



# Ruby 101

## Ender Ahmet Yurt

[www.enderahmetyurt.com](http://www.enderahmetyurt.com)

# Ender Ahmet Yurt

Full Stack Developer @ Sigortaci123

Açık kaynak geliştiricisi

Ruby Türkiye

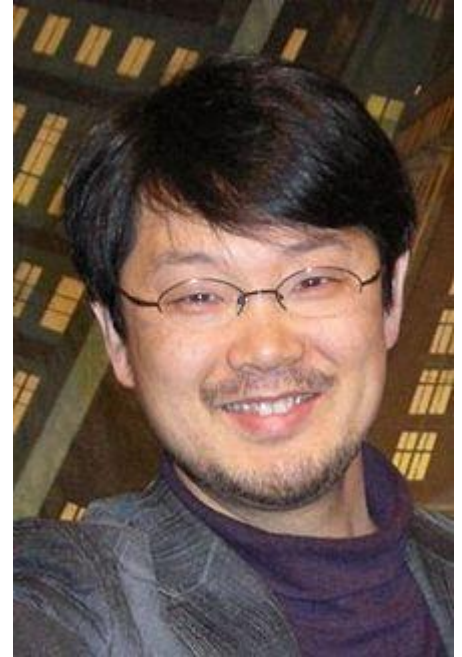
Twitter: ndr42

# İçindekiler

- Ruby
  - Nedir
  - Özellikleri
  - Diğer Uyarlamalar
  - Kurulum
- İlk Adımlar
  - Irb/Pry
  - İlk Program
- Temeller
- Kavramlar
- Daha fazlası
- Sorular / Teşekkürler

# Ruby

- Yukihiro Matsumoto (Matz)
- İlk versiyon 1995
- Amaç: “Dünyadaki tüm programcılara daha yaratıcı olmaları, programlamadan zevk almaları ve mutlu olmaları için yardım etmek”
- Açık kaynak kod <https://github.com/ruby/ruby>
- 2.4.0 (25.12.2016)



# Özellikleri

- Sade
- Her şey birer nesne
- Kod yazmak daha kolay
- Kolay okunabilir. İngilizce konuşur gibi
- Bir işi yapmanın birden fazla yolu var
- Dinamik bir dil
- Derlemeli değil, yorumlamalı bir dil
- Güçlü topluluk
  - Global
  - Türkiye [www.rubyturkiye.org](http://www.rubyturkiye.org)
- Ruby gems <https://rubygems.org>

# Diğer Uyarlamalar

- JRuby
- IronRuby
- MacRuby
- Rubinius

# Kurulum

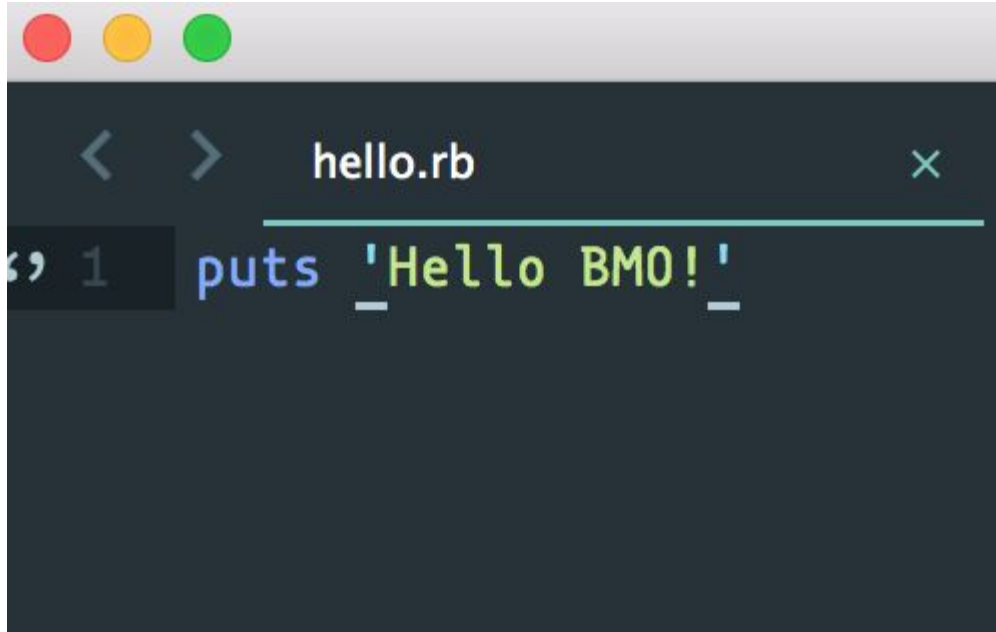
- **Osx**
  - Üstünde geliyor
  - RVM / RBENV
- **Linux**
  - `$ sudo apt-get install ruby<version>`
  - RVM / RBENV
- **Windows**
  - <https://rubyinstaller.org>

# Irb/Pry

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



# ilk program



```
< > hello.rb x  
1 puts 'Hello BM0!'
```

```
> ruby hello.rb  
Hello BM0!
```

# Temel IO işlemleri

- puts
- gets

# Kavramlar

- Değişkenler
- Sayılar
- Diziler
- Sözlükler
- Akış Kontrolü
- Döngüler
- Metotlar
- Sınıflar
- OO
- Modüller
- Proc Lambda

# Değişkenler

- Lokal değişkenler
- Global değişkenler
- Sınıf değişkenleri
- Örnek değişkenleri
- Sabitler
- Semboller

# Sayılar

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

# Diziler

- Oluşturma
- Eleman erişimi
- Eleman ekleme ve çıkarma
- Enumerable metotlar

# Sözlükler (Hash)

- Tanım
- Key-value pair
- Enumerable metotlar

# Akış Kontrolü

- True, False, Nil
- NOT (!)
- If-else-elsif-unless
- Case when
- Koşullu atama



# Döngüler

- for
- while
- until
- loop (break/next)
- each
- each\_with\_index

# Metotlar

- Tanım
- Parametre çeşitleri
- Return

# Sınıflar

```
2.3.1 :001 > puts 10.class  
Fixnum  
=> nil  
2.3.1 :002 > puts "ender".class.superclass  
Object  
=> nil
```

String -> Object -> BasicObject

# Sınıflar-2

- Metotlar
  - Public
  - Private
  - Protected
- Kalıtım

# Modüller

- Namespace
- Mixins
  - include
  - exclude

```
module Formuller
  def self.cevre(kenarlar=[])
    kenarlar.reduce(:+)
  end
end

class Ucgen
  attr_accessor :kenarlar
  def initialize(*kenarlar)
    @kenarlar = kenarlar
  end
end

class Dikdortgen
  attr_accessor :kenarlar
  def initialize(*kenarlar)
    @kenarlar = kenarlar
  end
end

ucgen = Ucgen.new(3,5,8)
dikdortgen = Dikdortgen.new(2,5,2,5)

puts Formuller::cevre(ucgen.kenarlar) # => 16
puts Formuller::cevre(dikdortgen.kenarlar) # => 14
```

# Bloklar

- Bloklar
  - Anonim fonksiyonlar
  - Metotlar tanımlanır ve tekrar tekrar kullanılır
  - Bloklar bir kere kullanılır, tanımlamaları yoktur
- Proc
- Lambda

## Bloklar-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

# => [1, 4, 9, 16]
```

## Bloklar-3

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



# Proc

- `Proc.new {}`
- `var = proc {}`

```
|proc_var = proc {"Hello there!"}  
puts proc_var.call  
# => Hello there!
```

# Lambda

- `l = lambda {}`
- `l.class # => Proc`
- `l = -> {}`

```
proc_var = proc {"Hello there!"}  
puts proc_var.call  
# => Hello there!
```

```
sum = lambda {|x,y| x+y}  
sum1 = -> x,y {x,y}  
  
puts sum.call(1,2)  
# => 3
```

# Proc vs Lambda

- Parametre sayısı
  - Proc'da önemli değil
  - Lambda'da önemli
- Metot dönüş biçimleri
  - Proc'da çağrılan yer devam etmez
  - Lambda'da çağrılan yerden devam eder

# Daha fazlası

- Dosya işlemleri
- Ruby kütüphaneleri ile işlemler
- Regex işlemleri
- Metaprogramming
- Web programming
  - Ruby on Rails
- Gerçek bir projede çalışmak

Sorular ?

Teşekkürler <3

[enderahmetyurt.com](http://enderahmetyurt.com)