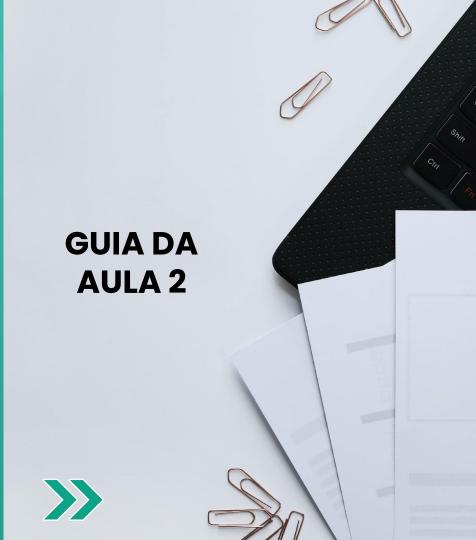


# Profissão: Analista de dados





## **COLETA DE DADOS I**







# Manipule arquivos TXT

Formato

Regex



Acompanhe aqui os temas que serão tratados na videoaula





#### 2.1. Formato

Um arquivo **texto** é um tipo de arquivo de **texto** sem uma estrutura definida (**não estruturado**).

Veja e seguir:





#### **Exemplo**: nubank.txt

%%writefile nubank.txt In []: Como você prefere falar com a gente? E-mail Tem alguma dúvida? Podemos te ajudar pelo nosso canal de email. meajuda@nubank.com.br Telefone Você pode ligar para o 0800 do Nubank a qualquer hora através do número abaixo. 0800 608 6236 Chat Precisa de uma ajuda agora? Entre em contato com nosso atendimento através do chat. Basta abrir o chat no app. Siga o @Nubank Saiba das novidades e receba dicas na nossas redes sociais e também na NuCommunity, a comunidade online oficial do Nubank.





#### **Exemplo**: nubank.txt

press@nu.bank

In [ ]:

Imprensa Reunimos todas **as** informações para você aqui.

Ouvidoria Já conversou conosco e mesmo assim não conseguiu resolver o que precisava? Nossa Ouvidoria pode avaliar seu caso.
0800 887 0463

ouvidoria@nubank.com.br

Atendemos em dias úteis das 9h às 18h horário de São Paulo/SP).

Parcerias

Se você tem uma proposta de patrocínio, parceria ou publicidade, fale conosco por aqui: marketing@nubank.com.br





#### **Exemplo**: Extrair e-mails de um arquivo de texto.

- Extrair as linhas do arquivo.

```
with open(file='./nubank.txt', mode='r', encoding='utf8') as arquivo:
    linhas = arquivo.readlines()
print(linhas)
```

- Limpar as linhas do caracter de nova linha '\n'

```
linhas = filter(lambda linha: linha != '\n', linhas)
linhas = map(lambda linha: linha.strip(), linhas)
linhas = list(linhas)
print(linhas)
```





#### - Extrair linhas com o texto \cdot\com'

```
linhas_com_email = filter(lambda linha: '.com' in linha, linhas)
linhas_com_email = list(linhas_com_email)
print(linhas_com_email)
```

#### - Extrair emails das linhas com o texto `.com'

```
emails_extraidos = []

for linha_com_email in linhas_com_email: palavras = linha_com_email.split(sep=' ')
emails = filter(lambda palavra: '@' in palavra, palavras) emails_extraidos =
emails_extraidos + list(emails) print(emails_extraidos)
```





### 2.2. Regex

É um algoritmo de busca de padrões em strings e é implementado nativamente em diversas linguagens de programação. Você pode ler mais sobre regex neste <u>link</u> e testar seu regex na ferramenta online deste <u>link</u>.

#### **import** re

lista\_padroes = re.findall('<string de busca>', texto)





**Exemplo**: Extrair e-mails de um arquivo de texto.

String de busca.

Para encontrar emails no arquivo de texto, vamos utilizar string de busca

'\S+@\S+', onde: \S+ encontra um sequencia de caracteres sem espaço;

@ encontra o caracter '@';

\S+ encontra um sequencia de caracteres sem espaço.





#### Código de extração

```
import re
with open(file='./nubank.txt', mode='r', encoding='utf8') as arquivo: texto =
arquivo.read()
emails_extraidos = re.findall('\S+@\S+', texto) print(emails_extraidos)
```

#### Código para salvar em um arquivo csv

```
In []:
    import csv

with open(file='./nubank.csv', mode='w', encoding='utf8') as arquivo:
    escritor_csv = csv.writer(arquivo, delimiter=';')
    escritor_csv.writerows(
        [['email']] + \
         list(map(lambda email_extraido: [email_extraido], emails_extraidos))
    )
```





#### **Exemplo**: Extrair perfil de redes sociais.

