



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS



Automação de tarefas com Python

Como fazer a integração de dados usando ferramentas da Google

Christopher Frige Lima Santos
Programa Queimadas
christopher.santos@inpe.br

Introdução

- **Objetivos da automação de tarefas**
- **Configurações no Google Cloud Plataform**
- **Configuração do ambiente de desenvolvimento**
- **Desenvolvimento do código**

Objetivos da automação de tarefas

- Economia de tempo e maior produtividade;
- Padronização eficiente;
- Integração facilitada.



Configurações no Google Cloud Platform

- Criação do projeto
- Ativação das APIs necessárias
- Criação de conta de serviço
- Download de credenciais



Google Cloud Platform



Google Sheets

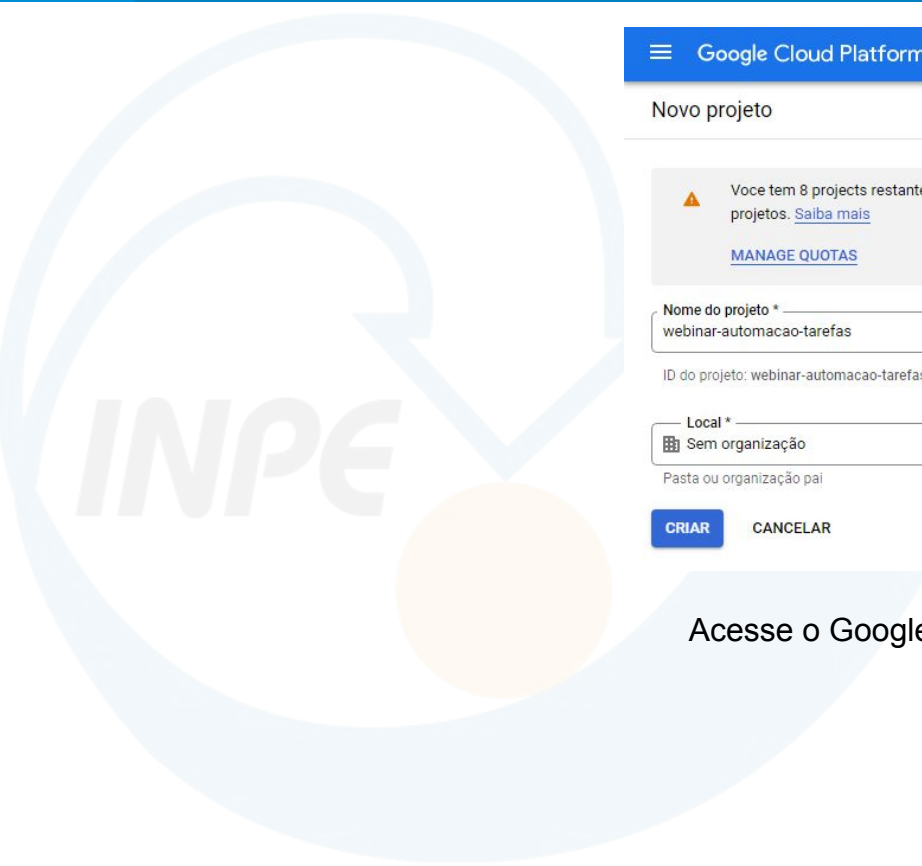


Google Drive



Google Docs

Configurações no Google Cloud Platform



Google Cloud Platform


Novo projeto

⚠️ Você tem 8 projects restantes na sua cota. Solicite um aumento ou exclua projetos. [Saiba mais](#)

[MANAGE QUOTAS](#)

Nome do projeto *
webinar-automacao-tarefas ?

ID do projeto: webinar-automacao-tarefas. Não é possível alterá-lo depois. [EDITAR](#)

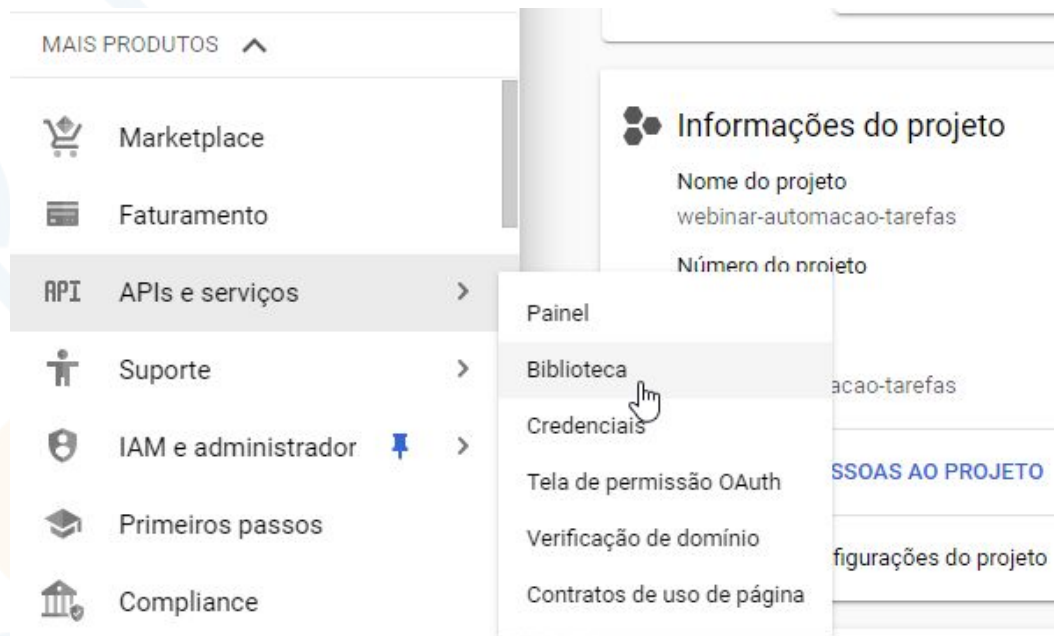
Local *
 Sem organização [PROCURAR](#)

