

```
In [1]:
import requests
from bs4 import BeautifulSoup

from datetime import datetime
from datetime import timedelta
from time import time

import json

import pandas as pd
```

## Extraction and Transformation

### 1. Definindo funções

Observando o url <https://www.in.gov.br/leiturajornal?data=30-06-2021&secao=do1>, podemos ver como navegar as diferentes datas. Precisamos apenas substituir a string 30-06-2021 for diferentes datas. Para tal, utilizei a biblioteca datetime.

Em seguida, com a inspeção de uma página, eu pude identificar que os links das publicações estavam na tag <script id: "param" class: "application/json">. O método find\_all de um objeto BeautifulSoup facilita a extração do conteúdo da tag em questão. Com essa finalidade, fiz requests das urls com diferentes datas e adicionei as informações prioritárias em um dicionário.

A função get\_publications utiliza as bibliotecas requests e BeautifulSoup para buscar as resoluções publicadas no Diário Oficial da União. A docstring da função, mostra como as publicações foram organizadas em um dicionário.

```
In [2]:
def get_publications(starting_date, num_days):
    """
    Busca no Diário Oficial da União todas as RESOLUÇÕES
    da ANVISA publicadas ao longo de [num_days] dias de uma [starting_date].
    Retorna um dicionário de datas com listas de publicações.
    As listas contém as seguintes informações: title, hierarchyList e url.

    Ex:
    publications = {
        datetime(YYYY, mm, dd): [
            {
                'title': 'RESOLUÇÃO-RE Nº 1.341, DE 30 DE ABRIL DE 2020',
                'hierarchyList': ['Ministério da Saúde',
                                'Agência Nacional de Vigilância Sanitária',
                                'Terceira Diretoria',
                                'Gerência de Produtos de Higiene,
                                Perfumes, Cosméticos e Saneantes'
                                ],
                'url': 'https://www.in.gov.br/en/web/dou/-/resolucao-re-n-1.341-de-30-de-abril-de-2020-254923025'
            },
            ...
        ],
        ...
    }

    [num_days] -> int
    [starting_date] -> datetime(Y, m, d) object
    ...

    url_base = r"https://www.in.gov.br/web/dou/-/"

    dates = [(datetime.today() - timedelta(x)).strftime('%d-%m-%Y') for x in range(num_days)]
    publications = {}
    for date in dates:
        publications[date] = []
        html = requests.get(r"https://www.in.gov.br/leiturajornal?data=" + date + r"&secao=do1").content
        soup = BeautifulSoup(html, 'html.parser')
        pubs_day = json.loads(soup.find('script', {'id':'params', 'type':'application/json'}).contents[0].replace('\n\t', ''))['jsonArray']

        for pub in pubs_day:
            if pub['artType'] == 'Resolução' and 'Agência Nacional de Vigilância Sanitária' in pub['hierarchyList']: ## Selecionando as resoluções da Anvisa
                publications[date].append({
                    'title': pub['title'],
                    'hierarchyList': pub['hierarchyList'],
                    'url': url_base + pub['urlTitle']
                })
    return publications
```

O próximo passo é encontrar as resoluções que contêm a chave: 'Deferir os registros e as petições dos produtos saneantes', e identificar como as informações que buscamos se encontram nas publicações. Ademais, inspecionei a página e localizei no corpo do documento a tag <p class: "dou\_paragraph">, que sempre acompanha as informações que procuramos.

A função a seguir utiliza requests e BeautifulSoup para abrir todas as publicações encontradas por get\_publications(), checar a string chave e - caso encontre a chave - retornar uma lista com todas as tags <p class: "dou\_paragraph"> encontradas.

```
In [3]:
def get_data(key, publications):
    """
    Busca em diferentes publicações uma chave, {key}.
    Se encontrada, busca na publicação uma a tag <p class: "dou_paragraph">
    e retorna o dicionário:

    Ex:
    raw_data = {
        'pub_title': [list_of_found_tag],
        'pub2_title': [list_of_found_tag],
        ...
    }

    ...
    raw_data = {}
    for date in publications:
        for pub in publications[date]:
            html = requests.get(pub['url']).content
            soup = BeautifulSoup(html, 'html.parser')
            if key in soup.get_text():
                print(pub['url'])
                required_data = soup.find_all('p', {'class':'dou-paraphraph'})
                data = []
                for tags in required_data:
                    data.append(tags)
                raw_data[pub['title']] = data
    return raw_data
```

As funções a seguir (structure\_product\_info e sort\_by\_business) estruturarão os dados encontrados em cada publicação em nested dictionaries. Quando aplicadas em uma lista de strings elas retornarão o seguinte dicionário:

```
Ex: {resolucao:
      {empresa:
       {auth: 'string', produtos: [{infos: 'string'}, ...]},
       empresa:{...}, ...},
      resolucao:
      {empresa:
       {auth: 'string', produtos: [{infos: 'string'}, ...]},
       empresa:{...}, ...},
```

...

```
In [4]: def structure_product_info(list_to_split):
    structured_list_of_dict = []

    product_dict = {
        'NOME DO PRODUTO E MARCA': None,
        'VERSÃO': None,
        'NUMERO DE PROCESSO': None,
        'NUMERO DE REGISTRO': None,
        'VENDA E EMPREGO': None,
        'VENCIMENTO': None,
        'APRESENTAÇÃO': None,
        'VALIDADE DO PRODUTO': None,
        'CATEGORIA': None,
        'ASSUNTO DA PETIÇÃO': None,
        'EXPEDIENTE DA PETIÇÃO': None
    }

    for item in list_to_split:
        for key in product_dict:
            if key in item:
                if key == 'NOME DO PRODUTO E MARCA' and product_dict[key] is not None:
                    structured_list_of_dict.append(product_dict)
                    product_dict = {k: None for k in product_dict}
                product_dict[key] = item.replace(key + ': ', '')

    structured_list_of_dict.append(product_dict)
    return structured_list_of_dict
```

```
In [5]: def sort_by_business(lista):
    aimed_data = [
        'NOME DO PRODUTO E MARCA',
        'VERSÃO',
        'NUMERO DE PROCESSO',
        'NUMERO DE REGISTRO',
        'VENDA E EMPREGO',
        'VENCIMENTO',
        'APRESENTAÇÃO',
        'VALIDADE DO PRODUTO',
        'CATEGORIA',
        'ASSUNTO DA PETIÇÃO',
        'EXPEDIENTE DA PETIÇÃO'
    ]

    business_dict = {}
    for item in lista:
        if 'NOME DA EMPRESA:' in item:
            try:
                business_dict[current_business]['produtos'] = structure_product_info(business_dict[current_business]['produtos'])
            except:
                pass

            current_business = item.replace("NOME DA EMPRESA: ", '')
            business_dict[current_business] = {}

        elif 'AUTORIZAÇÃO' in item:
            current_business_auth = item.replace("AUTORIZAÇÃO: ", '')
            business_dict[current_business]['auth'] = current_business_auth
            business_dict[current_business]['produtos'] = []

        elif any(aimed in item for aimed in aimed_data):
            business_dict[current_business]['produtos'].append(item)

    business_dict[current_business]['produtos'] = structure_product_info(business_dict[current_business]['produtos'])

    return business_dict
```

## 2.Extraindo os dados

```
In [6]: key = 'Deferir os registros e as petições dos produtos saneantes'
starting_date = datetime(2020, 6, 30)
num_days = 61

s = time()
print("Buscando todas as publicações...")
pubs = get_publications(starting_date, num_days)
print(f"A busca grossa levou {time() - s} segundos.")

s = time()
print(f"Selecionando apenas as resoluções que contêm a chave: \'{key}\' \n ...")
print('Seguem as publicações capturadas: ')
raw_data = get_data(key, pubs)
print(f"A busca fina levou {time() - s} segundos.")

Buscando todas as publicações...
A busca grossa levou 109.48577213287354 segundos.
Selecionando apenas as resoluções que contêm a chave: "Deferir os registros e as petições dos produtos saneantes"
...
Seguem as publicações capturadas:
https://www.in.gov.br/web/dou/-/resolucao-re-n-2.821-de-16-de-julho-de-2021-332707038
https://www.in.gov.br/web/dou/-/resolucao-re-n-2.690-de-8-de-julho-de-2021-331313194
https://www.in.gov.br/web/dou/-/resolucao-re-n-2.585-de-1-de-julho-de-2021-329794337
https://www.in.gov.br/web/dou/-/resolucao-re-n-2.495-de-24-de-junho-de-2021-328274955
https://www.in.gov.br/web/dou/-/resolucao-re-n-2.404-de-17-de-junho-de-2021-326852629
https://www.in.gov.br/web/dou/-/resolucao-re-n-2.314-de-10-de-junho-de-2021-325403737
https://www.in.gov.br/web/dou/-/resolucao-re-n-2.186-de-2-de-junho-de-2021-323929762
https://www.in.gov.br/web/dou/-/resolucao-re-n-2.115-de-27-de-maio-de-2021-323007247
A busca fina levou 470.642511293793 segundos.

A busca levou aproximadamente 10 minutos e conseguiu encontrar 8 publicações contendo a chave de busca.
```

```
In [7]: #saving raw_data in json for safekeeping
for resolucao in raw_data:
    raw_data[resolucao] = [str(l) for l in raw_data[resolucao]]

dump = open(r"json\raw_data.json", "w")
json.dump(raw_data, dump, indent = 6)

dump.close()
```

