

The background of the image is a blurred screenshot of a code editor. On the left, a file explorer sidebar is visible with several files listed. The main area of the editor shows Ruby code with syntax highlighting. The code includes requirements for 'File.expand\_path', 'spec\_helper', 'rspec/rails', 'capybara/rspec', and 'capybara/rails'. There are also comments in Portuguese, such as '# Prevent database truncation if the environment is production' and '# Requires supporting spec/support/ and its subdirectories'. The text 'DESENVOLVIMENTO COM AUTOMAÇÃO ROBÓTICA DE PROCESSOS - RPA' is overlaid in the center in a large, white, bold font.

# DESENVOLVIMENTO COM AUTOMAÇÃO ROBÓTICA DE PROCESSOS - RPA

## Texto base

# 5

## RegEx

Osvaldo Kotaro Takai & Ana Cristina dos Santos

### *Resumo*

*Aprender a trabalhar com Expressões Regulares para extrair dados de textos que tenham algum padrão. Para tanto, será utilizado o Automation 350 e a Linguagem Python.*

### 5.1. Introdução

O objetivo deste material é aprender a utilizar Expressões Regulares, ou RegEx, para extrair informações de textos que tenham algum padrão que permita identificar as porções de texto desejadas.

Como pano de fundo para aprender a fazer isso, serão descritos os passos de criação de um bot que consiga obter as indicações de livros apresentadas por seus colegas e guardá-las num arquivo texto.

### 5.2. Preparação

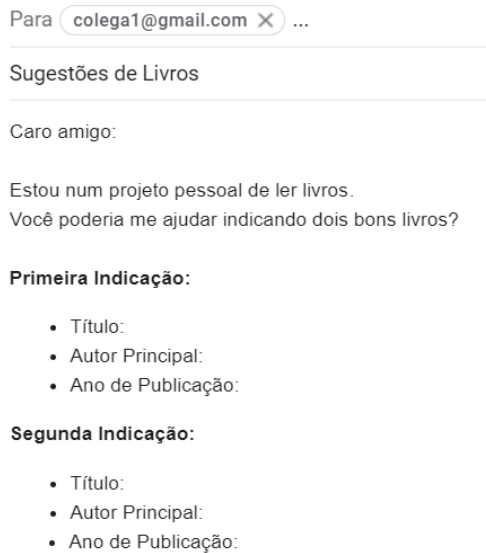
Assume-se aqui que o computador local onde o bot será construído e executado esteja devidamente configurado e com o Python 3 já instalado.

Com a conta do gmail criada anteriormente, deve-se solicitar aos colegas que indiquem dois livros seguindo os campos sugeridos (**Figura 5.1**). Esses campos estabelecem o padrão que será utilizado para extrair as informações dos livros.

Os e-mails retornados terão o aspecto similar ao da **Figura 5.2**. A parte inferior com fundo cinza contém informações desnecessárias para a extração dos dados de livros e, portanto, pode ser eliminada. Para tanto, é necessário encontrar um padrão que possa ser utilizado para dizer: “A partir deste ponto do texto pode ser eliminado”.

A palavra “Em” não pode ser esse padrão, pois pode ter nomes de livros que tenham “Em”. A data também não pode, pois ela é muito específica. O nome “fulano beltrano da silva”, foi escolhido que, neste caso, é o nome de quem pediu as indicações de livros.

**Figura 5.1: Exemplo de e-mail solicitando indicação de livros**



**Fonte: do autor, 2021**

**Figura 5.2: Aspecto dos e-mail retornados**

<p>Caro amigo:</p> <p>Seguem as minhas indicações:</p> <p>*Primeira Indicação:*</p> <ul style="list-style-type: none"><li>- Título: <b>*Pipeline de liderança*</b></li><li>- Autor Principal: <b>*CHARAN, Ram*</b></li><li>- Ano de Publicação: <b>*2018*</b></li></ul> <p>*Segunda Indicação:*</p> <ul style="list-style-type: none"><li>- Título: <b>Organizações exponenciais: Porque elas são 10 vezes melhores, mais rápidas e mais baratas que a sua (e o que fazer a respeito)</b></li><li>- Autor Principal: <b>ISMAIL, S</b></li><li>- Ano de Publicação: <b>2018</b></li></ul>
<p>Em dom., 15 de dez. de 2019 às 10:52, fulano beltrano da silva &lt;fulanob.silva.rpa@gmail.com&gt; escreveu:</p> <p>&gt; Caro amigo:</p> <p>&gt;</p> <p>&gt; Estou num projeto pessoal de ler livros em 2020.</p> <p>&gt; Você poderia me ajudar indicando dois bons livros?</p> <p>&gt;</p> <p>&gt; *Primeira Indicação:*</p> <p>&gt;</p> <p>&gt; - Título:</p> <p>&gt; - Autor Principal:</p> <p>&gt; - Ano de Publicação:</p> <p>&gt;</p> <p>&gt; *Segunda Indicação:*</p> <p>&gt;</p> <p>&gt; - Título:</p> <p>&gt; - Autor Principal:</p> <p>&gt; - Ano de Publicação:</p>

**Fonte: do autor, 2021**

O RegEx para fazer isso é:

(fulano beltrano da silva(.|\s)\*)