Pasta ou organização pai

[CRIAR](#) [CANCELAR](#)

Acesse o Google Cloud Plataform [aqui](#)

Configurações no Google Cloud Platform



Configurações no Google Cloud Platform

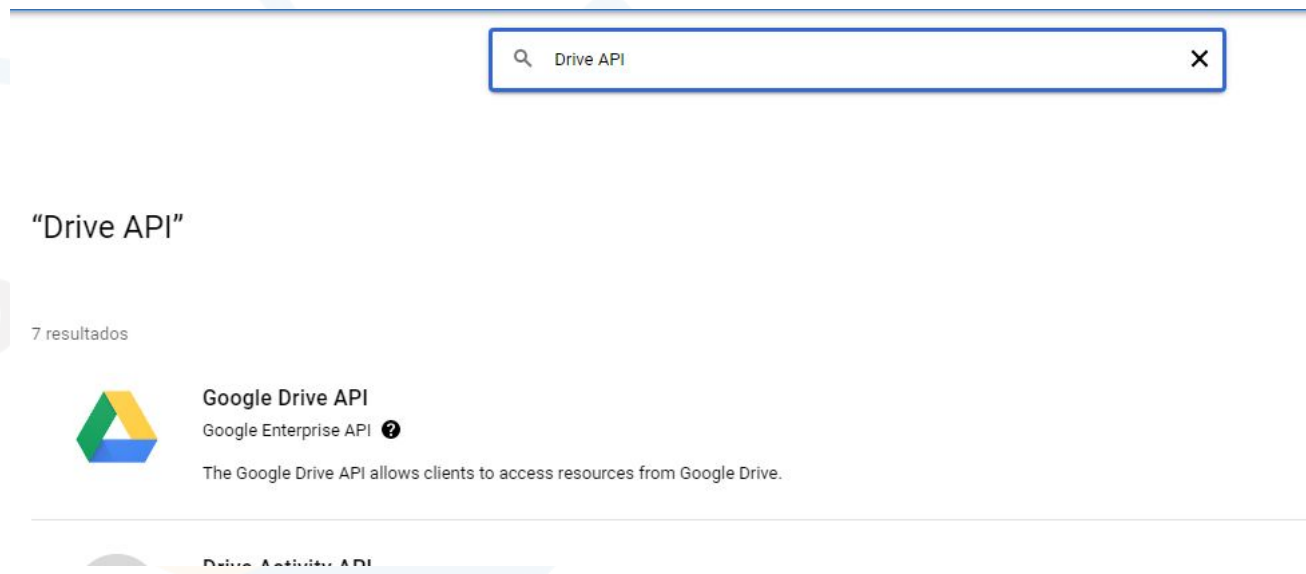
Olá! Esta é a Biblioteca de APIs

A Biblioteca de APIs contém documentos, links e uma experiência de pesquisa inteligente.

🔍 Drive API



Configurações no Google Cloud Platform



Configurações no Google Cloud Platform



Google Drive API

Google Enterprise API

The Google Drive API allows clients to access resources from Google Drive

ATIVAR

TESTAR ESTA API [↗](#)

Clique para ativar esta API

Configurações no Google Cloud Platform



Google Docs API

Google Enterprise API

Reads and writes Google Docs documents.

ATIVAR

TESTAR ESTA API [↗](#)

Clique para ativar esta API

Configurações no Google Cloud Platform



Google Sheets API

Google Enterprise API

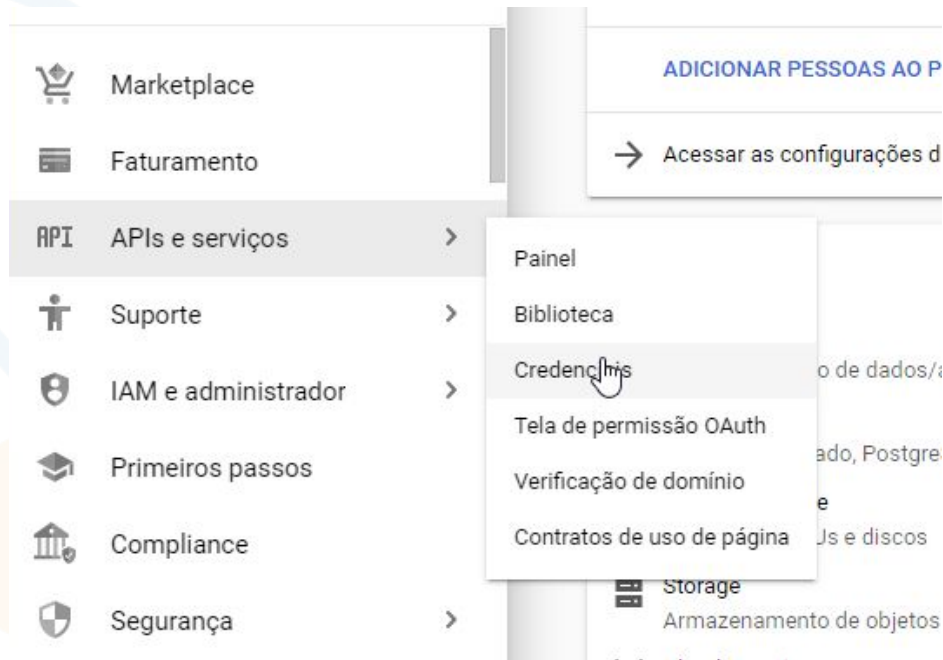
The Sheets API gives you full control over the content and appearance of your spreadsheet data.

