



**«Московский государственный технический университет  
имени Н.Э. Баумана»**

**(МГТУ им. Н.Э. Баумана)**

---

**ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ**

**КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)**

**НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА**

**О т ч е т**

**по лабораторной работе № 7**

**Дисциплина: Языки Интернет-программирования**

**Название лабораторной работы: Ruby. Классы, модули, примеси. Вариант 10.**

Студент гр. ИУ6-34Б

\_\_\_\_\_

К.Ю. Каташинский

\_\_\_\_\_

Преподаватель

\_\_\_\_\_

\_\_\_\_\_

Москва, 2020

# Часть 1

Условие задания:

*Дан символьный файл F. Записать в файл H с сохранением порядка следования те символы файла F, которым в этом файле предшествует буква «а». Автоматический тест программы обязательно должен проверять работу с файлами.*

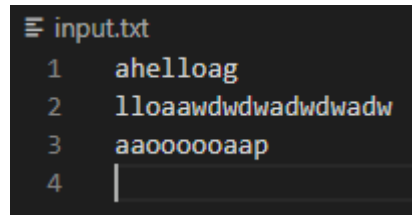
Код файла main.rb:

```
# frozen_string_literal: true
def open_file(file_name)
  array = File.readlines file_name
  File.open('output.txt', 'w') do |file|
    array.each do |str|
      (1..str.length).each do |i|
        if str[i - 1] == 'a'
          file.write(str[i])
        elsif str[i] == "\n"
          file.write(str[i])
        end
      end
    end
  end
end
```

Код файла test.rb:

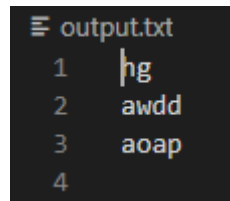
```
# frozen_string_literal: true
require 'minitest/autorun'
require './main.rb'
# Test class
class Test < MiniTest::Unit::TestCase
  def setup
    File.delete('output.txt') if File.exist?('output.txt')
  end
  def test_1
    File.open('input.txt', 'w') do |file|
      file.write("ahelloag\nlloaawdwdwadwdwadw\naaooooooooap\n")
    end
    open_file('input.txt')
    assert(File.exist?('output.txt'))
    array = File.readlines 'output.txt'
    assert_equal(array[0], "hg\n")
    assert_equal(array[1], "awdd\n")
    assert_equal(array[2], "aoap\n")
  end
  def test_2
    File.open('input.txt', 'w') do |file|
      file.write("aaff\nhelloasammaarat\n")
    end
    open_file('input.txt')
    assert(File.exist?('output.txt'))
    array = File.readlines 'output.txt'
    assert_equal(array[0], "af\n")
    assert_equal(array[1], "smart\n")
  end
end
```

Код файла user.rb:  
# frozen\_string\_literal: true  
require './main.rb'  
open\_file('input.txt')



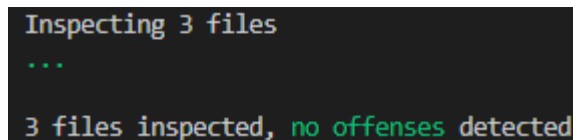
```
input.txt
1  ahelloag
2  lloaawdwadwdwadw
3  aaoooooaap
4  |
```

Рисунок 1 «Файл input.txt»



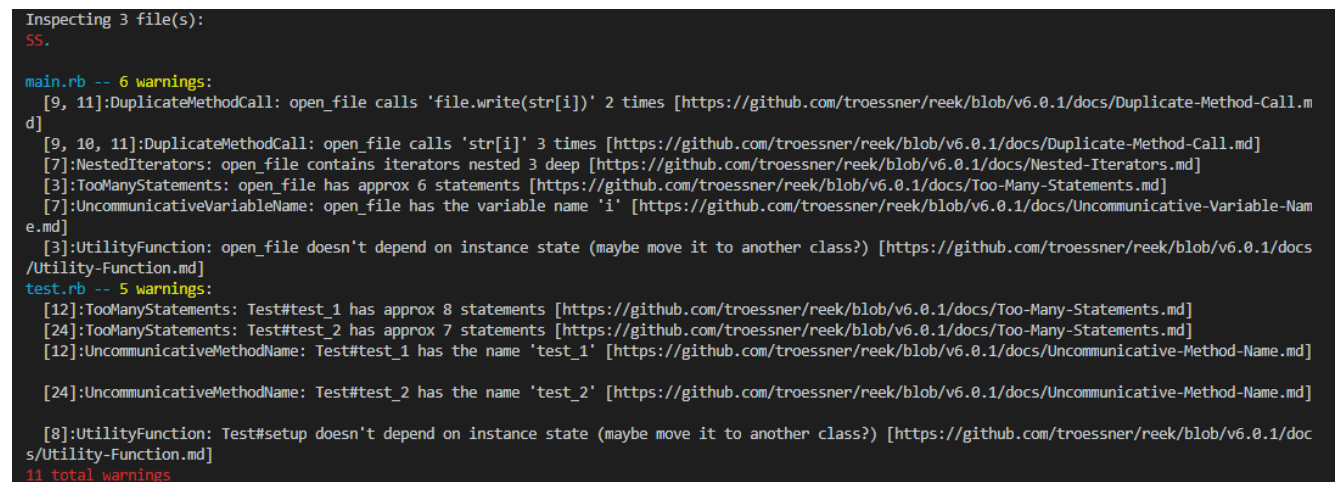
```
output.txt
1  hg
2  awdd
3  aoap
4  |
```

