Hello Ruby

Ender Ahmet Yurt

www.enderahmetyurt.com

Ender Ahmet Yurt

Full Stack Developer @ sigortaci123.com

Ruby Türkiye Organizer

Yakut Podcast Speaker

Open Source Developer

Context

- Ruby
  - What’s it?
  - Features
  - Installation
- First steps
  - Irb/Pry
  - First program
- Basics
- Concepts
- More…
- Questions

Ruby

- Yukihiro Matsumoto (Matz)
- First version 1995
- Aim: Make developers happy
- Open source https://github.com/ruby/ruby
- Ruby 2.5.0-preview1 (10.10.2017)

Features

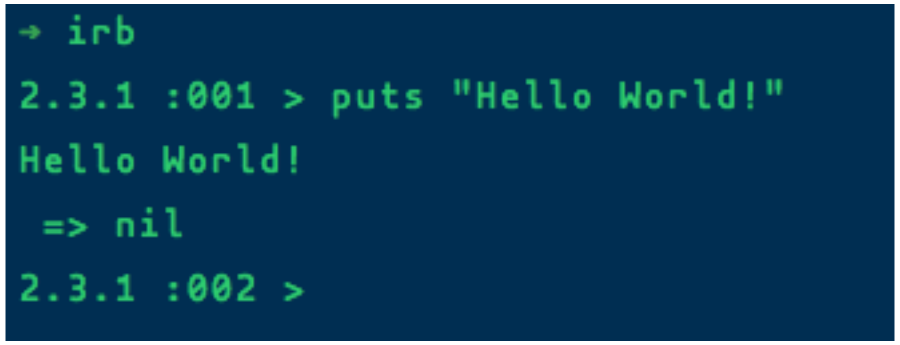
- Pure
- Everthing is an object
- Easy coding
- Readable like English
- More ways to do a job
- Dynamic language
- Scripting language
- Powerful communities
  - Global
  - Turkey rubyturkiye.org
- Ruby Gems rubygems.org

Installation

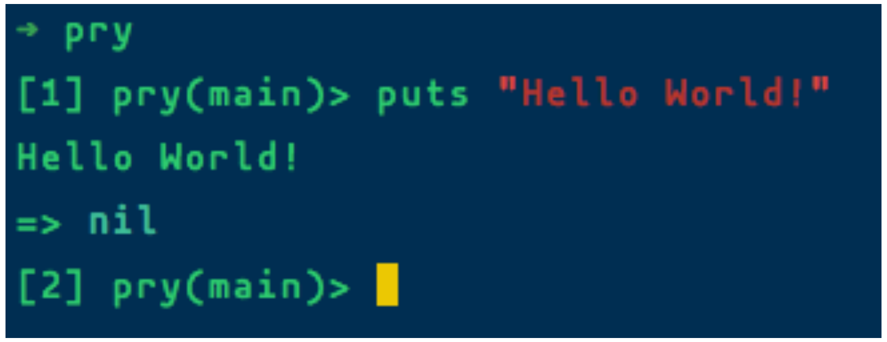
- OSX
  - Installed
  - RVM/RbEnv
- Linux
  - apt-get install ruby <version>
  - RVM/RbEnv
- Windows
  - rubyinstaller.org

Irb/Pry

- Irb (Interactive Ruby)
- Pry (Alternative Ruby Gem)
    - gem install pry

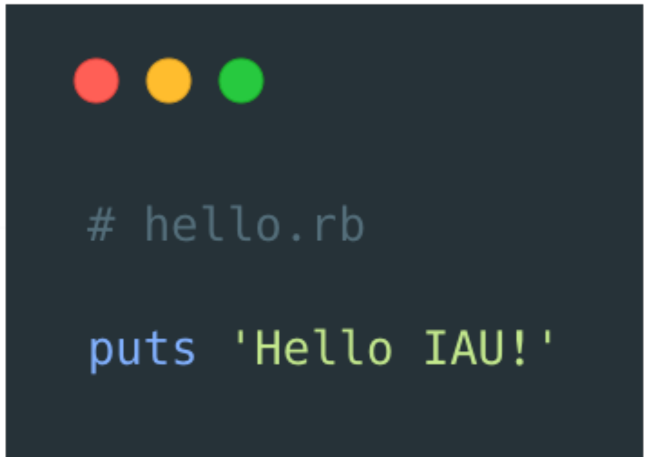


```
➜ irb
2.3.1 :001 > puts "Hello World!"
Hello World!
 => nil
2.3.1 :002 >
```



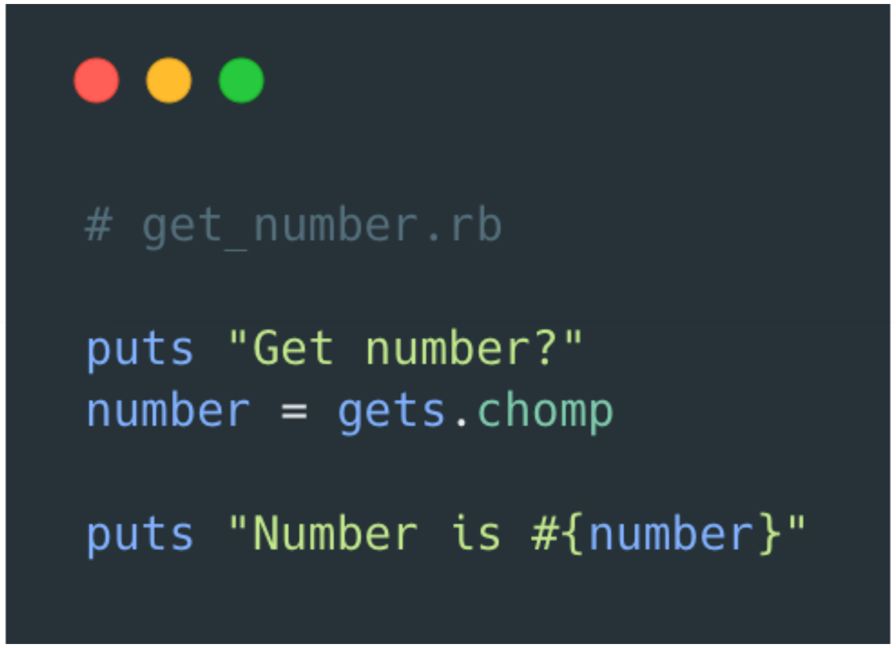
```
➜ pry
[1] pry(main)> puts "Hello World!"
Hello World!
=> nil
[2] pry(main)> █
```

First Program

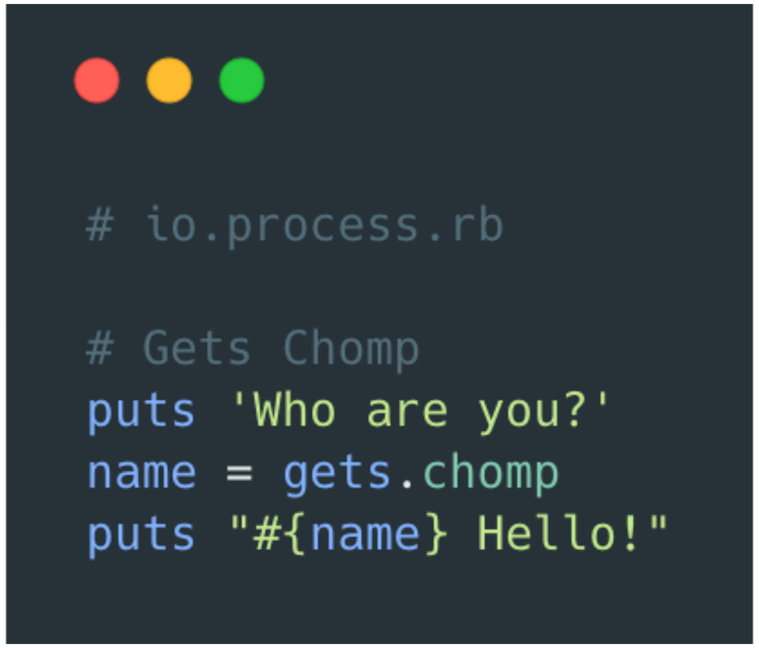


```ruby
# hello.rb

puts 'Hello IAU!'
```

Basic IO Processes



```ruby
# get_number.rb

puts "Get number?"
number = gets.chomp

puts "Number is #{number}"
```



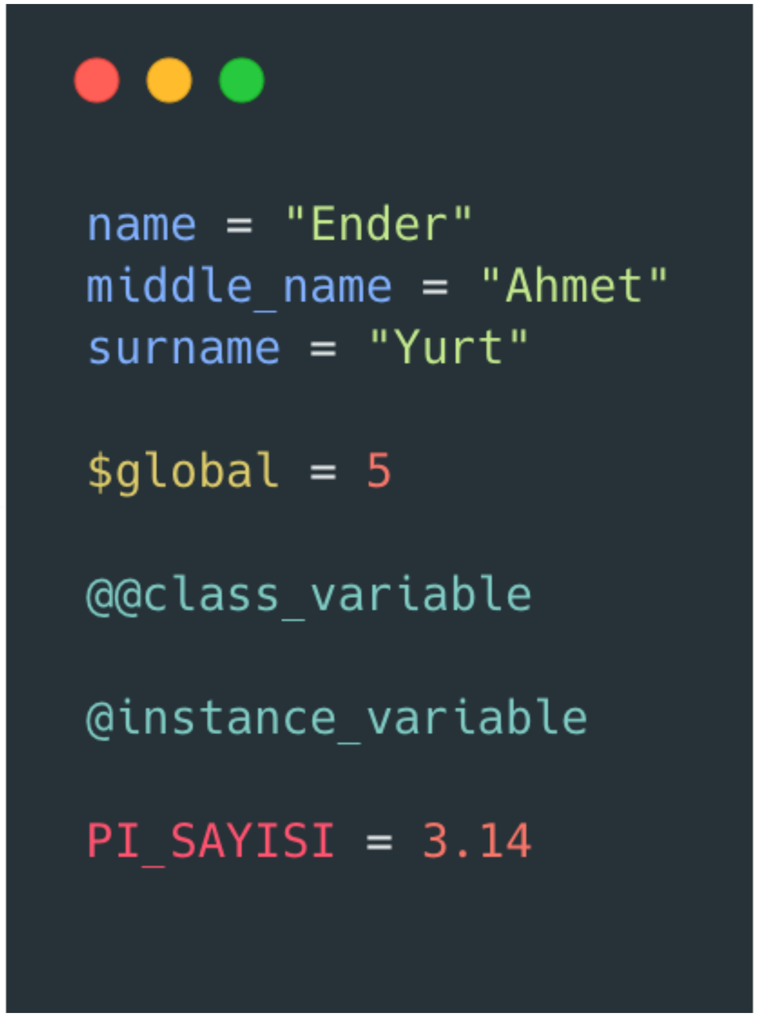
```ruby
# io.process.rb

# Gets Chomp
puts 'Who are you?'
name = gets.chomp
puts "#{name} Hello!"
```

Basics

- Variables
- Numbers
- Arrays
- Hashes
- Flows
- Loops
- Methods
- Classes
- OO
- Modules
- Proc/Lambda

Variables

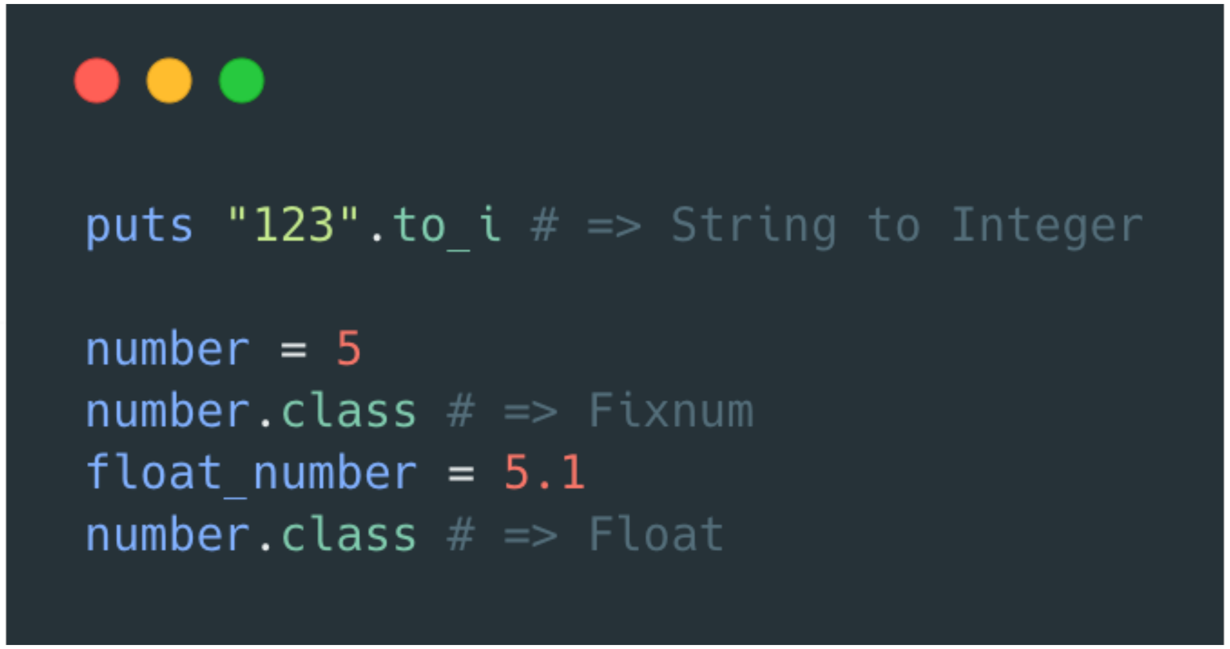
- Local variables
- Global variables
- Class variables
- Instance variables
- Constants
- Symbols



```
name = "Ender"
middle_name = "Ahmet"
surname = "Yurt"

$global = 5

@@class_variable

@instance_variable

PI_SAYISI = 3.14
```

Numbers

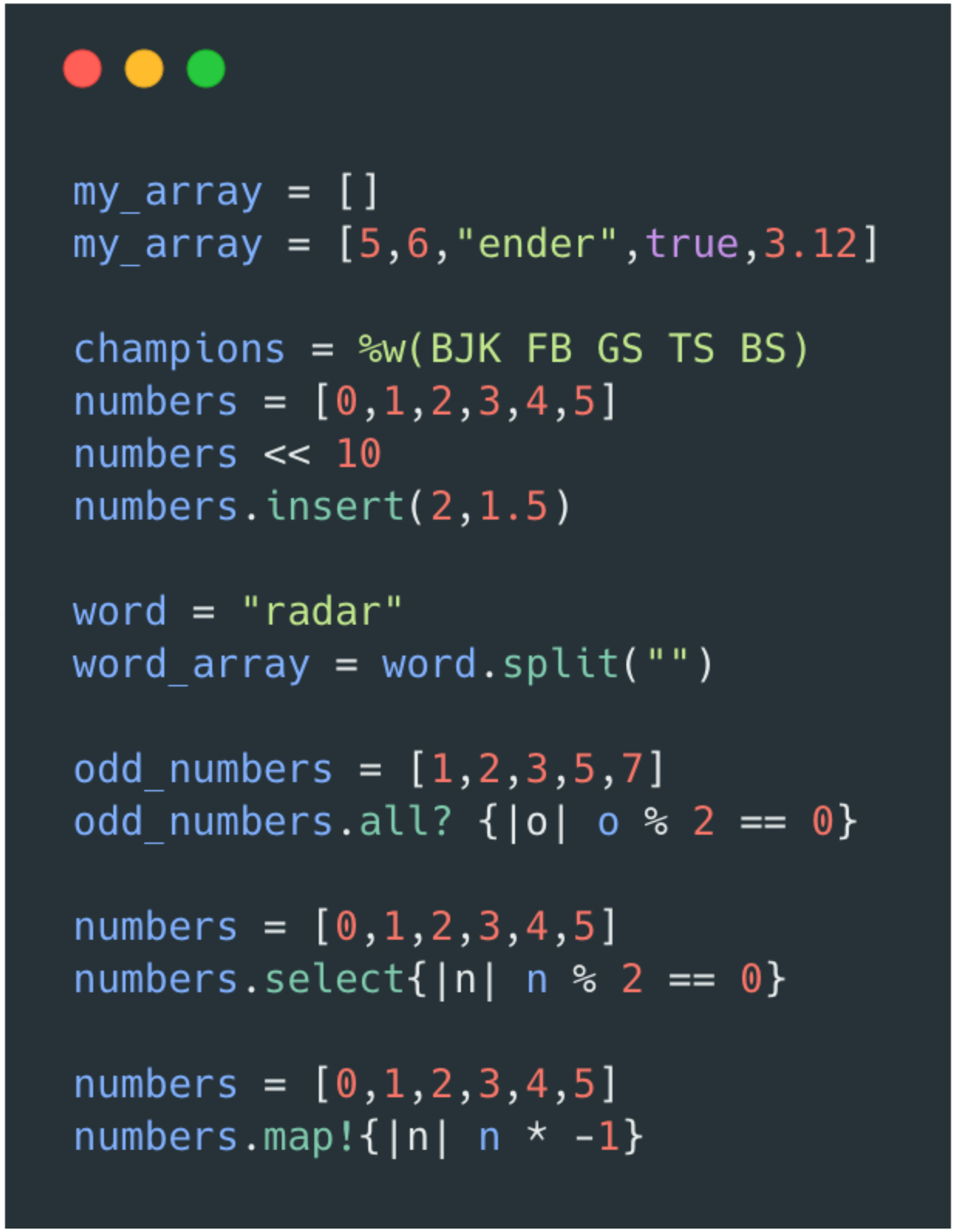
- Numeric
  - Integer
    - Fixnum
    - Bignum
  - Float
  - Complex
  - BigDecimal
  - Rational



```ruby
puts "123".to_i # => String to Integer

number = 5
number.class # => Fixnum
float_number = 5.1
number.class # => Float
```

