Inteligencia artificial y programación

Asistente de programación tipo “Copilot” con CodeGPT y Llama3 (Gratis y privado)

Autor: Sergi García

Actualizado Septiembre 2024

Licencia

Nomenclatura

A lo largo de este tema se utilizarán diferentes símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

📖 **Importante**

❗ **Atención**

💬 **Interesante**

ÍNDICE

1. Introducción **3**
1.1 ¿Qué software utilizaremos? **3**
2. Instalando Ollama **3**
2.1 Instalando Ollama en local 3
2.2 Instalando Ollama dockerizado 3
2.2.1 Instalando Ollama dockerizado, solo CPU 4
2.2.2 Instalando Ollama dockerizado, con GPU Nvidia 4
3. Obteniendo modelo "llama3" para Ollama **4**
3.1 Obteniendo el modelo Llama3 si instalaste tu Ollama en local 4
3.2 Obteniendo el modelo Llama3 si instalaste Ollama dockerizado 4
4. Instalando Visual Studio Code **4**
5. Instalando extensión CodeGPT para Visual Studio Code **4**
6. Configurando la extensión para uso como Chat **5**
7. Instalando otros modelos de Ollama **6**

ASISTENTE DE PROGRAMACIÓN TIPO "COPILOT" CON CODEGPT Y LLAMA 3 (GRATIS Y PRIVADO)

1. INTRODUCCIÓN

En el mundo de la programación, la demanda de herramientas eficientes y poderosas para agilizar el desarrollo de código es más evidente que nunca. La evolución constante de la tecnología y la creciente complejidad de los proyectos exigen soluciones innovadoras que no solo aceleren el proceso de codificación, sino que también mejoren la calidad y la precisión del código resultante.

La proliferación de herramientas de asistencia en programación, como Copilot o incluso ChatGPT, ha sido bien recibida en la comunidad de desarrollo de software gracias a su capacidad para acelerar el proceso de codificación. Sin embargo, surge un dilema significativo asociado con muchas de estas soluciones existentes: el costo de adquisición/suscripción y la preocupación por la privacidad (envías tu código a un tercero).

Para abordar estas inquietudes, en este documento ofrecemos una alternativa gratuita y "self-hosted" para garantizar la máxima privacidad.

1.1 ¿Qué software utilizaremos?

Para esta tarea, en resumen, utilizaremos el siguiente software:

- **Ollama**: https://ollama.ai
  - Software para descargar, preparar y lanzar distintos modelos LLMs de LLama.
  - El modelo "LLama 3". Lo descargaremos usando Ollama.
- **Visual Studio Code:** https://code.visualstudio.com/
  - Editor de código ligero creado por Microsoft, con multitud de plugins disponibles.
- **Extensión de Visual Studio Code "CodeGPT":** https://www.codegpt.co/
  - Extensión:https://marketplace.visualstudio.com/items?itemName=DanielSanMedium.dscodegpt&ssr=false.¡Cuidado! Hay otras extensiones con este nombre, recomendamos instalar desde el enlace.

> ! **Atención:** los requisitos mínimos pueden variar según el modelo utilizado. Por lo general siguen esta norma, según los billones (americanos) de parámetros: 7b -> 8GB de RAM, 13b -> 16 GB de RAM y 43b -> 32 GB de RAM

2. INSTALANDO OLLAMA

2.1 Instalando Ollama en local

Descarga tu versión de Ollama de https://ollama.ai/download y sigue las instrucciones de instalación según tu sistema operativo.

Por ejemplo, en sistemas Linux ejecuta como "root" los siguientes comandos:

```
apt update && apt install curl -y
curl -fsSL https://ollama.com/install.sh | sh
```

2.2 Instalando Ollama dockerizado

Para ver como instalar Ollama de forma dockerizado, seguiremos la siguiente entrada: https://ollama.ai/blog/ollama-is-now-available-as-an-official-docker-image

Si no tienes instalado Docker o no sabes utilizarlo, puedes encontrar más información en mi curso https://github.com/sergarb1/CursoIntroduccionADocker

2.2.1 Instalando Ollama dockerizado, solo CPU

Si queremos usar la imagen "CPU only" con este comando tenemos suficiente:

```
docker run -d -v ollama:/root/.ollama -p 11434:11434 --name ollama
ollama/ollama
```

2.2.2 Instalando Ollama dockerizado, con GPU Nvidia

Si tenemos una tarjeta gráfica Nvidia y queremos aprovechar su potencia para esta imagen dockerizada, debemos dar los siguientes pasos:

1) Instalar y configurar Nvidia container toolkit. Los distintos pasos están en https://docs.nvidia.com/datacenter/cloud-native/container-toolkit/latest/install-guide.html#installation
2) Una vez configurado, puedes poner en marcha el contenedor que utiliza la potencia de tu GPU con el siguiente comando:

```
docker run -d --gpus=all -v ollama:/root/.ollama -p 11434:11434 --name
ollama ollama/ollama
```

3. OBTENIENDO MODELO "LLAMA3" PARA OLLAMA

3.1 Obteniendo el modelo Llama3 si instalaste tu Ollama en local

En el caso de que hayas instalado Ollama en local, simplemente ejecuta estos dos comandos:

```
ollama pull llama3
ollama pull llama3:instruct
```

Con esto, estará todo listo para configurar la extensión de Visual Studio Code "CodeGPT".

3.2 Obteniendo el modelo Llama3 si instalaste Ollama dockerizado

En el caso de que hayas instalado Ollama de forma dockerizada, simplemente ejecuta estos dos comandos:

```
docker exec -it ollama ollama pull llama3
docker exec -it ollama ollama pull llama3:instruct
```

Con esto, estará todo listo para configurar la extensión de Visual Studio Code "CodeGPT".

4. INSTALANDO VISUAL STUDIO CODE

La instalación del editor Visual Studio Code es sencilla y está ampliamente documentada en tutoriales, videos, etc. y en multitud de formatos (ejecutable Windows, paquete ".deb", paquete Snap, etc.). Para esto, simplemente, te recomiendo que descargues el software de https://code.visualstudio.com/ y siguas las instrucciones.

5. INSTALANDO EXTENSIÓN CODEGPT PARA VISUAL STUDIO CODE

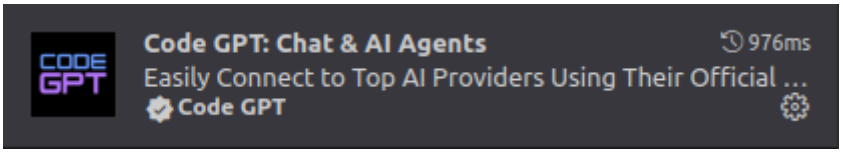
¡Cuidado! Hay otras extensiones con este nombre, recomendamos instalar desde el enlace.

https://marketplace.visualstudio.com/items?itemName=DanielSanMedium.dscodegpt&ssr=false

Además, también puedes instalarla si desde dentro de Visual Studio Code, pulsas la combinación de teclas "Control + P" y en la caja emergentes pegas este comando:

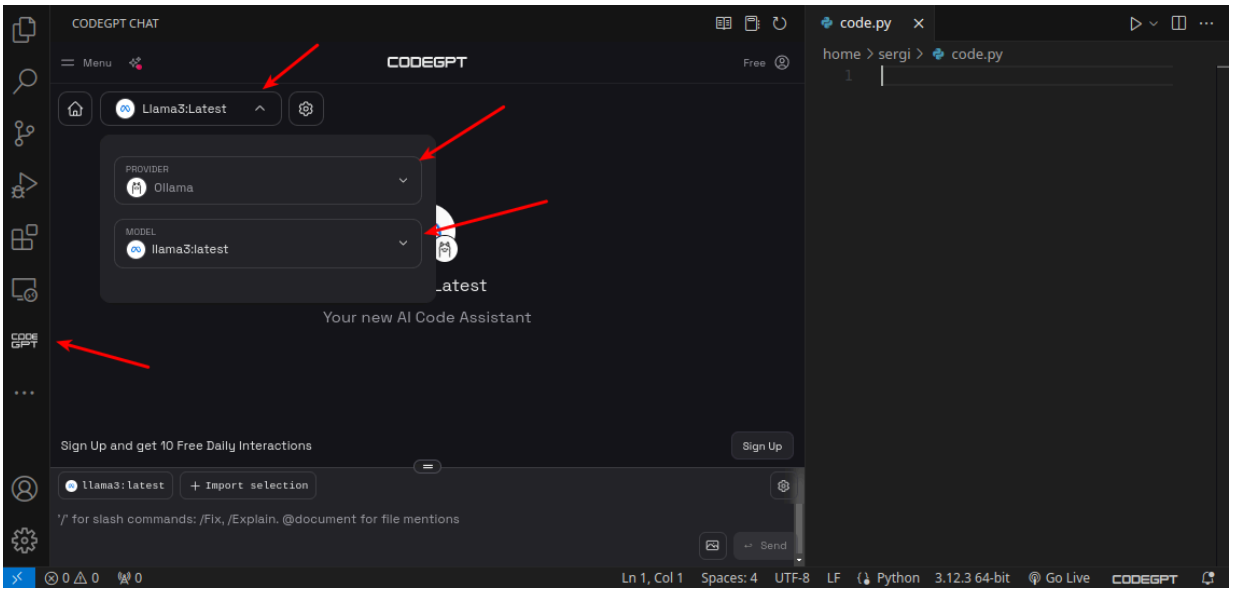
```
ext install DanielSanMedium.dscodegpt
```

