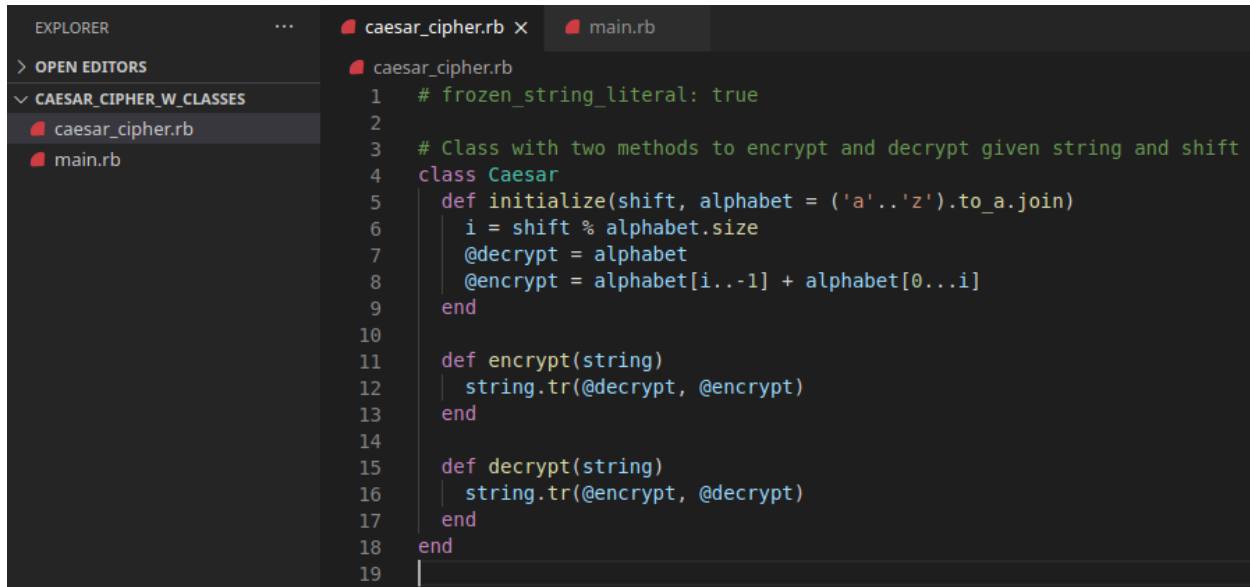


TASK 3

Caesar cipher

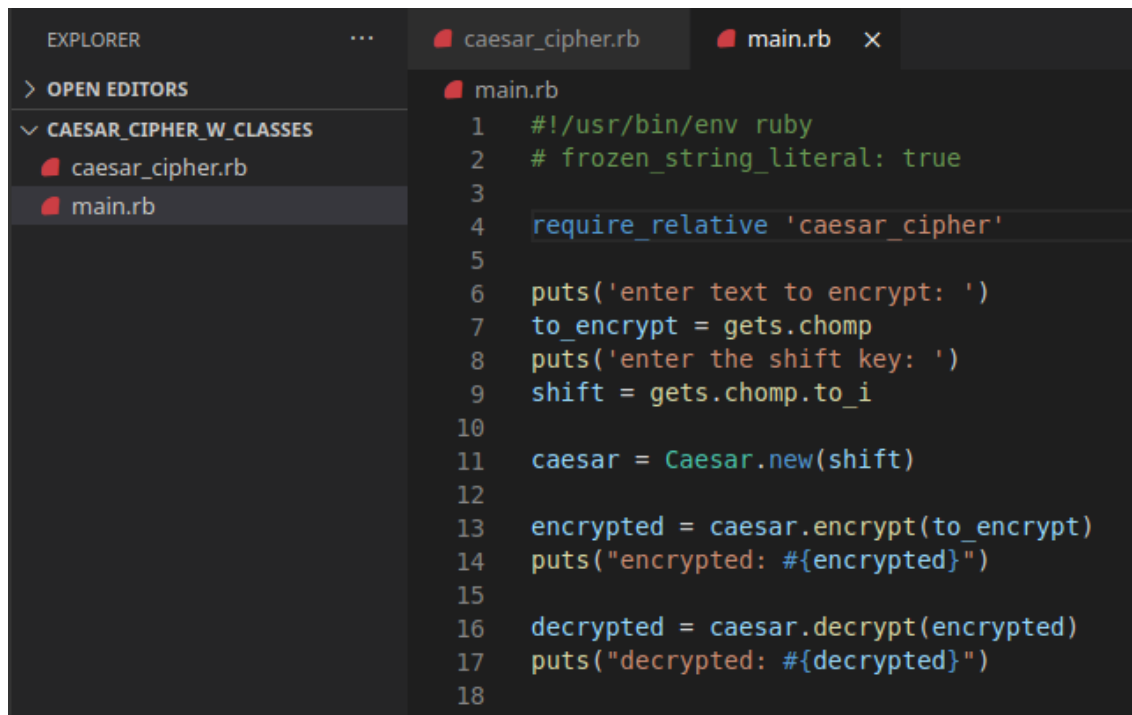
After rewriting Task2 Caesar cipher according to Object-Oriented programming paradigm:



```
EXPLORER  ...  caesar_cipher.rb x  main.rb
> OPEN EDITORS
v CAESAR_CIPHER_W_CLASSES
  caesar_cipher.rb
  main.rb

caesar_cipher.rb
1  # frozen_string_literal: true
2
3  # Class with two methods to encrypt and decrypt given string and shift
4  class Caesar
5    def initialize(shift, alphabet = ('a'..'z').to_a.join)
6      i = shift % alphabet.size
7      @decrypt = alphabet
8      @encrypt = alphabet[i..-1] + alphabet[0...i]
9    end
10
11    def encrypt(string)
12      string.tr(@decrypt, @encrypt)
13    end
14
15    def decrypt(string)
16      string.tr(@encrypt, @decrypt)
17    end
18  end
19
```

