Маслов А.Н. ИД23-1  
Сетевые Системы и Приложения

**Практическая работа - Простейшие TCP-клиент и эхо-сервер**

**Основные задания 1-4:**

server.py

import socket

print("СЕРВЕР: Запуск сервера...")

sock = socket.socket()

sock.bind(('', 9090))

print("СЕРВЕР: Сокет привязан к порту 9090.")

sock.listen(0)

print("СЕРВЕР: Начало прослушивания входящих подключений.")

conn, addr = sock.accept()

print(f"СЕРВЕР: Подключился клиент с адресом {addr}.")

msg = ''

while True:

data = conn.recv(1024)

if not data:

print("СЕРВЕР: Клиент прекратил отправку данных или отключился.")

break

print(f"СЕРВЕР: Получена порция данных от клиента: {data.decode()}")

msg += data.decode()

conn.send(data)

print("СЕРВЕР: Порция данных отправлена обратно клиенту.")

print("СЕРВЕР: Полное сообщение, полученное от клиента:")

print(msg)

conn.close()

print("СЕРВЕР: Клиент отключился.")

sock.close()

print("СЕРВЕР: Остановка сервера. Работа завершена.")

client.py

import socket

sock = socket.socket()

sock.setblocking(1)

print("КЛИЕНТ: Попытка соединения с сервером на адресе localhost:9090...")

sock.connect(('localhost', 9090))

print("КЛИЕНТ: Соединение с сервером установлено.")

msg = input("Введите строку для отправки серверу: ")

print("КЛИЕНТ: Отправка данных серверу...")

sock.send(msg.encode())

print("КЛИЕНТ: Данные успешно отправлены серверу.")

print("КЛИЕНТ: Ожидание ответа от сервера...")

data = sock.recv(1024)

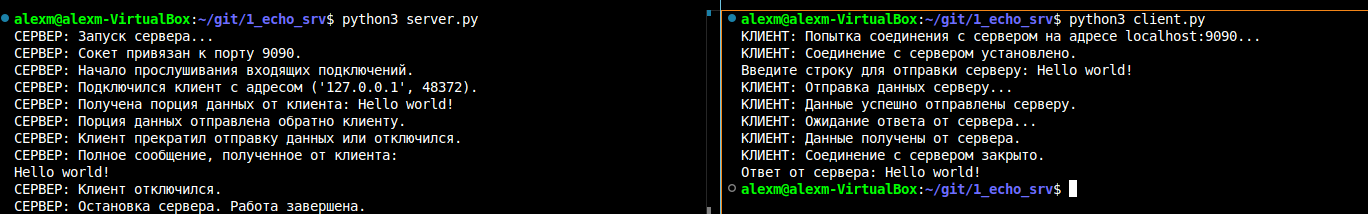
print("КЛИЕНТ: Данные получены от сервера.")

sock.close()

print("КЛИЕНТ: Соединение с сервером закрыто.")

print("Ответ от сервера:", data.decode())

Результат:

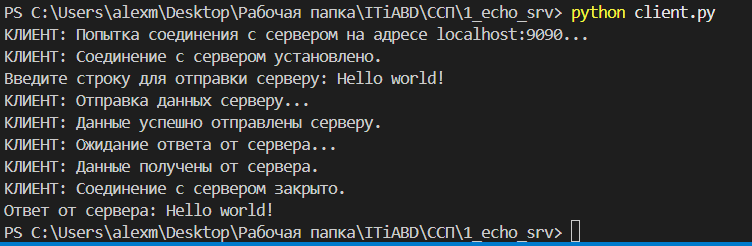


**Задания для самостоятельного выполнения:**

Задание 1:

Подключение с локальной машины.

Локальная машина:



Виртуальная машина:

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Задание 2:

client.py

import socket

sock = socket.socket()

sock.setblocking(1)

print("КЛИЕНТ: Попытка соединения с сервером на адресе localhost:9090...")

sock.connect(('localhost', 9090))

print("КЛИЕНТ: Соединение с сервером установлено.")

while True:

msg = input("Введите строку для отправки серверу: ")

if msg == "exit":

sock.close()

print("КЛИЕНТ: Соединение с сервером закрыто.")

break

print("КЛИЕНТ: Отправка данных серверу...")

sock.send(msg.encode())

print("КЛИЕНТ: Данные успешно отправлены серверу.")