Arrays

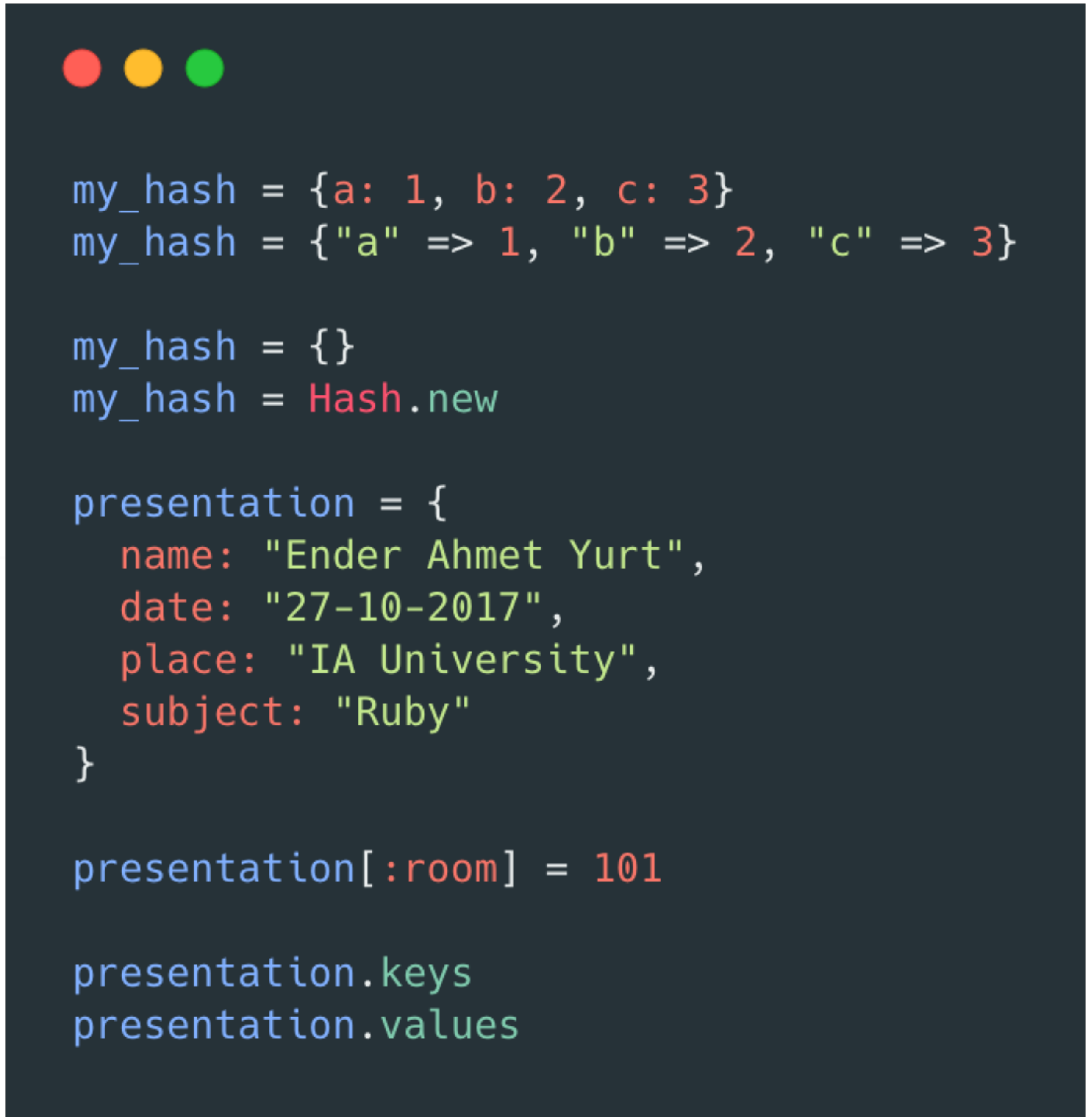
- Initialize
- Swap
- Join & Delete
- Enumerable methods



```
my_array = []
my_array = [5,6,"ender",true,3.12]

champions = %w(BJK FB GS TS BS)
numbers = [0,1,2,3,4,5]
numbers << 10
numbers.insert(2,1.5)

word = "radar"
word_array = word.split("")

odd_numbers = [1,2,3,5,7]
odd_numbers.all? {|o| o % 2 == 0}

numbers = [0,1,2,3,4,5]
numbers.select{|n| n % 2 == 0}

numbers = [0,1,2,3,4,5]
numbers.map!{|n| n * -1}
```

Hashes

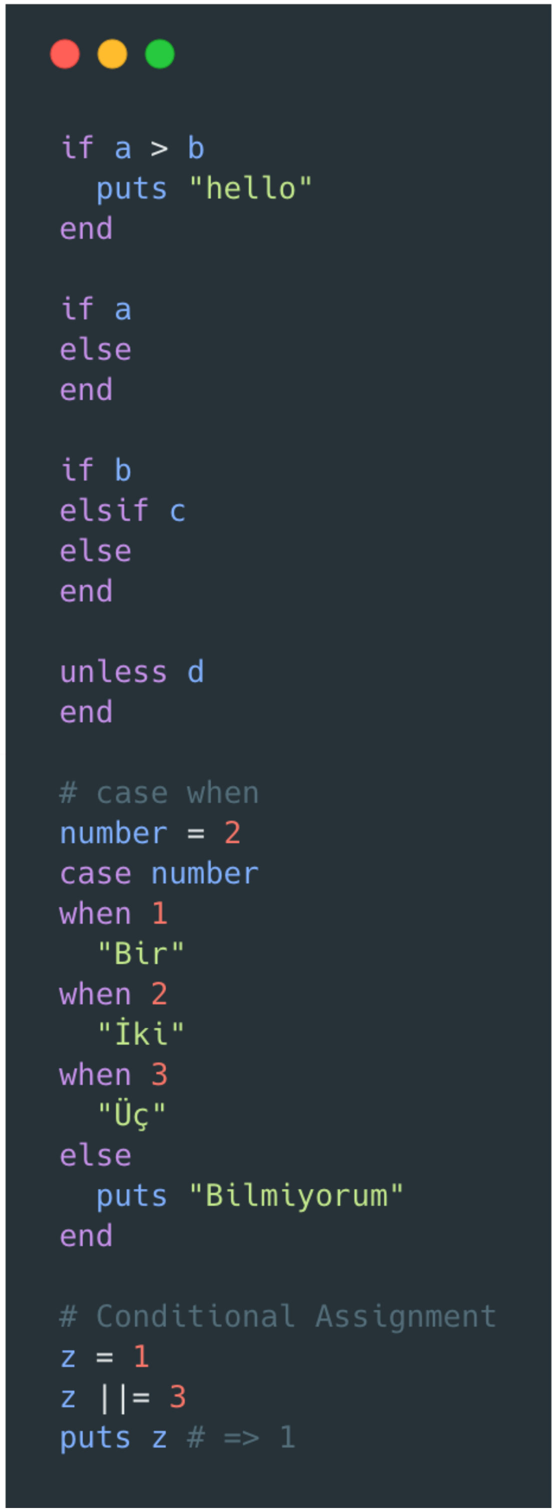
- What is it?
- Key-value pair
- Enumerable methods



```ruby
my_hash = {a: 1, b: 2, c: 3}
my_hash = {"a" => 1, "b" => 2, "c" => 3}

my_hash = {}
my_hash = Hash.new

presentation = {
  name: "Ender Ahmet Yurt",
  date: "27-10-2017",
  place: "IA University",
  subject: "Ruby"
}

presentation[:room] = 101

presentation.keys
presentation.values
```

Flows

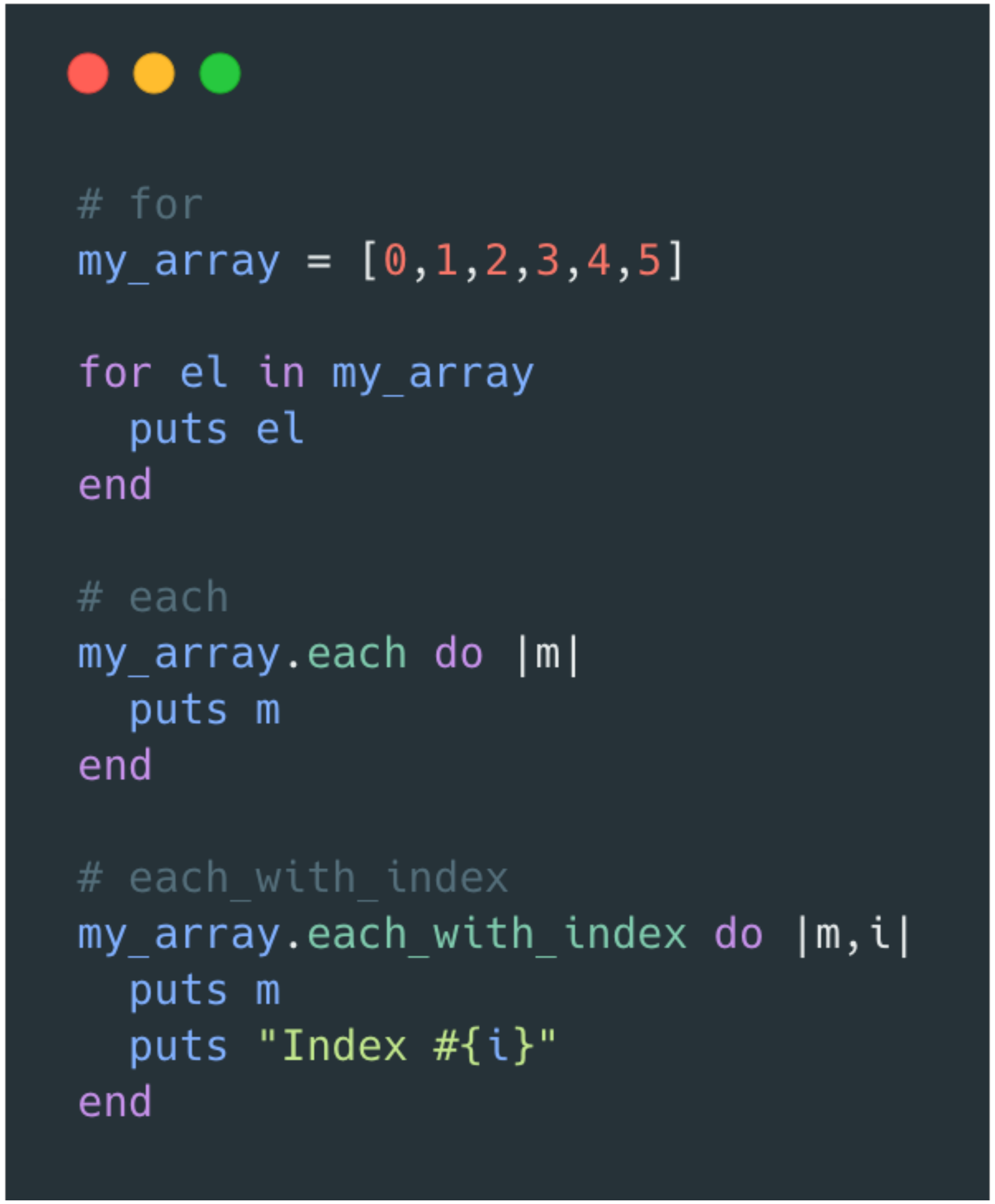
- True, False, Nil
- NOT (!)
- If-else-elsif-unless
- Case when
- Conditional Assignment



```
if a > b
  puts "hello"
end

if a
else
end

if b
elsif c
else
end

unless d
end

# case when
number = 2
case number
when 1
  "Bir"
when 2
  "İki"
when 3
  "Üç"
else
  puts "Bilmiyorum"
end

# Conditional Assignment
z = 1
z ||= 3
puts z # => 1
```

Loops

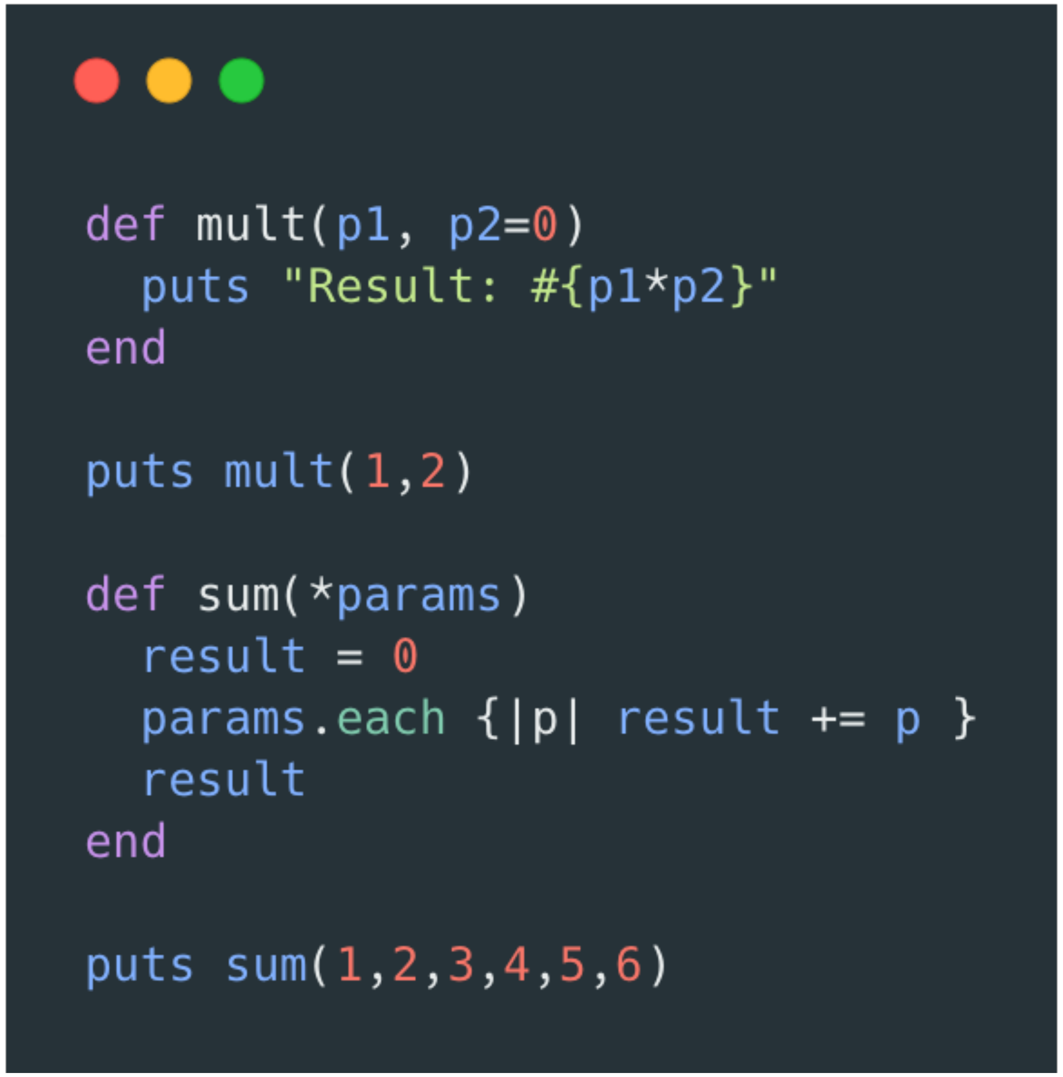
- for
- while
- until
- loop (next/break)
- each
- each_with_index



```ruby
# for
my_array = [0,1,2,3,4,5]

for el in my_array
  puts el
end

# each
my_array.each do |m|
  puts m
end

# each_with_index
my_array.each_with_index do |m,i|
  puts m
  puts "Index #{i}"
end
```

Methods

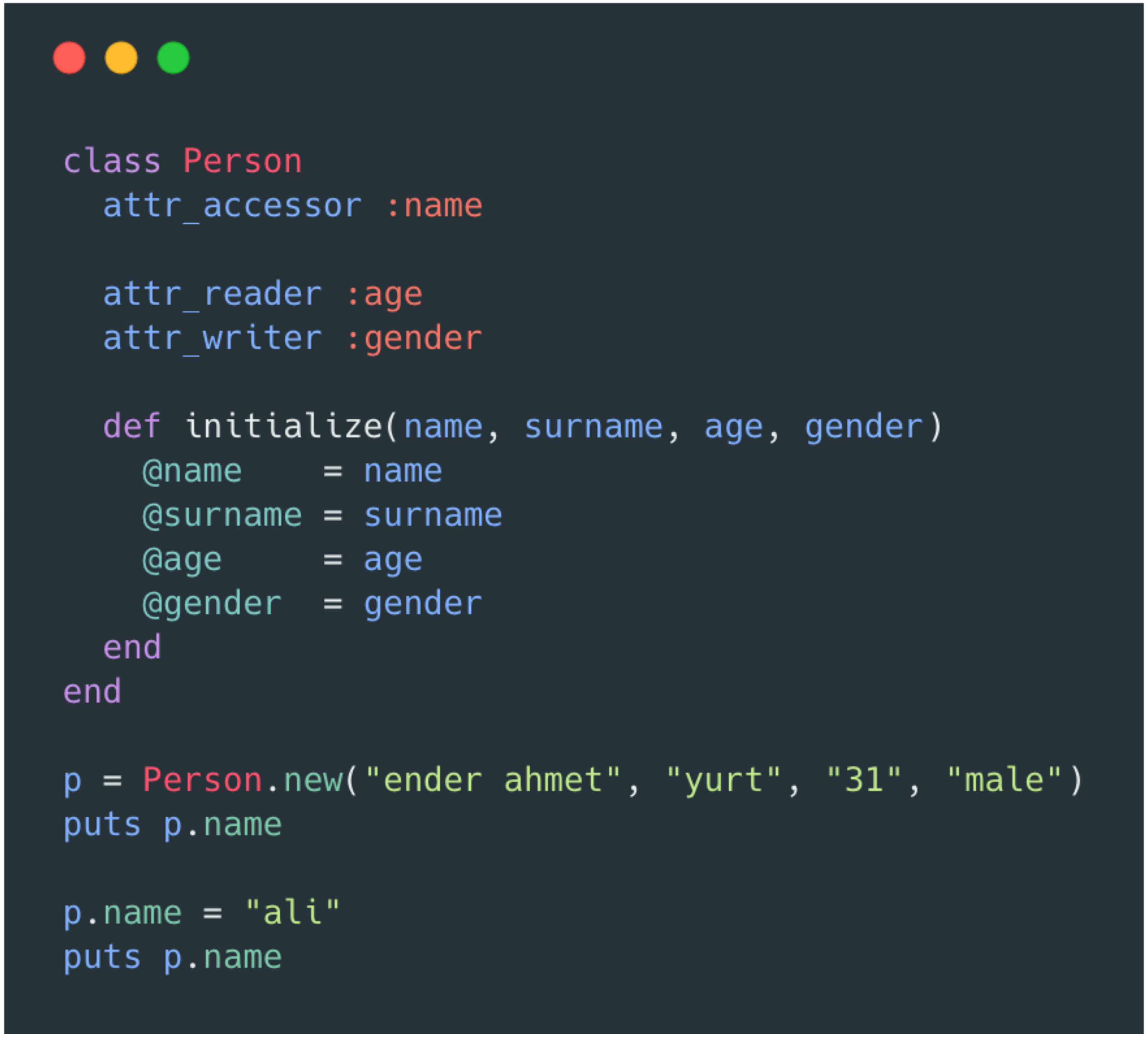
- What is it?
- Type of parameters
- Return



```ruby
def mult(p1, p2=0)
  puts "Result: #{p1*p2}"
end

puts mult(1,2)

def sum(*params)
  result = 0
  params.each {|p| result += p }
  result
end

puts sum(1,2,3,4,5,6)
```

