Ruby best practice to use double ‘

Strings – Other useful forms of declaration

```
myString = %&This is my String&

myString = %(This is my String)
myString = %[This is my String]
myString = %{This is my String}
```

Ruby docs

<<DOC
Este es un string
DOC

Strings – DOC

```
myText = <<DOC
Please Detach and return this coupon with your payment.
Do not send cash or coins.

Please write your name and account number on the check and
make checks payable to:

        Acme Corporation

Thank you for your business.
DOC
```

Concatenation

Strings – Concatenation

```
myString = "Welcome " + "to " + "Ruby!"
=> "Welcome to Ruby!"
```

```
myString = "Welcome " "to " "Ruby!"
=> "Welcome to Ruby!"
```

Strings – Concatenation

```
myString = "Welcome " << "to " << "Ruby!"
=> "Welcome to Ruby!"
```

```
myString = "Welcome ".concat("to ").concat("Ruby!")
=> "Welcome to Ruby!"
```

Strings – Accessing characters at

```
myString[3].chr
=> "c"
```

That still will be of class string
Using char returns only one character

```
 => "hell"
3.0.0 :007 > "hello"[0, 4].chr
 => "h"
3.0.0 :008 >
```

Substring
my_string[11, 4] -> first the index, then the length

Strings – Spaceship operator

```
"Apples" <=> "Apples"
=> 0


"Apples" <=> "Pears"
=> -1


"Pears" <=> "Apples"
=> 1
```

Compares type and content of it.
If there are no compatible, it returns nil.

In case of strings, it check which one is smaller

Comparing Strings

The spaceship operator can also compare strings. This is where a lot of people get tripped up. However, the important thing to remember is that the operator compares strings in ASCII order. So:

```
'abcdeeeeez' <=> 'cba'

=> -1
```

Changing string content

Strings – Changing a part of a string

```
myString = "Welcome to JavaScript!"

myString["JavaScript"]= "Ruby"
```

Only changes the first one
Also you can use
string[10] = “ruby”

```
myString.gsub("PHP", "Ruby")
=> "Welcome to Ruby Essentials!"
```

Strings – Gsub and Regex

```
1.  "a1".gsub(/\d/, "2")
2.
3.  # "a2"
```

REGEX

https://regex101.com/
https://regexr.com/

No nos pide ser master en regex pero si familiarizados.

Tarea
hacer ruby doc

después usar gsup con correo electrónico
cambiar a que el el primer carcater este bien el resto censurado, sino
cambiar una parte del email por otra palabra