oriented, Eh?
Code block is your new best friend


```
(5..10).each do |n|  
  print "#{n} is Even" if n.even?  
end
```

Ruby is very expressive

```
class String
  def palindrome?
    reverse == self
  end
end
```

“hi”.palindrome?

“tacocat”.palindrome?

Monkey Patching lets you dynamically modify things

Scripting

Who likes writing bash scripts?

Core, StdLib and Gems

- Batteries Included Core and Standard Library
 - <http://ruby-doc.org/>
- There is a gem for everything
 - <http://rubygems.org>
 - <http://awesome-ruby.com>
 - <https://www.ruby-toolbox.com>

`require 'digest/sha2'`

`Digest::SHA256.hexdigest "Hi"`

There's tons of stuff in the base installation

```
require 'chunky_png'
png = ChunkyPNG::Image.new 32,32
32.times do |x|
  32.times do |y|
    color = ['red', 'green', 'blue'].sample
    png[x,y] = ChunkyPNG::Color(color)
  end
end
png.save 'random.png'
```

There's a gem for that

Let's Write A _____

Domain Specific Language

“a programming language of limited expressiveness
focused on a particular domain”

HTTP -> Sinatra

```
require 'sinatra'
```

```
get '/' do  
  'Hello World'  
end
```

```
get '/:name' do  
  "Hi #{params['name']}"  
end
```

Music -> Sonic Pi

```
play :C  
sleep 0.5  
play :D  
sleep 0.5  
play :E
```

Lets Create a CPU