def is\_valid\_password(password):

if len(password) < 8:

return False, 'Password must be at least 8 characters long.'

if not re.search(r'\d', password):

return False, 'Password must include at least one number.'

if not re.search(r'[!@#$%^&\*(),.?":{}|<>]', password):

return False, 'Password must include at least one special character.'

return True, 'Password is valid.'

# Example Test Cases

test\_passwords = ['abc123', 'abc12345', 'abc123@', 'mypassword1', 'Pass123!', '12345678', 'MyPass@']

for pwd in test\_passwords:

valid, message = is\_valid\_password(pwd)

print(f'Password: {pwd} - {message}')

Test Code Document – Password Validation Logic

def validate\_password(password):

if len(password) < 8:

return False, "Password must be at least 8 characters long"

if not re.search(r"\d", password):

return False, "Password must contain at least one digit"

if not re.search(r"[!@#$%^&\*(),.?\":{}|<>`]", password):

return False, "Password must contain at least one special character"

return True, "Password is valid"

def test\_passwords():

print(valid, msg)

print(valid, msg)

print(valid, msg)

print(valid, msg)

print(valid, msg)

print(valid, msg)

print(valid, msg)

if \_\_name\_\_ == "\_\_main\_\_":