Assignment No-5

Name- Prathamesh Gokulkar

Rollno-33

Code

```
import sys


class Graph:
    def __init__(self, nodes):
        self.nodes = nodes  # List of nodes
        self.graph = {}     # Dictionary to store adjacency list for graph


    def add_edge(self, u, v, weight):
        if u not in self.graph:
            self.graph[u] = {}
        if v not in self.graph:
            self.graph[v] = {}
        self.graph[u][v] = weight
        self.graph[v][u] = weight



    def dijkstra(self, start_node):
        # Dictionary to store shortest path from start_node to other nodes
        shortest_paths = {node: sys.maxsize for node in self.nodes}
        shortest_paths[start_node] = 0

        # Dictionary to store the best previous node to get the shortest path
        previous_nodes = {node: None for node in self.nodes}

        visited = set()
```

```
nodes = self.nodes.copy()

while nodes:
    # Get node with the smallest distance from start_node
    min_node = None
    for node in nodes:
        if node in visited:
            continue
        if min_node is None:
            min_node = node
        elif shortest_paths[node] < shortest_paths[min_node]:
            min_node = node

    if min_node is None:
        break

    # Nodes adjacent to the current node
    neighbors = self.graph[min_node].items()

    for neighbor, weight in neighbors:
        tentative_value = shortest_paths[min_node] + weight
        if tentative_value < shortest_paths[neighbor]:
            shortest_paths[neighbor] = tentative_value
            previous_nodes[neighbor] = min_node

    visited.add(min_node)
    nodes.remove(min_node)
```

```
        return previous_nodes, shortest_paths

    # Function to display the shortest path from start_node to each node
    def print_shortest_path(self, start_node):
        previous_nodes, shortest_paths = self.dijkstra(start_node)

        print(f"Shortest paths from {start_node}:")
        for node in self.nodes:
            if node == start_node:
                continue
            path = []
            current_node = node
            while current_node is not None:
                path.append(current_node)
                current_node = previous_nodes[current_node]
            path.reverse()
            print(f"Path to {node}: {' -> '.join(path)}, Total cost: {shortest_paths[node]}")

nodes = ['A', 'B', 'C', 'D', 'E']

graph = Graph(nodes)
graph.add_edge('A', 'B', 4)
graph.add_edge('A', 'C', 2)
graph.add_edge('B', 'C', 1)
graph.add_edge('B', 'D', 5)
graph.add_edge('C', 'D', 8)
graph.add_edge('C', 'E', 10)
graph.add_edge('D', 'E', 2)
```

start_node = 'A'

graph.print_shortest_path(start_node)

output:

```
PS C:\Users\HP> & C:/Users/HP/AppData/Local/Microsoft/WindowsApps/python3.12.exe c:/Users/HP/OneDrive/Desktop/appt.py
Shortest paths from A:
Path to B: A -> C -> B, Total cost: 3
Path to C: A -> C, Total cost: 2
Path to D: A -> C -> B -> D, Total cost: 8
Path to E: A -> C -> B -> D -> E, Total cost: 10
```

Assignment No-3

Name- Prathamesh Gokulkar

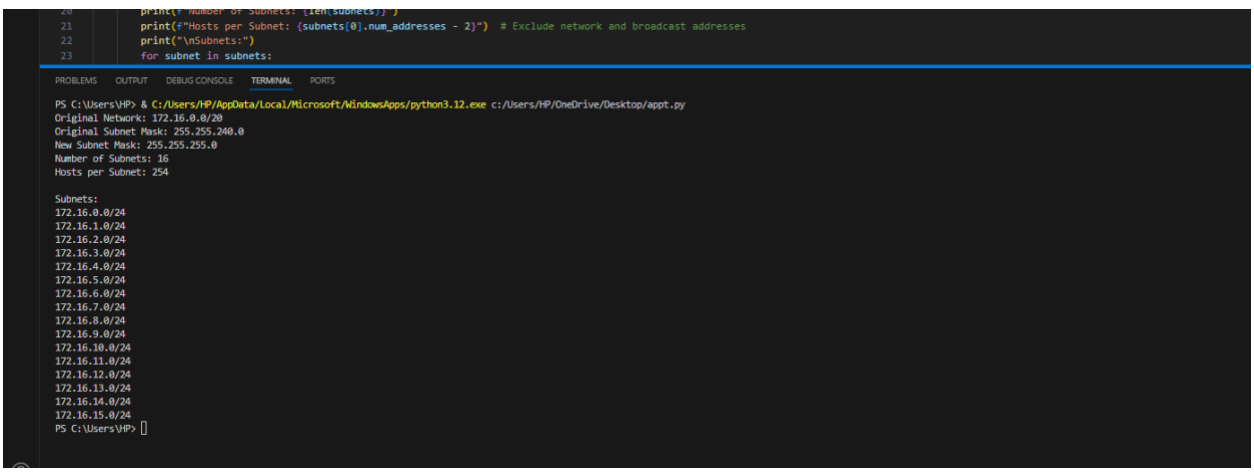
Rollno-33

Code

```
import ipaddress


def calculate_subnet(network, subnet_bits):
    try:
        # Convert the network to an IPv4Network object
        net = ipaddress.IPv4Network(network)

        # Calculate the new subnet mask
        new_prefix = net.prefixlen + subnet_bits
        if new_prefix > 32:
            raise ValueError("Invalid subnet bits. Subnet prefix length cannot exceed 32.")

        # Create subnets
        subnets = list(net.subnets(new_prefix=new_prefix))

        # Display results
        print(f"Original Network: {network}")
        print(f"Original Subnet Mask: {net.netmask}")
        print(f"New Subnet Mask: {subnets[0].netmask}")
        print(f"Number of Subnets: {len(subnets)}")
        print(f"Hosts per Subnet: {subnets[0].num_addresses - 2}")  # Exclude network and broadcast
addresses
        print("\nSubnets:")
        for subnet in subnets:
```

```
        print(subnet)

    except ValueError as e:
        print(f"Error: {e}")

if __name__ == "__main__":
    # Example usage
    network = "172.16.0.0/20"  # Define the network in CIDR notation
    subnet_bits = 4  # Number of bits to use for subnetting
    calculate_subnet(network, subnet_bits)
```

ouput:



Assignment No-9

Name- Prathamesh Gokulkar

Rollno-33

Code

```
import socket


def dns_lookup(ip_address=None, domain_name=None):
    try:
        if ip_address:
            # Perform reverse DNS lookup
            result = socket.gethostbyaddr(ip_address)
            return f"IP Address: {ip_address} -> Domain Name: {result[0]}"
        elif domain_name:
            # Perform forward DNS lookup
            result = socket.gethostbyname(domain_name)
            return f"Domain Name: {domain_name} -> IP Address: {result}"
        else:
            return "Please provide either an IP address or a domain name."
    except socket.herror as e:
        return f"Error: Unable to resolve {ip_address or domain_name}. {e}"
    except socket.gaierror as e:
        return f"Error: {e}"


if __name__ == "__main__":
    print("DNS Lookup Program")


    # Input from user
    choice = input("Do you want to lookup by (1) IP Address or (2) Domain Name? Enter 1 or 2: ")
```

```
    if choice == "1":
        ip_address = input("Enter the IP address: ")
        result = dns_lookup(ip_address=ip_address)
        print(result)
    elif choice == "2":
        domain_name = input("Enter the domain name: ")
        result = dns_lookup(domain_name=domain_name)
        print(result)
    else:
        print("Invalid choice. Please run the program again.")
```

OUPUT:

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

PS C:\Users\HP> & C:/Users/HP/AppData/Local/Microsoft/WindowsApps/python3.12.exe c:/Users/HP/OneDrive/Desktop/cn/server.py
DNS Lookup Program
Do you want to lookup by (1) IP Address or (2) Domain Name? Enter 1 or 2: 1
Enter the IP address: 8.8.8.8
IP Address: 8.8.8.8 -> Domain Name: dns.google
PS C:\Users\HP>
```

Assignment No-7

Name- Prathamesh Gokulkar

Rollno-33

Code

Server.py

```
import socket


def send_file(filename, conn):
    with open(filename, "rb") as file:
        while chunk := file.read(1024):
            conn.send(chunk)
    print(f"File {filename} sent successfully!")


def start_server(host='127.0.0.1', port=65432):
    server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    server_socket.bind((host, port))
    server_socket.listen(1)
    print(f"Server listening on {host}:{port}...")


    conn, addr = server_socket.accept()
    print(f"Connected to {addr}")


    filename = conn.recv(1024).decode()


    try:
        send_file(filename, conn)
    except FileNotFoundError:
        print(f"File {filename} not found.")
        conn.send(b"ERROR: File not found.")
```

```
    conn.close()

if __name__ == "__main__":
    start_server()
```

client.py

```
import socket

def receive_file(filename, host='127.0.0.1', port=65432):
    client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    client_socket.connect((host, port))
    client_socket.send(filename.encode())

    with open("received_" + filename, "wb") as file:
        while True:
            data = client_socket.recv(1024)
            if not data:
                break
            file.write(data)

    print(f"File {filename} received and saved as received_{filename}")
    client_socket.close()

if __name__ == "__main__":
    filename = input("Enter the filename to download: ")
    receive_file(filename)
```

ouput:

```
C:\Users\HP\OneDrive\Desktop\cn>python server.py

Server listening on 127.0.0.1:65432...