Рисунок 2 «Файл output.txt»



```
Inspecting 3 files
...
3 files inspected, no offenses detected
```

Рисунок 3 «Результаты проверки Rubosop»



```
Inspecting 3 file(s):
SS.

main.rb -- 6 warnings:
[9, 11]:DuplicateMethodCall: open_file calls 'file.write(str[i])' 2 times [https://github.com/troessner/reek/blob/v6.0.1/docs/Duplicate-Method-Call.md]
[9, 10, 11]:DuplicateMethodCall: open_file calls 'str[i]' 3 times [https://github.com/troessner/reek/blob/v6.0.1/docs/Duplicate-Method-Call.md]
[7]:NestedIterators: open_file contains iterators nested 3 deep [https://github.com/troessner/reek/blob/v6.0.1/docs/Nested-Iterators.md]
[3]:TooManyStatements: open_file has approx 6 statements [https://github.com/troessner/reek/blob/v6.0.1/docs/Too-Many-Statements.md]
[7]:UncommunicativeVariableName: open_file has the variable name 'i' [https://github.com/troessner/reek/blob/v6.0.1/docs/Uncommunicative-Variable-Name.md]
[3]:UtilityFunction: open_file doesn't depend on instance state (maybe move it to another class?) [https://github.com/troessner/reek/blob/v6.0.1/docs/Utility-Function.md]
test.rb -- 5 warnings:
[12]:TooManyStatements: Test#test_1 has approx 8 statements [https://github.com/troessner/reek/blob/v6.0.1/docs/Too-Many-Statements.md]
[24]:TooManyStatements: Test#test_2 has approx 7 statements [https://github.com/troessner/reek/blob/v6.0.1/docs/Too-Many-Statements.md]
[12]:UncommunicativeMethodName: Test#test_1 has the name 'test_1' [https://github.com/troessner/reek/blob/v6.0.1/docs/Uncommunicative-Method-Name.md]
[24]:UncommunicativeMethodName: Test#test_2 has the name 'test_2' [https://github.com/troessner/reek/blob/v6.0.1/docs/Uncommunicative-Method-Name.md]
[8]:UtilityFunction: Test#setup doesn't depend on instance state (maybe move it to another class?) [https://github.com/troessner/reek/blob/v6.0.1/docs/Utility-Function.md]
11 total warnings
```

Рисунок 4 «Результаты проверки Reek»

## Часть 2

Условие задания:

*Разработать и реализовать иерархию классов для описанных объектов предметной области, используя механизмы наследования. Проверить ее на тестовом примере, с демонстрацией всех возможностей разработанных классов на конкретных данных. Объект — больной. Параметры: фамилия, возраст. Методы: инициализирующий, вывода на экран фамилии и возраста. Объект — больной. Параметры: фамилия, возраст, год последней диспансеризации. Методы: инициализирующий, определения года следующей диспансеризации, исходя из того, что диспансеризация должна проходиться каждые 3 года. В тестирующей программе обеспечить автоматическую проверку того, что созданные объекты действительно соответствуют заданной иерархии классов.*

Код файла main.rb:

```
# frozen_string_literal: true
# Ill class
class Ill
  def initialize(surname, age)
    @surname = surname
    @age = age
  end
  def write
    puts 'Фамилия больного - ' + @surname
    puts 'Возраст - ' + @age.to_s
  end
end
# IllUpgrade class
class IllUpgrade < Ill
  def initialize(surname, age, year)
    super(surname, age)
    @year = year.to_i
  end
  def determine
    puts 'Год следующей диспансеризации - ' + (@year + 3).to_s
  end
end
```

