(/)

You just released the advanced tasks of this project. Have fun!

0x06. Regular expression

Regex   **DevOps**

👤 By: Sylvain Kalache

⚙ Weight: 1

📅 Project will start Jan 3, 2024 6:00 AM, must end by Jan 5, 2024 6:00 AM

✓ Checker was released at Jan 3, 2024 6:00 PM

☑ An auto review will be launched at the deadline

Concepts

For this project, we expect you to look at this concept:

- Regular Expression (/concepts/29)

Background Context

For this project, you have to build your regular expression using Oniguruma, a regular expression library that which is used by Ruby by default. Note that other regular expression libraries sometimes have different properties.

Because the focus of this exercise is to play with regular expressions (regex), here is the Ruby code that you should use, just replace the regexp part, meaning the code in between the `//` :

🔍

Help

```
sylvain@ubuntu$ cat example.rb
#!/usr/bin/env ruby
puts ARGV[0].scan(/127.0.0.[0-9]/).join
sylvain@ubuntu$
sylvain@ubuntu$ ./example.rb 127.0.0.2
127.0.0.2
sylvain@ubuntu$ ./example.rb 127.0.0.1
127.0.0.1
sylvain@ubuntu$ ./example.rb 127.0.0.a
```

Resources

Read or watch:

- Regular expressions - basics (/rltoken/6VeaVMauglxcFAwA27TBdQ)
- Regular expressions - advanced (/rltoken/rntjh3-3S86zt0Qy28L10w)
- Rubular is your best friend (/rltoken/RGkVuw1IZ_hoCCbLsiOAhg)
- Use a regular expression against a problem: now you have 2 problems (/rltoken/Vwm8lpMUGa4x_FBtIyUQ8g)
- Learn Regular Expressions with simple, interactive exercises (/rltoken/XsQ6rzS1uy-E6bnswUqlKg)

Requirements

General

- Allowed editors: `vi`, `vim`, `emacs`
- All your files will be interpreted on Ubuntu 20.04 LTS
- All your files should end with a new line
- A `README.md` file, at the root of the folder of the project, is mandatory
- All your Bash script files must be executable
- The first line of all your Bash scripts should be exactly `#!/usr/bin/env ruby`
- All your regex must be built for the Oniguruma library

Quiz questions

Great! You've completed the quiz successfully! Keep going! (Show quiz)

Tasks 🔍

0. Simply matching School   mandatory

Your test string:

```
Holberton
holberton
H0lberton
aaahHolberton
Holbertonaaa
```

Wrap words ☑   Show invisibles ☐   Ruby version 2.1.5 ◇

Match result:

```
Holberton
holberton
H0lberton
aaahHolberton
Holbertonaaa
```

Requirements:

- The regular expression must match `School`
- Using the project instructions, create a Ruby script that accepts one argument and pass it to a regular expression matching method

Example:

```
sylvain@ubuntu$ ./0-simply_match_school.rb School | cat -e
School$
sylvain@ubuntu$ ./0-simply_match_school.rb "Best School" | cat -e
School$
sylvain@ubuntu$ ./0-simply_match_school.rb "School Best School" | cat -e
SchoolSchool$
sylvain@ubuntu$ ./0-simply_match_school.rb "Grace Hopper" | cat -e
$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x06-regular_expressions`
- File: `0-simply_match_school.rb`

[☐ Done?]   [Help]   [Check your code]   [>_ Get a sandbox]

1. Repetition Token #0   [mandatory]

Your test string:

```
hbn
hbtn
hbttn
hbtttn
hbttttn
hbtttttn
hbttttttn
```

Wrap words ☑   Show invisibles ☐   Ruby version 2.1.5 ◇

Match result:

```
hbn
hbtn
hbttn
hbtttn
hbttttn
hbtttttn
hbttttttn
```

Requirements:

- Find the regular expression that will match the above cases
- Using the project instructions, create a Ruby script that accepts one argument and pass it to a regular expression matching method

Repo:
(7)

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x06-regular_expressions`
- File: `1-repetition_token_0.rb`

[ ] Done?   Help   Check your code   >_ Get a sandbox

2. Repetition Token #1   mandatory

Your test string:

```
htn
hbtn
hbbtn
hbbbtn
```

Wrap words ✓   Show invisibles ☐   Ruby version 2.1.5 ⌄

Match result:

```
htn
hbtn
hbbtn
hbbbtn
```

Requirements:

- Find the regular expression that will match the above cases
- Using the project instructions, create a Ruby script that accepts one argument and pass it to a regular expression matching method

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x06-regular_expressions`
- File: `2-repetition_token_1.rb`

[ ] Done?   Help   Check your code   >_ Get a sandbox

3. Repetition Token #2   mandatory

Your test string:

```
hbn
hbtn
hbttn
hbtttn
hbttttn
```

Wrap words ✓   Show invisibles ☐   Ruby version 2.1.5 ⌄

Match result:

```
hbn
hbtn
hbttn
hbtttn
hbttttn
```

Requirements:

- Find the regular expression that will match the above cases

- Using the project instructions, create a Ruby script that accepts one argument and pass it to a **(/)** regular expression matching method

---

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x06-regular_expressions`
- File: `3-repetition_token_2.rb`

[☐ Done?]   [Help]   [Check your code]   [>_ Get a sandbox]

4. Repetition Token #3   [mandatory]

Requirements:

- Find the regular expression that will match the above cases
- Using the project instructions, create a Ruby script that accepts one argument and pass it to a regular expression matching method
- Your regex should not contain square brackets

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x06-regular_expressions`
- File: `4-repetition_token_3.rb`

[☐ Done?]   [Help]   [Check your code]   [>_ Get a sandbox]

5. Not quite HBTN yet   [mandatory]

Requirements:

- The regular expression must be exactly matching a string that starts with `h` ends with `n` and can have any single character in between
- Using the project instructions, create a Ruby script that accepts one argument and pass it to a regular expression matching method

Example:
(/)

```
sylvain@ubuntu$ ./5-beginning_and_end.rb 'hn' | cat -e
$
sylvain@ubuntu$ ./5-beginning_and_end.rb 'hbn' | cat -e
hbn$
sylvain@ubuntu$ ./5-beginning_and_end.rb 'hbtn' | cat -e
$
sylvain@ubuntu$ ./5-beginning_and_end.rb 'h8n' | cat -e
h8n$
sylvain@ubuntu$
$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x06-regular_expressions`
- File: `5-beginning_and_end.rb`

☐ Done?   Help   Check your code   >_ Get a sandbox

6. Call me maybe   `mandatory`

This task is brought to you by a professional advisor Neha Jain (/rltoken/GqwvXAvTXR_JXqyTvZ4AzQ), Senior Software Engineer at LinkedIn.

Requirement:

- The regular expression must match a 10 digit phone number

Example:

```
sylvain@ubuntu$ ./6-phone_number.rb 4155049898 | cat -e
4155049898$
sylvain@ubuntu$ ./6-phone_number.rb " 4155049898" | cat -e
$
sylvain@ubuntu$ ./6-phone_number.rb "415 504 9898" | cat -e
$
sylvain@ubuntu$ ./6-phone_number.rb "415-504-9898" | cat -e
$
sylvain@ubuntu$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x06-regular_expressions`
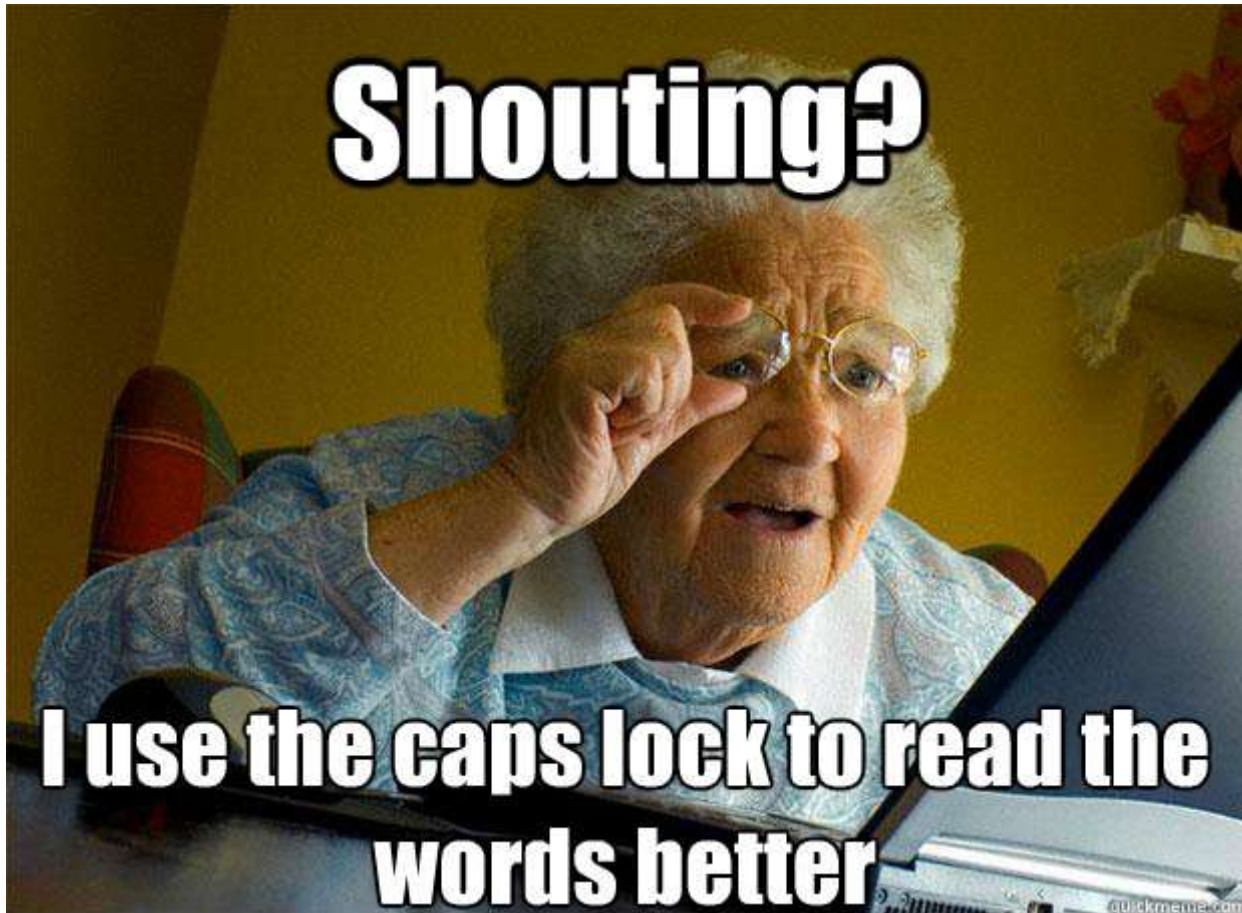- File: `6-phone_number.rb`

(7)Done? | Help | Check your code | >_ Get a sandbox

7. OMG WHY ARE YOU SHOUTING?

`mandatory`

Requirement:

- The regular expression must be only matching: capital letters

Example:

```
sylvain@ubuntu$ ./7-OMG_WHY_ARE_YOU_SHOUTING.rb "I realLy hOpe VancouvEr posseSs Yummy Soft
vAnilla Dupper Mint Ice Nutella cream" | cat -e
ILOVESYSADMIN$
sylvain@ubuntu$ ./7-OMG_WHY_ARE_YOU_SHOUTING.rb "WHAT do you SAY?" | cat -e
WHATSAY$
sylvain@ubuntu$ ./7-OMG_WHY_ARE_YOU_SHOUTING.rb "cannot read you" | cat -e
$
sylvain@ubuntu$
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x06-regular_expressions`
- File: `7-OMG_WHY_ARE_YOU_SHOUTING.rb`

(/)Done?   Help   Check your code   >_ Get a sandbox

---

8. Textme   #advanced

This exercise was prepared for you by Guillaume Plessis (/rltoken/q-XxV5dl95Qn0BBR7vvg4g), VP of Infrastructure at TextMe. It is something he uses daily. You can thank Guillaume for his project on Twitter (/rltoken/Vz-iLl3_Of797s9TsjCxJQ).

For this task, you'll be taking over Guillaume's responsibilities: one afternoon, a TextMe VoIP Engineer comes to you and explains she wants to run some statistics on the TextMe app text messages transactions.

Requirements:

- Your script should output: `[SENDER],[RECEIVER],[FLAGS]`
    - The sender phone number or name (including country code if present)
    - The receiver phone number or name (including country code if present)
    - The flags that were used

You can find a [log file here] (http://intranet-projects-files.s3.amazonaws.com/holbertonschool-sysadmin_devops/78/text_messages.log).

Example:

```
$ ./100-textme.rb 'Feb 1 11:00:00 ip-10-0-0-11 mdr: 2016-02-01 11:00:00 Receive SMS [SMSC:SY
BASE1] [SVC:] [ACT:] [BINF:] [FID:] [from:Google] [to:+16474951758] [flags:-1:0:-1:0:-1] [ms
g:127:This planet has - or rather had - a problem, which was this: most of the people on it
were unhappy for pretty much of the time.] [udh:0:]'
Google,+16474951758,-1:0:-1:0:-1
$
$
$ ./100-textme.rb 'Feb 1 11:00:00 ip-10-0-64-10 mdr: 2016-02-01 11:00:00 Receive SMS [SMSC:S
YBASE2] [SVC:] [ACT:] [BINF:] [FID:] [from:+17272713208] [to:+19172319348] [flags:-1:0:-1:0:
-1] [msg:136:Orbiting this at a distance of roughly ninety-two million miles is an utterly i
nsignificant little blue green planet whose ape-descended] [udh:0:]'
+17272713208,+19172319348,-1:0:-1:0:-1
$
$ ./100-textme.rb 'Feb 1 11:00:00 ip-10-0-64-11 mdr: 2016-02-01 11:00:00 Sent SMS [SMSC:SYBA
SE1] [SVC:backendtextme] [ACT:] [BINF:] [FID:] [from:18572406905] [to:14022180266] [flags:-
1:0:-1:-1:-1] [msg:136:Far out in the uncharted backwaters of the unfashionable end of the w
estern spiral arm of the Galaxy lies a small unregarded yellow sun.] [udh:0:]'
18572406905,14022180266,-1:0:-1:-1:-1
$
$
$ ./100-textme.rb 'Feb 1 11:00:00 ip-10-0-64-11 mdr: 2016-02-01 11:00:00 Sent SMS [SMSC:SYBA
SE1] [SVC:backendtextme] [ACT:] [BINF:] [FID:] [from:12392190384] [to:19148265919] [flags:-
1:0:-1:-1:-1] [msg:99:life forms are so amazingly primitive that they still think digital wa
tches are a pretty neat idea.] [udh:0:]'
12392190384,19148265919,-1:0:-1:-1:-1
$
```

Repo:
(7)

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x06-regular_expressions`
- File: `100-textme.rb`

☐ Done?   Help   Check your code   >_ Get a sandbox