

Expressões Regulares (2)

Aula 12

Curso: BIG863 - Basic Python Programming for Ecologists

Professora: Dra. Cecilia F. Fiorini

Supervisor: Prof. Dr. Fernando A. O. Silveira

<https://meet.google.com/zdi-ueoz-nsr>, 31 de maio de 2023



Roteiro^{*}

1 Prática

2 Exercícios

^{*}Conteúdo adaptado a partir de material desenvolvido pelo Prof. Zanoni Dias e disponível em <https://ic.unicamp.br/mc102>.

Prática

Regexone

Exercício 1

- Escreva um programa para determinar se uma string representa um número (inteiro ou real) válido.
- Exemplos de números válidos:
 - 10
 - +5
 - 3
 - 10.3
 - 0.80
 - 2.8033

Exercício 1 - Resposta

```
1 import re
2 regexp = r'^[+-]?[0-9]+(\.[0-9]+)?$'
3 while True:
4     número = input()
5     if not(número):
6         break
7     if re.search(regexp, número):
8         print("OK")
9     else:
10        print("ERRO")
```

Exercício 2

- Escreva um programa para determinar se uma string representa um número de telefone (fixo ou celular) válido.
- Exemplos de números de telefones válidos:
 - (31) 3123-4567
 - (031) 91234 5678
 - 313123-4567
 - (031)91234 5678
 - (031)3123-4567
 - 031912345678
 - (31)31234567
 - 31 91234 5678
 - 3131234567
 - 3191234 5678

Exercício 2 - Resposta

```
1 import re
2
3 ddd = r'^(0?[1-9]{2}[- ]?|\(0?[0-9]{2}\) ?) '
4 tel = r'[2-9]?[0-9]{4}[- ]?[0-9]{4}$'
5 regexp = ddd + tel
6
7 while True:
8     telefone = input()
9     if not(telefone):
10         break
11     if re.search(regexp, telefone):
12         print("OK")
13     else:
14         print("ERRO")
```

Exercício 3

- Com base no exercício anterior, escreva uma função que recebe como parâmetro uma string que representa um número de telefone (fixo ou celular). Caso o número não seja válido, sua função deve retornar None. Caso contrário, ela deve retornar uma string no formato (XX) XXXX-XXXX (no caso de telefone fixo) ou (XX) XXXXX-XXXX (no caso de telefone celular), onde X representa um dígito do telefone.

Exercício 3 - Resposta

```
1 import re
2
3 def padroniza_telefone(telefone):
4     ddd = r'^ (0?[1-9]{2}[- ]?|\ (0?[0-9]{2}\ ) ?)'
5     tel = r'([2-9]?[0-9]{4})[- ]?([0-9]{4})$'
6     regexp = ddd + tel
7
8     if re.search(regexp, telefone):
9         return None
10
11     digitos = re.sub(r'[^\d]', '', telefone)
12     grupos = r'^0?([0-9]{2})([0-9]{4,5})([0-9]{4})$'
13     formato = r'(\1) \2-\3'
14
15     return re.sub(grupos, formato, digitos)
```