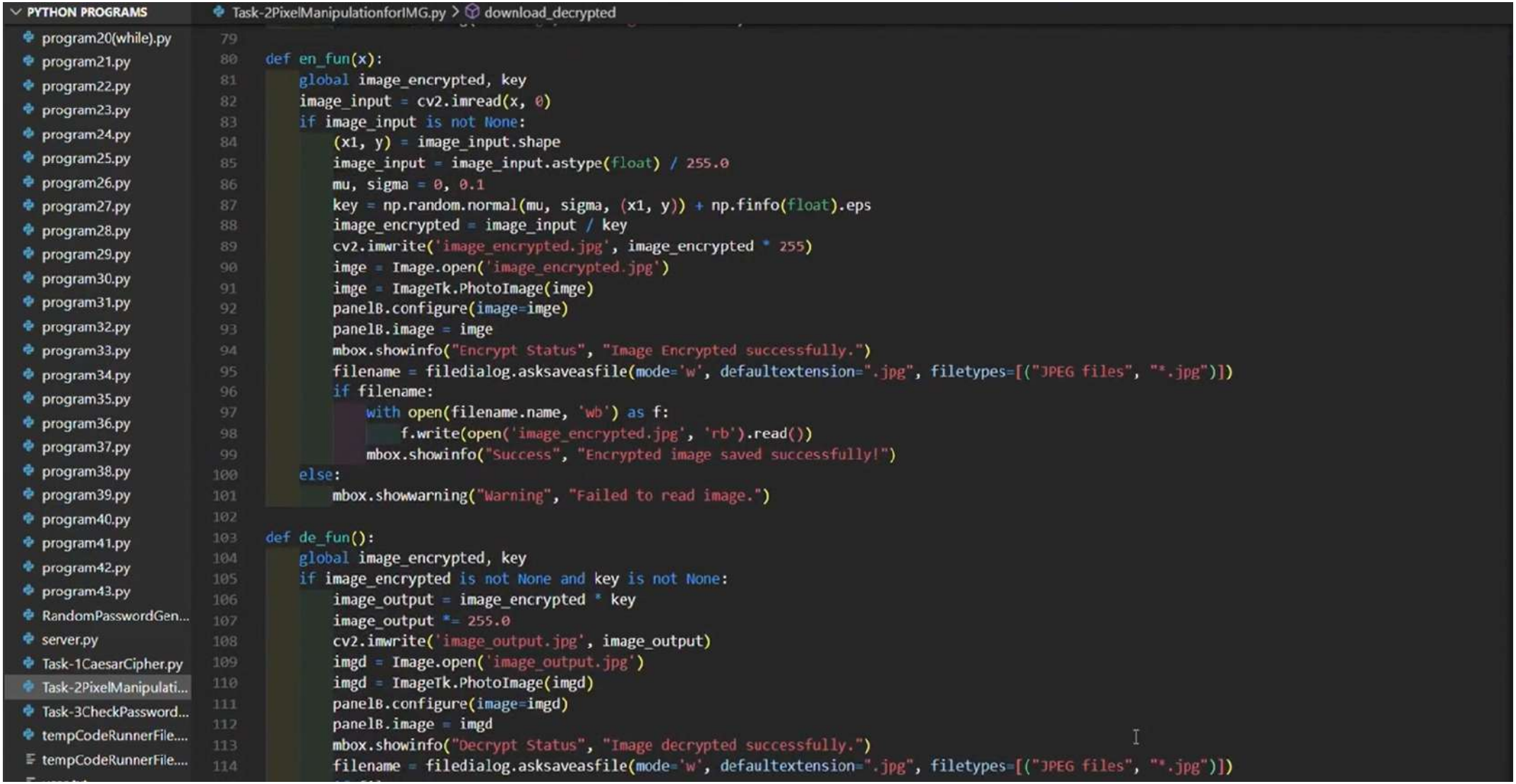
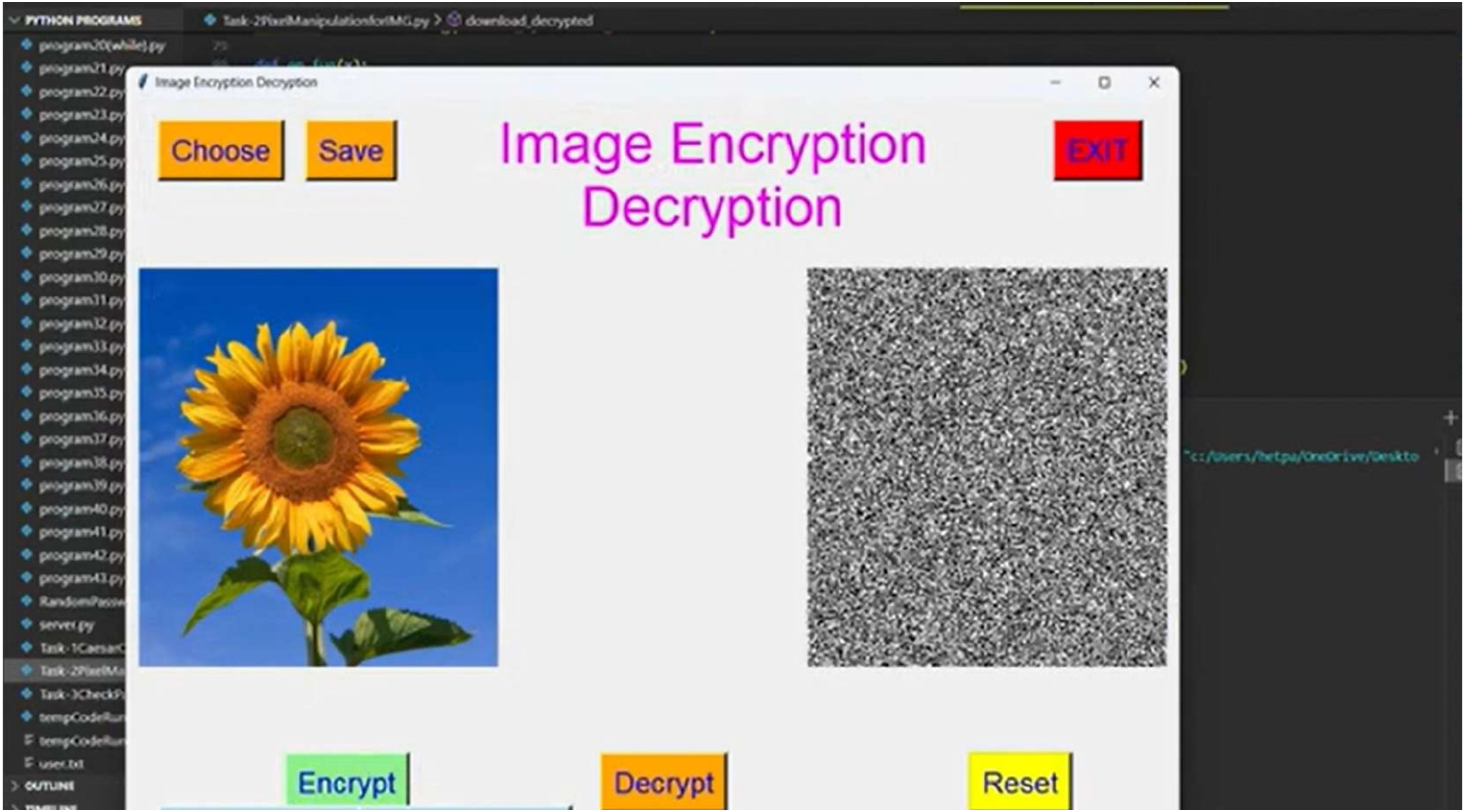