ATIVAR

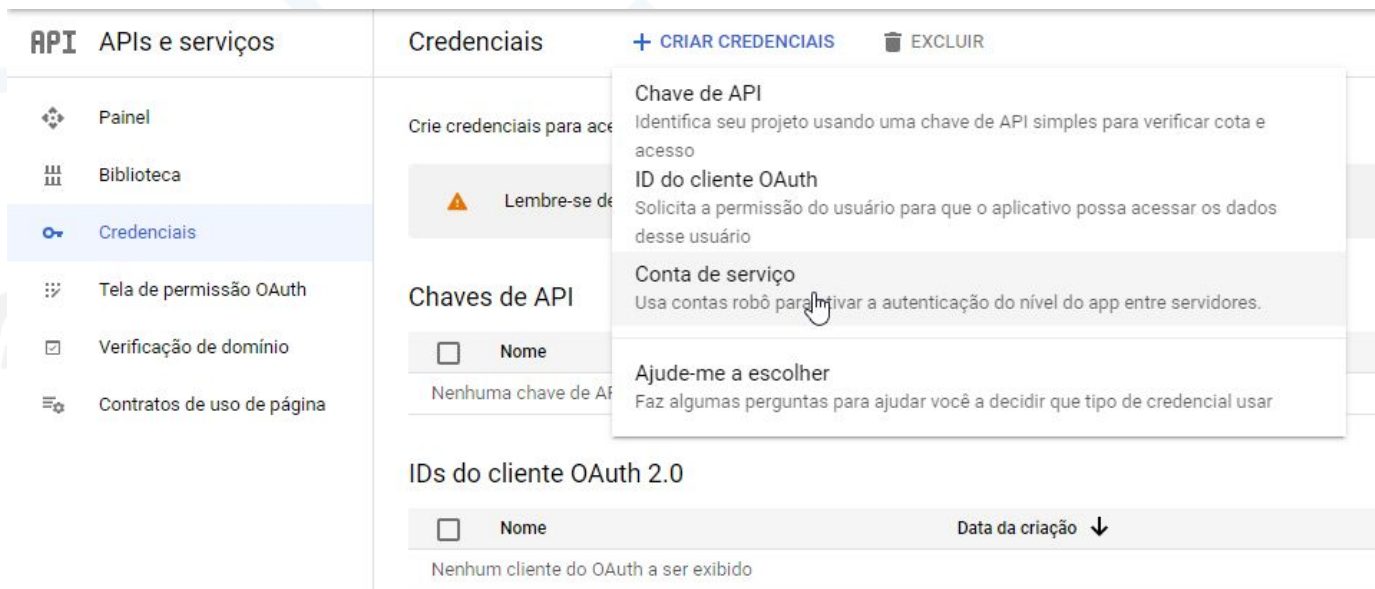
TESTAR ESTA API [↗](#)

Clique para ativar esta API

Configurações no Google Cloud Platform



Configurações no Google Cloud Platform



The screenshot shows the Google Cloud Platform 'APIs e serviços' (APIs and services) page. The left sidebar contains a menu with the following items: 'Painel', 'Biblioteca', 'Credenciais' (highlighted), 'Tela de permissão OAuth', 'Verificação de domínio', and 'Contratos de uso de página'. The main content area is titled 'Credenciais' and includes a '+ CRIAR CREDENCIAIS' button and an 'EXCLUIR' button. Below this, there is a section 'Chaves de API' (API Keys) with a table header 'Nome' and a sub-section 'IDs do cliente OAuth 2.0' with a table header 'Nome' and 'Data da criação'. A dropdown menu is open, showing options: 'Chave de API' (Identifica seu projeto usando uma chave de API simples para verificar cota e acesso), 'ID do cliente OAuth' (Solicita a permissão do usuário para que o aplicativo possa acessar os dados desse usuário), 'Conta de serviço' (Usa contas robô para permitir a autenticação do nível do app entre servidores), and 'Ajude-me a escolher' (Faz algumas perguntas para ajudar você a decidir que tipo de credencial usar).

APIs e serviços

- Painel
- Biblioteca
- Credenciais**
- Tela de permissão OAuth
- Verificação de domínio
- Contratos de uso de página

Credenciais + CRIAR CREDENCIAIS EXCLUIR

Crie credenciais para acessar

Lembre-se de

Chaves de API

Nome

Nenhuma chave de API

IDs do cliente OAuth 2.0

Nome Data da criação ↓

Nenhum cliente do OAuth a ser exibido

Chave de API
Identifica seu projeto usando uma chave de API simples para verificar cota e acesso

ID do cliente OAuth
Solicita a permissão do usuário para que o aplicativo possa acessar os dados desse usuário

Conta de serviço
Usa contas robô para permitir a autenticação do nível do app entre servidores.

Ajude-me a escolher
Faz algumas perguntas para ajudar você a decidir que tipo de credencial usar

Configurações no Google Cloud Platform

Criar conta de serviço

1 Detalhes da conta de serviço

Nome da conta de serviço *

service-account-webinar

Nome de exibição para esta conta de serviço

ID da conta de serviço *

service-account-webinar

@webinar-automacao-tarefas.iam.gservi



Descrição da conta de serviço

Descreva como a conta de serviço será usada

CRIAR E CONTINUAR

2 Conceda a essa conta de serviço acesso ao projeto (opcional)

3 Conceda aos usuários acesso a essa conta de serviço (opcional)

CONCLUIR

CANCELAR

Configurações no Google Cloud Platform

Contas de serviço

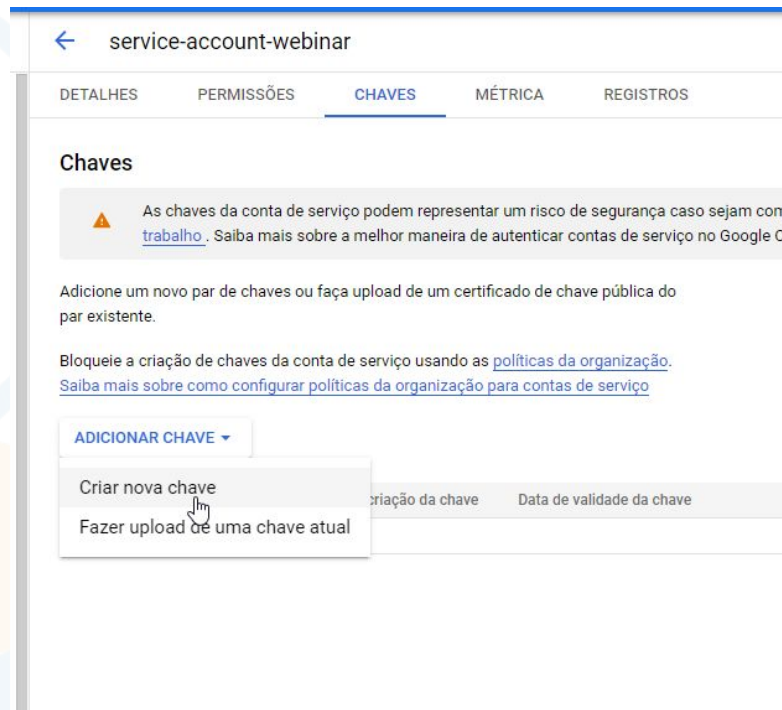


E-mail



service-account-webinar@webinar-automacao-tarefas.iam.gserviceaccount.com

Configurações no Google Cloud Platform



← service-account-webinar

DETALHES PERMISSÕES **CHAVES** MÉTRICA REGISTROS

Chaves

⚠ As chaves da conta de serviço podem representar um risco de segurança caso sejam comprometidas. Saiba mais sobre a melhor maneira de autenticar contas de serviço no Google Cloud.

Adicione um novo par de chaves ou faça upload de um certificado de chave pública do par existente.

Bloqueie a criação de chaves da conta de serviço usando as [políticas da organização](#). Saiba mais sobre [como configurar políticas da organização para contas de serviço](#).

ADICIONAR CHAVE ▾

- Criar nova chave
- Fazer upload de uma chave atual

Criação da chave	Data de validade da chave
------------------	---------------------------

Configurações no Google Cloud Platform

Criar chave privada para "service-account-webinar"

É feito o download de um arquivo contendo a chave privada. Armazene o arquivo com segurança porque essa chave não pode ser recuperada em caso de perda.

Tipo de chave

☒ JSON

Recomendado

☐ P12

Para compatibilidade com versões anteriores usando código com o formato P12

CANCELAR

CRIAR

INPE

Configuração do ambiente de desenvolvimento

- Instalação do Python (versão $\leq 3.9.X$)
 - Tutorial para a instalação em [Windows](#) ou [Linux](#);
- Criação de ambiente virtual (virtualenv) com o Python;
 - Instruções detalhadas [aqui](#);
- Instalação dos pacotes necessários com o pip;
- Copiar arquivo de credenciais para dentro do projeto local;

Configuração do ambiente de desenvolvimento

PROBLEMAS SAÍDA TERMINAL CONSOLE DE DEPURAÇÃO

```
PS C:\Programacao\INPE\Webinar-Automacao-Google> python -m virtualenv env
```

Configuração do ambiente de desenvolvimento

PROBLEMAS SAÍDA TERMINAL CONSOLE DE DEPURAÇÃO

```
PS C:\Programacao\INPE\Webinar-Automacao-Google> .\env\Scripts\activate
```

Configuração do ambiente de desenvolvimento

```
PROBLEMAS  SAÍDA  TERMINAL  CONSOLE DE DEPURACÃO

(env) PS C:\Programacao\INPE\Webinar-Automacao-Google> pip list
Package      Version
-----
pip          21.3.1
setuptools   58.3.0
wheel        0.37.0
(env) PS C:\Programacao\INPE\Webinar-Automacao-Google> 
```

