درس ۸

Ruby

map · select · reduce

app.rb

```ruby
nums.select(&:even?)
    .map { |n| n * n }
    .sum
```

یک کورس رایگان از MyWebUniversity

برنامه‌نویسی Ruby

درس ۸

ابزارهای قوی Enumerable



map، select، reject، reduce، sort_by، group_by، tally، each_slice و زنجیر کردن

reject و map، select

map هر عنصر را تبدیل می‌کند و یک آرایهٔ نو برمی‌گرداند. select عنصرهایی را نگه می‌دارد که ...

این‌ها هرگز اصل را تغییر نمی‌دهند، پس امن و آسان برای زنجیر کردن هستند.

map.rb

```ruby
nums = [1, 2, 3, 4, 5, 6]
nums.map { |n| n * n }        # [1, 4, 9, 16, 25, 36]
nums.select(&:even?)          # [2, 4, 6]
nums.reject(&:even?)          # [1, 3, 5]
nums.find { |n| n > 3 }       # 4
nums.any? { |n| n > 5 }       # true
```

01

reduce، sum و minmax

reduce همهٔ عنصرها را با بردن یک جمع‌کننده به یک مقدار ترکیب می‌کند. sum، min، max و minmax ...

count با یک بلاک آیتم‌های مطابق را می‌شمارد.

reduce.rb

```ruby
[1, 2, 3, 4].reduce(0) { |acc, n| acc + n }   # 10
[1, 2, 3, 4].reduce(:*)                        # 24
scores.sum / scores.size.to_f                 # average
scores.minmax                                 # [62, 98]
scores.count { |s| s >= 90 }
```

02

sort_by، min_by و max_by

sort_by بر اساس مقداری که بلاک برمی‌گرداند مرتب می‌کند. برای ترتیب نزولی از یک عدد منفی یا ...

min_by و max_by عنصری با کوچک‌ترین یا بزرگ‌ترین مقدار را پیدا می‌کنند، مثلاً بهترین شاگرد.

sort.rb

```ruby
students.sort_by { |s| -s[:score] }            # highest first
students.sort_by { |s| [s[:city], s[:name]] }  # two keys
students.max_by { |s| s[:score] }[:name]       # "Mariam"
%w[pear fig banana].sort_by(&:length)         # ["fig", "pear", "banana"]
```

03

partition و group_by، tally

group_by یک hash از گروه‌ها می‌سازد. tally می‌شمارد هر مقدار چند بار ظاهر می‌شود. partition ...

این‌ها جای سطرهای زیادی از حلقه‌ها در زبان‌های دیگر را می‌گیرند.

group.rb

```ruby
students.group_by { |s| s[:city] }
# {"Kabul"=>[...], "Herat"=>[...]}

%w[a b a c a].tally          # {"a"=>3, "b"=>1, "c"=>1}

pass, fail = scores.partition { |s| s >= 50 }
```

04

زنجیر کردن همه

method ها را به یک خط لوله زنجیر کنید: کلمه‌ها را بخوانید، یکسان کنید، بشمارید، مرتب کنید، ...

each_slice و each_cons آیتم‌ها را در گروه‌ها پردازش می‌کنند، و lazy ترتیب‌های عظیم یا ...

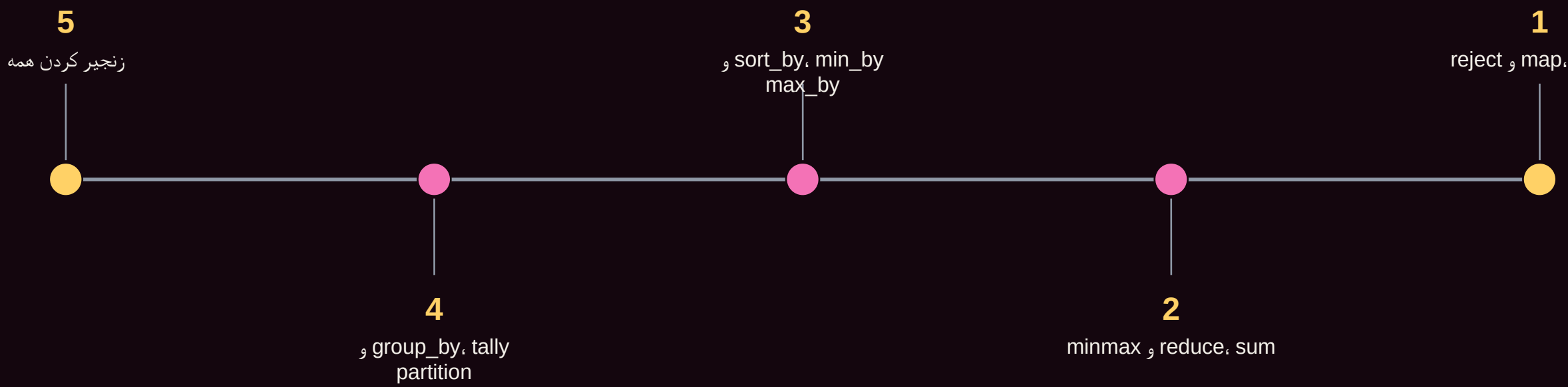
top_words.rb

```ruby
File.read("story.txt")
    .downcase
    .scan(/[a-z']+/)
    .tally
    .sort_by { |_, n| -n }
    .first(5)
    .each { |w, n| puts "#{w.ljust(10)} #{n}" }

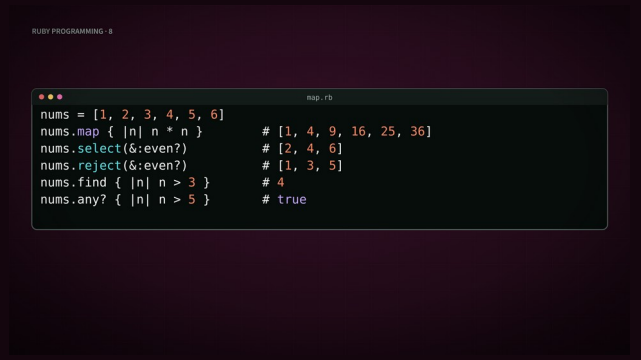
(1..).lazy.select(&:even?).first(3)     # [2, 4, 6]
```

05

نقشهٔ درس


5
زنجیر کردن همه
4
group_by، tally و
partition
3
sort_by، min_by و
max_by
2
reduce، sum و
minmax
1
map، select و
reject

منابع و مطالعهٔ بیشتر





تشکر از تماشای شما

MyWebUniversity.com

آموزش رایگان تکنالوژی از سال ۲۰۰۰ — انگلیسی و دری

@MyWebUniversity-Dari · @MyWebUniversity