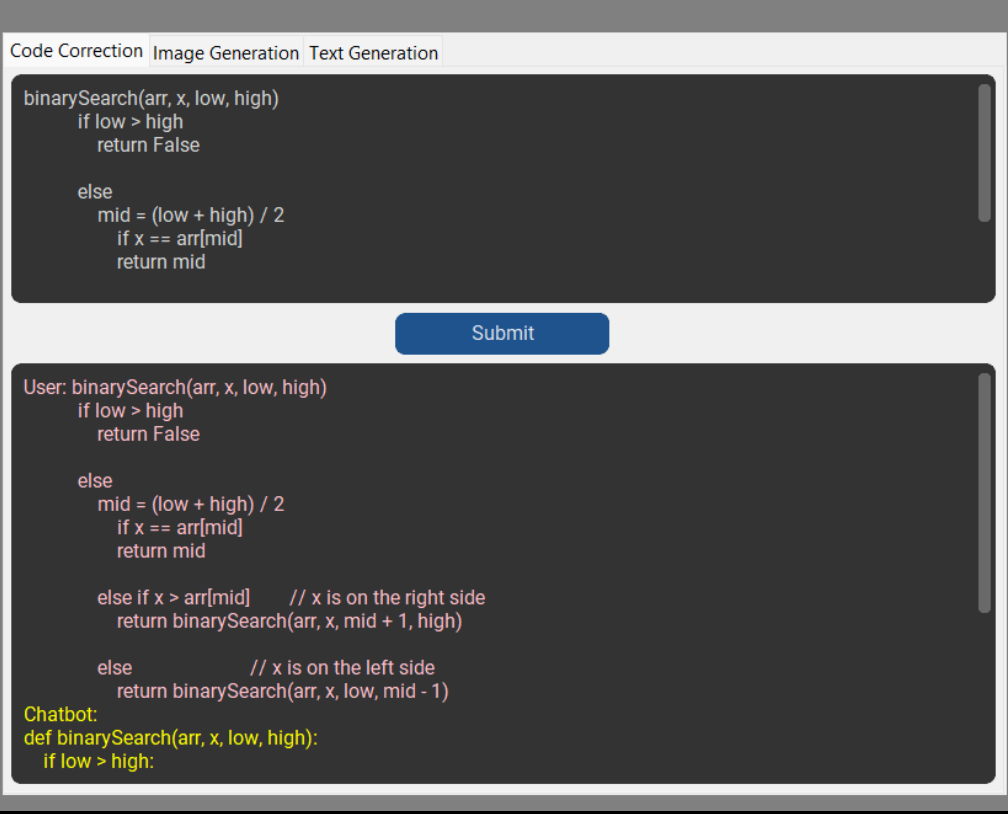


# HOW TO USE! 1

## ChatBots OpenAI APIs

First Bot we see is code correction, it is prompt to fix any python code to best of its ability also it can even write python code from just algorithm provided. It uses ChatGPT for its working.

Write the algorithm or code that needs fixing in the box provided and click [Submit] (See Fig. 1.1,1.2).



The screenshot shows a web interface with three tabs: "Code Correction", "Image Generation", and "Text Generation". The "Code Correction" tab is active. It contains a text area with the following Python code:

```
binarySearch(arr, x, low, high)
if low > high
    return False

else
    mid = (low + high) / 2
    if x == arr[mid]
        return mid
```

Below the text area is a blue "Submit" button. The response area below the button shows the corrected code:

```
User: binarySearch(arr, x, low, high)
if low > high
    return False

else
    mid = (low + high) / 2
    if x == arr[mid]
        return mid

    else if x > arr[mid]    // x is on the right side
        return binarySearch(arr, x, mid + 1, high)

    else                  // x is on the left side
        return binarySearch(arr, x, low, mid - 1)

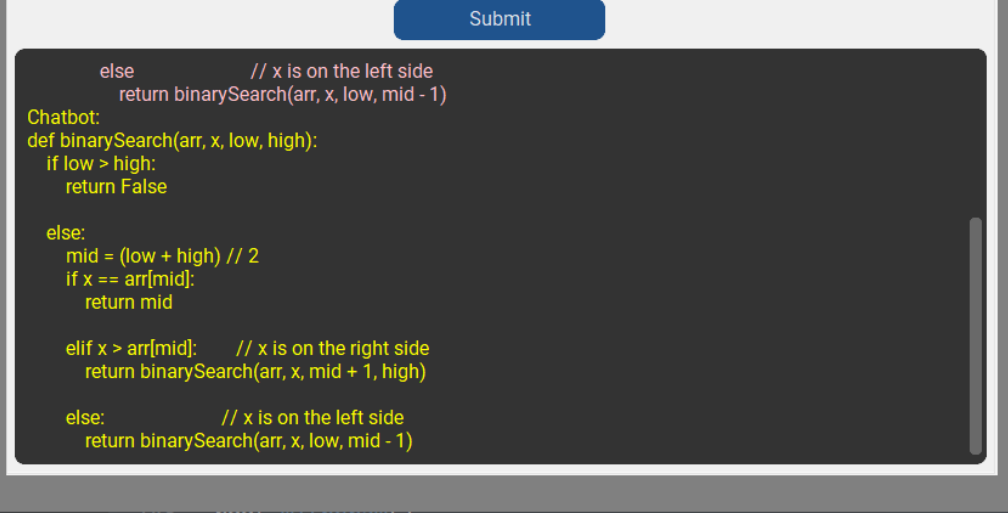
Chatbot:
def binarySearch(arr, x, low, high):
    if low > high:
        return False

    else:
        mid = (low + high) // 2
        if x == arr[mid]:
            return mid

    elif x > arr[mid]:    // x is on the right side
        return binarySearch(arr, x, mid + 1, high)

    else:                // x is on the left side
        return binarySearch(arr, x, low, mid - 1)
```

Fig. 1.1



The screenshot shows the same web interface as Fig. 1.1, but with a different code correction example. The "Code Correction" tab is active. It contains a text area with the following Python code:

```
else
    // x is on the left side
    return binarySearch(arr, x, low, mid - 1)

Chatbot:
def binarySearch(arr, x, low, high):
    if low > high:
        return False

    else:
        mid = (low + high) // 2
        if x == arr[mid]:
            return mid

    elif x > arr[mid]:    // x is on the right side
        return binarySearch(arr, x, mid + 1, high)

    else:                // x is on the left side
        return binarySearch(arr, x, low, mid - 1)
```

Fig. 1.2

2

Second Bot is Image Generation it uses user input and sends it to DALL-E for image generation, a URL is generated and urllib and Pillow libraries are used to display the image.

Write the description of image in the first text box, request sometimes takes time considering DALL-E takes time to process request and sometimes servers are dealing with more traffic, (See Fig. 2.1,2.2,2.3).

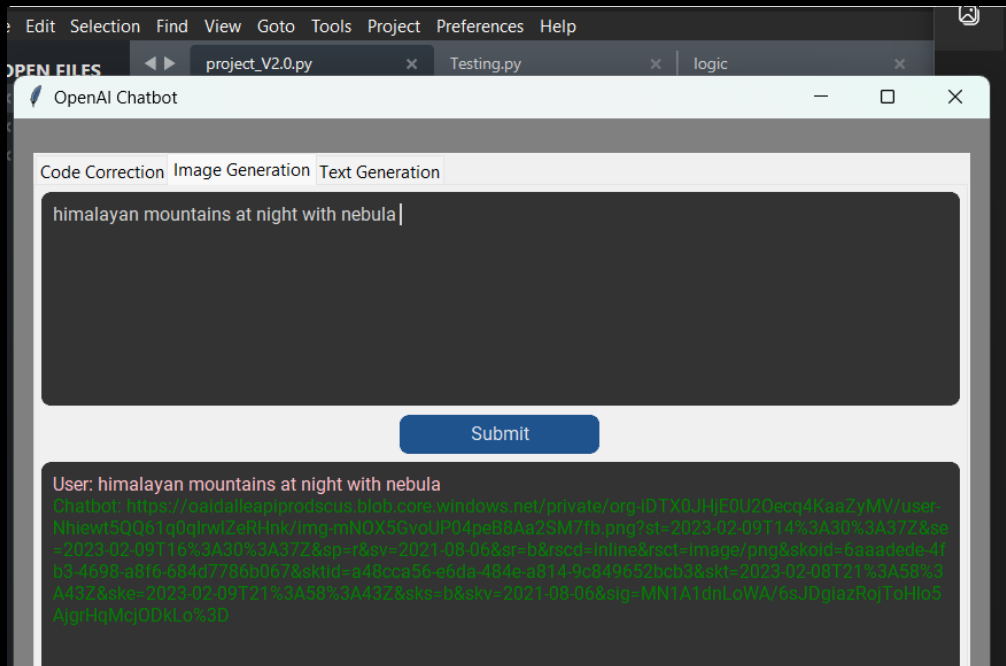


Fig. 2.1

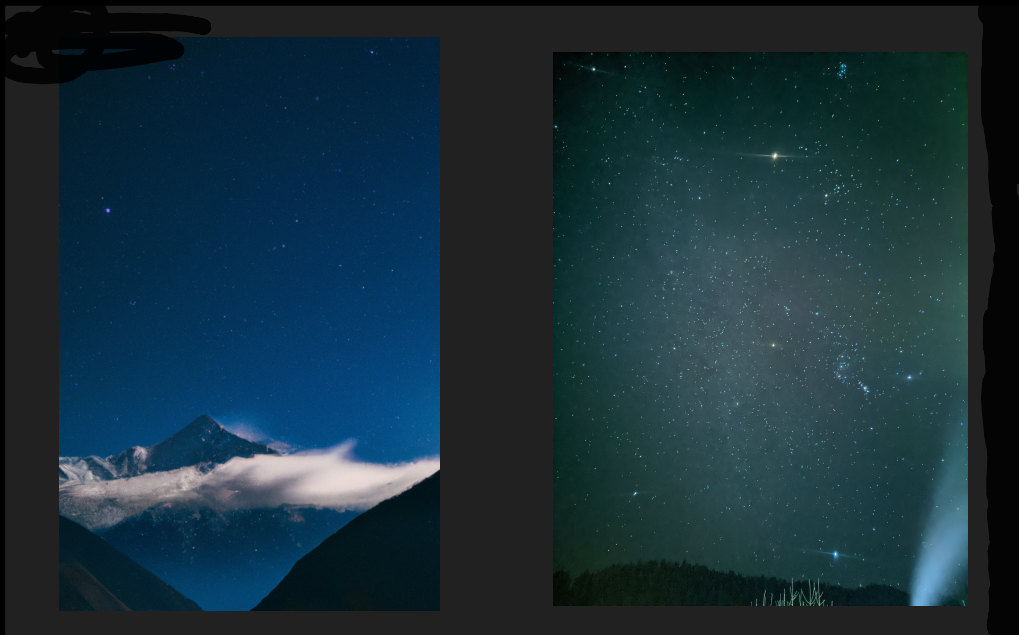


Fig. 2.2

Fig. 2.3

### 3

Third Bot is Text Generation, it reads user input and sends it to GPT-3 servers and gets the result. It can write anything from a rap song to essay about the above images.

Write your request in the text box and click [Submit]. (See Fig 3).

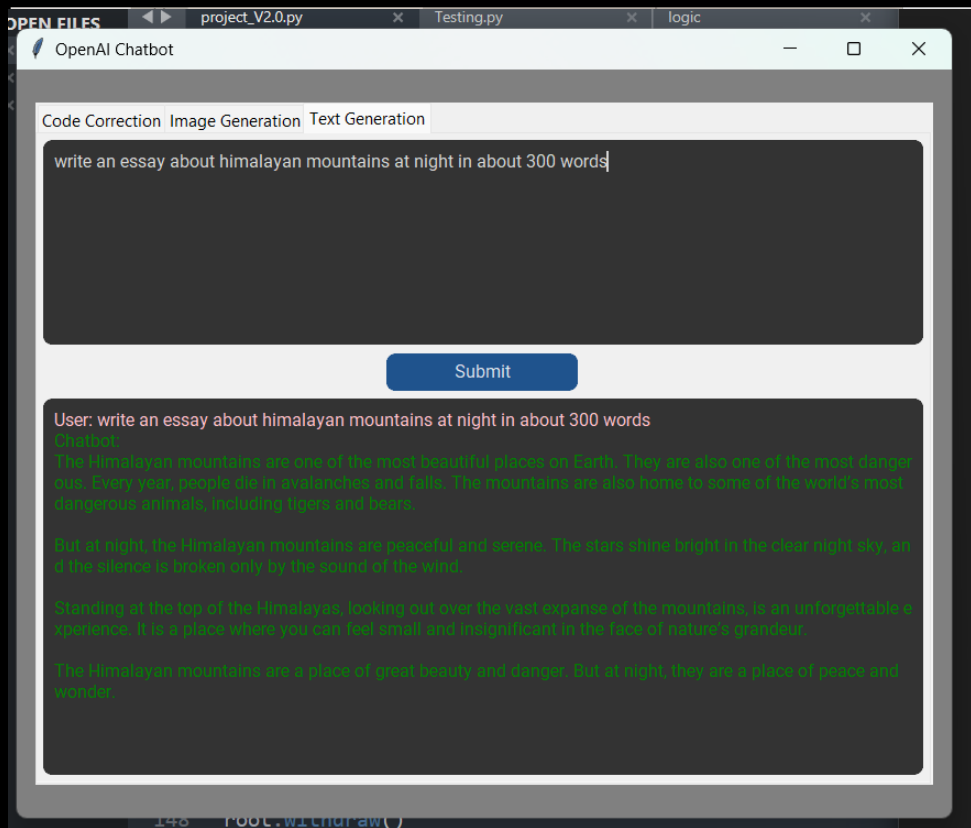


Fig. 3