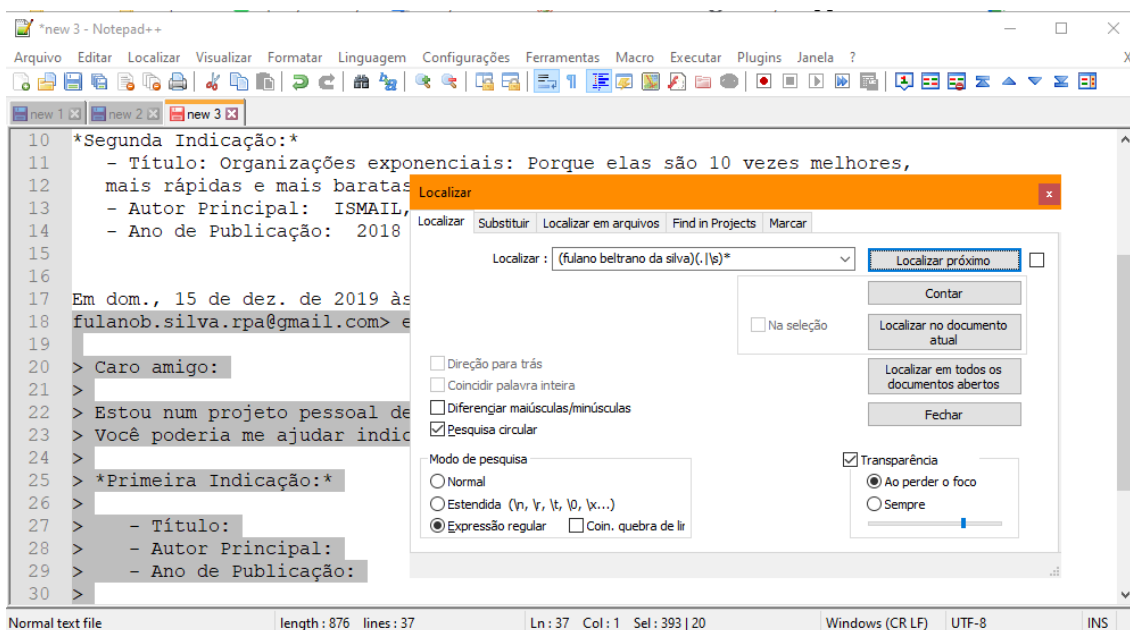
A expressão regular acima diz o seguinte:

Busque no texto “fulano beltrano da silva” seguido de qualquer caracter “(.|\s)\*”.

- O ponto, “.” indica qualquer caractere normal.
- O “\s” indica caracteres em branco.
- O “\*” indica zero ou mais vezes o que estiver entre parênteses.
- O “|” indica ou.
- Assim, “(fulano beltrano da silva(.|\s)\*)” define o texto que tenha “fulano beltrano da silva” seguido de zero ou mais caracteres normais ou branco.

Os caracteres encontrados devem ser substituídos por uma string vazia “”. Pode-se testar se essa expressão regular funciona como esperado usando o notepad++ (NOTEPAD++, 2021). Para tanto, o texto da **Figura 5.2** pode ser copiado e colado no notepad++ e realizar a operação de Localizar conforme a **Figura 5.3**.

Figura 5.3: Notepad++ para testar a expressão regular



Fonte: do autor, 2021

Esse mesmo teste pode ser feito no site <https://regexr.com> utilizando o browser FireFox (Figura 5.4); é possível que em outros navegadores o site não funcione.

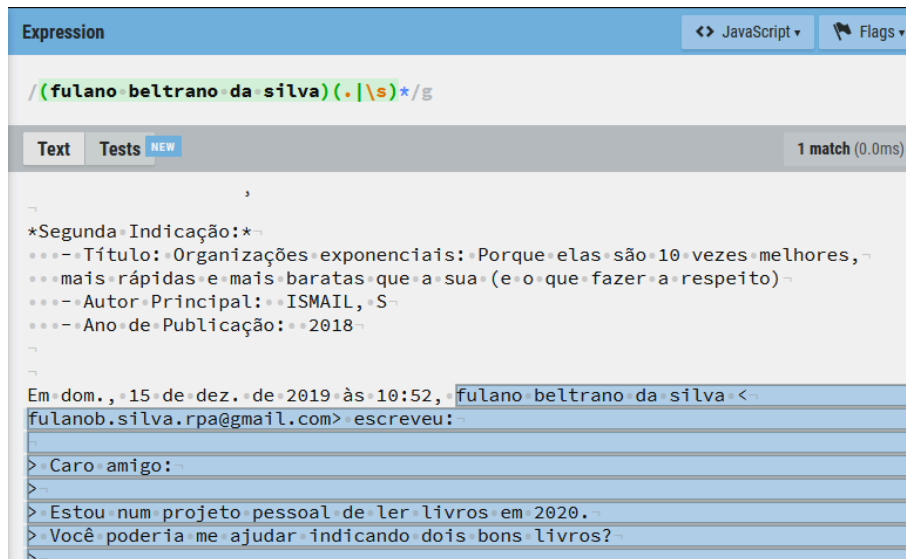
Agora, do texto que sobrou, deseja-se eliminar todos os caracteres de mudança de linhas “\r” (*carriage return*) e “\n” (*line feed*) para conseguir extrair títulos longos e que possuam mudanças de linhas, como no título da segunda indicação:

**Organizações exponenciais: Porque elas são 10 vezes melhores,  
mais rápidas e mais baratas que a sua (e o que fazer a respeito)**

Deseja-se, também, eliminar os “\*” para extrair somente os dados dos campos necessários.

Figura 5.4 - Site para testar expressões regulares no browser Firefox





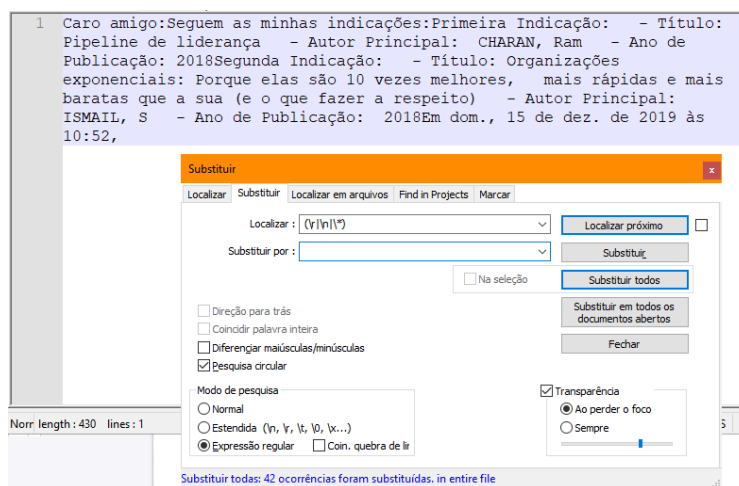
Fonte: do autor, 2021

O RegEx para fazer isso é:

`(\r|\n|\\*)`

O “\\*” indica o caracteres “\*” para não confundir com a indicação de zero ou mais vezes. Os caracteres encontrados serão substituídos por uma string vazia “”. O teste dessa expressão regular no notepad++ pode ser vista na **Figura 5.5**.

**Figura 5.5: Teste da expressão regular**



Fonte: do autor, 2021

Deseja-se agora, limpar o texto removendo os múltiplos espaços em branco:

Caro amigo:Seguem as minhas indicações:Primeira Indicação: -  
Título: Pipeline de liderança - Autor Principal: CHARAN, Ram -  
Ano de Publicação: 2018Segunda Indicação: - Título: Organizações

exponenciais: Porque elas são 10 vezes melhores, mais rápidas e  
mais baratas que a sua (e o que fazer a respeito) - Autor Principal:  
ISMAIL, S - Ano de Publicação: 2018 Em dom., 15 de dez. de 2019  
às 10:52,

O RegEx para isso é:

`\s+`

- “\s” indica espaço em branco.
- “+” indica uma ou mais vezes.

O texto encontrado será substituído por um caractere branco “ ”:

Caro amigo: Seguem as minhas indicações: Primeira Indicação: -  
Título: Pipeline de liderança - Autor Principal: CHARAN, Ram - Ano  
de Publicação: 2018 Segunda Indicação: - Título: Organizações  
exponenciais: Porque elas são 10 vezes melhores, mais rápidas e  
mais baratas que a sua (e o que fazer a respeito) - Autor Principal:  
ISMAIL, S - Ano de Publicação: 2018 Em dom., 15 de dez. de 2019  
às 10:52,

Recomenda-se que realize o teste no notepad++ ou no Firefox.

Finalmente chegou a hora de extrair todos os títulos encontrados no texto. O  
RegEx para isso é:

`Título:\s*(.+?)\s*-\sAutor`

Ou seja:

- Buscar por todas as ocorrências de “Título:”;
- Seguidos por zero ou mais caracteres em branco: “\s\*”;
- Seguidos por um ou mais caracteres do título propriamente dito: “(.+?)”;
  - “?” Indica *lazy* ao invés de *greedy*.
- Finalizado por “\s\*-\sAutor”.

Esta expressão selecionará apenas os títulos conforme pode ser verificado no texto  
em **bold** abaixo:

Caro amigo: Seguem as minhas indicações: Primeira Indicação: -  
**Título: Pipeline de liderança - Autor Principal: CHARAN, Ram - Ano**  
**de Publicação: 2018 Segunda Indicação: - Título: Organizações**  
**exponenciais: Porque elas são 10 vezes melhores, mais rápidas e**  
**mais baratas que a sua (e o que fazer a respeito) - Autor Principal:**  
**ISMAIL, S - Ano de Publicação: 2018 Em dom., 15 de dez. de 2019**  
**às 10:52,**

Caso o *lazy* (preguiçoso), “?”, não tivesse sido especificado, a seleção seria *greedy*  
(gulosa) e selecionaria todo o texto do primeiro “Título:” até o último “Autor”:

Caro amigo: Seguem as minhas indicações: Primeira Indicação: -  
**Título: Pipeline de liderança - Autor Principal: CHARAN, Ram -**  
**Ano de Publicação: 2018 Segunda Indicação: - Título: Organizações**  
**exponenciais: Porque elas são 10 vezes melhores, mais rápidas e**

**mais baratas que a sua (e o que fazer a respeito) - Autor Principal:**  
ISMAIL, S - Ano de Publicação: 2018Em dom., 15 de dez. de 2019  
às 10:52,

A mesma estratégia para extrair “Títulos” pode ser utilizada para extrair o “Autor Principal”. O RegEx para isso é:

Autor Principal:\s\*(.+?)\s\*-\sAno

Ou seja:

- Buscar por todas as ocorrências de “Autor Principal:”;
- Seguidos por zero ou mais caracteres em branco: “\s\*”;
- Seguidos por um ou mais caracteres que conterá o autor principal: “(.+?)”;
  - “?” Indica *lazy* ao invés de *greedy*.
- Finalizado por “\s\*-\sAno”.

