**(1..num).select {|x| num % x == 0}.count == 2** # boolean statement if prime

**Math.sqrt(num) % 1 == 0** # Boolean for perfect square

**array.reduce(:+)** # return sum of an array of integers

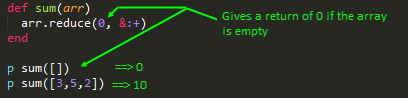
**string.chars.map {|x| x.to\_i}.reduce(:+)** # return sum of a string of integers

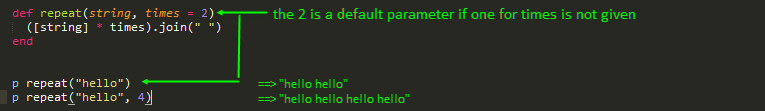
**array.each\_slice(n).to\_a** # returns array of sub arrays count == n

**(1..num).select {|x| num % x == 0}** # return array of all factors

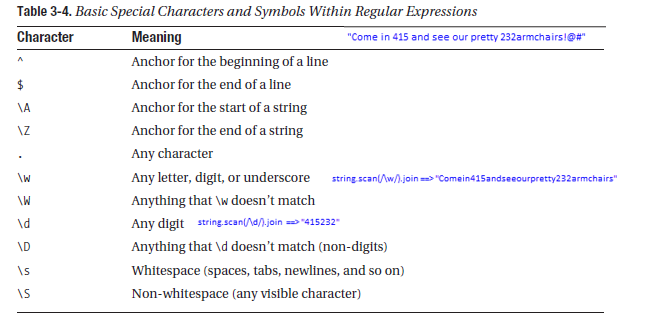
**p %w{Come in and see our pretty armchairs}** ==>["Come", "in", "and", "see", "our", "pretty", "armchairs"]

**p %q{Come in and see our pretty armchairs}** ==> "Come in 415 and see our pretty 232armchairs!@#"

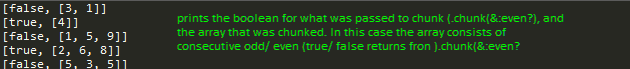




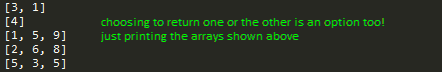






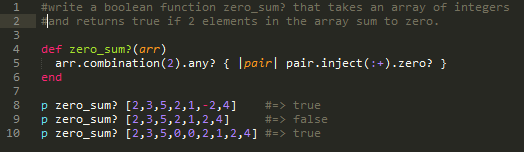




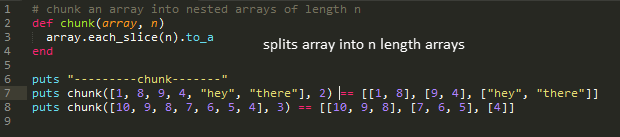




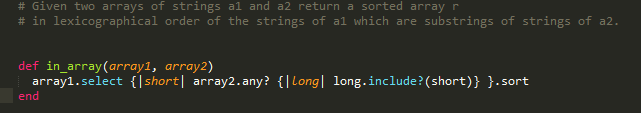


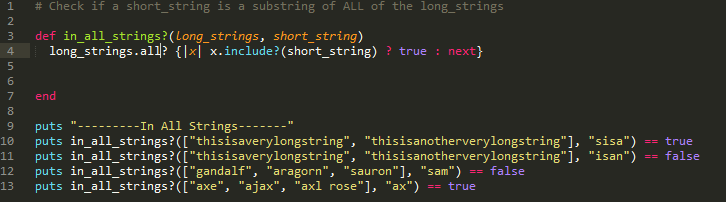


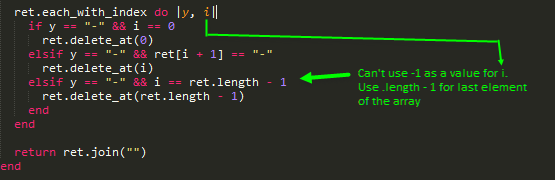


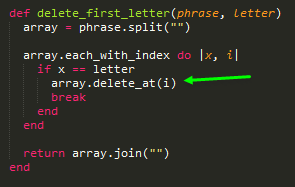


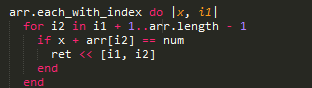


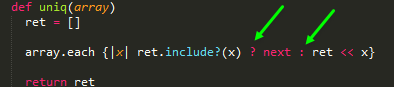


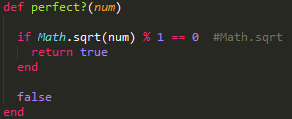




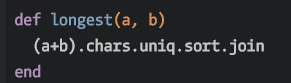








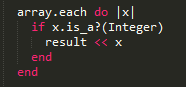
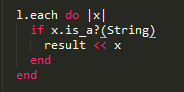
To find square root, need Math module. Call method sqrt on object number



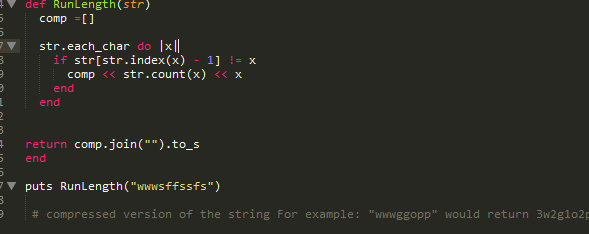
str + other\_str → new\_str

.chars #Returns an array of characters in str. This is a shorthand for str.each\_char.to\_a.

.join #returns string

To test the class (String/ Integer)and sent only those/ remove only those. .reject and .select



if str[str.index(x) - 1] != x # the str.index(x) only gives value for first occurrence ie for w, only 0 for all iter

**string.each\_char**

**for x in 1..+1.0/0.0**

**alpha = ("a".."z").to\_a**

**str1.scan(x).count**

**vowels = "aeiou".chars**

**array = string.split(" ").join("").split("")** #string with spaces to an array of individual letters w/o spaces