Код файла test.rb:

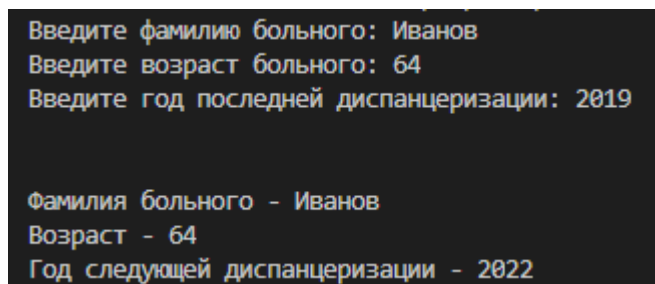
```
# frozen_string_literal: true
require 'minitest/autorun'
require './main.rb'
# Test class
class Test < MiniTest::Unit::TestCase
  def test_1
    surname = 'Ivanov'
    age = 57
    year = 2020
    ill = IllUpgrade.new(surname, age, year)
    assert(ill.is_a?(Ill))
    assert(ill.instance_of?(IllUpgrade))
  end
  def test_2
    surname = 'Petrov'
    age = 60
    ill = Ill.new(surname, age)
    assert(ill.instance_of?(Ill))
  end
end
```

Продолжение кода файла test.rb:

```
end  
end
```

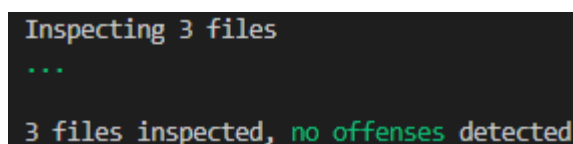
Код файла user.rb:

```
# frozen_string_literal: true  
require './main.rb'  
if Gem.win_platform?  
  Encoding.default_external = Encoding.find(Encoding.locale_charmap)  
  Encoding.default_internal = __ENCODING__  
  [STDIN, STDOUT].each do |io|  
    io.set_encoding(Encoding.default_external, Encoding.default_internal)  
  end  
end  
print 'Введите фамилию больного: '  
surname = gets.chomp  
print 'Введите возраст больного: '  
age = gets.chomp  
print 'Введите год последней диспансеризации: '  
year = gets.chomp  
puts "\n\n"  
ill = IllUpgrade.new(surname, age, year)  
ill.write  
ill.determine
```



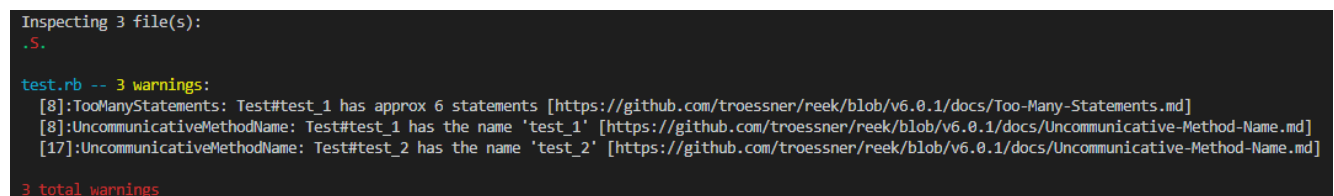
```
Введите фамилию больного: Иванов  
Введите возраст больного: 64  
Введите год последней диспансеризации: 2019  
  
Фамилия больного - Иванов  
Возраст - 64  
Год следующей диспансеризации - 2022
```

Рисунок 5 «Результаты выполнения программы»



```
Inspecting 3 files  
...  
3 files inspected, no offenses detected
```

Рисунок 6 «Результаты проверки Rubosop»



```
Inspecting 3 file(s):  
.S.  
  
test.rb -- 3 warnings:  
[8]:TooManyStatements: Test#test_1 has approx 6 statements [https://github.com/troessner/reek/blob/v6.0.1/docs/Too-Many-Statements.md]  
[8]:UncommunicativeMethodName: Test#test_1 has the name 'test_1' [https://github.com/troessner/reek/blob/v6.0.1/docs/Uncommunicative-Method-Name.md]  
[17]:UncommunicativeMethodName: Test#test_2 has the name 'test_2' [https://github.com/troessner/reek/blob/v6.0.1/docs/Uncommunicative-Method-Name.md]  
  
3 total warnings
```

Рисунок 7 «Результаты проверки Reek»

## Вывод

В ходе данной лабораторной работы были изучены способы работы с файлами и классами в Ruby.