PYTHON PROGRAMS
program20(while).py
program21.py
program22.py
program23.py
program24.py
program25.py
program26.py
program27.py
program28.py
program29.py
program30.py
program31.py
program32.py
program33.py
program34.py
program35.py
program36.py
program37.py
program38.py
program39.py
program40.py
program41.py
program42.py
program43.py
RandomPasswordGen...
server.py
Task-1CaesarCipher.py
Task-2PixelManipulati...
Task-3CheckPassword...
tempCodeRunnerFile....
tempCodeRunnerFile....
Task-2PixelManipulationforIMG.py > download_decrypted
def en_fun(x):
global image_encrypted, key
image_input = cv2.imread(x, 0)
if image_input is not None:
(x1, y) = image_input.shape
image_input = image_input.astype(float) / 255.0
mu, sigma = 0, 0.1
key = np.random.normal(mu, sigma, (x1, y)) + np.finfo(float).eps
image_encrypted = image_input / key
cv2.imwrite('image_encrypted.jpg', image_encrypted * 255)
imge = Image.open('image_encrypted.jpg')
imge = ImageTk.PhotoImage(imge)
panelB.configure(image=imge)
panelB.image = imge
mbox.showinfo("Encrypt Status", "Image Encrypted successfully.")
filename = filedialog.asksaveasfile(mode='w', defaultextension=".jpg", filetypes=[("JPEG files", "*.jpg")])
if filename:
with open(filename.name, 'wb') as f:
f.write(open('image_encrypted.jpg', 'rb').read())
mbox.showinfo("Success", "Encrypted image saved successfully!")
else:
mbox.showwarning("Warning", "Failed to read image.")
def de_fun():
global image_encrypted, key
if image_encrypted is not None and key is not None:
image_output = image_encrypted * key
image_output *= 255.0
cv2.imwrite('image_output.jpg', image_output)
imgd = Image.open('image_output.jpg')
imgd = ImageTk.PhotoImage(imgd)
panelB.configure(image=imgd)
panelB.image = imgd
mbox.showinfo("Decrypt Status", "Image decrypted successfully.")
filename = filedialog.asksaveasfile(mode='w', defaultextension=".jpg", filetypes=[("JPEG files", "*.jpg")])


PYTHON PROGRAMS
Image Encryption Decryption
Choose
Save
Image Encryption
Decryption
EXIT
Encrypt
Decrypt
Reset


PYTHON PROGRAMS
Task-29PixelManipulationforIMG.py > download_decrypted
Image Encryption Decryption
Choose
Save
Image Encryption
Decryption
EXIT