Figure 1 – Caesar cipher program using OOP paradigm



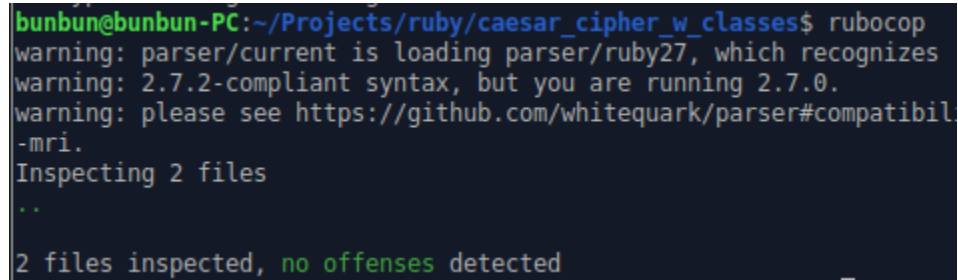
```
EXPLORER  ...  caesar_cipher.rb  main.rb x
> OPEN EDITORS
v CAESAR_CIPHER_W_CLASSES
  caesar_cipher.rb
  main.rb

main.rb
1  #!/usr/bin/env ruby
2  # frozen_string_literal: true
3
4  require_relative 'caesar_cipher'
5
6  puts('enter text to encrypt: ')
7  to_encrypt = gets.chomp
8  puts('enter the shift key: ')
9  shift = gets.chomp.to_i
10
11  caesar = Caesar.new(shift)
12
13  encrypted = caesar.encrypt(to_encrypt)
14  puts("encrypted: #{encrypted}")
15
16  decrypted = caesar.decrypt(encrypted)
17  puts("decrypted: #{decrypted}")
18
```

Figure 2 - main file for output

The encryption, decryption and alphabet values are now initialized inside the class.
Main.rb file is for output only with the import for the cipher class.

Fixed mistakes with rubocop, no offenses detected.

A terminal window screenshot showing the execution of the rubocop command. The prompt is 'bunbun@bunbun-PC:~/Projects/ruby/caesar_cipher_w_classes\$'. The output includes three warnings about the parser and Ruby version, followed by 'Inspecting 2 files', a progress indicator '..', and finally '2 files inspected, no offenses detected'.