Configuração do ambiente de desenvolvimento

```
# Bibliotecas usadas no projeto
- Google Client for Python
- Pandas
- Matplotlib
- SQLAlchemy
- Psycopg2 (conexão com PostgreSQL)
```

```
## Instalação direta usando o arquivo "requirements":
pip install -r requirements.txt
```

Configuração do ambiente de desenvolvimento

```
## Instalação biblioteca por biblioteca:
```

```
### Google
```

```
pip install --upgrade google-api-python-client google-auth-http2 google-auth-oauthlib
```

```
### Acesso ao banco de dados
```

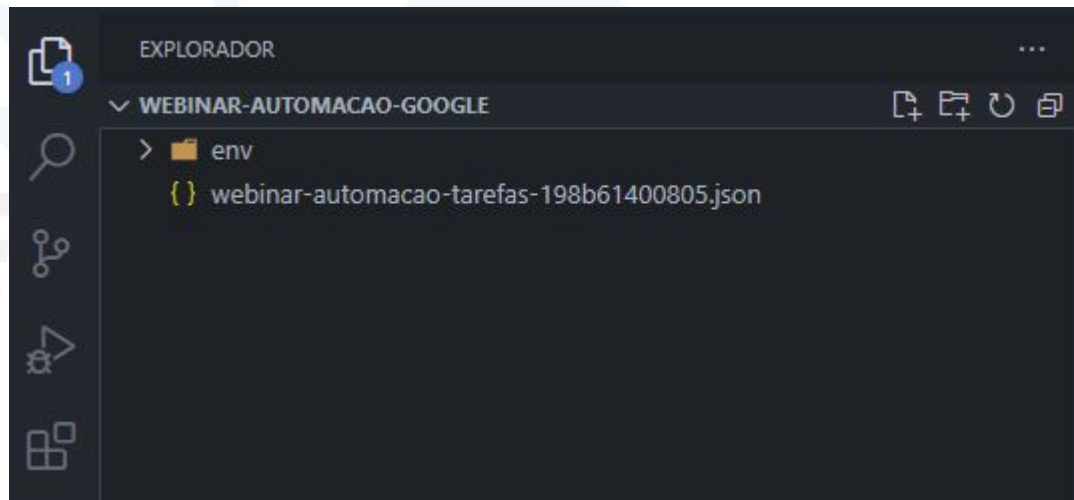
```
pip install psycopg2 sqlalchemy
```

```
### Tratamento e visualização de dados
```

```
pip install pandas matplotlib
```

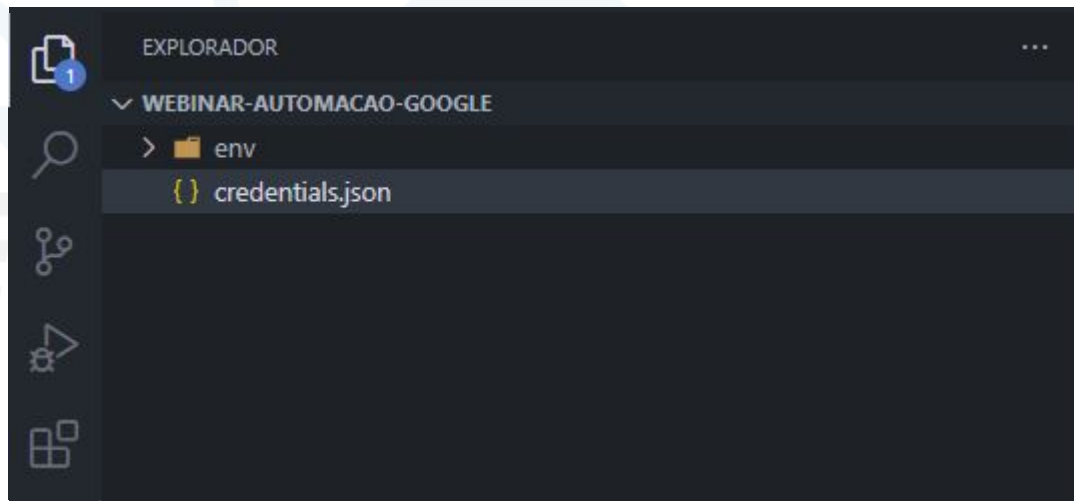
Configuração do ambiente de desenvolvimento

- Renomear arquivo de credenciais baixado do GCP



Configuração do ambiente de desenvolvimento

- Renomear arquivo de credenciais baixado do GCP



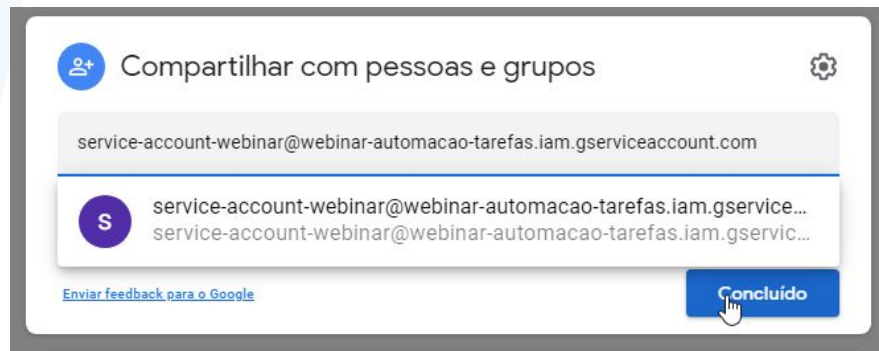
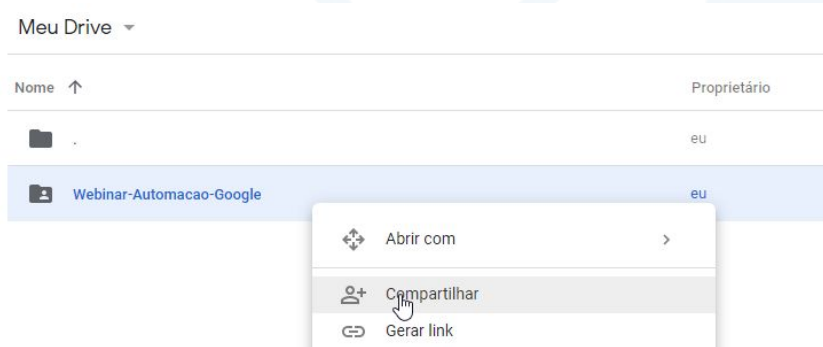
Configuração do ambiente de desenvolvimento

- Copiar email referente a conta de serviço, e permitir o acesso dele à:
 - Pasta principal do projeto no Google Drive;
 - Planilha que será utilizada no Google Sheets;

```
{ } credentials.json X
{ } credentials.json > ...
1  {
2    "type": "service_account",
3    "project_id": "webinar-automacao-tarefas",
4    "private_key_id": "198b614008056158a85e11e6a78ac9b9287f8c4e",
5    "private_key": "-----BEGIN PRIVATE KEY-----\nMIIEvQIBADANBgkqhkiG9w0BAQEEFAASCBKcwggSIAQEAQoTRAQDds
6    "client_email": "service-account-webinar@webinar-automacao-tarefas.iam.gserviceaccount.com",
7    "client_id": "112447423230711295280",
8    "auth_uri": "https://accounts.google.com/o/oauth2/auth",
9    "token_uri": "https://oauth2.googleapis.com/token",
10   "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
11   "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata/x509/service-account-webinar
12 }
13
```

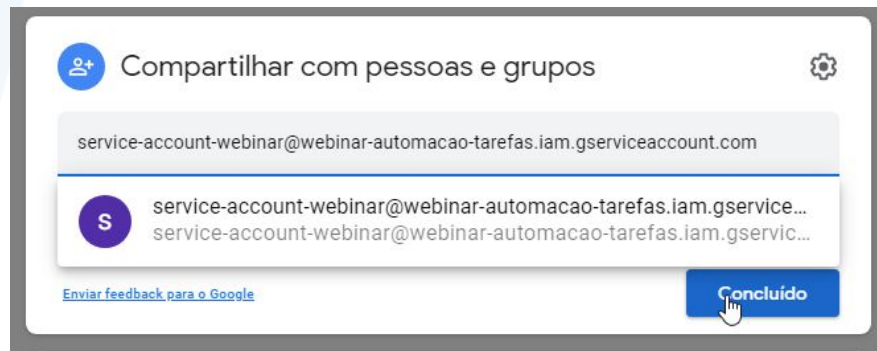
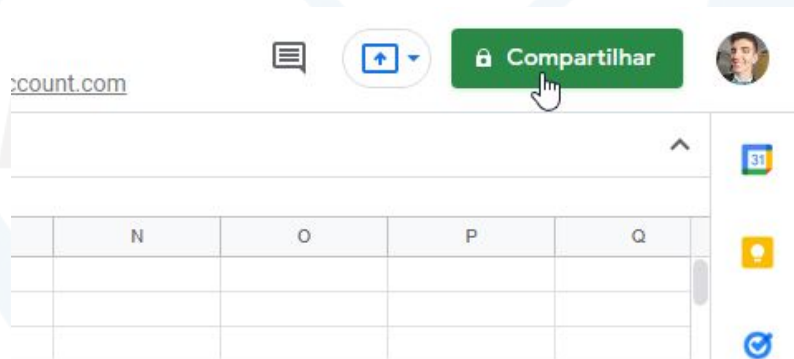
Configuração do ambiente de desenvolvimento

- Copiar email referente a conta de serviço, e permitir o acesso dele à:
 - Pasta principal do projeto no Google Drive;
 - Planilha que será utilizada no Google Sheets;



Configuração do ambiente de desenvolvimento

- Copiar email referente a conta de serviço, e permitir o acesso dele à:
 - Pasta principal do projeto no Google Drive;
 - Planilha que será utilizada no Google Sheets;

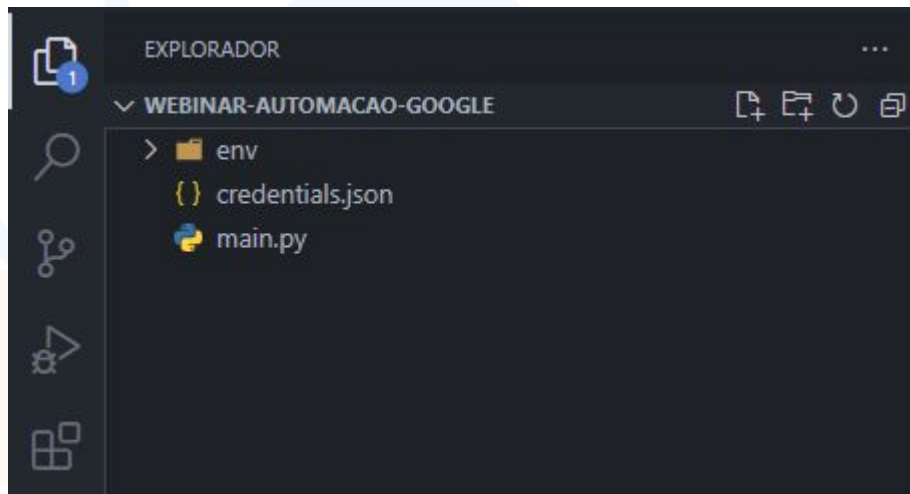


Desenvolvimento do código

- Requisitos
- Autenticação
- Coleta de dados
- Geração de gráficos
- Integração com os serviços da Google
 - Drive, Docs, Sheets;
- Execução e testes

