REPOSITORY = ‘https://github.com/betrakiss/ruby-retrospective-2015-2’

# Още 20 неща, които научих

# Част 2

# 0. Default поведение пред nil поведение

# 1. Освен да си пиша константи в класовете, е хубаво и да ги правя константи.. (freeze)

# 2. Може да се вкарват променливи в placeholder-ите с хеш

# 3. arity и parameters спестяват дооста код

# 4. Hash-овете могат да сочат към функции!

# 5. Една идея как да подобря регулярните си изрази

# 6. Мога да връзвам методи след end -> end.join(…;)

# 7. Вместо да се вграждат класове, може да се дефинират така: class Outer::Inner

# 8. Трябва да се внимава с референциите (масив, хеш), едното домашно беше счупено заради един забравен dup

# 9. Правя твърде много променливи, които често са нужни само за един ред

# 10. Подравняването на приравнявания, например, е много по-четимо, ако са повече

# 11. flat\_map комбинира flatten и map .. каква изненада ;d

# 12. attr\_reader могат да бъдат изреждани на един ред

# 13. Може да се match-ва регулярен израз и с [regex] (квадратни скоби)

# 14. Как се ползва sprintf

# 15. Chomp връща nil, ако се използва върху празен стринг (а не празен стринг, както си мислех)

# 16. Мисля, че вече имам по-добра представа как да си пакетирам класовете заедно (от ObjectStore проекта)

# 17. Символът \ позволява да си пренасяш стринговете

# 18. Arity връща отрицателна стойност, когато има сплат параметър

# 19. Мога да сплитвам и по регулярен израз!