Started on	Tuesday, 4 November 2025, 11:53 AM
State	Finished
Completed on	Tuesday, 4 November 2025, 12:02 PM
Time taken	9 mins 13 secs
Marks	7.00/10.00
Grade	**70.00** out of 100.00

Question 1

Complete

Mark 1.00 out of 1.00

In a high-performance Python network server, why might one prefer selectors over threading?

- a. Selectors automatically handle SSL negotiation.
- b. Selectors allow handling thousands of sockets using one thread via multiplexing.
- c. Threads are slower for single-client workloads.
- d. Selectors create new threads for each I/O event.

Question 2

Complete

Mark 1.00 out of 1.00

In Python's selectors module, which object type does the register() method return?

- a. SelectorKey
- b. SelectorEvent
- c. PollEvent
- d. SelectorResult

Question 3

Complete

Mark 1.00 out of 1.00

What happens if you call bind() on a TCP socket that is already bound?

- a. It silently overwrites the old binding.
- b. It rebinds to the new address.
- c. It raises OSError: [Errno 22] Invalid argument.
- d. It closes the old socket automatically.

Question 4

Complete

Mark 0.00 out of 1.00

What happens if you call sock.recv(1024) on a non-blocking socket when there's no data to read?

- a. The function returns an empty byte string.
- b. It raises BlockingIOError.
- c. It blocks until data arrives.
- d. It raises EOFError.

Question 5

Complete

Mark 1.00 out of 1.00

What is the correct way to make a secure SSL-wrapped socket client in Python?

- a. ssl.create_default_context().wrap_socket(socket, server_hostname='example.com')
- b. ssl.SSLSocket(socket)
- c. ssl.connect(socket, cert=True)
- d. ssl.wrap_socket(socket)

Question 6

Complete

Mark 1.00 out of 1.00

When a TCP socket is created using: socket.socket(socket.AF_INET, socket.SOCK_STREAM). which of the following accurately describes the socket's default behavior?

- a. It is non-blocking and connects automatically.
- b. It is blocking and requires bind() and listen() before accept().
- c. It is non-blocking and immediately ready for read/write.
- d. It is blocking and automatically connects to localhost.

Question 7

Complete

Mark 0.00 out of 1.00

When using asyncio for socket operations, which call integrates sockets into the event loop for asynchronous use?

- a. socket.create_connection_async()
- b. loop.add_socket()
- c. loop.sock_accept()
- d. asyncio.open_connection()

Question 8

Complete

Mark 1.00 out of 1.00

Which method should be used to allow multiple concurrent clients in a TCP server using threads?

a. Use socket.connect_ex() inside a loop.
b. Spawn a new thread per client inside the accept() loop.
c. socket.listen(5)
d. socket.listen(1)

Question 9

Complete

Mark 0.00 out of 1.00

Which of the following correctly describes the role of: sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)

a. It disables the Nagle algorithm.
b. It allows a socket to bind to a port in TIME_WAIT state.
c. It allows multiple sockets to listen on the same port concurrently.
d. It enables non-blocking mode.

Question 10

Complete

Mark 1.00 out of 1.00

Which of the following statements about UDP sockets in Python is false?

a. You can use sendto() and recvfrom().
b. They are connectionless.
c. Calling connect() fixes the destination for future send() calls.
d. UDP guarantees delivery order.