Esta expressão regular resultará na seleção abaixo:

Caro amigo:Seguem as minhas indicações:Primeira Indicação: -  
Título: Pipeline de liderança - **Autor Principal: CHARAN, Ram - Ano**  
de Publicação: 2018Segunda Indicação: - Título: Organizações  
exponenciais: Porque elas são 10 vezes melhores, mais rápidas e  
mais baratas que a sua (e o que fazer a respeito) - **Autor Principal:**  
**ISMAIL, S - Ano de Publicação: 2018**Em dom., 15 de dez. de 2019  
às 10:52,

Para extrair os anos de publicação pode-se utilizar a seguinte RegEx:

Ano de Publicação:\s\*(\d{4})

Ou seja:

- Buscar por todas as ocorrências de “Ano de Publicação:”;
- Seguidos por zero ou mais caracteres em branco: “\s\*”;
- Seguidos por quatro dígitos que conterá o ano de publicação: “(\d{4})”.

Esta expressão regular resultará na seleção abaixo:

Caro amigo:Seguem as minhas indicações:Primeira Indicação: -  
Título: Pipeline de liderança - Autor Principal: CHARAN, Ram - **Ano**  
**de Publicação: 2018**Segunda Indicação: - Título: Organizações  
exponenciais: Porque elas são 10 vezes melhores, mais rápidas e  
mais baratas que a sua (e o que fazer a respeito) - Autor Principal:  
ISMAIL, S - **Ano de Publicação: 2018**Em dom., 15 de dez. de 2019  
às 10:52,

### 5.3. Criação de 3 bots

Agora, com o entendimento dos RegEx necessários, pode-se criar os bots. O primeiro bot, Bot 1, guarda os e-mails com as sugestões de livros como um arquivo texto num diretório previamente especificado.



O segundo bot, Bot 2, processa esses arquivos textos e extrai as indicações de livros no seguinte formato:

Título/tAutor Principal/tAno/n

Onde:

- “\t” representa o caractere de tabulação (TAB)
- “\n” representa o caractere line feed (ENTER)

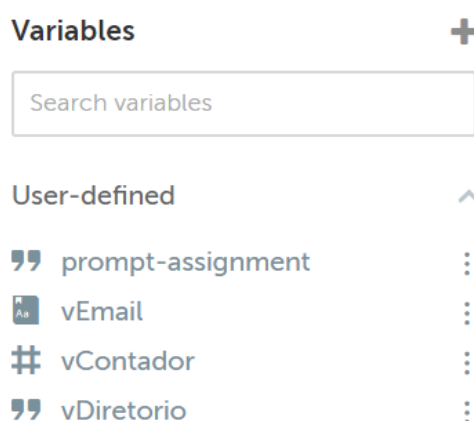
O terceiro bot executa Bot 1 e depois Bot 2.

### 5.3.1 Bot 1

#### 5.3.1.1 Variáveis

Este bot precisa utilizar as variáveis descritas na **Figura 5.6**:

**Figura 5.6: Variáveis utilizadas no Bot 1**



**Fonte: do autor, 2021**

A variável vEmail é usada para armazenar os conteúdos de um e-mail (**Figura 5.7**). A variável é do tipo dicionário, o que permite acessar cada parte de um e-mail a partir de índices previamente definidos conforme trabalhado na Parte 3 da Unidade 1.

**Figura 5.7: Configuração da variável vEmail**

The screenshot shows the 'Edit variable' dialog box. At the top right are 'Cancel' and 'Apply' buttons. The 'Type' dropdown is set to 'Dictionary' and the 'Subtype' dropdown is set to 'String'. The 'Name' field contains 'vEmail' with a 'Max characters = 50' label below it. The 'Description (optional)' field contains 'An unordered group of key/value pairs' with a 'Max characters = 255' label below it. There are three checkboxes: 'Use as input' (unchecked), 'Use as output' (unchecked), and 'Constant (read-only)' (unchecked). The 'Default value (optional)' section shows 'This dictionary is empty' above a light gray bar with a '+' icon.

Fonte: do autor, 2021

A variável vContador é utilizada para numerar o nome dos arquivos textos:

- arqEmail-1.eml para o primeiro e-mail.
- arqEmail-2.eml para o segundo e-mail e assim por diante.

A sua configuração é bastante simples como pode ser observado na **Figura 5.8**. A variável é numérica com valor inicial 0.

**Figura 5.8: Configuração da variável vContador**

The screenshot shows the 'Edit variable' dialog box. At the top right are 'Cancel' and 'Apply' buttons. The 'Type' dropdown is set to 'Number'. The 'Name' field contains 'vContador' with a 'Max characters = 50' label below it. The 'Description (optional)' field contains 'Digits' with a 'Max characters = 255' label below it. There are three checkboxes: 'Use as input' (unchecked), 'Use as output' (unchecked), and 'Constant (read-only)' (unchecked). The 'Default value (optional)' field contains '0' with increment (+) and decrement (-) buttons to its right.

Fonte: do autor, 2021

Por fim, a variável vDiretorio guarda o caminho para o diretório dos arquivos textos (**Figura 5.9**).

A variável é do tipo string. Vale chamar à atenção aqui que o valor padrão desta variável deve ser aquela em que se deseja guardar os arquivos de e-mails lidos. O importante é garantir que esse diretório exista e que o seu caminho esteja completo a partir da raiz do disco.

**Figura 5.9: Configuração da variável vDiretório**

**Edit variable** Cancel Apply

Type  
String

Name  
vDiretorio  
Max characters = 50

Description (optional)  
Text  
Max characters = 255

☐ Use as input

☐ Use as output

☐ Constant (read-only)

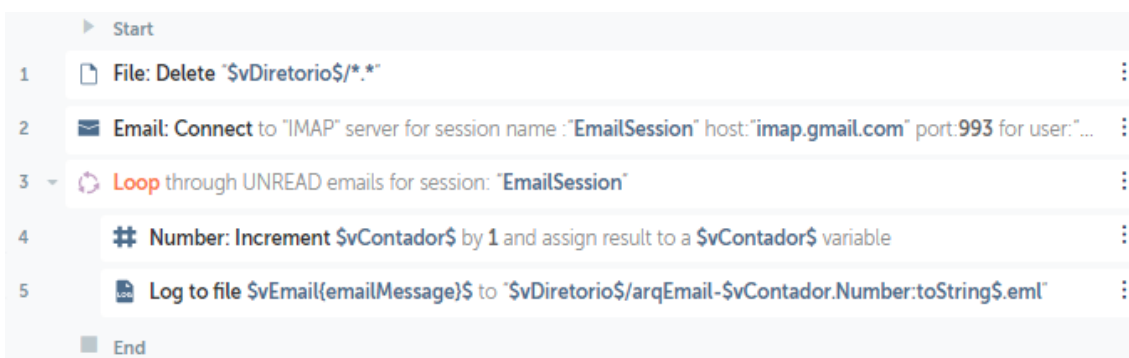
Default value  
C:\Users\USUARIO\Desktop\Curso RPA\07 - RegEx\docs

Fonte: do autor, 2021

### 5.3.1.2 Ações

As ações deste primeiro bot devem ser arrastadas para a sua lista conforme a **Figura 5.10**.

**Figura 5.10: Configuração da lista de ações do Bot 1**



Fonte: do autor, 2021

#### 5.3.1.2.1 Ação File: Delete

O diretório vDiretorio pode conter arquivos de alguma execução anterior desse bot. Para garantir que os arquivos colocados lá sejam da execução atual, é importante remover quaisquer arquivos existentes nesse diretório. Isso pode ser realizado pela ação **File: Delete** (**Figura 5.11**).

É importante observar que a opção Date foi selecionada. O filtro define que os arquivos que serão removidos devem ter sido criados há um dia. Isso foi feito porque a versão atual dessa ação gera erros se não estiver configurada desta forma.

**Figura 5.11: Configuração da ação Ação File: Delete**

**File: Delete**  
Deletes a file

File

e.g. C:\MyDoc\\*.doc

☐ Size

Size (KB)

☒ Date

Created

☒ Is within last (days)

☐ Is between

Start date (MM/DD/YY)

End date (MM/DD/YY)

☐ Is before (MM/DD/YY)

Fonte: do autor, 2021

### 5.3.1.2.2 Ação Email: Connect

As configurações desta ação são exatamente iguais às utilizadas na Parte 3 da Unidade 1 (Figura 5.12).

**Figura 5.12: Configuração da ação Email: Connect**

Session name

e.g. Session1 or S1

Connect to

☐ Outlook

☒ Email server

Host

eg: outlook.office365.com, etc.

Port

eg: 993, 995 etc.

Username

@gmail.com

Password

☒ Use secure connection(SSL/TLS)

Protocol

☒ IMAP

☐ POP3

Fonte: do autor, 2021

### 5.3.1.2.3 Ação Loop

Esta ação permite visitar cada e-mail não lido da caixa de e-mail configurada na ação anterior, **Email: Connect**. A sua configuração pode ser visualizada na **Figura 5.13**.

**Figura 5.13: Configuração da ação Loop**

### Loop

Repeats the actions in a loop until a break

#### Loop Type

☒ Iterator

#### Iterator

Email  
For each mail in mail box

Iterator for each mail in mail box

#### Session name

"" EmailSession (x)

#### Type of email to get

☐ ALL

☐ READ

☒ UNREAD

For POP3 protocol all message will be fetched

#### From a specific folder (optional)

"" Inbox (x)

e.g. Inbox/folder1;Inbox/folder2 or Inbox/test\*. For POP3 fetching from Inbox only

#### When subject contains (optional)

"" "Sugestões de Livros" (x)

e.g. subject1;subject2

#### From specific senders (optional)

"" (x)

e.g. john@abc.com;Mary@xyz.com

#### When received date is on or after (optional)

Choose a variable (x)

#### When received date is before (optional)

Choose a variable (x)

#### Message format

☐ HTML

☒ PLAINTEXT

#### Assign the current value to variable (optional)

Multiple variables Dictionary

vEmail (x)

☐ While

**Fonte: do autor, 2021**

A primeira coisa que pode ser observada nesta configuração é o iterador selecionado: “Email Foreach mail in mail box”. Isso permite que o loop visite cada e-mail da caixa de e-mail configurada com o nome da seção “EmailSession” configurada na ação anterior.

O tipo de e-mail a ser obtido é UNREAD, ou seja, os e-mails não lidos. A caixa de e-mail selecionada é o Inbox, ou seja, a Caixa de Entrada. Esta opção é interessante porque permite selecionar outra caixa que não seja a caixa de entrada caso o usuário do e-mail tenha criado uma pasta para agrupar seus e-mails.

Outro campo interessante é o campo “When subject contains (optional)”. Ela permite especificar quais dos e-mail não lidos serão obtidos. Neste caso, serão somente os e-mail não lidos que contenham no seu campo assunto o seguinte texto: “Sugestões de Livros”. Existem outras opções de filtro que foram deixadas em branco por não ser importante para este bot.

A opção “Message format” foi deixada em PLAINTEXT. Isso significa que o e-mail visitado será obtido como um texto plano, ou seja, sem nenhuma marcação de formatação.

