Условия и циклы в Ruby

If, If-else, Case, While, Until, Loop

Создание простых проверок

```ruby
products_count = 10

if products_count < 1
    puts 'We are out of products'
end

puts 'We are out of products' if products_count < 1

p products_count > 1, products_count <= 1, products_count >= 1, products_count != 1, products_count == 1
#=> true, false, true, true, false

if !(products_count < 1)
    puts 'We have enough products'
end

puts 'We have enough products' unless products_count < 1
```

Проверка нескольких условий

```
18  if products_count < 1
19      puts 'We are out of products'
20  elsif products_count == 10
21      puts 'We have many products'
22  else
23      puts 'We have enough products'
24  end
25
26  country = 'FR'
27
28  puts 'We have enough products in FR' if products_count > 1 && country == 'FR'
29
30  age = 10
31
32  if age == 10 || age == 20
33      puts 'The age is 10 or 20'
34  end
35
36  puts age == 10 || age == 20 ? 'The age is 10 or 20' : 'The age isn\'t 10 or 20'
```

Условный оператор Case

```ruby
 1  puts (1..10) === 5, (1..10).include?(5) #=> true, true
 2
 3  puts (/abc/ === 'abcdef'), /abc/.match?('abcdef') #=> true, true
 4
 5  puts String === 'foo', 'foo'.is_a?(String) #=> true, true
 6
 7  ['1', 2, :a].any?(Integer) #=> puts true
 8
 9  def check_percent(percent)
10      case percent
11      when 0
12          'No percent'
13      when 1..20
14          'Low percent'
15      when 21..50
16          'Good percent'
17      when 51..80
18          'Very good percent'
19      when 81..100
20          'Excellent percent'
21      else
22          'Wrong percent'
23      end
24  end
```

Запись case условий через if

```ruby
def check_percent_short(percent)
    if 0..20 === percent
        'Low percent'
    elsif 21..80 === percent
        'Good percent'
    elsif 81..100 === percent
        'Excellent percent'
    else
        'Wrong percent'
    end
end

puts check_percent(30)
puts check_percent(130)

puts check_percent(30)
puts check_percent(130)
```

Примеры циклов в Ruby

```ruby
def iterate_example(&block)
    param = 5

    block.call(param)

    puts
end

iterate_example do |param|
    while param > 0
        print param, ' ' #=> 5 4 3 2 1
        param -= 1
    end
end


iterate_example do |param|
    begin
        print param, ' ' #=> 5 4 3 2 1
        param -= 1
    end while param != 0
end

iterate_example do |param|
    until param == 0
        print param, ' ' #=> 5 4 3 2 1
        param -= 1
    end
end
```

Примеры циклов. Продолжение

```
31  iterate_example do |param|
32      loop do
33          print param, ' ' #=> 5 6 7 8 9
34          param += 1
35
36          break if param == 10
37      end
38  end
39
40  iterate_example do |param|
41      for i in 1..param
42          break if i > 3
43          print i, ' ' #=> 1 2 3
44      end
45
46      puts
47
48      (1..param).each do |i|
49          next if i < 3
50
51          print i, ' ' #=> 3 4 5
52      end
53
54      puts
55
56      param.times do |i|
57          print i, ' ' #=> 0 1 2 3 4
58      end
59  end
```

Полезные ссылки

https://www.rubyguides.com/ruby-tutorial/ruby-if-else/ – Описание условный конструкций;

https://www.rubyguides.com/2015/10/ruby-case/ – Описание case оператора;

https://dev.to/baweaver/understanding-ruby-triple-equals-2p9c – О понимании оператора "===";

https://www.tutorialspoint.com/ruby/ruby_loops.htm – Статья про циклы.

Конец! Спасибо!