Client Sends Message to Server (Client)

Description
This problem focuses on the reverse of a common
server-client interaction: the client sends a message
to the server. The server needs to correctly receive
this message and handle the connection.

Input
No direct input is provided for this problem since
the communication is initiated by the client through
a network connection. The server is set to listen on
localhost (127.0.0.1) at port 12345. The client will
connect to the server and send a specific message.

Output
The server should output to the console the message
it receives from the client. The complete output of
program is as follows (you do not need to write the
unit test as it is provided in the skeleton):
- Client output:
connect called with: call(('127.0.0.1', 12345))
send called with: call(b'Hello, Server!')
close called with: call()

Method
Implement the server and client programs as
described. The client will send a message to the
server after establishing a connection. The server,
upon accepting the connection, should receive the
message, print the expected output, and then close
the connection.
- Server Program: The server listens on a specified
localhost and port, accepts a client connection,
receives a message, prints the message to the console
in the specified format, and then closes the
connection. The server must handle at least one
client connection and print the received message
before shutting down for this problem.
- Client Program: The client connects to the server
using the specified host and port, sends a message to
the server, and then closes the connection.

Evaluation

The solution will be evaluated on the following criteria:
– The server must correctly start, accept a client connection, receive the correct message, print it in the specified format, and close the connection.
– The client must successfully connect to the server, send the message, and close the connection.
– Although unit tests are provided for both server and client, the primary focus for evaluation will be the correct interaction and message exchange between the client and server rather than the unit tests themselves.