# Core Ruby 41st Batch




## Navigation

Home

- My home
  - Site pages
  - My profile
  - Current course
    - PORPC101-41C
      - Participants
      - General
      - 5 April - 11 April
      - 12 April - 18 April
      - 19 April - 25 April
      - 26 April - 2 May
      - 3 May - 9 May
      - Week 5 Tutorial
      - Week 5: Exercises
      - Week 5: Forum
      - L5E2 Spec Change
      - **Some useful Ruby methods**
        - Week 5: Quiz
      - 10 May - 16 May
      - 17 May - 23 May
      - 24 May - 30 May
      - 31 May - 6 June
      - 7 June - 13 June
  - My courses

## Settings

- Course administration


## Some useful Ruby methods

**obj.inspect -> string**

The **Object** class has an **inspect** method that returns a string containing a human-readable representation of obj. If not overridden, uses the **to_s** method to generate the string.

```
puts [ 1, 2, 3..4, 'five' ].inspect
```

**ObjectSpace.each_object( h class_or_mod i ) {| obj | block } -> fixnum**

The Ruby runtime system needs to keep track of all known objects. This information is made accessible via the **ObjectSpace.each_object** method. It returns the number of objects found. Objects of Fixnums, Symbols, true, false, and nil are never returned.

```
ObjectSpace.each_object do |obj|

printf "%20s: %s\n", obj.class, obj.inspect

end
```

If you specify a class or module as a parameter to each_object, only objects of that type will be returned.

**enum.partition {| obj | block } -> [ true_array, false_array ]**

The **partition** method of module **Enumerable**, divides a collection into two parts.

When partition is called and passed a block, it returns two arrays, the first containing the elements of **enum** for which the block evaluates to true, the second containing the rest.

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

odd_even = nums.partition {|x| x%2 ==1}

puts odd_even
```

### enum.any? h {| obj | block } -> true or false

The quantifier method **any?** of module **Enumerable** makes it easier to test the nature of a collection.

```
nums = [1,2,3,4,5,6,7,8,9,nil]

# Are any of these numbers even?

flag = nums.any? {|x| x%2 ==0}

puts flag # true
```

It passes each element of the collection to the given block. The method returns **true** if the block ever returns a value other than **false** or **nil**. In the absence of a block, this simply tests the truth value of each element. That is, a block **{|x| x}** is added implicitly.

```
flag1 =nums.any?

# list contains at least one true value

# (non-nil or non-false)

puts flag1 # true
```

### enum.all? h {| obj | block } -> true or false

The quantifier method **all?** of module **Enumerable** makes it easier to test the nature of a collection.

```
nums = [1,2,3,4,5,6,7,8,9,nil]

# Are all of these numbers even?

flag = nums.all? {|x| x%2 ==0}

puts flag # false
```

It passes each element of the collection to the given block. The method returns **true** if the block never returns **false** or **nil**. In the absence of a block, this simply tests the truth value of each element. That is, a block **{|x| x}** is added implicitly.

```
flag1 =nums.all?

# list contains no falses or nils

puts flag1 # false
```

Last modified: Friday, 22 June 2012, 10:45 PM