Ruby Monstas

Session 10

Agenda

- Symbols
- Ranges
- Exercises

Symbols

How to write a symbol?

```
irb(main):001:0> :name
=> :name
irb(main):002:0> :address
=> :address
irb(main):003:0> :any_string
=> :any_string
```

Symbols are like strings, but immutable

```
irb(main):010:0> puts "hello world".object_id
280
=> nil
irb(main):011:0> puts "hello world".object_id
880
=> nil
irb(main):012:0> puts :"hello world".object_id
628
=> nil
irb(main):013:0> puts :"hello world".object_id
628
=> nil
```

When to use symbols?

- The string needn't be changed at runtime.
- The string doesn't need methods of class String.

```
irb(main):001:0> full_names = {
irb(main):002:1* :marion => 'Marion Schleifer',
irb(main):003:1* :fabian => 'Fabian Mettler',
irb(main):004:1* :ferdinand => 'Ferdinand Niedermann',
irb(main):005:1* :thomas => 'Thomas Ritter'
irb(main):006:1> }
=> {:marion=>"Marion Schleifer", :fabian=>"Fabian Mettler", :ferdinand=>"Ferdinand
Niedermann", :thomas=>"Thomas Ritter"}

irb(main):007:0> full_names[:marion]
=> "Marion Schleifer"
```

Ranges

What are ranges?

- Set of values with beginning & end
- Can be constructed with a type that are comparable with the ⇔ operator
  - i.e. integers, floats, doubles, strings etc.

How to write a range?

- irb(main):003:0> range = (1..9)

  => 1..9

 - Range with numbers from 1 to 9

- irb(main):004:0> other_range = (1...9)

  => 1...9

 - Range with numbers from 1 to 8

Methods on ranges

- begin

  irb(main):006:0> range.begin

  => 1

- cover?

  irb(main):007:0> letter_range = ("a".."z")

  => "a".."z"

  irb(main):008:0> letter_range.cover?("c")

  => true

Methods on ranges

- to_a

irb(main):009:0> range = (1..9)

=> 1..9

irb(main):010:0> range.to_a

=> [1, 2, 3, 4, 5, 6, 7, 8, 9]

irb(main):011:0>

Methods on ranges

- each

  irb(main):020:0> range.each do |number|

  irb(main):021:1* print number + 1

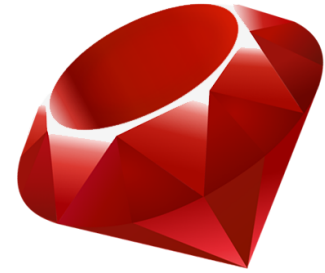 irb(main):022:1> end

  2345678910=> 1..9

Documentation

Check out the Ruby documentation for more range methods:

http://ruby-doc.org/core-2.2.0/Range.html

Time to practice

Let's get to it!