Por fim, o campo “Assign the current value to variable (optional)” permite indicar que a variável que receberá o conteúdo do e-mail visitado será o vEmail que é uma variável do tipo dicionário.

#### 5.3.1.2.4 Ação Number: Increment

Esta ação serve apenas para incrementar o valor da variável vContador (**Figura 5.14**). Relembrando, esta variável foi iniciada com o valor 0 e é utilizada para definir os nomes dos arquivos textos contendo cada um, conteúdos de e-mails lidos.

**Figura 5.14: Configuração da ação Number: Increment**

**Number: Increment**

Increments a number by specified value

Enter number

# \$vContador\$ (x)

Enter increment value

# 1 (x)

Increments number by value (e.g. 1)

---

Assign the output to variable

# vContador (x)

**Fonte: do autor, 2021**

#### 5.3.1.2.5 Ação Log to file

Esta ação permite guardar o conteúdo do e-mail lido no diretório especificado anteriormente (**Figura 5.15**).



**Figura 5.15: Configuração da ação Log to file**

**Log to file**

Logs any text into a file

File path

” ” \$vDiretorio\$/arqEmail-\$vContador.Number.toString\$.eml (x) Browse...

Enter text to log

” ” \$vEmail(emailMessage)\$ (x)

☐ Append timestamp

When logging

☐ Append to existing log file

☒ Overwrite existing log file

Encoding

UTF8

Fonte: do autor, 2021

O campo **File path** contém a seguinte string:

**\$vDiretorio\$/arqEmail-\$vContador.Number.toString\$.eml**

O conteúdo da variável vDiretorio é concatenado com “/arqEmail-” que, por sua vez é concatenado com o conteúdo da variável vContador convertido para String e, por fim, é concatenado com “.eml”.

Ou seja, numa primeira iteração, o nome do arquivo que será:

C:\Users\USUARIO\Desktop\Curso RPA\07 - RegEx\docs\arqEmail-1.eml

## 5.3.2 Bot 2

### 5.3.2.1 Variáveis

Este bot precisará de duas variáveis conforme pode ser visto na **Figura 5.16**.

**Figura 5.16: Variáveis do Bot 2**

**Variables** +

Search variables

User-defined ^

” ” prompt-assignment :

” ” vDiretorio :

Fonte: do autor, 2021

A variável vDiretorio tem a mesma configuração do vDiretorio do Bot 1 (**Figura 5.17**).

**Figura 5.17: Configuração da variável vDiretorio**

**Edit variable** [Cancel] [Apply]

Type: String

Name: vDiretorio (Max characters = 50)

Description (optional): Text (Max characters = 255)

☐ Use as input

☐ Use as output

☐ Constant (read-only)

Default value: C:\Users\USUARIO\Desktop\Curso RPA\07 - RegEx\docs

**Fonte: do autor, 2021**

Essa variável guarda o caminho completo para o diretório dos arquivos textos:

- Conteúdos dos e-mails (criado pelo Bot 1) e
- Conteúdo das indicações (que será criado por este bot, Bot 2).

### 5.3.2.2 Python script: Open

Esta ação define em Python, o código que visita cada arquivo **arqEmail-\*.eml** no diretório vDiretorio e extrai os Títulos, Autores Principais e Anos de Publicação dos livros indicados. O script em Python pode ser vista na **Figura 5.18**.

A linha 1 especifica que o Script em Python irá trabalhar com a codificação utf-8 (MCINGVALE, 2007).

A linha 2 realiza a importação da biblioteca que permite utilizar as funções para trabalhar com Expressões Regulares, RegEx.

A linha 3 realiza a importação da biblioteca que contém funções que permitem manipular arquivos no sistema operacional.

A explicação da função **obterLivros** da linha 5 a 22 será realizada após a explicação da função **criarArqDeIndicacoes** que se inicia na linha 24. Isso porque a função **criarArqDeIndicacoes** é a função chamada inicialmente pelo bot 2 que, por sua vez, chama a função **obterLivros**.

Assim, na linha 24 inicia a definição da função **criarArqDeIndicacoes** que é responsável por percorrer cada arquivo contendo o corpo do e-mail lido pelo Bot 1, que está no diretório vDiretorio, extrair as indicações de livros, guardá-las num único arquivo de indicações e salvá-lo no mesmo diretório vDiretorio.

Explicando em detalhes, na linha 25 foi chamada a função **open** com os seguintes parâmetros:

1. **diretorio + "\\indicacoes.txt"**: define o caminho completo do arquivo "indicacoes.txt" que será criado.
2. **'w+'**: define que o arquivo será criado com permissões de leitura e escrita.
3. **encoding='utf8'**: define que o arquivo será salvo com a codificação utf-8 (MCINGVALE, 2007).

Na linha 27 é inicializada com uma string vazia a variável na qual serão guardadas as indicações.

A instrução **for** da linha 28 permite percorrer todos os arquivos existentes no diretório e, para cada arquivo, verifica se ele tem a extensão **".eml"** (linha 29); se tiver, então esse arquivo é aberto para leitura **'r'** (linha 30).

Ainda dentro do bloco **if** do laço **for**, o arquivo que acabou de ser aberto é lido e o seu conteúdo concatenado com o conteúdo da variável **indicacoes** (**+=**) e, por fim o arquivo é fechado.

Importante observar que a leitura das indicações contidas no arquivo é feita chamando-se a função **obterLivros** que será descrita posteriormente.