## 3.Estruturando os dados

raw\_data contém o corpo de todas as resoluções dos meses de maio e junho. As informações ainda estão dentro das tags e em formato de texto. A seguir limparei o corpo das tags e particionarei as informações em dicionários utilizando as funções criadas anteriormente.

```
In [8]: clean_data = {}
for resolucao in raw_data:
    clean_data[resolucao] = [str(l).replace("</p>", '').replace("<p class='dou-paragraph'>", '') for l in raw_data[resolucao]]
    clean_data[resolucao] = sort_by_business(clean_data[resolucao])

#saving clean_data in json for safekeeping
#Este json pode ser salvo e facilmente atualizado, sem ocupar muito espaço.
#Esta seria uma etapa crucial para a manutenção do Banco de Dados.
dump = open(r"json\clean_data.json", "w")
json.dump(clean_data, dump, indent = 6)

dump.close()
```

## 4.Criando um DataFrame

In [9]:

```
df = pd.DataFrame(columns = [
    'RESOLUCAO',
    'EMPRESA',
    'AUTORIZACAO',
    'MARCA',
    'PROCESSO',
    'REGISTRO',
    'VENDA E EMPREGO',
    'VENCIMENTO',
    'APRESENTACAO',
    'VALIDADE PRODUTO',
    'CATEGORIA',
    'ASSUNTO PETICAO',
    'EXPEDIENTE PETICAO',
    'VERSAO'
])
```

Usei a biblioteca Pandas para criar um dataframe com as informações estruturadas. Como a informação limpa estava em nested dicionaries, o df foi construído de forma recursiva para cada nova linha.

In [10]:

```
idx = 0
errors = []
for resolucao in clean_data:
    for empresa in clean_data[resolucao]:
        auth = clean_data[resolucao][empresa]['auth']
        for produto in clean_data[resolucao][empresa]['produtos']:
            try:
                df.loc[idx] = [
                    resolucao,
                    empresa,
                    auth,
                    produto['NOME DO PRODUTO E MARCA'],
                    produto['NUMERO DE PROCESSO'],
                    produto['NUMERO DE REGISTRO'],
                    produto['VENDA E EMPREGO'],
                    produto['VENCIMENTO'],
                    produto['APRESENTAÇÃO'],
                    produto['VALIDADE DO PRODUTO'],
                    produto['CATEGORIA'],
                    produto['ASSUNTO DA PETIÇÃO'],
                    produto['EXPEDIENTE DA PETIÇÃO'],
                    produto['VERSAO']
                ]
                idx += 1
            except Exception as e:
                if [resolucao, empresa] in errors:
                    pass
                else:
                    errors.append([resolucao, empresa])
```

Finalmente, salvei o DataFrame com 1528 entradas em csv e xlsx.

