**Stegno.py**

from PIL import Image

def message\_to\_binary(message):

binary\_message = ''.join(format(ord(char), '08b') for char in message)

return binary\_message

def encode\_image(image\_path, message):

img = Image.open(image\_path)

binary\_message = message\_to\_binary(message)

if len(binary\_message) > img.size[0] \* img.size[1] \* 3:

raise Exception("Message is too long to be encoded in the image.")

index = 0

encoded\_pixels = img.getdata()

encoded\_list = list(encoded\_pixels)

for i in range(len(encoded\_list)):

pixel = list(encoded\_list[i])

for j in range(3):

if index < len(binary\_message):

pixel[j] = pixel[j] & 0xFE | int(binary\_message[index])

index += 1

else:

break

encoded\_list[i] = tuple(pixel)

encoded\_image = Image.new(img.mode, img.size)

encoded\_image.putdata(encoded\_list)

encoded\_image.save("encoded\_image.png")

print("Image encoded successfully.")

def decode\_image(encoded\_image\_path):

encoded\_image = Image.open(encoded\_image\_path)

binary\_message = ""

for pixel in encoded\_image.getdata():

for value in pixel:

binary\_message += str(value & 1)

message = ""

for i in range(0, len(binary\_message), 8):

byte = binary\_message[i:i+8]

message += chr(int(byte, 2))

if message[-5:] == "#####":

break

print("Decoded message:", message[:-5])

# Example usage:

message = "Sanjay 8+1"

encode\_image("original\_image.png", message)

decode\_image("encoded\_image.png")

**Decode.py**

from PIL import Image # Import the Image module from Pillow

# Define the function to decode the image

def decode\_image(encoded\_image\_path):

encoded\_image = Image.open(encoded\_image\_path)

binary\_message = ""

for pixel in encoded\_image.getdata():

for value in pixel:

binary\_message += str(value & 1)

message = ""

for i in range(0, len(binary\_message), 8):

byte = binary\_message[i:i+8]

message += chr(int(byte, 2))

if message[-5:] == "#####":

break

print("Decoded message:", message[:-5])

# Path to the encoded image

encoded\_image\_path = "encoded\_image.png"

# Call the function to decode the image and print the secret message

decode\_image(encoded\_image\_path)