Classes

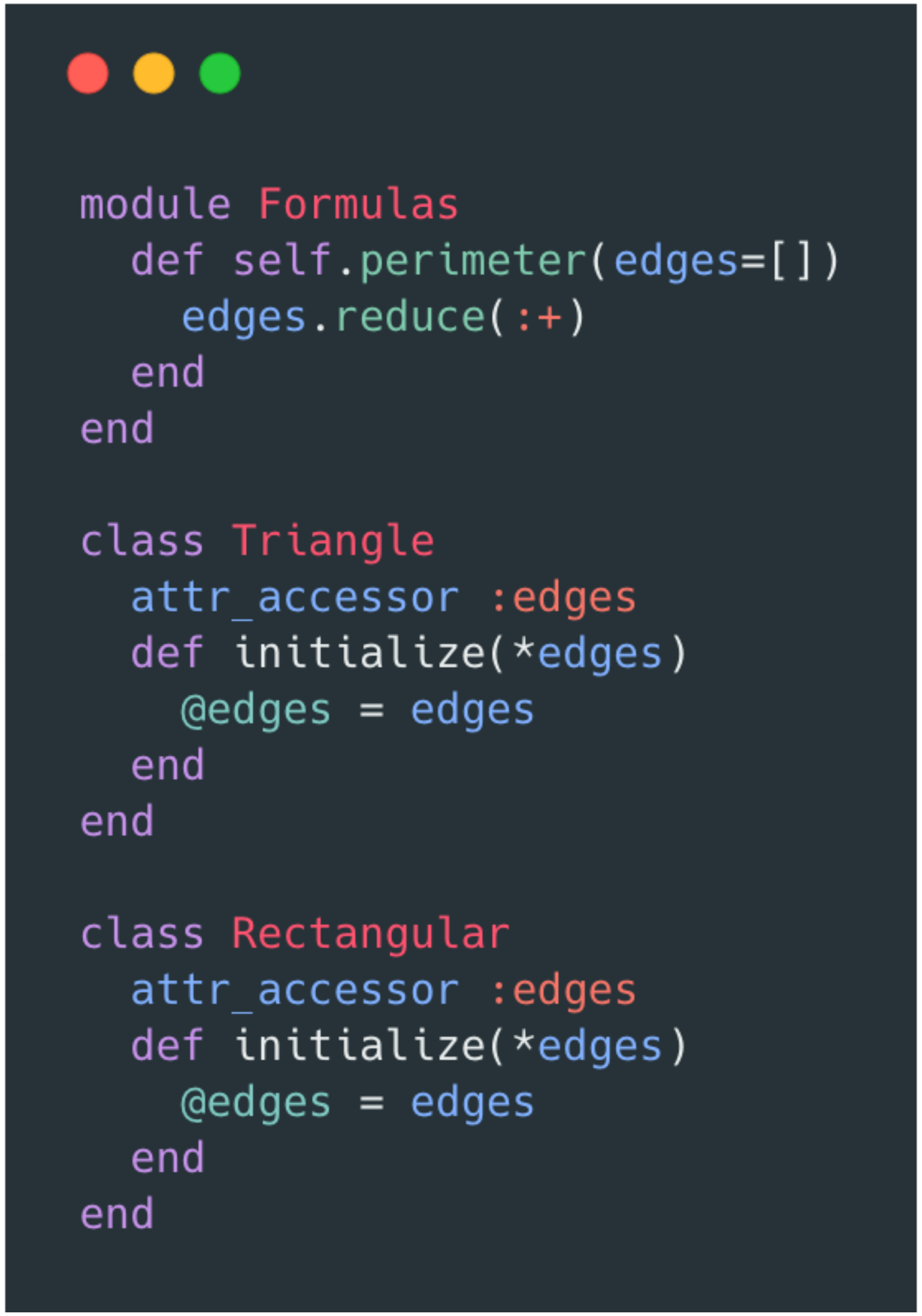
- Basics
- Attributes
- Methods
- Inheritance



```
class Person
  attr_accessor :name

  attr_reader :age
  attr_writer :gender

  def initialize(name, surname, age, gender)
    @name    = name
    @surname = surname
    @age     = age
    @gender  = gender
  end
end

p = Person.new("ender ahmet", "yurt", "31", "male")
puts p.name

p.name = "ali"
puts p.name
```

Modules

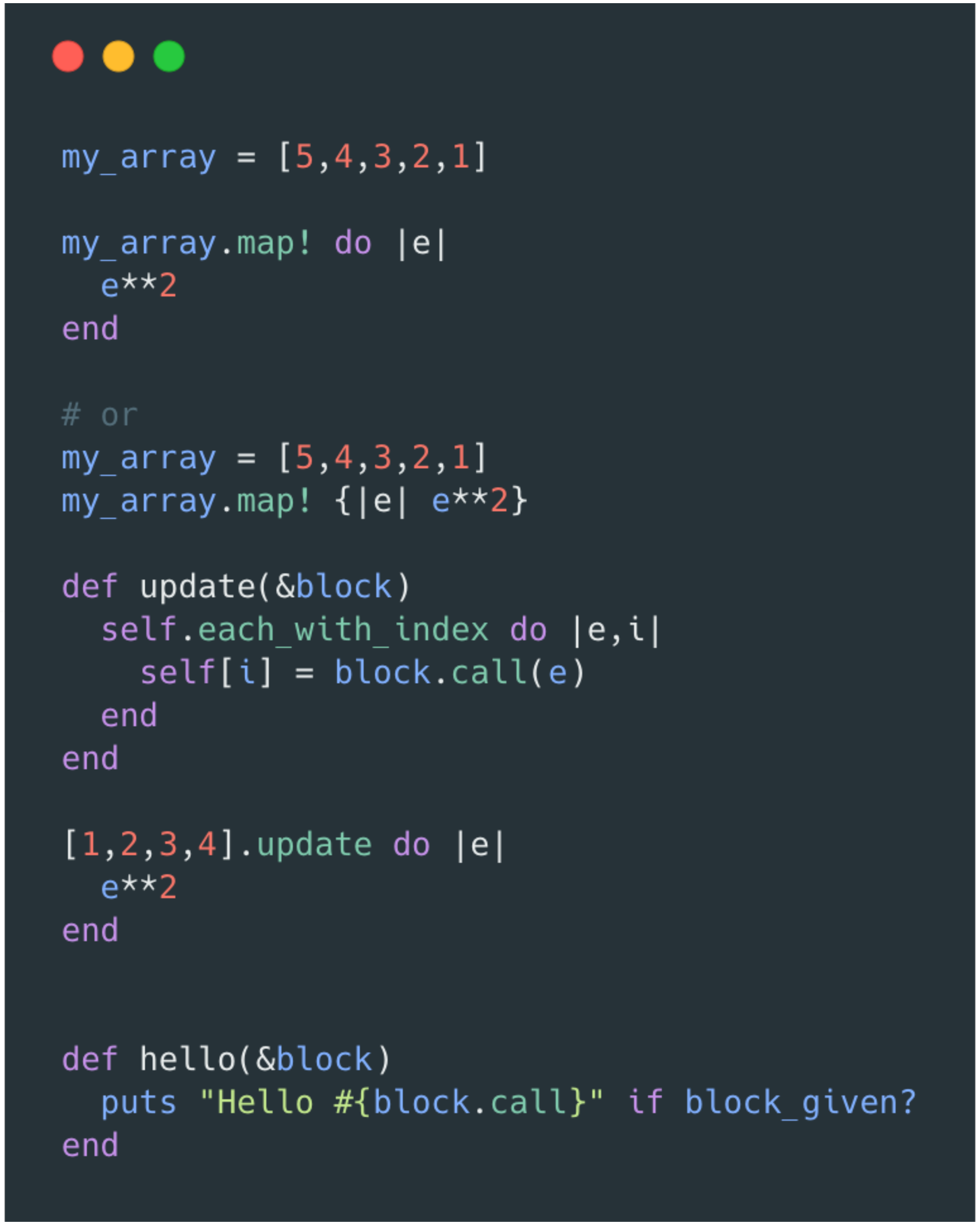
- Namespace
- Mixins
    - include
    - exclude



```
module Formulas
  def self.perimeter(edges=[])
    edges.reduce(:+)
  end
end

class Triangle
  attr_accessor :edges
  def initialize(*edges)
    @edges = edges
  end
end

class Rectangular
  attr_accessor :edges
  def initialize(*edges)
    @edges = edges
  end
end
```

Blocks

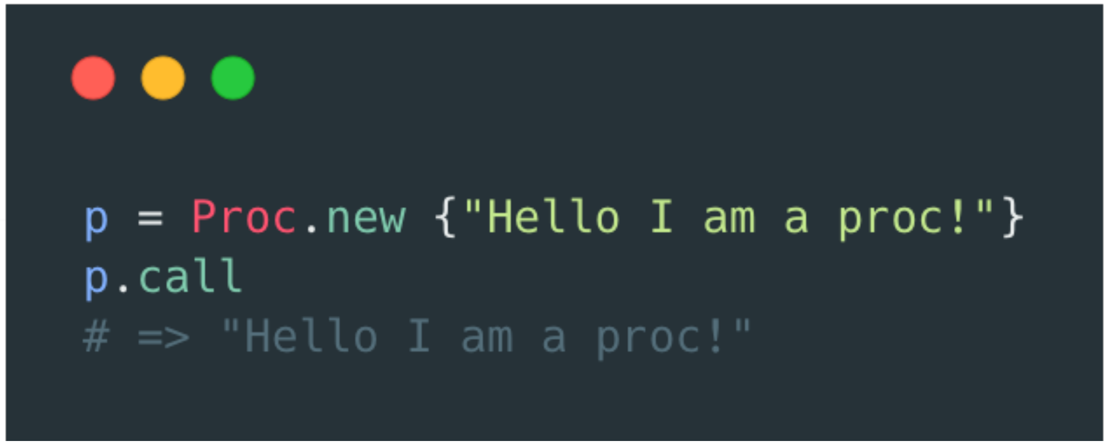
- Anonymous functions
- {}
- yield
- Procs
- Lambda



```ruby
my_array = [5,4,3,2,1]

my_array.map! do |e|
  e**2
end

# or
my_array = [5,4,3,2,1]
my_array.map! {|e| e**2}

def update(&block)
  self.each_with_index do |e,i|
    self[i] = block.call(e)
  end
end

[1,2,3,4].update do |e|
  e**2
end


def hello(&block)
  puts "Hello #{block.call}" if block_given?
end
```

Procs

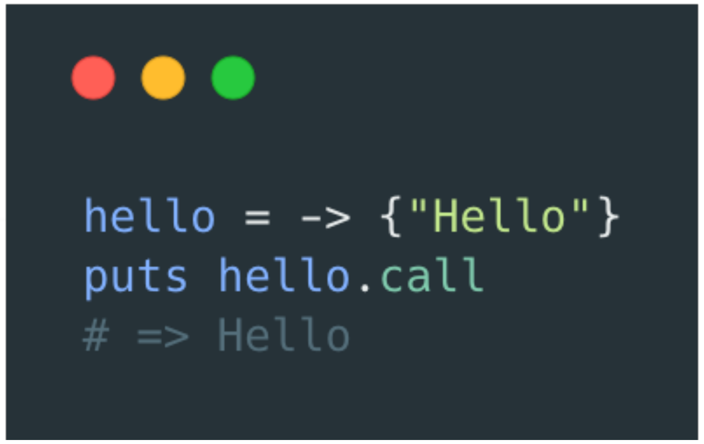
- Block is not an object
- Proc is an object
- p = Proc.new {}



```
p = Proc.new {"Hello I am a proc!"}
p.call
# => "Hello I am a proc!"
```

Lambda

- A Proc object
- l = lambda {}



```
hello = -> {"Hello"}
puts hello.call
# => Hello
```

Proc vs Lambda

- Number of parameters
  - Not important for Procs
  - Important for Lambdas
- Return formats for methods
  - Procs not continue
  - Lambdas continues

More

- Error handling
- File process
- RubyGems
- Regex
- Metaprogramming
- Web programming
  - Ruby on Rails
- Developing a real product

Questions?

💎 Thanks ❤️

enderahmetyurt.com

rubyturkiye.org
yakutrb.com