



PYTHON
ACADEMY



COMO INTEGRAR O CHATGPT EM SEU CÓDIGO PYTHON

Aprenda a integrar o famoso e poderoso ChatGPT a seus códigos Python de maneira rápida e fácil.

[PYTHONACADEMY.COM.BR](https://pythonacademy.com.br)

Este ebook foi gerado por



Crie Ebooks técnicos incríveis em minutos com IA

Conheça a 1ª IA Especializada na criação de Ebooks **com código!**



Chega de formatar código no Google Docs

Deixe que nossa IA faça o trabalho pesado



Syntax Highlight



Adicione Banners Promocionais



Edite em Markdown em Tempo Real



Infográficos feitos por IA

TESTE AGORA

PRIMEIRO CAPÍTULO 100% GRÁTIS

Salve salve Pythonistas 🙌

Eu tenho certeza que você já sabe o que é o ChatGPT, talvez o que não saiba é como integrá-lo ao seu código Python.

O ChatGPT é uma ferramenta criada pela OpenAI que permite a criação de bots de conversação utilizando a técnica de aprendizado profundo conhecida como Transformers.

Essa tecnologia foi desenvolvida com o objetivo de criar chatbots cada vez mais inteligentes e que possam conversar com os usuários de forma mais natural e humanizada.

O ChatGPT é baseado no modelo GPT-3, que é utilizado pela OpenAI em diversas aplicações de inteligência artificial.

O GPT-3 foi treinado com uma quantidade enorme de dados e é capaz de gerar textos com uma qualidade impressionante, muitas vezes passando por um ser humano.

Para utilizar o ChatGPT em seus projetos, é necessário fazer a integração do SDK em seu código Python.

Felizmente, a própria OpenAI disponibiliza um pacote de instalação no `pip`, tornando o processo de integração bastante simples.

Nesse artigo você vai ver como é simples, então vamos nessa!

Instalando o SDK do OpenAI

Antes de começarmos a utilizar o ChatGPT, é necessário instalar o SDK fornecido pela OpenAI. Para isso, basta executar o seguinte comando no terminal:

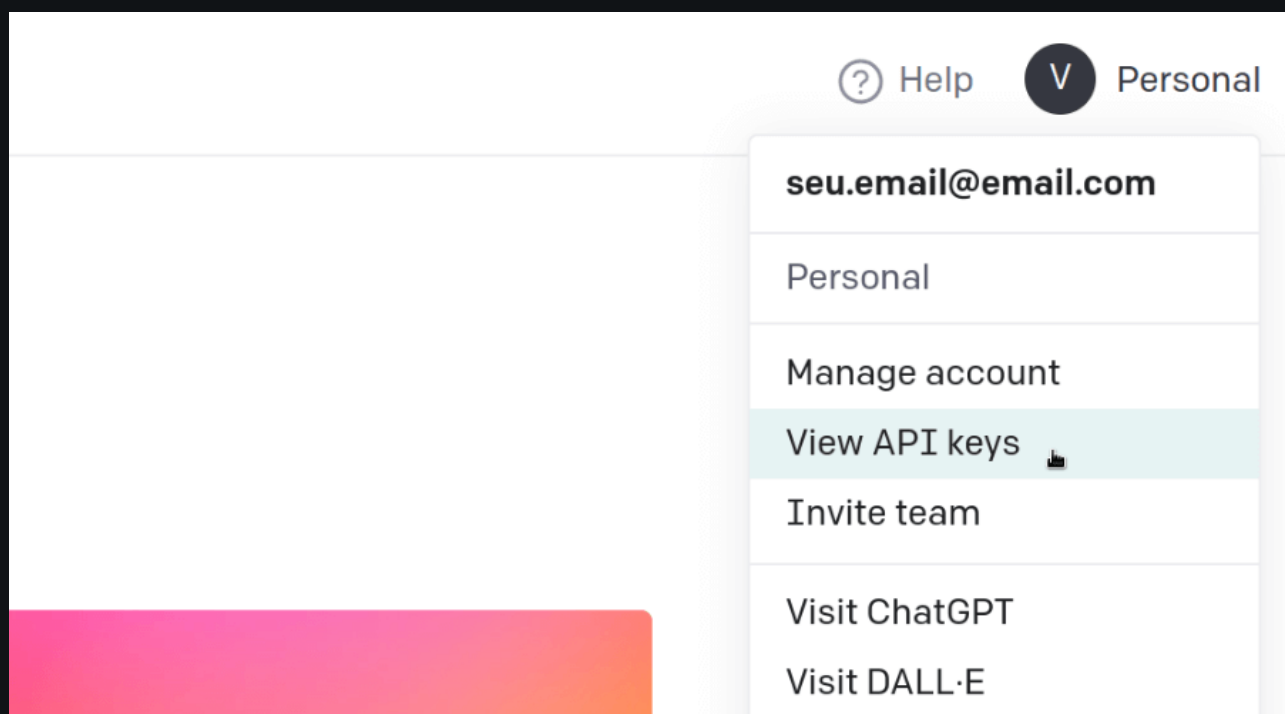
```
pip install openai
```

Com o SDK instalado, precisamos gerar uma chave de API para poder interagir com o ChatGPT.

Gerando sua chave de API

Primeiro, acesse <https://platform.openai.com> e crie uma conta, caso ainda não tenha acesso.

Após logar, vá no canto superior direito, na opção “View API keys”:



Em seguida, clique no botão “+ Create new secret key”, opcionalmente dê um nome à esta chave e **copie o código** que será gerado.

O código será similar a este:

Create new secret key

Please save this secret key somewhere safe and accessible. For security reasons, **you won't be able to view it again** through your OpenAI account. If you lose this secret key, you'll need to generate a new one.

sk-qubp49i0d3eYVJ4QpE6XT3BlbkFJWQEIc4MtaGuOKxm30



Done

Guarde-o muito bem e não compartilhe com ninguém (muito menos suba-o no Github 😱)

Ele não será mais apresentado a você. Caso não o salve, terá que gerar uma nova chave.

P.S.: a chave acima já foi deletada! 😊

Ao criar sua conta, você terá um crédito para utilizar bastante a API do ChatGPT durante um tempo pré-determinado.

Após isso, será necessário pagar para utilizá-lo, portanto aproveite seus créditos!

Agora, vamos para a parte do código!



*Estou desenvolvendo o **DevBook**, uma plataforma que usa IA para gerar ebooks técnicos profissionais. Te convido a conhecer!*

Crie Ebooks técnicos incríveis em minutos com IA

Conheça a 1ª IA Especializada na criação de Ebooks **com código!**



Chega de formatar código no Google Docs

Deixe que nossa IA faça o trabalho pesado

Syntax Highlight

Adicione Banners Promocionais

Edite em Markdown em Tempo Real

Infográficos feitos por IA

TESTE AGORA! PRIMEIRO CAPÍTULO 100% GRÁTIS

Criando uma aplicação simples com o ChatGPT

Para ilustrar como utilizar o ChatGPT em uma aplicação Python, vamos criar um chatbot simples que responde a algumas perguntas relacionadas a programação.

O objetivo aqui é apenas mostrar como utilizar a API do ChatGPT, não se trata de uma aplicação funcional em si.

```

import openai

openai.api_key = "SUA_API_KEY"

def conversar(texto):
    resposta = openai.ChatCompletion.create(
        model="gpt-3.5-turbo",
        messages=[
            {"role": "system", "content": "Suponha que você é um ins-
            trutor de Python."},
            {"role": "user", "content": texto},
        ],
        temperature=0.7,
    )
    return resposta['choices'][0]['message']['content']

while True:
    prompt = input("Usuário: ")
    resposta_chatgpt = conversar(prompt)
    print("Bot: " + resposta_chatgpt)

```

Agora vamos à explicação: - Primeiro, em `openai.api_key = "SUA_API_KEY"`, você deve colar a chave de API que foi gerada pela Plataforma da OpenAI (ali em cima no nosso tutorial 🙌) - Nesse exemplo, utilizamos a API `ChatCompletion` da OpenAI para enviar um texto para o ChatGPT. - O parâmetro `model` define qual o modelo será utilizado para processar sua requisição ([acesse aqui documentação](#) para ver quais estão disponíveis). - O parâmetro `messages` define os “atores” da sua requisição. Em `"role": "system"` você pode definir algum contexto à sua requisição (por exemplo, você pode pedir para o ChatGPT se comportar como um instrutor de um curso, ou um professor de gastronomia, ou um entrevistador técnico). Já em `"role": "user"` vai o texto de entrada que você quer perguntar ao ChatGPT. - O parâmetro `temperature` define qual “temperatura” de amostragem usar, entre 0 e 2. Valores mais altos, como `0,8`, tornarão a saída mais aleatória, enquanto valores mais baixos, como `0,2`, a tornarão mais focada e determinística.

O modelo retorna uma resposta, que é então exibida na tela.

Por exemplo, fazendo a seguinte pergunta:

Como posso definir um Decorator em Python?

A saída será a seguinte:

```
Bot: Em Python, um decorator é uma função que recebe outra função como
argumento e retorna uma nova função.
O objetivo é adicionar funcionalidades à função original sem modificá-
la diretamente. A sintaxe básica para
definir um decorator em Python é a seguinte:
{% raw %}```
def decorator_func(original_func):
    def wrapper_func(*args, **kwargs):
        # Código executado antes da função original
        result = original_func(*args, **kwargs)
        # Código executado após a função original
        return result
    return wrapper_func
```${% endraw %}

... continua
```

Muito fera! 🚀

## Conclusão

Parabéns Pythonista!

Você acabou de fazer seu primeiro código de integração com o ChatGPT utilizando Python! 🙌🙌🙌



Integrar o ChatGPT em aplicações Python é uma tarefa bastante simples, graças ao SDK fornecido pela OpenAI.

Com essa tecnologia, é possível criar chatbots cada vez mais inteligentes e humanizados, o que pode ser muito útil em diversos cenários, como atendimento ao cliente e suporte técnico.

Com esse tutorial, você já tem o conhecimento básico para começar a criar aplicações utilizando o ChatGPT em seus projetos em Python.

Lembre-se de seguir as melhores práticas de programação e de sempre documentar seu código para facilitar o trabalho de outros desenvolvedores que eventualmente trabalhem no mesmo projeto.

Nos vemos na próxima!

Não se esqueça de conferir!



DevBook

# Crie Ebooks técnicos em minutos com IA

Conheça a 1ª IA Especializada na criação de Ebooks **com código!**



Chega de formatar código no Google Docs



 Syntax Highlight

 Infográficos feitos por IA

 Adicione Banners Promocionais

Deixe que nossa IA faça o trabalho pesado

 Edite em Markdown em Tempo Real

**TESTE AGORA** 

 PRIMEIRO CAPÍTULO 100% GRÁTIS