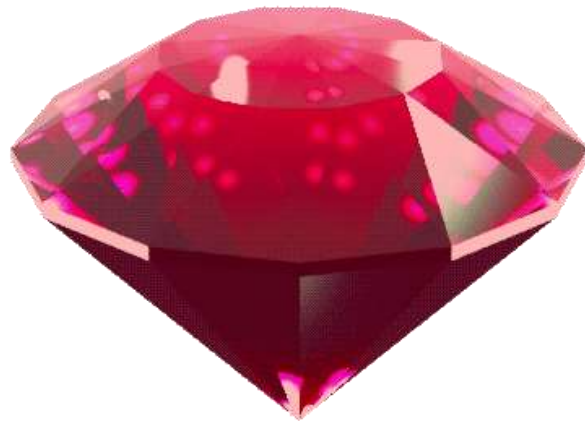


بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِیْمِ



Ruby

تهیه کنندگان:

زهرة سعيدزاده - هانیه سوری

دانشگاه کردستان

تاریخ ارائه:

۱۳۹۷/۰۳  
/۰۲

# عناوین



- ❖ تاریخچه
- ❖ ویژگی‌ها
- ❖ ۶ جزء اصلی
- ❖ کاربردها



# Ruby



نسخه فعلی ۱, ۵, ۲

مفسری

مک‌روبی، جی‌روبی، روبینوس، هات‌روبی، آیرن‌روبی و ام‌روبی

تحت حمایت شرکت‌های: Heroku, new Relik, Engine Yard

# Ruby



open source

Portable

Dynamic

نحو کوتاه و انعطاف پذیر

مدیریت استثنا

محیط برنامه‌سازی مناسب: JetBrains

پایگاه داده: Sybase و MySQL، DB Oracle

# Ruby



- Data (1)
- Primitive Operations (2)
- Sequence Control (3)
- Data Control (4)
- Storage Management (5)
- Operating Environment (6)





Ruby

Data Types

- 1) String
- 2) Numeric
  - a) Integer
  - b) Float
- 3) Hash
- 4) Symbol
- 5) Array
- 6) Range

THE  
**Ruby**



# Ruby

## Data Types

### String

#### ► Specification

► Attribute

variable size

متجانس

حداکثر تعداد اجزاء: نامحدود

سازمان منطقی بین اجزاء: خطی

► Value

► Operation

`*` , `+` , `<=>` , `==` , `=~` , `to_s` , `to_i`





Ruby

Data Types

## String

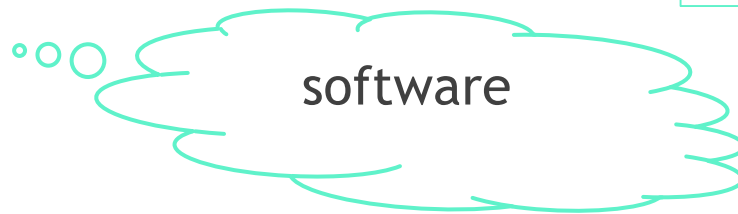
### ► Implementation

► Storage representation



LinkedRepresentation

► Algorithms





# Ruby

## Data Types

### Numeric

- 1) Integer
- 2) Float

#### ► Specification

- Attribute
- Value
- Operation

تعداد Component ها نا محدودیت

`% , * , + , << , <=> , == , to_s , to_i`

#### ► Implementation

- Storage representation
- Algorithms

software



# Ruby

## Data Types

### Hash

#### ► Specification

► Attribute

► Value

► Operation

#### ► Implementation

► Storage representation

► Algorithms

variable size  
 نامتجانس  
 دستیابی به یک جز: کلید  
 حداکثر تعداد اجزاء: نامحدود  
 سازمان منطقی بین اجزاء: خطی

**== , =>, .delete, .keys, .values, .invert, .length**

LinkedRepresentation

software



# Ruby

## Data Types

### Symbol

► **Specification**

- Attribute
- Value
- Operation

Fixed size

<=> , ==, =

► **Implementation**

- Storage representation
- Algorithms

Sequential Representation

software



# Ruby

## Data Types

### Array

#### ► Specification

- Attribute
- Value

#### ► Operation

#### ► Implementation

- Storage representation
- Algorithms

variable size  
نامتجانس

**\* , + , - , << , .insert, delete\_at, <=> , eql?, .sort**

LinkedRepresentation

software



# Ruby

## Data Types

### Range

#### ► Specification

- Attribute
- Value
- Operation

Fixed size  
متجانس

==, ===, eql?