Connected to ('127.0.0.1', 52151)

File tr.txt sent successfully!

C:\Users\HP\OneDrive\Desktop\cn>python client.py

Enter the filename to download: tr.txt

File tr.txt received and saved as received_tr.txt
```

Assignment No-6

Name- Prathamesh Gokulkar

Rollno-33

Code

Server

```
import socket


def start_server():
    server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    server_socket.bind(('localhost', 12345))
    server_socket.listen(1)
    print("Server is listening for connections...")


    while True:
        try:
            client_socket, addr = server_socket.accept()
            print(f"Connection from {addr} has been established!")


            data = client_socket.recv(1024)
            if data:
                print(f"Received from client: {data.decode('utf-8')}")
                client_socket.sendall(b"Hello, Client!")
            else:
                print("No data received from client.")


            client_socket.close()
        except Exception as e:
            print(f"Error: {e}")
```

```
if __name__ == "__main__":
    start_server()
```

client

```
import socket

def start_client():
    try:
        client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        print("Connecting to server...")

        client_socket.connect(('localhost', 12345))
        print("Connected to server.")

        client_socket.sendall(b"Hello, Server!")
        print("Message sent to server.")

        data = client_socket.recv(1024)
        print(f"Received from server: {data.decode('utf-8')}")

    except Exception as e:
        print(f"Error: {e}")
    finally:
        client_socket.close()
        print("Client socket closed.")

