

ENVIAR E-MAIL PYTHON GMAIL

CONFIGURAÇÃO DO SERVIÇO DO GOOGLE

<https://myaccount.google.com/lesssecureapps?pli=1>
(<https://myaccount.google.com/lesssecureapps?pli=1>)

IMPORTAR MÓDULOS

```
In [1]: import smtplib
import mimetypes
import os

from email import encoders
from email.mime.text import MIMEText
from email.mime.audio import MIMEAudio
from email.mime.base import MIMEBase
from email.mime.image import MIMEImage
from email.mime.multipart import MIMEMultipart
```

E-MAIL TEXTO

```
In [2]: def emailgoogle(destinatario, titulo, mensagemTXT):
        '''FUNÇÃO PARA ENVIAR E-MAIL TEXTO'''

        try:
            # CONFIGURAÇÕES DO SERVIDOR DE E-MAIL
            smtp_ssl_host = 'smtp.gmail.com'
            smtp_ssl_port = '465'

            # CONFIGURAÇÕES DA CONTA LOGIN E SENHA
            username = 'pythontouso@gmail.com'
            password = '2020Abacaxi2020'

            # E-MAIL DO REMETENTE
            from_addr = 'pythontouso@gmail.com'

            # E-MAIL DO DESTINATÁRIO
            to_addrs = destinatario

            # USAR O MÓDULO MENETEXT PARA MONTAR O TEMPLATES DA MENSAGEM
            mensagem = MIMEText(mensagemTXT)
            mensagem['subject'] = titulo
            mensagem['from'] = from_addr
            mensagem['to'] = ', '.join(to_addrs)

            # CRIAR A CONEXÃO COM O SERVIDOR DE E-MAIL
            server = smtplib.SMTP_SSL(smtp_ssl_host, smtp_ssl_port)

            # FAZER O LOGIN NO SERVIDOR DE E-MAIL
            server.login(username, password)

            # ENVIAR A MENSAGEM
            server.sendmail(from_addr, to_addrs, mensagem.as_string())

            # SAIR DO SERVIDOR
            server.quit()

            print('E-mail enviado.')
        except:
            print('Erro ao enviar e-mail')
```

TESTE ENVIO DE E-MAIL

```
In [8]: lista = '''pythontouso@gmail.com
pythontouso@gmail.com
pythontouso@gmail.com'''

lista = lista.split('\n')
```

```
In [9]: lista
```

```
Out[9]: ['pythontouso@gmail.com', 'pythontouso@gmail.com', 'pythontouso@gmail.com']
```

```
In [10]: emailgoogle(lista, 'Teste', 'Boa tarde!\nTeste de e-mail com python\n: )')
```

E-mail enviado.

```
In [11]: emailgoogle(lista[0], 'Teste', 'Boa tarde!\nMais um teste de e-mail com python\n:
```

E-mail enviado.

```
In [12]: contador = 0
for email in lista:
    emailgoogle(email, 'Teste {0}'.format(contador), 'Boa tarde!\nMais um loop c
    contador += 1
```

E-mail enviado.

E-mail enviado.

E-mail enviado.

E-MAIL TEXTO COM ANEXO

```

In [34]: def emailgoogleAnexo(destinatario, titulo, mensagemTXT, anexo):
    '''FUNÇÃO PARA ENVIAR E-MAIL TEXTO'''

    try:
        def adiciona_anexo(msg, filename):
            '''VERIFICA SE O ARQUIVO EXISTE E ANEXA O ARQUIVO NO E-MAIL'''

            if not os.path.isfile(filename):
                print(filename, 'Arquivo não encontrado')
                return 0

            # VERIFICA O TIPO DO ARQUIVO
            ctype, encoding = mimetypes.guess_type(filename)

            if ctype is None or encoding is not None:
                ctype = 'application/octet-stream'

            maintype, subtype = ctype.split('/', 1)

            # ANEXA O ARQUIVO CONFORME O TIPO DO ARQUIVO
            if maintype == 'text':
                with open(filename) as f:
                    mime = MIMEText(f.read(), _subtype=subtype)
            elif maintype == 'image':
                with open(filename, 'rb') as f:
                    mime = MIMEImage(f.read(), _subtype=subtype)
            elif maintype == 'audio':
                with open(filename, 'rb') as f:
                    mime = MIMEAudio(f.read(), _subtype=subtype)
            elif maintype == 'csv':
                with open(filename, 'rb') as f:
                    mime = MIMEText(f.read(), _subtype=subtype)
            else:
                with open(filename, 'rb') as f:
                    mime = MIMEBase(maintype, subtype)
                    mime.set_payload(f.read())

                encoders.encode_base64(mime)

            mime.add_header('Content-Disposition', 'attachment', filename=filename)

            msg.attach(mime)

        # CONFIGURAÇÕES DO SERVIDOR DE E-MAIL
        smtp_ssl_host = 'smtp.gmail.com'
        smtp_ssl_port = '465'

        # CONFIGURAÇÕES DA CONTA LOGIN E SENHA
        username = 'pythontouso@gmail.com'
        password = '2020Abacaxi2020'

        # E-MAIL DO REMETENTE
        from_addr = 'pythontouso@gmail.com'

        # E-MAIL DO DESTINATÁRIO
        to_addrs = destinatario

        # USAR O MÓDULO MENETEXT PARA MONTAR O TEMPLATES DA MENSAGEM
        mensagem = MIMEMultipart()
        mensagem['subject'] = titulo

```

```

mensagem['from'] = from_addr
mensagem['to'] = ', '.join(to_addrs)

# ADICIONAR MENSAGEM
mensagem.attach(MIMEText(mensagemTXT))

# ADICIONAR O ANEXO
vl = adiciona_anexo(mensagem, anexo)

if vl != 0:
    # CRIAR A CONEXÃO COM O SERVIDOR DE E-MAIL
    server = smtplib.SMTP_SSL(smtp_ssl_host, smtp_ssl_port)

    # FAZER O LOGIN NO SERVIDOR DE E-MAIL
    server.login(username, password)

    # ENVIAR A MENSAGEM
    server.sendmail(from_addr, to_addrs, mensagem.as_string())

    # SAIR DO SERVIDOR
    server.quit()

print('E-mail enviado.')
except:
    print('Erro ao enviar e-mail')

```

TESTE E-MAIL COM ANEXO

In [30]: lista

Out[30]: ['pythontouso@gmail.com', 'pythontouso@gmail.com', 'pythontouso@gmail.com']

In [31]: os.listdir()

Out[31]: ['.ipynb_checkpoints',
'2020_alunos.csv',
'2020 Nomes.txt',
'backup',
'BANCO DE DADOS.ipynb',
'banco.db',
'banco_backup.sql',
'enve',
'ENVIAR E-MAIL PYTHON GMAIL.ipynb',
'ESTATÍSTICA INICIAL.ipynb',
'MANIPULAÇÃO DE ARQUIVOS.ipynb',
'requerimentos.txt']

In [35]: emailgoogleAnexo([lista[0]], 'Teste anexo', 'Teste de anexo', '2020_alunos.csv')

E-mail enviado.

In [33]: for email in lista:
emailgoogleAnexo([email], 'Teste anexo for', 'Teste de anexo', '2020_alunos.csv')

E-mail enviado.

E-mail enviado.

E-mail enviado.

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []: