

Uso de GitHub Copilot con Python

3 minutos

En unidades anteriores, se mostró cómo configurar Copilot y se mencionó cómo puede ayudarle a ser más rápido como desarrollador que empieza a escribir código.

En esta unidad, vamos a analizar cómo Copilot puede ayudarle con proyectos existentes y con tareas más complicadas.

Desarrollo con GitHub Copilot

A menudo, cuando compilamos proyectos, es necesario asegurarnos de que nuestro código esté actualizado y sea actualizado continuamente. Además, es posible que tengamos que corregir los errores que surjan o agregar nuevas características para mejorar su funcionalidad y facilidad de uso. Vamos a explorar algunas maneras de realizar actualizaciones con GitHub Copilot.

Ingeniería rápida

Aunque GitHub Copilot puede sugerir código a medida que escribe, también puede crear consultas para generar sugerencias útiles. Una consulta, que sería nuestra entrada, es una colección de instrucciones o directrices que ayudan a generar código. La consulta es útil para generar respuestas específicas de Copilot. La consulta puede ser un comentario que dirija a Copilot para que genere código por usted o escribir código para que Copilot lo autocomplete.

La calidad de la salida de Copilot depende de la forma en que se elabora la consulta. El diseño de una consulta eficaz es, por lo tanto, fundamental para garantizar lograr los resultados deseados. Por ejemplo, si hace la consulta siguiente:

```
Python
```

```
# Create an API endpoint
```

Dado que la consulta es ambigua y vaga, es posible que el resultado de GitHub Copilot no sea lo que necesita. Por ejemplo, podría usar un marco que no conozca o un punto de conexión que requiera datos que no reconozca. Por ejemplo, si hace la consulta siguiente:

Python

```
# Create an API endpoint using the FastAPI framework that accepts a JSON payload in a POST request
```

Esta última consulta es específica, clara y permite a GitHub Copilot comprender el objetivo y el ámbito de la tarea. También puede proporcionar contexto y ejemplos a Copilot mediante comentarios y código. Hacer una buena consulta garantiza que el modelo genere una salida de alta calidad.

Procedimientos recomendados con GitHub Copilot

Copilot impulsa la productividad, pero requiere algunos procedimientos recomendados para garantizar la calidad. Algunos procedimientos recomendados al usar Copilot son:

1. Mantenga las consultas simples y agregue componentes más elaborados a medida que va avanzando, por ejemplo:

text

```
create an HTML form with a text field and button
```

Luego, profundice más en la consulta para obtener sugerencias más específicas:

text

```
Add an event listener to the button to send a POST request to /generate endpoint and display response in a div with id "result"
```

Alterne entre sugerencias, puede hacerlo mediante `Ctrl+Enter` (o `Cmd+Enter` en un equipo Mac). Obtendrá varias sugerencias de Copilot y puede elegir la mejor salida.

Si no logra avanzar o no obtiene los resultados que desea, puede replantear la consulta o empezar a escribir código para que Copilot lo autocomplete.

❗ Nota

GitHub Copilot usa los archivos abiertos en el editor de texto como contexto adicional. Esto es útil porque proporciona información adicional a la consulta o al código que podría estar

escribiendo. Si necesita que GitHub Copilot proporcione sugerencias basadas en otros archivos, recuerde abrirlos para obtener sugerencias más precisas.

Módulo completado:

Desbloquear logro