if __name__ == "__main__":
    start_client()
```

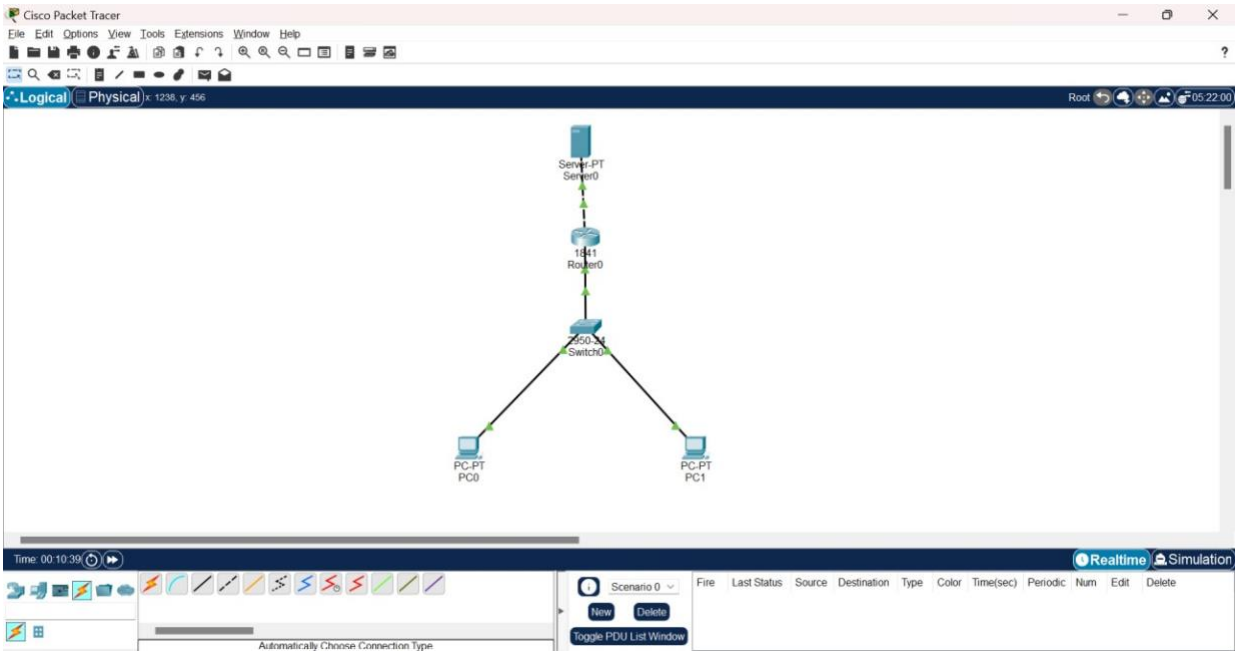
ouput:

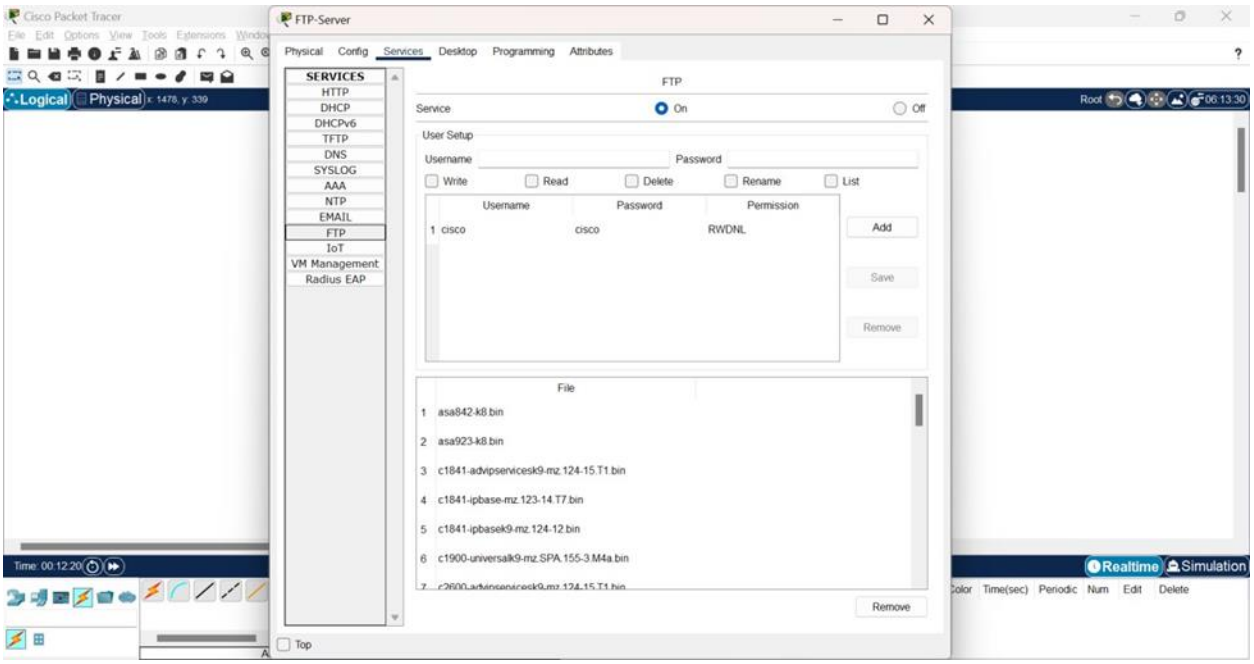
```
C:\Users\HP\OneDrive\Desktop\cn>python server.py

Server is listening for connections...

Connection from ('127.0.0.1', 52095) has been established!

Received from client: Hello, Server!

C:\Users\HP\OneDrive\Desktop\cn>python client.py

Connecting to server...

Connected to server.

Message sent to server.

Received from server: Hello, Client!