#### ► Implementation

- Storage representation
- Algorithms

software



# Ruby

`+, -, *, /, %, =, +=, -=`

`*=, /=, %=, **=, **`

Operations

`x += y #x = x + y`

`x **= y #x = x ** y`

`x, y = 10, 20`



`x = 10  
y = 20`

`x, y = y, x`

`#x = 20 y = 10`





# Ruby

Operations

== != > < >= <= <=> .eql? ===

Example:

```
puts 3 == 3.0 #true
```

# but

```
puts 3.eql?(3.0) #false
```

```
puts 1<=>2 # -1    puts 1<=>1 # 0    puts 1<=>0 # 1
```

```
puts (1..100) === 99 # true
```





Ruby

and &&

or ||

not !

Operations

```
if (a>b && b<10) || (a<b && b>10)
```

```
puts a
```

```
end
```



Sequence Control

## Precedence

() before \*\* before \* / % before + -



کاربردها

۶ جز اصلی

ویژگی‌ها



Ruby

## If/else/elsif/unless statement

Example:

```
num = 8
if num == 3
  puts "Number is 3"
elsif num == 7
  puts "Number is 7"
else
  puts "Not found"
end #outputs "Not found"
```

Sequence Control



کاربردها

۶ جز اصلی

ویژگی‌ها



# Ruby

## If/else/elsif/unless statement

Example:

```
a = 42
unless a < 10
  puts "Yes"
else
  puts "No"
end
# outputs "Yes"
```

```
a = 42
puts "Yes" if a > 10
puts "Yes" unless a < 10
```

Sequence Control



case statement



۲۱ از ۳۴

۶ جز اصلی

ویژگی‌ها



```
age = 18
```

```
case age
```

```
when 1, 2, 3
```

```
  puts "Little baby"
```

```
when 4, 5
```

```
  puts "child"
```

```
else
```

```
  puts "Not a baby"
```

```
end #outputs "Not a baby"
```

Ruby

Sequence Control



# Ruby

## Loops

```
x = 0
while x < 10
  puts x
  x += 1
end
```

```
a = 0
until a > 10
  puts "a = #{a}"
  a += 2
end
```

Sequence Control



# Ruby

## Loops

```
for i in (1..10)
  break if i > 8
  next if i % 2 == 0
  puts i
end
```

```
x = 0
loop do
  puts x
  x += 1
  break if x > 10
end
```

Sequence Control



کاربردها

۶ جز اصلی

ویژگی‌ها



# Ruby

## Loops

```
arr = [2, 4, 6]
sum = 0
arr.each do |x|
  sum += x
end
puts sum # 12
```

```
10.times do
  puts "Hi"
end
```

Sequence Control





# Ruby

Data Control

Local:

```
def calc(x)
  y = 2
  puts x*y
end
puts y
```

# output: "undefined local variable or method 'y'"



کاربردها

۶ جز اصلی

ویژگی‌ها



# Ruby

Data Control

Global:

```
$x = 42
def change
  $x = 8
end
change
puts $x
# output: 8
```



نمونه کد

کاربردها

۶ جز اصلی

ویژگی‌ها



# Ruby

Class  
Instance

Data Control

```
class Person
  def initialize (age) @age = age end
  def show
    puts “#{@age} years #{days_lived} days”
  end
  private def days_lived
    @age * 365
  end
end
```

۲۷ از ۳۴



# Ruby

## Storage Management

- ❑ مدیریت حافظه خودکار و پویا
- ❑ جمع‌آوری زباله خودکار
- ❑ برای شیء‌های کوچک heap و برای شیء‌های بزرگ  
malloc extra memory outside of heaps



Ruby

Operating Environment

- ▶ Batch Processing
- ▶ Interactive

puts  
gets  
prints

۲۹ از ۳۴





کاربردها



۶ جز اصلی



ویژگی‌ها



Ruby

## Framework

Ruby on Rails

David Heinemeier Hansson (DHH) ۲۰۰۵

سیناترا

پادرینو

# Ruby



ویژگی‌ها

۶ جز اصلی

کاربردها



در حوزه وب

شبیه‌سازی

مدل‌سازی سه بعدی

رباتیک

شبکه‌بندی

علم تلفن

سیستم‌های مدیریت سیستم

و ...





# Ruby



- Hulu ◀
- Bloomberg ◀
- Shopify ◀
- Airbnb ◀
- SlideShare ◀

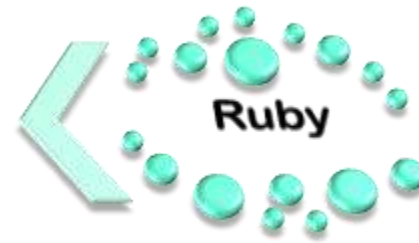




کاربردها

۶ جز اصلی

ویژگی‌ها



# Ruby

```
def fact n
  if n <= 1
    1
  else
    n * fact(n-1)
  end
end
puts "fact = #{fact 5}"
```