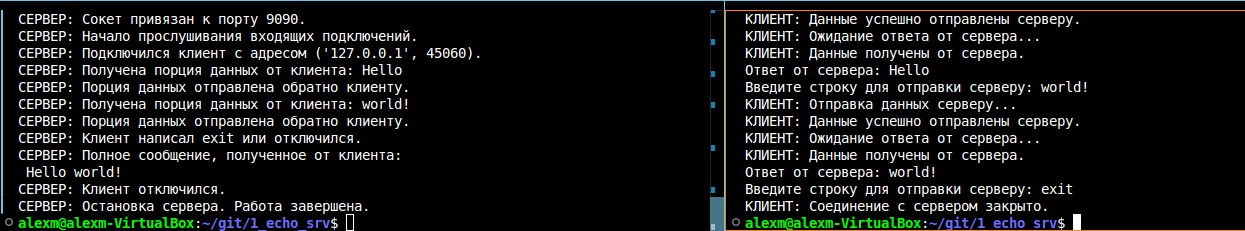
print("КЛИЕНТ: Ожидание ответа от сервера...")

data = sock.recv(1024)

print("КЛИЕНТ: Данные получены от сервера.")

print("Ответ от сервера:", data.decode())

Результат:



Задание 3:

server.py

import socket

print("СЕРВЕР: Запуск сервера...")

sock = socket.socket()

sock.bind(('', 9090))

print("СЕРВЕР: Сокет привязан к порту 9090.")

sock.listen(0)

print("СЕРВЕР: Начало прослушивания входящих подключений.")

shutdown\_requested = False

while not shutdown\_requested:

conn, addr = sock.accept()

print(f"СЕРВЕР: Подключился клиент с адресом {addr}.")

msg = ''

try:

while True:

data = conn.recv(1024)

if not data:

print("СЕРВЕР: Клиент прекратил отправку данных или отключился.")

break

decoded\_data = data.decode().strip().lower()

print(f"СЕРВЕР: Получена порция данных: {decoded\_data}")

if decoded\_data == "shutdown":

conn.send(b"Server shutting down...")

shutdown\_requested = True

break

msg += decoded\_data

conn.send(data)

print(f"СЕРВЕР: Полное сообщение от клиента: {msg}")

except ConnectionResetError:

print(f"СЕРВЕР: Клиент {addr} отключился неожиданно.")

finally:

conn.close()

print(f"СЕРВЕР: Соединение с {addr} закрыто.")

sock.close()

print("СЕРВЕР: Остановка сервера. Работа завершена.")

client.py

import socket

sock = socket.socket()

sock.setblocking(1)

print("КЛИЕНТ: Попытка соединения с сервером на адресе localhost:9090...")

sock.connect(('localhost', 9090))

print("КЛИЕНТ: Соединение с сервером установлено.")

while True:

msg = input("Введите строку для отправки серверу: ")

if msg == "exit":

sock.close()

print("КЛИЕНТ: Соединение с сервером закрыто.")

break

print("КЛИЕНТ: Отправка данных серверу...")

sock.send(msg.encode())

print("КЛИЕНТ: Данные успешно отправлены серверу.")

print("КЛИЕНТ: Ожидание ответа от сервера...")

data = sock.recv(1024)

print("КЛИЕНТ: Данные получены от сервера.")

print("Ответ от сервера:", data.decode())

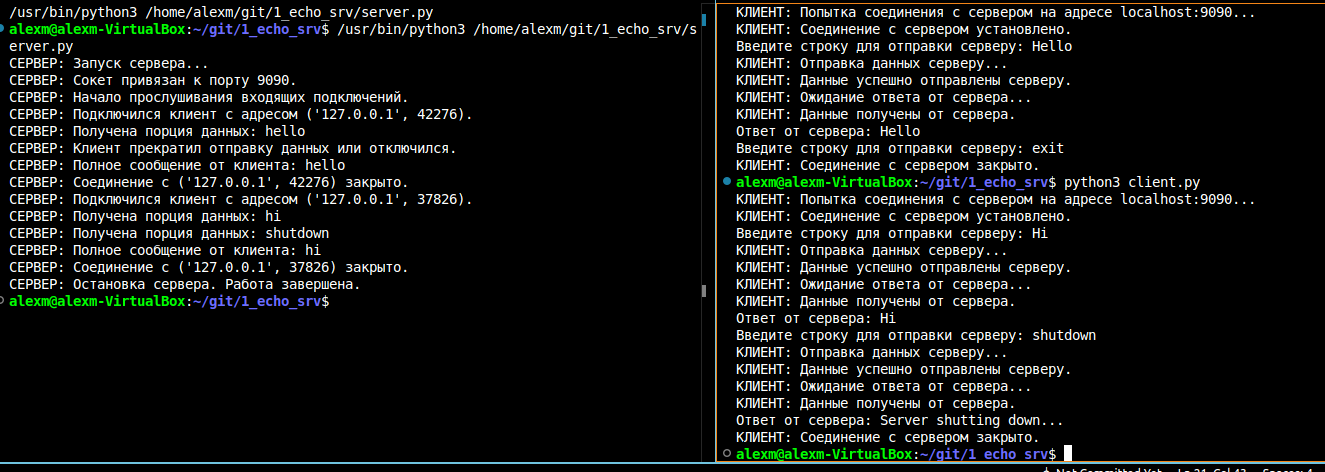
if msg == "shutdown":

sock.close()

print("КЛИЕНТ: Соединение с сервером закрыто.")

break

Результат:



Задание 4: