

Bansilal Ramnath Agarwal Charitable Trust's
Vishwakarma Institute of Technology,Pune-37

(An Autonomous Institute of Savitribai Phule Pune University)

Assignment 10:



Department of Artificial Intelligence and Data Science

Division	A
Batch	1
Rollno	26
Name	Jineshwari Bagul

CODE :

Sender.py

```
import socket
import os
import time

# ♦ Receiver IP and Port (Change as per your network)
SERVER_IP = "192.168.1.100" # Change to the receiver's IP
SERVER_PORT = 5005

# ♦ File to Send (Update the path)
FILE_TO_SEND = r"C:\Users\jines\OneDrive\Desktop\SY SEM
2\CN\sample.txt" # Change this path

# ♦ UDP Config
CHUNK_SIZE = 1024 # 1KB chunk size

def send_file(filename):
    """Send a file to the receiver using UDP."""
    if not os.path.isfile(filename):
        print(f"Error: File '{filename}' not found!")
        return

    sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

    # Get file size
    file_size = os.path.getsize(filename)

    # Send filename and size
    sock.sendto(f"{os.path.basename(filename)}|{file_size}".encode(),
                (SERVER_IP, SERVER_PORT))

    # Open file and send in chunks
    with open(filename, "rb") as f:
        bytes_sent = 0
        while True:
            chunk = f.read(CHUNK_SIZE)
            if not chunk:
                break # EOF
```

```

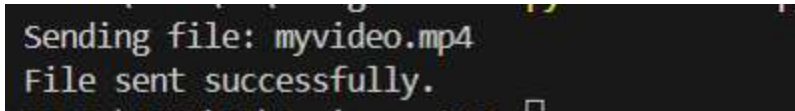
sock.sendto(chunk, (SERVER_IP, SERVER_PORT))
bytes_sent += len(chunk)

# Wait for ACK
try:
    sock.settimeout(0.5) # 500ms timeout
    ack, _ = sock.recvfrom(1024)
    if ack.decode() != "ACK":
        print("ACK not received, resending last chunk")
        bytes_sent -= len(chunk)
        f.seek(bytes_sent)
except socket.timeout:
    print("Timeout! Resending last chunk")
    bytes_sent -= len(chunk)
    f.seek(bytes_sent)

print(f"✅ File '{filename}' sent successfully.")
sock.close()

if __name__ == "__main__":
    send_file(FILE_TO_SEND)

```



```

Sending file: myvideo.mp4
File sent successfully.

```

Receiver.py

```

import socket
import os

# ♦ Receiver IP and Port (Listen for incoming files)
RECEIVER_IP = "0.0.0.0" # Listen on all network interfaces
RECEIVER_PORT = 5005

# ♦ UDP Config
CHUNK_SIZE = 1024 # 1KB chunk size

def receive_file():

```

```

"""Receive a file over UDP and save it."""
sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
sock.bind((RECEIVER_IP, RECEIVER_PORT))
print(f"🔊 Listening for incoming files on
{RECEIVER_IP}:{RECEIVER_PORT}...")

# Receive file metadata
data, sender_addr = sock.recvfrom(1024)
filename, file_size = data.decode().split("|")
file_size = int(file_size)

# Create the received file
save_path = os.path.join(os.getcwd(), filename)
with open(save_path, "wb") as f:
    bytes_received = 0

    while bytes_received < file_size:
        chunk, sender_addr = sock.recvfrom(CHUNK_SIZE)
        f.write(chunk)
        bytes_received += len(chunk)

    # Send ACK to sender
    sock.sendto("ACK".encode(), sender_addr)

print(f"✅ File '{filename}' received successfully and saved as '{save_path}'.")
sock.close()

if __name__ == "__main__":
    receive_file()

```

```

C:\Users\user\Documents\Assignment10> python receiver.py
Listening on port 5005...
Receiving file: myvideo.mp4 from ('192.168.245.126', 57856)
File received and saved as received_myvideo.mp4
C:\Users\user\Documents\Assignment10>

```