Challenge 1

Muhammad Fadli Azhar

Muhammad Azka Aysar

This code is a Python script that sends a file to a server using a UDP socket. The code first sets up the server address as `('127.0.0.1', 5000)` and creates a UDP socket with `socket.socket(socket.AF_INET, socket.SOCK_DGRAM)`. The socket is then connected to the server using `CLIENT_SOCKET.connect(SERVER_ADDRESS)`.

The code then defines a function `system_interrupt()` that closes the socket and exits the program.

Next, the code sets the buffer size for reading and writing data to the socket as `BUF_SIZE = 1024`. It then prompts the user to input the filename to be sent to the server using `filename = input('Input filename to be sent: ')`.

The code then gets the file size using `os.path.getsize(filename)` and sends the size to the server using `CLIENT_SOCKET.send(str(filesize).encode())`.

The code opens the file with `with open(filename, 'rb') as f:` and reads the file in chunks of `BUF_SIZE` bytes with `data = f.read(BUF_SIZE)`. It sends each chunk to the server with `CLIENT_SOCKET.send(data)`. If an interruption is detected (by pressing Ctrl+C), the `system_interrupt()` function is called to close the socket and exit the program.

After the file is sent, the code waits for the server to send back the percentage of the file received using `percentage = float(CLIENT_SOCKET.recv(BUF_SIZE).decode())`. If a timeout occurs while waiting for the server's response, it prints "Server is down". Otherwise, it prints the percentage received with `print(f"Server received {percentage}% of the file")`. If an interruption is detected, the `system_interrupt()` function is called to close the socket and exit the program.