In [11]:

```
df.to_csv(r'structured\saneantes_anvisa.csv', index = False, sep = ';')
df.to_excel(r'structured\saneantes_anvisa.xlsx', index = False, sheet_name = 'Dados')
```

In [62]:

df

Out[62]:

	RESOLUCAO	EMPRESA	AUTORIZACAO	MARCA	PROCESSO	REGISTRO	VENDA E EMPREGO	VENCIMENTO	APRESENTACAO	VALIDADE PRODUTO	CATEGORIA	ASSUNTO PETICAO	EXPEDIENTE PETICAO	VERSAO
0	RESOLUÇÃO RE Nº 2.821, DE 16 DE Julho DE 2021	Archote Indústria Química Ltda	3.00524-6	DIA% DESINFETANTE USO GERAL	25351.704205/2017- 65	3.0524.0040.001- 3	PRODUTO DE VENDA LIVRE	03/2028	FRASCO DE PLASTICO TRANSPARENTE	24 Meses	3205061 DESINFETANTE PARA USO GERAL	389 REG. SANEANTES - Alteração de Rotulagem de...	1897959/21- 4	EUCALIPTO
1	RESOLUÇÃO RE Nº 2.821, DE 16 DE Julho DE 2021	Archote Indústria Química Ltda	3.00524-6	DIA% DESINFETANTE USO GERAL	25351.704205/2017- 65	3.0524.0040.002- 1	PRODUTO DE VENDA LIVRE	03/2028	GALAO PLASTICO	24 Meses	3205061 DESINFETANTE PARA USO GERAL	389 REG. SANEANTES - Alteração de Rotulagem de...	1897959/21- 4	EUCALIPTO
2	RESOLUÇÃO RE Nº 2.821, DE 16 DE Julho DE 2021	Archote Indústria Química Ltda	3.00524-6	DIA% DESINFETANTE USO GERAL	25351.704205/2017- 65	3.0524.0040.003- 1	PRODUTO DE VENDA LIVRE	03/2028	FRASCO DE PLASTICO TRANSPARENTE	24 Meses	3205061 DESINFETANTE PARA USO GERAL	389 REG. SANEANTES - Alteração de Rotulagem de...	1897959/21- 4	PINHO
3	RESOLUÇÃO RE Nº 2.821, DE 16 DE Julho DE 2021	Archote Indústria Química Ltda	3.00524-6	DIA% DESINFETANTE USO GERAL	25351.704205/2017- 65	3.0524.0040.004- 8	PRODUTO DE VENDA LIVRE	03/2028	GALAO PLASTICO	24 Meses	3205061 DESINFETANTE PARA USO GERAL	389 REG. SANEANTES - Alteração de Rotulagem de...	1897959/21- 4	PINHO
4	RESOLUÇÃO RE Nº 2.821, DE 16 DE Julho DE 2021	Archote Indústria Química Ltda	3.00524-6	DIA% DESINFETANTE USO GERAL	25351.704205/2017- 65	3.0524.0040.005- 6	PRODUTO DE VENDA LIVRE	03/2028	FRASCO DE PLASTICO TRANSPARENTE	24 Meses	3205061 DESINFETANTE PARA USO GERAL	389 REG. SANEANTES - Alteração de Rotulagem de...	1897959/21- 4	LAVANDA
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1523	RESOLUÇÃO RE Nº 2.115, DE 27 DE Maio DE 2021	WHITE CLEAN SANEANTES EIRELI-ME	3.05990-7	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN	25351.034348/2021- 84	3.5990.0004.001- 8	PRODUTO DE USO PROFISSIONAL OU DE VENDA RESTRITA	05/2031	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN ...	24 Meses	3205061 DESINFETANTE PARA USO GERAL	30020 REG. SANEANTES - Registro de produtos sa...	None	None
1524	RESOLUÇÃO RE Nº 2.115, DE 27 DE Maio DE 2021	WHITE CLEAN SANEANTES EIRELI-ME	3.05990-7	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN	25351.034348/2021- 84	3.5990.0004.002- 6	PRODUTO DE USO PROFISSIONAL OU DE VENDA RESTRITA	05/2031	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN ...	24 Meses	3205061 DESINFETANTE PARA USO GERAL	30020 REG. SANEANTES - Registro de produtos sa...	None	None
1525	RESOLUÇÃO RE Nº 2.115, DE 27 DE Maio DE 2021	WHITE CLEAN SANEANTES EIRELI-ME	3.05990-7	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN	25351.034348/2021- 84	3.5990.0004.003- 4	PRODUTO DE USO PROFISSIONAL OU DE VENDA RESTRITA	05/2031	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN ...	24 Meses	3205061 DESINFETANTE PARA USO GERAL	30020 REG. SANEANTES - Registro de produtos sa...	None	None
1526	RESOLUÇÃO RE Nº 2.115, DE 27 DE Maio DE 2021	WHITE CLEAN SANEANTES EIRELI-ME	3.05990-7	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN	25351.034348/2021- 84	3.5990.0004.004- 2	PRODUTO DE USO PROFISSIONAL OU DE VENDA RESTRITA	05/2031	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN ...	24 Meses	3205061 DESINFETANTE PARA USO GERAL	30020 REG. SANEANTES - Registro de produtos sa...	None	None
1527	RESOLUÇÃO RE Nº 2.115, DE 27 DE Maio DE 2021	WHITE CLEAN SANEANTES EIRELI-ME	3.05990-7	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN	25351.034348/2021- 84	3.5990.0004.005- 0	PRODUTO DE USO PROFISSIONAL OU DE VENDA RESTRITA	05/2031	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN ...	24 Meses	3205061 DESINFETANTE PARA USO GERAL	30020 REG. SANEANTES - Registro de produtos sa...	None	None

1528 rows x 14 columns

## 5.Observações

Uma breve observação dos dados obtidos deixa explícito algumas possíveis modificações. Primeiro, eu faria uma normalização na unidade de tempo da validade, regularizando-as em [meses]. Também, adicionaria uma coluna [url] para cada resolução. Esta adição facilitaria a aquisição dos documentos no site da união para o cliente.

Loading 

### 1.Tabela Fato e Tabela Dimensão

As colunas [CATEGORIA] e [ASSUNTO PETICAO] podem ser resumidas a um código. Então elas podem ser substituídas por seu respectivo código e adicionadas como tabelas complementares à tabela fato.

In [45]:

```
cat_id = []
cat_name = []
cat_replace_dict = {}
for item in df.CATEGORIA.unique():
    cat = item.split(' ', 1)
    cat_id.append(cat[0])
    cat_replace_dict[item] = cat[0]
    cat_name.append(cat[1])
```

In [46]:

```
subj_id = []
subj_name = []
subj_replace_dict = {}
for item in df['ASSUNTO PETICAO'].unique():
    subj = item.split(' ', 1)
    subj_id.append(subj[0])
    subj_replace_dict[item] = subj[0]
    subj_name.append(subj[1])
```

In [59]:

```
fato = df.replace(cat_replace_dict)
fato.replace(subj_replace_dict, inplace = True)
fato.rename(columns = {'MARCA': 'SANEANTE'}, inplace = True)
fato = fato[['SANEANTE', 'CATEGORIA', 'ASSUNTO PETICAO', 'RESOLUCAO', 'EMPRESA', 'AUTORIZACAO', 'PROCESSO',
            'REGISTRO', 'VENDA E EMPREGO', 'VENCIMENTO', 'APRESENTACAO',
            'VALIDADE PRODUTO', 'EXPEDIENTE PETICAO', 'VERSAO']]
```

#### 1.1.Tabela Fato

In [61]:

fato

Out[61]:

	SANEANTE	CATEGORIA	ASSUNTO PETICAO	RESOLUCAO	EMPRESA	AUTORIZACAO	PROCESSO	REGISTRO	VENDA E EMPREGO	VENCIMENTO	APRESENTACAO	VALIDADE PRODUTO	EXPEDIENTE PETICAO	VERSAO
0	DIA% DESINFETANTE USO GERAL	3205061	389	RESOLUÇÃO RE Nº 2.821, DE 16 DE Julho DE 2021	Archote Indústria Química Ltda	3.00524-6	25351.704205/2017-65	3.0524.0040.001-3	PRODUTO DE VENDA LIVRE	03/2028	FRASCO DE PLASTICO TRANSPARENTE	24 Meses	189795/21-4	EUCALIPTO
1	DIA% DESINFETANTE USO GERAL	3205061	389	RESOLUÇÃO RE Nº 2.821, DE 16 DE Julho DE 2021	Archote Indústria Química Ltda	3.00524-6	25351.704205/2017-65	3.0524.0040.002-1	PRODUTO DE VENDA LIVRE	03/2028	GALAO PLASTICO	24 Meses	189795/21-4	EUCALIPTO
2	DIA% DESINFETANTE USO GERAL	3205061	389	RESOLUÇÃO RE Nº 2.821, DE 16 DE Julho DE 2021	Archote Indústria Química Ltda	3.00524-6	25351.704205/2017-65	3.0524.0040.003-1	PRODUTO DE VENDA LIVRE	03/2028	FRASCO DE PLASTICO TRANSPARENTE	24 Meses	189795/21-4	PINHO
3	DIA% DESINFETANTE USO GERAL	3205061	389	RESOLUÇÃO RE Nº 2.821, DE 16 DE Julho DE 2021	Archote Indústria Química Ltda	3.00524-6	25351.704205/2017-65	3.0524.0040.004-8	PRODUTO DE VENDA LIVRE	03/2028	GALAO PLASTICO	24 Meses	189795/21-4	PINHO
4	DIA% DESINFETANTE USO GERAL	3205061	389	RESOLUÇÃO RE Nº 2.821, DE 16 DE Julho DE 2021	Archote Indústria Química Ltda	3.00524-6	25351.704205/2017-65	3.0524.0040.005-6	PRODUTO DE VENDA LIVRE	03/2028	FRASCO DE PLASTICO TRANSPARENTE	24 Meses	189795/21-4	LAVANDA
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1523	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN	3205061	30020	RESOLUÇÃO RE Nº 2.115, DE 27 DE Maio DE 2021	WHITE CLEAN SANEANTES EIRELI-ME	3.05990-7	25351.034348/2021-84	3.5990.0004.001-8	PRODUTO DE USO PROFISSIONAL OU DE VENDA RESTRITA	05/2031	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN ...	24 Meses	None	None
1524	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN	3205061	30020	RESOLUÇÃO RE Nº 2.115, DE 27 DE Maio DE 2021	WHITE CLEAN SANEANTES EIRELI-ME	3.05990-7	25351.034348/2021-84	3.5990.0004.002-6	PRODUTO DE USO PROFISSIONAL OU DE VENDA RESTRITA	05/2031	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN ...	24 Meses	None	None
1525	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN	3205061	30020	RESOLUÇÃO RE Nº 2.115, DE 27 DE Maio DE 2021	WHITE CLEAN SANEANTES EIRELI-ME	3.05990-7	25351.034348/2021-84	3.5990.0004.003-4	PRODUTO DE USO PROFISSIONAL OU DE VENDA RESTRITA	05/2031	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN ...	24 Meses	None	None
1526	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN	3205061	30020	RESOLUÇÃO RE Nº 2.115, DE 27 DE Maio DE 2021	WHITE CLEAN SANEANTES EIRELI-ME	3.05990-7	25351.034348/2021-84	3.5990.0004.004-2	PRODUTO DE USO PROFISSIONAL OU DE VENDA RESTRITA	05/2031	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN ...	24 Meses	None	None
1527	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN	3205061	30020	RESOLUÇÃO RE Nº 2.115, DE 27 DE Maio DE 2021	WHITE CLEAN SANEANTES EIRELI-ME	3.05990-7	25351.034348/2021-84	3.5990.0004.005-0	PRODUTO DE USO PROFISSIONAL OU DE VENDA RESTRITA	05/2031	DESINFETANTE HOSPITALAR BAC 640 H WHITE CLEAN ...	24 Meses	None	None

1528 rows × 14 columns

#### 1.2.Tabela Dimensão Assunto

In [63]:

```
dim_subj = pd.DataFrame({'ASSUNTO PETICAO': subj_id, 'NOME ASSUNTO': subj_name})
df_dim_subj.set_index('ASSUNTO')
```

Out[63]:

ASSUNTO	NOME ASSUNTO
389	REG. SANEANTES - Alteração de Rotulagem de Pro...
331	REG. SANEANTES - Nova versão de Produto
30020	REG. SANEANTES - Registro de produtos saneantes
396	REG. SANEANTES - Alteração (Inclusão Ou Exclus...
392	REG. SANEANTES - Novo Prazo de Validade de Pro...
3873	Registro de Produto de Risco 2 - Jardinagem Am...
3881	Registro de Produto de Risco 2 - Inseticida de...
3883	Registro de Produto de Risco 2 - Desinfetante ...
3782	REG. SANEANTES - Retificação de Publicação de ...
330	REG. SANEANTES - Modificação de Fórmula de Pro...
332	REG. SANEANTES - Nova Embalagem de Produto
390	REG. SANEANTES - Mudança de Nome de Produto
335	REG. SANEANTES - Cancelamento de Registro de P...

**NOME ASSUNTO**

<b>312</b>	REG. SANEANTES - Mudança de Categoria de Produto
<b>3928</b>	Registro de Produto de Risco 2 - Desinfetante ...
<b>387</b>	Registro de Produto de Risco 2 - Detergentes e...
<b>30014</b>	REG. SANEANTES - Desistência de petição/proces...
<b>3882</b>	Registro de Produto de Risco 2 - Desinfetante ...
<b>3929</b>	Registro de Produto de Risco 2 - Desinfetante ...

**1.3 Tabela Dimensão Categoria**

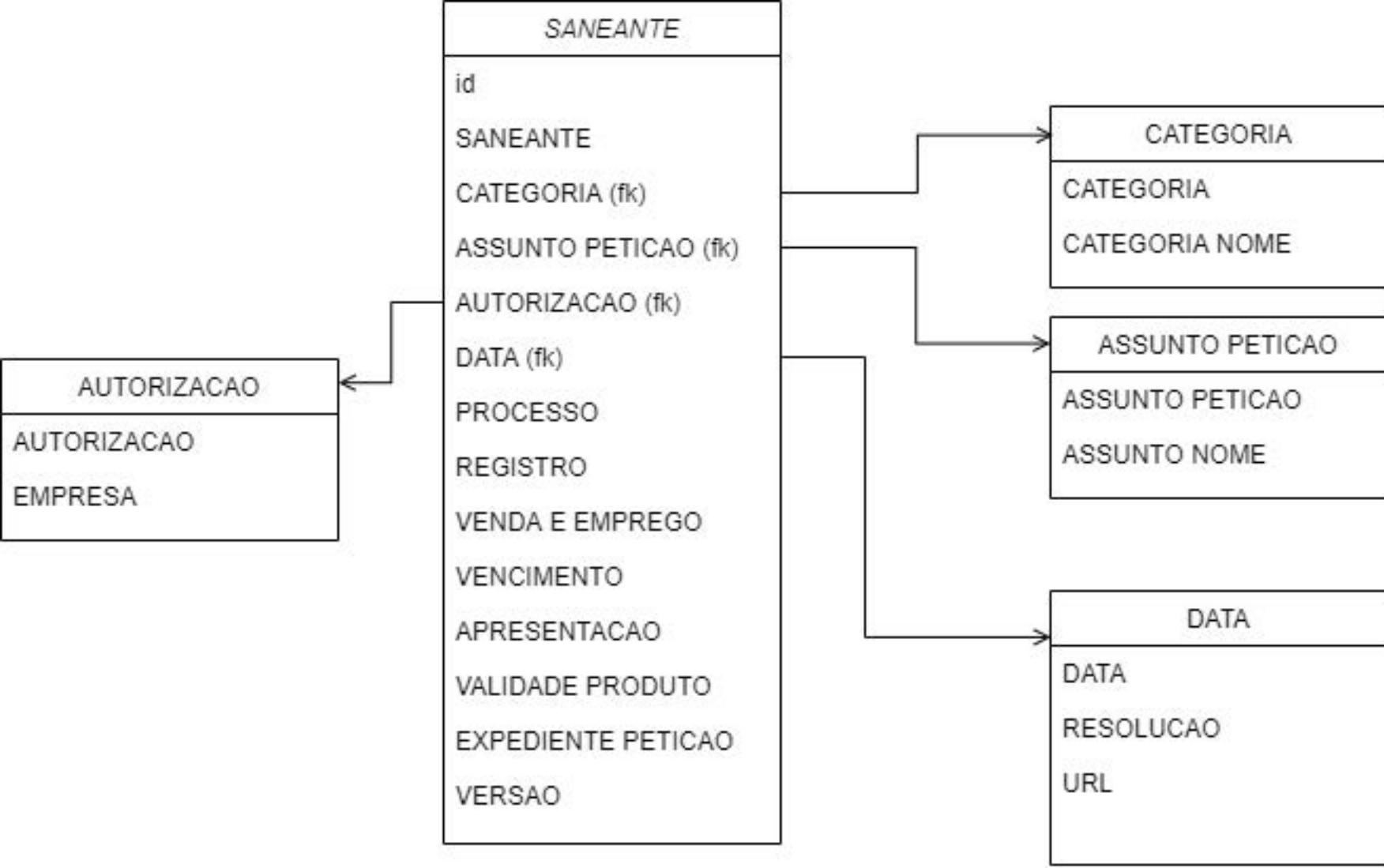
```
In [64]: df_dim_cat = pd.DataFrame({'CATEGORIA': cat_id, 'NOME CATEGORIA': cat_name})
df_dim_cat.set_index('CATEGORIA')
```

```
Out[64]:
```