Ao final, linhas 34 e 35, a função **criarArqDeIndicacoes** escreve as indicações no arquivo de indicações e fecha o arquivo.

**Figura 5.18: Script em Python para extrair as indicações**

```

1  # -*- coding: utf-8
2  import re
3  import os
4
5  def obterLivros(fonte, diretorio, arqEmail):
6      fonte = re.sub("(fulano beltrano da silva(.\s*))", "", fonte) # Remove texto desnecessário.
7      fonte = re.sub("\n\s*", "", fonte) # Elimina os "pula de linhas" e " ".
8      fonte = re.sub("\s+", " ", fonte) # Remover múltiplos espaços.
9
10     # Apenas para depuração
11     arquivoAux = open(diretorio + "\\ " + arqEmail + "-DEBUG.txt", 'w+', encoding='utf8')
12     arquivoAux.write(fonte)
13     arquivoAux.close()
14
15     aux1 = re.findall("Título:\s*(.+?)\s*-\s*Autor", fonte) # Obtém os Títulos.
16     aux2 = re.findall("Autor Principal:\s*(.+?)\s*-\s*Ano", fonte) # Obtém os Principais.
17     aux3 = re.findall("Ano de Publicação:\s*(\d{4})", fonte) # Obtém os Anos de Publicação.
18
19     livros = ""
20     for i in range(len(aux1)):
21         livros += aux1[i].strip() + '\t' + aux2[i].strip() + '\t' + aux3[i].strip() + '\n'
22     return livros
23
24 def criarArqDeIndicacoes(diretorio):
25     arquivoIndicacoes = open(diretorio + "\\indicacoes.txt", 'w+', encoding='utf8')
26
27     indicacoes = ""
28     for arqEmail in os.listdir(diretorio):
29         if arqEmail.endswith(".eml"):
30             arquivo = open(diretorio + "\\ " + arqEmail, 'r', encoding='utf8')
31             indicacoes += obterLivros(arquivo.read(), diretorio, arqEmail)
32             arquivo.close()
33
34     arquivoIndicacoes.write(indicacoes)
35     arquivoIndicacoes.close()

```

Fonte: do autor, 2021

A função **obterLivros** (linhas de 5 a 22) utiliza as expressões regulares estudadas na seção 2, Preparação, para extrair as indicações a partir do conteúdo dos e-mails que são passados no parâmetro **fonte**.

O parâmetro **diretorio** e o **arqEmail** são utilizados para poder salvar os resultados intermediários à título de depuração, isto é, para ver se a função está realizando corretamente o seu trabalho. Para a finalidade deste bot, após verificado que ele está funcionando corretamente, esses parâmetros e as linhas de 10 a 13 podem ser removidos.

A linha 6 utiliza a expressão regular que remove a parte final do e-mail que não contém nenhuma indicação de livros.

A linha 7 utiliza a expressão regular que elimina todos os caracteres de “quebra de linhas” e os asteriscos.

A linha 8 utiliza a expressão regular que remove múltiplos espaçamentos.

Como explicado anteriormente, as linhas de 10 a 13 servem ao propósito de depuração que, neste caso, simplesmente salva o resultado da aplicação das expressões regulares anteriores num arquivo com final “-DEBUG.txt”.

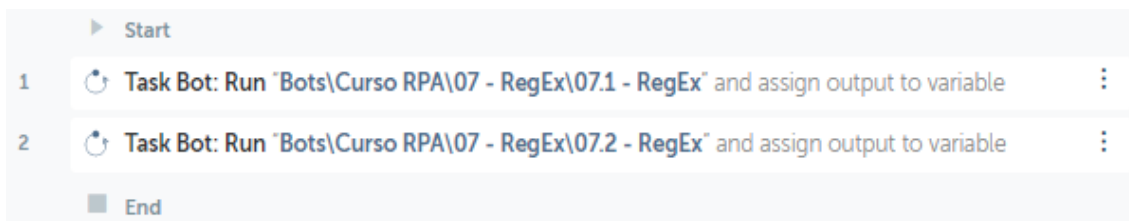
As linhas de 15 a 17 utilizam a função **findall** e as expressões regulares estudadas anteriormente para extrair o Título, Auto Principal e Ano de Publicação da **fonte**.

Como em **aux1**, **aux2** e **aux3** podem existir mais de uma indicação, as linhas de 19 a 22 colocam cada indicação de livros numa linha, sendo que cada linha terá o título, o autor principal e ano de publicação separados por um caractere de tabulação ('\t').

### 5.3.3 Bot 3

Por fim, este último bot simplesmente chamará os outros dois bots anteriores (**Figura 5.19**).

**Figura 5.19: Ações do Bot 3**



**Fonte: do autor, 2021**

A ação Task Bot: Run permite executar um bot configurado. Seguem as configurações dessas duas linhas nas **Figuras 5.20** e **5.21** respectivamente.

**Figura 5.20: Configuração da ação Task Bot para chamar o Bot 1**

**Task Bot: Run**

Runs the selected task bot.

Required bot agent version: 20.11 or above

Task Bot to run

Current Task Bot   **Control Room file**   Variable

Bots\Curso RPA\07 - RegEx\07.1 - RegEx ✕ Choose...

Input values ↻

*This bot has no input values*

Repetition (optional)

Do not repeat ▼

☐ Delay between repetitions

Minutes (optional)

Seconds (optional)

☐ Upon error, start next repetition

---

Save the outcome to a variable (optional)

*This has no output*

**Fonte: do autor, 2021**

Figura 5.21: Configuração da ação Task Bot para chamar o Bot 2

**Task Bot: Run**


Runs the selected task bot.

Required bot agent version: 20.11 or above

Task Bot to run

Current Task Bot   **Control Room file**   Variable

Bots\Curso RPA\07 - RegEx\07.2 - RegEx   X   Choose...

Input values 

This bot has no input values

Repetition (optional)

Do not repeat ▼

☐ Delay between repetitions

Minutes (optional)

##

Seconds (optional)

##

☐ Upon error, start next repetition

---

Save the outcome to a variable (optional)

This has no output

Fonte: do autor, 2021

#### 4. Considerações finais

Esta lição, muito próxima dos bots que são implementados no mundo real, permitiu entender e aplicar as expressões regulares que são muito utilizadas no desenvolvimento de programas e bots.

O domínio de expressões regulares pode abreviar drasticamente o tempo de desenvolvimento e de codificação pelo fato dessas expressões permitirem a definição de padrões relativamente complexos e utilizá-los na remoção e/ou extração de informações no formato texto.



## Referências

AUTOMATION ANYWHERE AUTOMATION 360. **Using dictionary variable for email properties.** Disponível em: <<https://docs.automationanywhere.com/bundle/enterprise-v2019/page/enterprise-cloud/topics/aae-client/bot-creator/commands/cloud-using-email-properties.html>>. Acesso em: 21 jul. 2021.

AUTOMATION ANYWHERE COMMUNITY EDITION. **Formulário para obtenção de acesso à versão community edition do automation anywhere gratuita.** São José – EUA. Disponível em: <<https://www.automationanywhere.com/products/enterprise/community-edition>>. Acesso em: 21 jun. 2021.

AUTOMATION ANYWHERE UNIVERSITY. **Introdução ao automation anywhere.** São José – EUA. Disponível em: <<https://apeople.automationanywhere.com/s/getting-started>>. Acesso em: 21 jun. 2021a.

AUTOMATION ANYWHERE UNIVERSITY. **Trilhas de aprendizagem.** São José – EUA. Disponível em: <<https://university.automationanywhere.com/training/rpa-learning-trails/>>. Acesso em: 21 jun. 2021b.

AUTOMATION ANYWHERE UNIVERSITY. **Email server setting.** Disponível em: <<https://docs.automationanywhere.com/bundle/enterprise-v2019/page/enterprise-cloud/topics/aae-client/bot-creator/commands/cloud-configuring-mail-server.html>>. Acesso em: 23 jul. 2021c.

BANIN, S. L. **Python 3: conceitos e aplicações: uma abordagem didática.** São Paulo: Érica, 2018. Disponível em: <<https://integrada.minhabiblioteca.com.br/#/books/9788536530253/>>. Acesso em: 23 jul. 2021.

CHICONI, N. O que é ASCII, UNICODE e UTF-8. CCM, 2020. Disponível em: <<https://br.ccm.net/faq/9956-o-que-e-ascii-unicode-e-utf-8>>. Acesso em: 21 jul. 2021.

CHANDRA, R. V.; VARANASI, B. S. **Python requests essentials: learn how to integrate your applications seamlessly with web services using python requests.** Packt Publishing, 2015.

DIGICERT. **The ultimate guide: what is SSL, TLS and HTTPS?** Disponível em: <<https://www.websecurity.digicert.com/security-topics/what-is-ssl-tls-https>>. Acesso em: 23 jul. 2021.

ELMAN, J.; LAVIN, M. **Django essencial: usando REST, websockets e backbone.** São Paulo: Novatec, 2015.

GOOGLE. **Ajuda do administrador do google workspace:** controle o acesso a apps menos seguros. Disponível em: <<https://support.google.com/a/answer/6260879?hl=pt-BR>>. Acesso em: 21 jul. 2021.

JARMUL, K.; LAWSON, R. **Python web scraping**. 2. ed. Birmingham: Packt Publishing, 2017.

LOPES, M. D. e LIMA, W. R. **Análise do índice de massa corporal de funcionários de uma instituição de ensino superior**. EFDeportes.com, Revista Digital. Buenos Aires, ano 18, n. 181, jun. 2013. Disponível em: <<https://www.efdeportes.com/efd181/analise-do-indice-de-massa-corporal-de-funcionarios.htm>>. Acesso em: 21 jul. 2021.

MCINGVALE, FRANK. **All about python and unicode**. trad. Menezes, Nilo. PythonBrasil, 2007. Disponível em: <<https://wiki.python.org.br/TudoSobrePythoneUnicode>>. Acesso em: 29 jul. 2021.

MICROSOFT. **O que são IMAP e POP?** Disponível em: <<https://support.microsoft.com/pt-br/office/o-que-s%C3%A3o-imap-e-pop-ca2c5799-49f9-4079-aefe-ddca85d5b1c9>>. Acesso em: 23 jul. 2021.

PYTHON BRASIL. **Instalando o python 3 no windows**. Disponível em: <<https://python.org.br/instalacao-windows/>>. Acesso em: 24 jul. 2021.

NOTEPAD++. **What is notepad++**. Disponível em: <<https://notepad-plus-plus.org/>>. Acesso em: 28 jun. 2021.

WDG AUTOMATION – AN IBM COMPANY. **7 pilares essenciais para projetos de RPA bem-sucedidos**. São Paulo: Newsletter WDG. Disponível em: <<https://www.wdgautomation.com/7-pilares-essenciais-para-projetos-de-rpa-bem-sucedidos/>>. Acesso em: 21 jun. 2021.