# Documentação do Programa de Captura de Anexos de Email

Este é um programa Python que faz o download de anexos de email de uma conta Gmail usando a biblioteca `imap\_tools`. Ele foi projetado para funcionar com base em configurações definidas em um arquivo JSON de configuração.

# Pré-requisitos

Antes de executar o programa, você deve garantir que os seguintes requisitos sejam atendidos:

1. Ter uma conta Gmail configurada com permissões de acesso para aplicativos menos seguros.
2. Instalar a biblioteca `imap\_tools`. Você pode instalá-la usando o seguinte comando: `pip install imap\_tools`
3. Ter um arquivo JSON de configuração com as informações necessárias (consulte o arquivo `appconfig.json`).

# Configuração

O programa requer um arquivo JSON de configuração chamado `appconfig.json` que deve estar localizado no diretório especificado. O arquivo de configuração deve conter as seguintes informações:

* - `USUARIO`: O endereço de email da conta Gmail.
* - `SENHA`: A senha da conta Gmail.
* - `DE`: O endereço de email do remetente cujos e-mails serão filtrados.
* - `PASTA\_LOG`: O diretório onde os registros de log serão armazenados.
* - `PASTA\_DESTINO`: O diretório onde os anexos dos e-mails serão salvos.

# Funcionamento

O programa faz o seguinte:

1. Carrega as configurações do arquivo JSON de configuração.
2. Inicia uma conexão com a caixa de entrada do Gmail usando as credenciais fornecidas.
3. Define as variáveis para os diretórios de anexos e logs.
4. Inicia um loop infinito para verificar periodicamente a caixa de entrada.

**Dentro do loop:**

* - Filtra os e-mails com base no remetente especificado no arquivo de configuração.
* - Obtém a data atual.
* - Verifica os anexos de cada email e faz o download de anexos não duplicados para o diretório de anexos.
* - Registra as informações do anexo no arquivo de log, evitando duplicatas.
* - Move os e-mails processados para a pasta "Feito".
* - Aguarda 30 segundos antes de verificar novamente a caixa de entrada.

# Exemplo de Uso

1. Configure o arquivo `appconfig.json` com as informações necessárias.
2. Execute o programa Python.