CATEGORIA	NOME CATEGORIA
<b>3205061</b>	DESINFETANTE PARA USO GERAL
<b>3205045</b>	DESINFETANTE PARA PISCINAS
<b>3210014</b>	ALGICIDA
<b>3222030</b>	DEINCRUSTANTE ALCALINO
<b>3222045</b>	SANITIZANTE PARA INDÚSTRIA ALIMENTÍCIA
<b>3206017</b>	INSETICIDA DE VENDA LIVRE
<b>3206025</b>	INSETICIDA PARA EMPRESAS ESPECIALIZADAS
<b>3203018</b>	DETERGENTE PROFISSIONAL DESINCRUSTANTE ÁCIDO
<b>3207021</b>	RATICIDA PARA EMPRESAS ESPECIALIZADAS
<b>3207013</b>	RATICIDA DE VENDA LIVRE
<b>3222029</b>	DEINCRUSTANTE ÁCIDO
<b>3211042</b>	DETERGENTE DESENGORDURANTE
<b>3222021</b>	LIMPADOR DE USO GERAL
<b>3222020</b>	DETERGENTE ENZIMÁTICO
<b>4300212</b>	DESINFETANTE DE ALTO NÍVEL
<b>3222039</b>	LAVA ROUPAS
<b>3222040</b>	TIRO MANCHAS
<b>3103033</b>	ÁGUA SANITÁRIA
<b>3205029</b>	DESINFETANTE HOSPITALAR PARA SUPERFÍCIES FIXAS...
<b>3222019</b>	JARDINAGEM AMADORA
<b>3202038</b>	REMOVEDOR
<b>3222033</b>	LIMPA PISOS
<b>3205053</b>	DESINFETANTE PARA INDÚSTRIA ALIMENTÍCIA E AFINS
<b>3222035</b>	LAVA LOUÇAS
<b>3208011</b>	REPELENTE
<b>3222047</b>	SANITIZANTE PARA TECIDOS E ROUPAS
<b>3103084</b>	DESENGRAXANTE
<b>3102025</b>	AMACIANTE DE TECIDOS E ROUPAS
<b>3101010</b>	DETERGENTE PARA USO GERAL
<b>3211062</b>	DESINFETANTE PARA HORTIFRUTÍCOLAS
<b>4300215</b>	ALVEJANTE CLORADO
<b>3222049</b>	DESINFETANTE PARA TECIDOS E ROUPAS
<b>3103071</b>	DETERGENTE PARA LAVAR ROUPAS
<b>3102017</b>	ALVEJANTE
<b>4300213</b>	DETERGENTE PARA USO ESPECÍFICO
<b>4300217</b>	PRODUTO PARA TRATAMENTO DE PISCINAS
<b>3211051</b>	DESINFETANTE DE ÁGUA PARA CONSUMO HUMANO
<b>4300211</b>	DESINFETANTE DE NÍVEL INTERMEDIÁRIO
<b>3222050</b>	DESINFETANTE PARA ROUPAS HOSPITALARES
<b>3222022</b>	LIMPA ALUMÍNIO
<b>3103013</b>	DESODORIZANTE AMBIENTAL
<b>3222051</b>	DESINFETANTE PARA USO ESPECÍFICO
<b>3211032</b>	NEUTRALIZADOR DE ODORES COM AÇÃO ANTIMICROBIANA
<b>3102092</b>	SABÃO
<b>3103092</b>	DETERGENTE AUTOMOTIVO
<b>3102033</b>	DETERGENTE ANTIFERRUGINOSO
<b>3207031</b>	DETERGENTE PARA LAVAR LOUÇAS

**2.Observações**

É possível criar outras duas tabelas dimensão. As colunas [AUTORIZACAO] e [EMPRESA] seriam resumidas em uma tabela, deixando apenas [AUTORIZACAO] na tabela fato. Ademais, poderíamos substituir a coluna [RESOLUCAO] pela respectiva data e criar uma tabela dimensão com as colunas *data*, *título* e *url* da mesma. Estas modificações podem ser observadas no diagrama de modelagem de dados.





## PROCESSO COMPLETO EM PYTHON