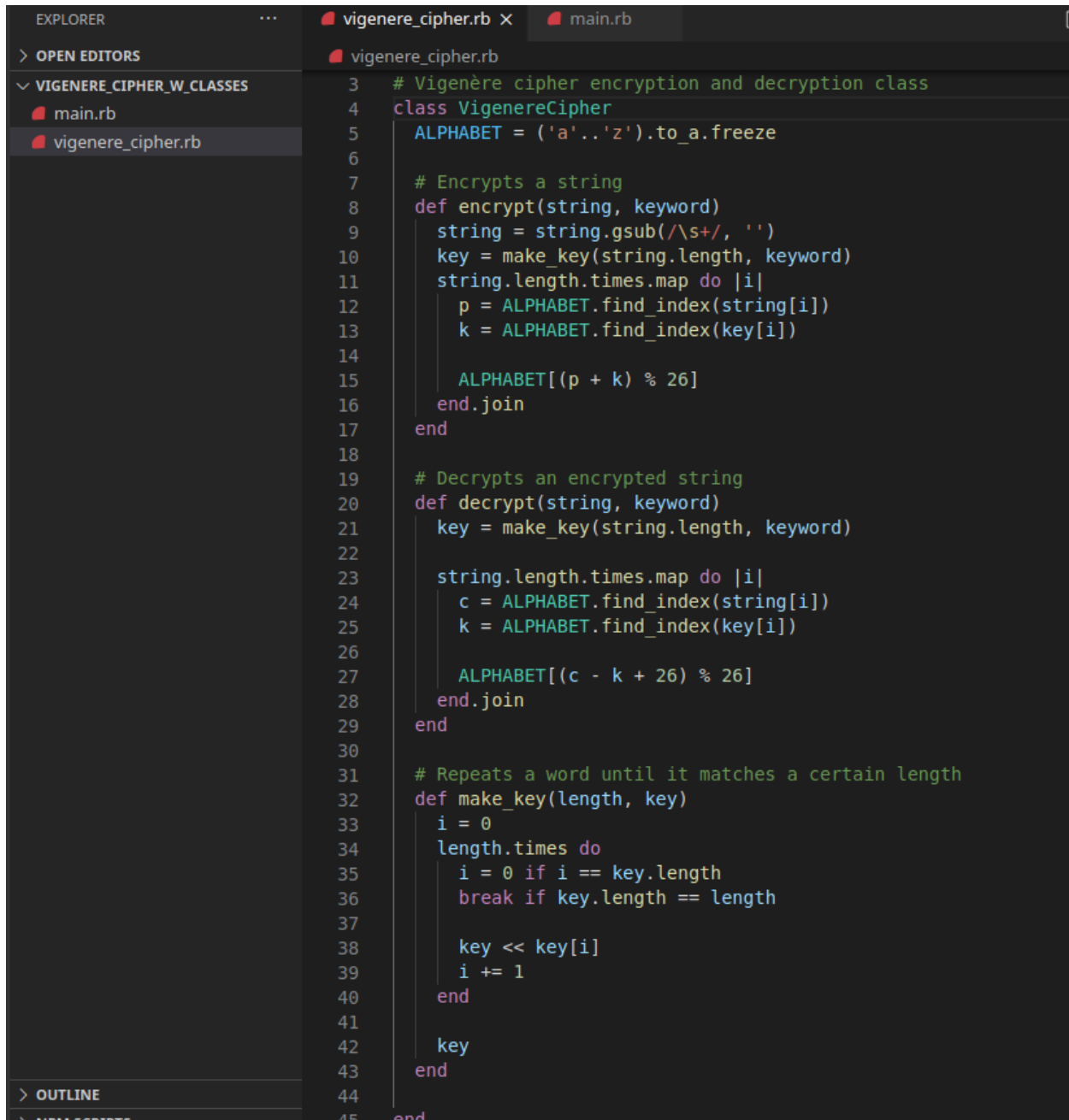
```
bunbun@bunbun-PC:~/Projects/ruby/caesar_cipher_w_classes$ rubocop
warning: parser/current is loading parser/ruby27, which recognizes
warning: 2.7.2-compliant syntax, but you are running 2.7.0.
warning: please see https://github.com/whitequark/parser#compatibil
-mri.
Inspecting 2 files
..
2 files inspected, no offenses detected
```

Figure 3 - rubocop result

The outputs are the same as in task 2.

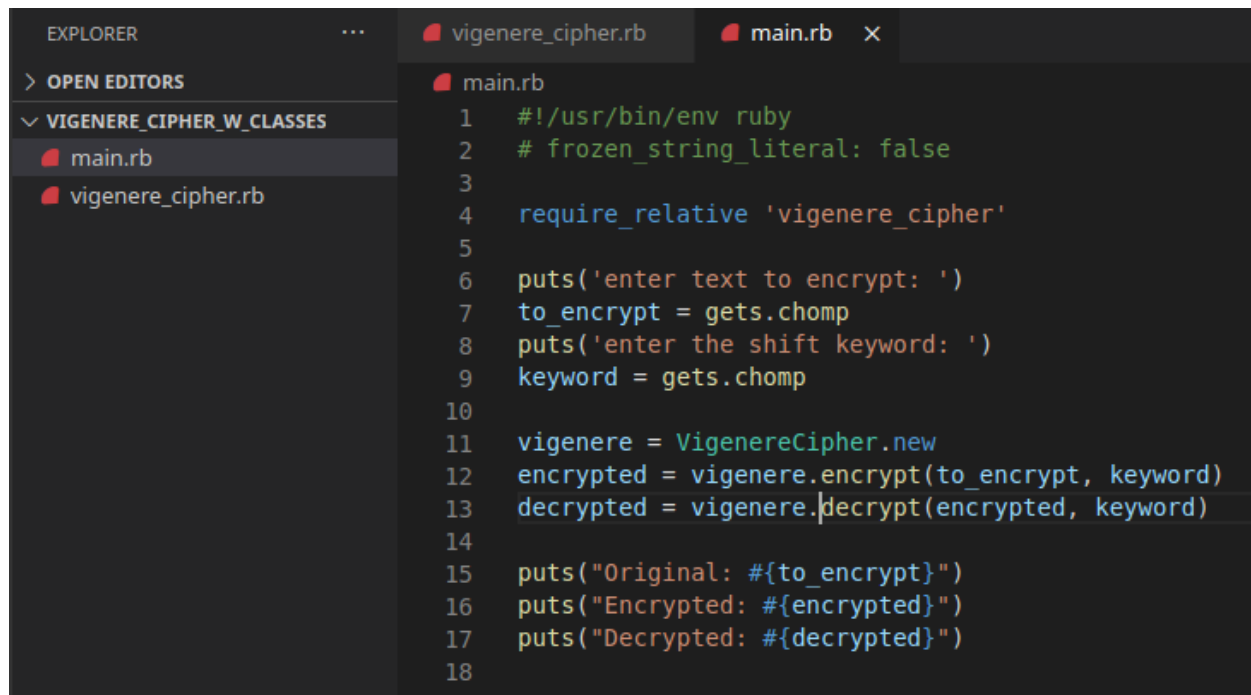
Vigenere cipher

After rewriting task 2 Vigenere cipher according to Object-Oriented programming paradigm:



```
EXPLORER  ...  vigenere_cipher.rb x  main.rb
> OPEN EDITORS
  VIGENERE_CIPHER_W_CLASSES
    main.rb
    vigenere_cipher.rb
  vigenere_cipher.rb
3  # Vigenère cipher encryption and decryption class
4  class VigenereCipher
5    ALPHABET = ('a'..'z').to_a.freeze
6
7    # Encrypts a string
8    def encrypt(string, keyword)
9      string = string.gsub(/\s+/, '')
10     key = make_key(string.length, keyword)
11     string.length.times.map do |i|
12       p = ALPHABET.find_index(string[i])
13       k = ALPHABET.find_index(key[i])
14
15       ALPHABET[(p + k) % 26]
16     end.join
17   end
18
19   # Decrypts an encrypted string
20   def decrypt(string, keyword)
21     key = make_key(string.length, keyword)
22
23     string.length.times.map do |i|
24       c = ALPHABET.find_index(string[i])
25       k = ALPHABET.find_index(key[i])
26
27       ALPHABET[(c - k + 26) % 26]
28     end.join
29   end
30
31   # Repeats a word until it matches a certain length
32   def make_key(length, key)
33     i = 0
34     length.times do
35       i = 0 if i == key.length
36       break if key.length == length
37
38       key << key[i]
39       i += 1
40     end
41
42     key
43   end
44
45 end
```

Figure 4 – Vigenere cipher program using OOP paradigm

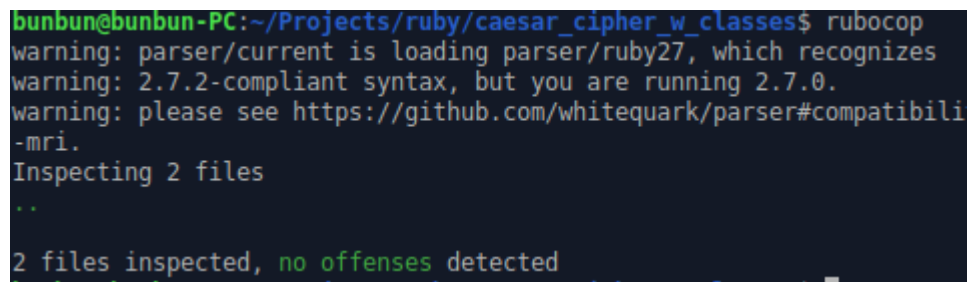


```
1  #!/usr/bin/env ruby
2  # frozen_string_literal: false
3
4  require_relative 'vigenere_cipher'
5
6  puts('enter text to encrypt: ')
7  to_encrypt = gets.chomp
8  puts('enter the shift keyword: ')
9  keyword = gets.chomp
10
11  vigenere = VigenereCipher.new
12  encrypted = vigenere.encrypt(to_encrypt, keyword)
13  decrypted = vigenere.decrypt(encrypted, keyword)
14
15  puts("Original: #{to_encrypt}")
16  puts("Encrypted: #{encrypted}")
17  puts("Decrypted: #{decrypted}")
18
```

Figure 5 - main file for output

Everything works in the same way as in task 2, the only difference is that all of the methods are in a class now and it is called from the main file where the output is handled.

Fixed mistakes with rubocop, no offenses detected.



```
bunbun@bunbun-PC:~/Projects/ruby/caesar_cipher_w_classes$ rubocop
warning: parser/current is loading parser/ruby27, which recognizes
warning: 2.7.2-compliant syntax, but you are running 2.7.0.
warning: please see https://github.com/whitequark/parser#compatibili
-mri.
Inspecting 2 files
..
2 files inspected, no offenses detected
```

Figure 6 - rubocop result

The outputs are the same as in task 2.