Becoming a Rubyist with Style

Implicit Returns

```ruby
# Less preferred
def get_avg(num_1, num_2)
    return (num_1 + num_2) / 2
end

# Preferred by a Rubyist
def get_avg(num_1, num_2)
    (num_1 + num_2) / 2
end
```

Omitting parentheses for method calls with no arguments

```ruby
def say_hi
    puts "hi"
end

# Less preferred
say_hi()

# Preferred by a Rubyist
say_hi
```

Use single line conditionals when possible

```ruby
raining = true

# Less preferred
if raining
    puts "don't forget an umbrella!"
end
```

```
# Preferred by a Rubyist
puts "don't forget an umbrella!" if raining
```

Use built-in methods

```
num = 6

# Less preferred
p num % 2 == 0

# Preferred by a Rubyist
p num.even?
```

```
people = ["Joey", "Bex", "Andrew"]

# Less preferred
p people[people.length - 1]

# Preferred by a Rubyist
p people[-1]
p people.last
```

Use enumerables to iterate

```
# Less preferred
def repeat_hi(num)
    i = 0
    while i < num
        puts "hi"
        i += 1
    end
end

# Preferred by a Rubyist
def repeat_hi(num)
    num.times { puts "hi" }
end
```

```ruby
# Less preferred
def all_numbers_even?(nums)
    nums.each do |num|
        return false if num % 2 != 0
    end

    true
end

# Preferred by a Rubyist
def all_numbers_even?(nums)
    nums.all? { |num| num.even? }
end
```