Si todo va bien, tu Visual Studio Code tendrá instalada esta extensión:



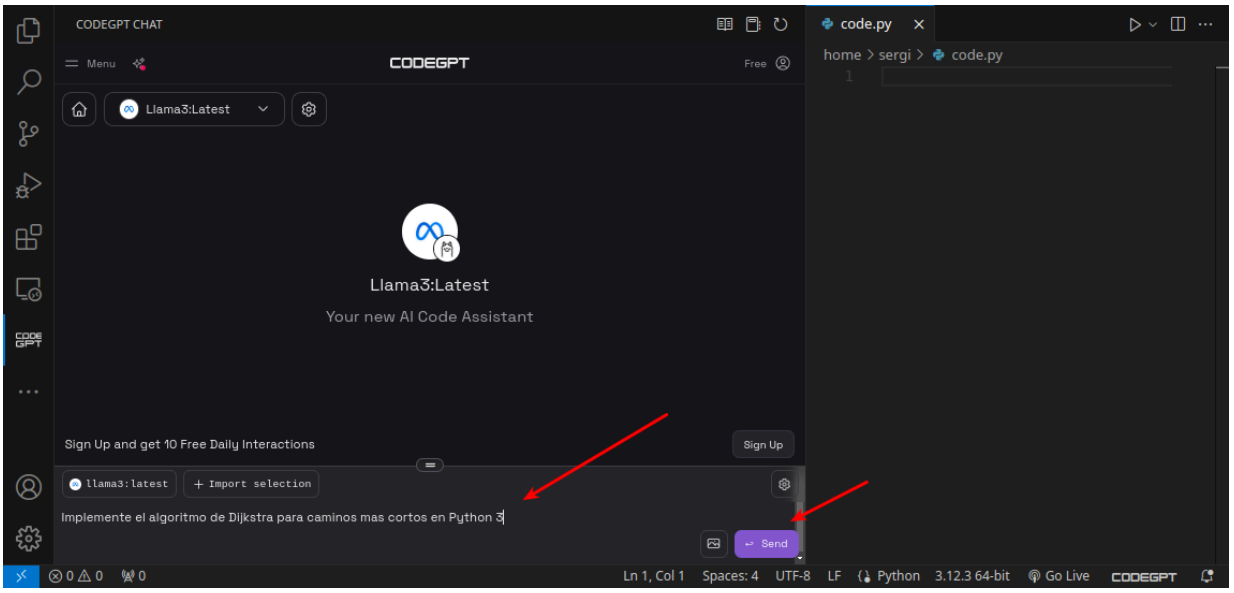
6. CONFIGURANDO LA EXTENSIÓN PARA USO COMO CHAT

Una vez lista la extensión, ábrela e indica que estás usando "Ollama" y el modelo "Llama3", como se puede ver en la siguiente imagen:

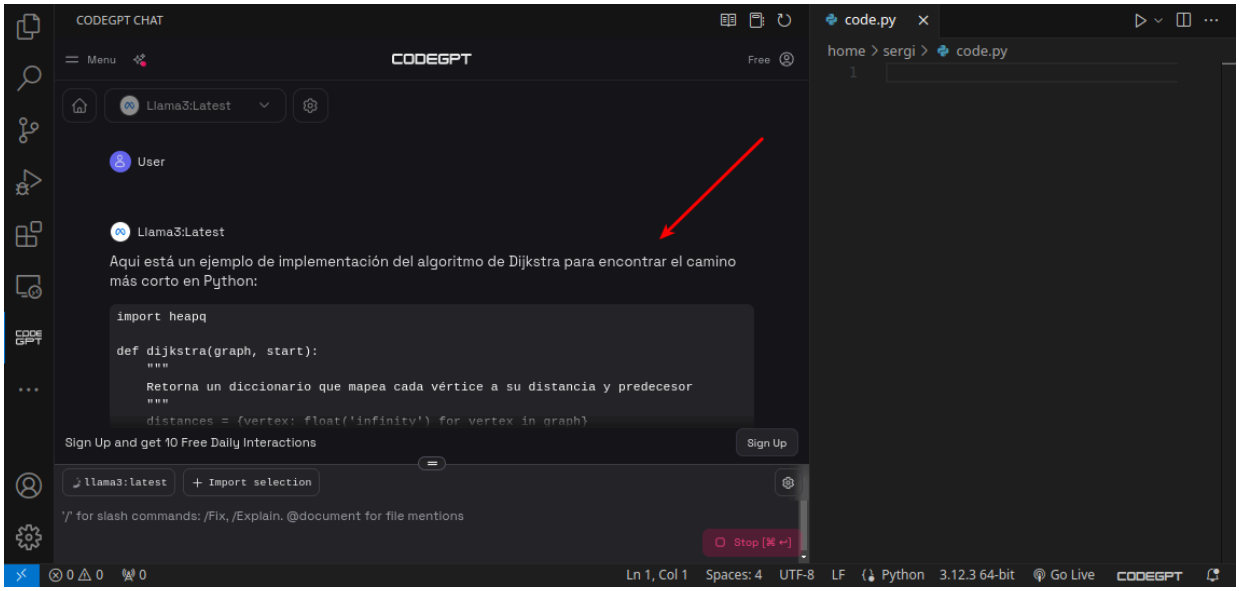


Con esta configuración, tendrás un chat similar a "Chat GPT" para tu código.

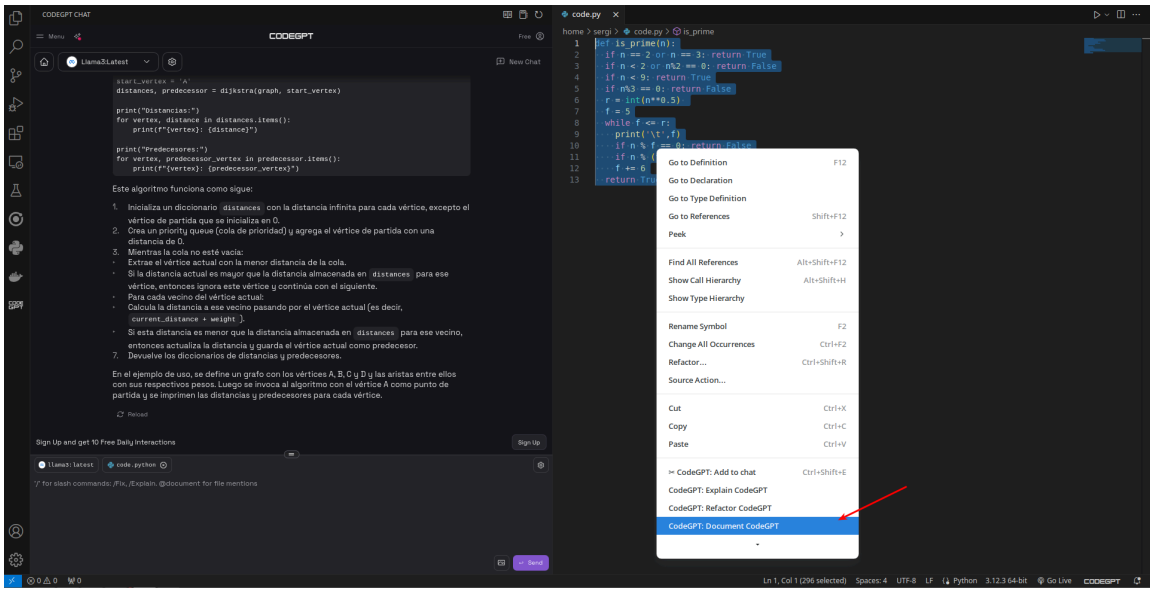
Aquí un ejemplo de uso del Chat:



Aquí una porción de la respuesta obtenida:



También se puede, simplemente, seleccionando código y pulsando botón derecho:



7. INSTALANDO OTROS MODELOS DE OLLAMA

En este ejemplo hemos utilizado el modelo "Llama3", pero realmente pueden utilizarse otros modelos como "mistral" u otros que surjan en el futuro (incluso en Ollama se puede importar cualquier modelo en formato GGUF tal como indican aquí https://github.com/jmorganca/ollama)

Un ejemplo para añadir "codellama":

a) Si tienes una instalación local de Ollama, puedes añadirlo con:

```
ollama pull mistral
ollama pull mistral:instruct
```

b) Si tienes una instalación dockerizada de Ollama, puedes añadirlo con:

```
docker exec -it ollama ollama pull mistral
docker exec -it ollama ollama pull mistral:instruct
```