Desenvolvimento do código

- Criação de arquivo **.py** para desenvolver o código



Desenvolvimento do código

```
# IMPORTAÇÕES
from googleapiclient.discovery import build
from google.oauth2 import service_account
from googleapiclient.http import MediaFileUpload

from sqlalchemy import create_engine, pool
import pandas as pd
import matplotlib.pyplot as plt

from datetime import datetime
import os
```

Desenvolvimento do código

```
# AUTENTICAÇÃO POR MEIO DE ARQUIVO (CONTA DE SERVIÇO)
SCOPES = ['https://www.googleapis.com/auth/spreadsheets', 'https://www.googleapis.com/auth/drive',
          'https://www.googleapis.com/auth/documents']

creds = service_account.Credentials.from_service_account_file('credentials.json', scopes=SCOPES)
```

[Ver lista de escopos para APIs do Google](#)

Desenvolvimento do código

```
# CONSULTA AO BANCO E GERAÇÃO DE DF
sql=''
SELECT
name_0 as nome_pais,
count(1) as n_focos
FROM
    public.focos_operacao
WHERE
    data_pas>='20220101' -- A partir do início de 2022
    and continente_id=8 -- Filtra por países da América do Sul
    and cod_sat='AQUA_M-T'
group by name_0;
'''

BD_CONNECTION = os.getenv('BD_CONNECTION') # string no formato: "username:password@host:port/dbname"
ENGINE = create_engine(f'postgresql+psycopg2://{BD_CONNECTION}', poolclass=pool.NullPool)

df = pd.read_sql(sql, ENGINE)
df_list = df.values.tolist()
```

Desenvolvimento do código

```
# GERA O GRÁFICO
fig = plt.figure(figsize=(12, 3.15))
ax = fig.add_axes([0,0,1,1])

x_label = df.nome_pais
ax.yaxis.grid(zorder=0)
ax.bar(x_label, df['n_focos'], color='#f7cb4d', width=0.7, zorder=2)

plt.xlabel('Países')
plt.ylabel('Nº Focos')
plt.xticks(x_label)

ax.get_figure().savefig(f"grafico_gerado.png", format='png', bbox_inches = "tight")
```

Desenvolvimento do código

- Instanciação de serviços que serão utilizados no projeto;

```
# SERVICES DAS FERRAMENTAS GOOGLE
service_sheet = build('sheets', 'v4', credentials=creds)
service_drive = build('drive', 'v3', credentials=creds)
service_docs = build('docs', 'v1', credentials=creds)
```

- O uso será sempre *serviço + método*.

Desenvolvimento do código

```
# ATUALIZA A PLANILHA
service_sheet.spreadsheets().values().update(
    spreadsheetId='17zwhbAIyMv92N8wnueMx8N9fcFypN2kXj48cGaw5Cqc',
    range='Página1!A1', valueInputOption="RAW",
    body={"values": df_list}
).execute()

service_sheet.spreadsheets().values().update(
    spreadsheetId='17zwhbAIyMv92N8wnueMx8N9fcFypN2kXj48cGaw5Cqc',
    range='Página1!D5', valueInputOption="RAW",
    body={"values": [['Atualizado em: '],[date_now]]}
).execute()
```

docs.google.com/spreadsheets/d/**17zwhbAIyMv92N8wnueMx8N9fcFypN2kXj48cGaw5Cqc**/edit#gid=0

Integracao-api-google

Arquivo Editar Ver Inserir Formatar Dados Ferramentas Extensões Ajuda

100% R\$ % .0 .00 123 Padrão (Ari...) 10 B I A

	A	B	C	D	E	F	G	H	I
1	Argentina	5709							
2	Bolivia	200							
3	Brasil	1744							

spreadsheetId

Desenvolvimento do código

	A	B	C	D	E
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					

	A	B	C	D	E
1	Argentina	5709			
2	Bolivia	200			
3	Brasil	1744			
4	Chile	67			
5	Colombia	3327			
6	Ecuador	213			
7	Guyana	59			
8	Paraguay	2714			
9	Peru	302			
10	Suriname	6			
11	Uruguay	33			
12	Venezuela	3392			
13					
14					

Atualizado em:
21/01/2022 11:41

Desenvolvimento do código

```
# CRIA DOCUMENTO USANDO O BASE
doc_base = service_drive.files().list(q="name='DOC BASE'", spaces='drive', fields='files(id)').execute()
doc_id_base = doc_base['files'][0]['id']
body = {'name': 'DOC NOVO - ' + date_now}
response = service_drive.files().copy(fileId=doc_id_base, body=body).execute()
doc_id = response.get('id')
```

Desenvolvimento do código

Meu Drive > Webinar-Automacao-Google ▾ 👤

Nome ↑

📁 saida

📄 DOC BASE 👤

Meu Drive > Webinar-Automacao-Google ▾ 👤

Nome ↑

📁 saida

📄 DOC BASE 👤

📄 DOC NOVO - 20/01/2022 13:37 👤

Desenvolvimento do código

```
# ENVIAR A IMAGEM PRO DRIVE
service_drive.files().list(q="mimeType='application/vnd.google-apps.folder' and name='Webinar-Automacao-Google'",
                           fields='files(id)').execute()
output_folder = service_drive.files().list(q="mimeType='application/vnd.google-apps.folder' and name='saida'",
                                           fields='files(id)',).execute()
img_folder_id = output_folder['files'][0]['id']


media = MediaFileUpload('grafico_gerado.png', mimetype='image/png', resumable=True)
service_drive.files().create(body={'name': 'grafico_gerado.png', 'parents': [img_folder_id]},
                             media_body=media,
                             fields='id').execute()
print (f'Subindo a imagem para o drive!')
```


Desenvolvimento do código

Meu Drive > Webinar-Automacao-Google > saida ▾ 👤

Meu Drive > Webinar-Automacao-Google > saida ▾ 👤

Nome ↑

 grafico_gerado.png 👤

Desenvolvimento do código

```
#ATUALIZA OS TEXTOS DO DOCS
texts = {
    'data_doc': date_now,
    'num_max_focos': str(df['n_focos'].max()),
    'pais_max_focos': df['nome_pais'][df['n_focos'] == df['n_focos'].max()][0]
}

requests_text=[]
for key in texts.keys():
    if texts[key] != '' and texts[key] is not None:
        requests_text.append({
            'replaceAllText': {
                'containsText': {
                    'text': '{{'+key+'}}',
                    'matchCase': 'true'
                },
                'replaceText': texts[key],
            }
        })

service_docs.documents().batchUpdate(documentId=doc_id, body={'requests': requests_text}).execute()
print('Textos alterados!')
```

Desenvolvimento do código

```
# ATUALIZA AS IMAGENS DO DOCS

# Método para pegar o id "local" da imagem no documento base (campo 'objectId', inicia com "kix")
# print( id_imagens = service_docs.documents().get(documentId=doc_id, fields='inlineObjects').execute() )

# {'nome_da_imagem_no_drive.png': 'id_imagem_doc_base'}
objects = {'grafico_gerado.png': 'kix.71znzotqim9j'}

requests_img = []
images = service_drive.files().list(q="mimeType='image/png' and '"+img_folder_id+"' in parents",
                                     spaces='drive',
                                     fields='files(id, name, webContentLink)').execute()

for img in images['files']:
    if img['name'] in objects.keys():
        print('Substituindo a imagem ' + img['name'])
        service_drive.permissions().create(body={"role": "reader", "type": "anyone"}, fileId=img['id']).execute()
        requests_img = [{
            'replaceImage': {
                'imageObjectId': objects[img['name']],
                'uri': img['webContentLink']
            },
        },
        ]
    service_docs.documents().batchUpdate(documentId=doc_id, body={'requests': requests_img}).execute()
print('Imagens alteradas!')
```

Desenvolvimento do código

RELATÓRIO DE `{{data_doc}}`

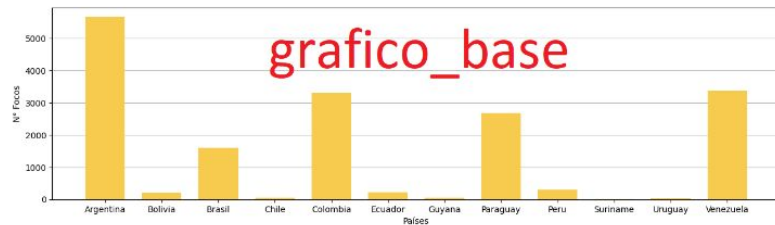


Gráfico que mostra que o maior número de focos foi `{{num_max_focos}}`, tendo sido registrado no país `{{pais_max_focos}}`.

RELATÓRIO DE 21/01/2022 10:35

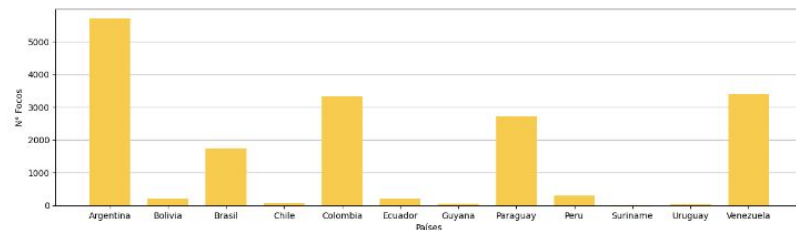


Gráfico que mostra que o maior número de focos foi **5709**, tendo sido registrado no país **Argentina**.

Conclusão

- Processo recompensador após o código estar pronto;
- Ideal para colocar no crontab;
- Inúmeras possibilidades de aplicações.

Download de arquivos utilizados: bit.ly/ArquivosWebinarINPE



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS



Obrigado pela Atenção!



INPE - Instituto Nacional de Pesquisas Espaciais
Programa Queimadas
Monitoramento por Satélites

Christopher Frige Lima Santos
Programa Queimadas
christopher.santos@inpe.br
www.inpe.br/queimadas