class Fish

# class\_variable

@@all\_fish = []

# attr\_reader :name

# attr\_writer :name

attr\_accessor :name # creates a getter and a setter

def self.all\_fish

@@all\_fish

end

def self.random\_state\_of\_being

[true, false].sample # randomly returns one of these values

end

# factory method

def self.make\_nemo

# puts self # context or receiver of the method

Fish.new("nemo", self.random\_state\_of\_being) # Fish and self are the same in this method

end

# factory method for randomly named fish

def self.make\_random\_fish

name = ""

5.times do

name << ('a'..'z').to\_a.sample

end

Fish.new(name, self.random\_state\_of\_being)

end

# def self.new(\*someargs)

# # some code that creates an instance

# instance.initialize(\*someargs)

# end

def initialize(name, lost = Fish.random\_state\_of\_being)

puts self

@name = name

@lost = lost # boolean

@@all\_fish << self

end

def lost? # ruby style denotes that you should have a question mark on boolean methods

# puts self

@lost

end

def find

@lost = false

end

# this is what attr\_reader does

# def name

# @name

# end

# attr\_writer

# def name=(name)

# @name = name

# end

end