

HOW TO TEST CODE

MANUALLY:

Manually Testing a Login Function:

```
def login(username, password):  
    # Example credentials  
    stored_username = "user123"  
    stored_password = "pass123"  
  
    if username == stored_username and password == stored_password:  
        return "Login successful"  
    else:  
        return "Login failed"
```

Test Plan for login Function:

Objective: Verify that the login function correctly validates the username and password against stored credentials.

Scope: The function login in the provided code.

Testing Type: White-Box Testing (Manual)

Test Approach: Execute the function with various test inputs and compare the actual output to the expected output.

Test Cases:

Test Case ID	Description	Input	Expected Result	Steps	Status
TC1	Correct Username and Password	username="user123" password="pass123"	"Login successful"	1. Call login ("user123", "pass123"). 2. Verify the output is "Login successful".	Pass

TC2	Incorrect Username	username= "wrong user" password= "pass123"	"Login failed"	1. Call login ("wrong user", "pass123"). 2. Verify the output is "Login failed".	Pass
TC3	Incorrect Password	username= "user123" password= "wrong pass"	"Login failed"	1. Call login ("user123", "wrong pass"). 2. Verify the output is "Login failed".	Pass
TC4	Both Username and Password Incorrect	username= "wrong user" password= "wrong pass"	"Login failed"	1. Call login ("wrong user", "wrong pass"). 2. Verify the output is "Login failed".	Pass
TC5	Empty Username and Password	Username= "" password= ""	"Login failed"	1. Call login ("", ""). 2. Verify the output is "Login failed".	Pass
TC6	Empty Username	username= "" password= "pass123"	"Login failed"	1. Call login ("", "pass123"). 2. Verify the output is "Login failed"	Pass

TC7	Empty Password	username="user123" password=""	"Login failed"	1. Call login ("user123", ""). 2. Verify the output is "Login failed".	Pass
-----	----------------	-----------------------------------	----------------	---	------

Manually Executing the Test Cases:

Python Script for Manual Testing:

```
# Define the function to be tested
def login(username, password):
    # Example credentials
    stored_username = "user123"
    stored_password = "pass123"

    if username == stored_username and password == stored_password:
        return "Login successful"
    else:
        return "Login failed"

# Manually execute the test cases

# Test Case 1: Correct Username and Password
input_username = "user123"
input_password = "pass123"
expected_output = "Login successful"
actual_output = login(input_username, input_password)
print(f"Test Case 1 - Input: ({input_username}, {input_password}) | Expected: {expected_output} | Actual: {actual_output} | Result: {'Pass' if actual_output == expected_output else 'Fail'}")

# Test Case 2: Incorrect Username
input_username = "wronguser"
input_password = "pass123"
expected_output = "Login failed"
actual_output = login(input_username, input_password)
print(f"Test Case 2 - Input: ({input_username}, {input_password}) | Expected: {expected_output} | Actual: {actual_output} | Result: {'Pass' if actual_output == expected_output else 'Fail'}")
```

```
# Test Case 3: Incorrect Password
input_username = "user123"
input_password = "wrongpass"
expected_output = "Login failed"
actual_output = login(input_username, input_password)
print(f"Test Case 3 - Input: ({input_username}, {input_password}) |
Expected: {expected_output} | Actual: {actual_output} | Result:
{'Pass' if actual_output == expected_output else 'Fail'}")

# Test Case 4: Both Username and Password Incorrect
input_username = "wronguser"
input_password = "wrongpass"
expected_output = "Login failed"
actual_output = login(input_username, input_password)
print(f"Test Case 4 - Input: ({input_username}, {input_password}) |
Expected: {expected_output} | Actual: {actual_output} | Result:
{'Pass' if actual_output == expected_output else 'Fail'}")

# Test Case 5: Empty Username and Password
input_username = ""
input_password = ""
expected_output = "Login failed"
actual_output = login(input_username, input_password)
print(f"Test Case 5 - Input: ({input_username}, {input_password}) |
Expected: {expected_output} | Actual: {actual_output} | Result:
{'Pass' if actual_output == expected_output else 'Fail'}")

# Test Case 6: Empty Username
input_username = ""
input_password = "pass123"
expected_output = "Login failed"
actual_output = login(input_username, input_password)
print(f"Test Case 6 - Input: ({input_username}, {input_password}) |
Expected: {expected_output} | Actual: {actual_output} | Result:
{'Pass' if actual_output == expected_output else 'Fail'}")

# Test Case 7: Empty Password
input_username = "user123"
input_password = ""
expected_output = "Login failed"
actual_output = login(input_username, input_password)
print(f"Test Case 7 - Input: ({input_username}, {input_password}) |
Expected: {expected_output} | Actual: {actual_output} | Result:
{'Pass' if actual_output == expected_output else 'Fail'}")
```

Outcome:

```
PS C:\Users\AB> & "C:/Python 3.11/python.exe" c:/Users/AB/Desktop/new.py
Test Case 1 - Input: (user123, pass123) | Expected: Login successful | Actual: Login successful | Result: Pass
Test Case 2 - Input: (wronguser, pass123) | Expected: Login failed | Actual: Login failed | Result: Pass
Test Case 3 - Input: (user123, wrongpass) | Expected: Login failed | Actual: Login failed | Result: Pass
Test Case 4 - Input: (wronguser, wrongpass) | Expected: Login failed | Actual: Login failed | Result: Pass
Test Case 5 - Input: (, ) | Expected: Login failed | Actual: Login failed | Result: Pass
Test Case 6 - Input: (, pass123) | Expected: Login failed | Actual: Login failed | Result: Pass
Test Case 7 - Input: (user123, ) | Expected: Login failed | Actual: Login failed | Result: Pass
PS C:\Users\AB> █
```

Test Report:

Test Results:	
Passed Test Cases:	7
Failed Test Cases:	0
Total Test Cases:	7