Client socket closed.
```

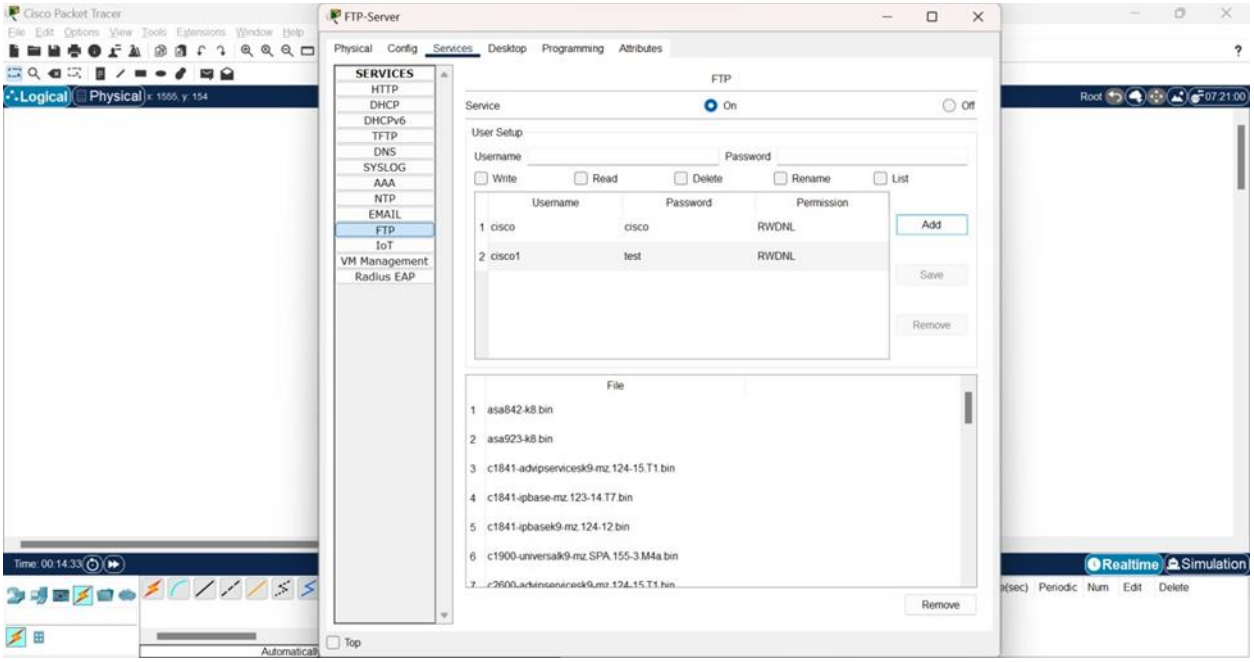
Assignment no-8

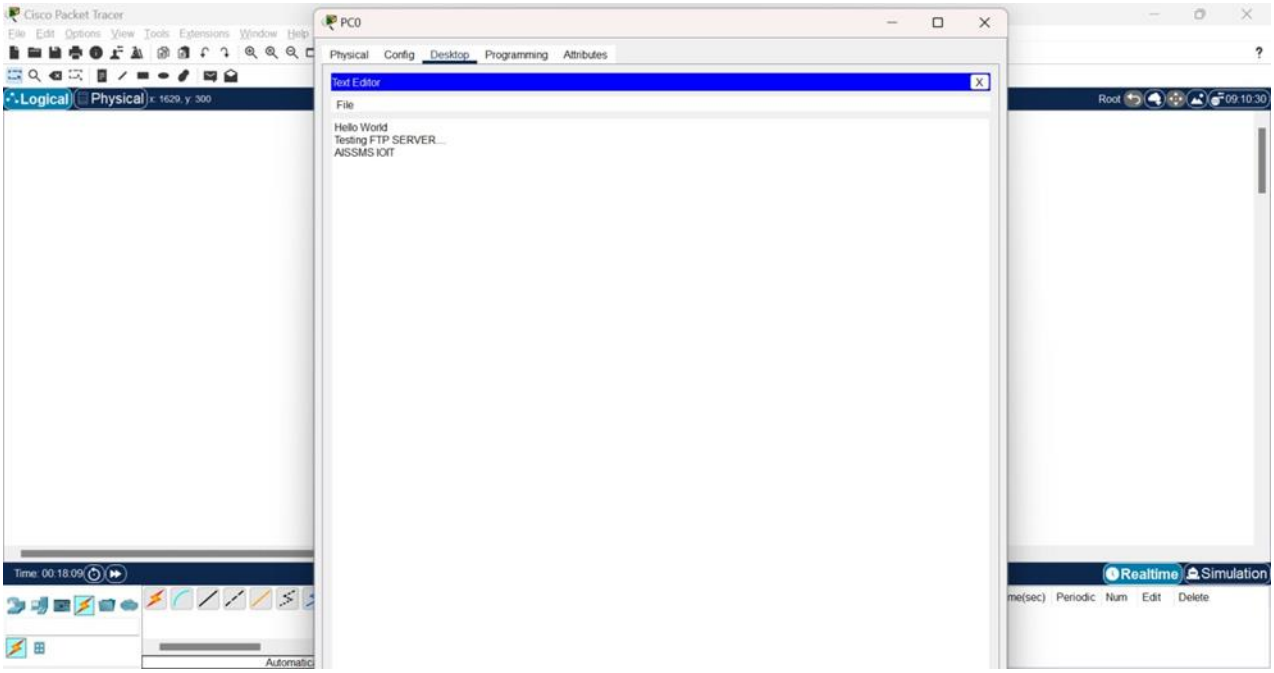
Name- Prathamesh Gokulkar

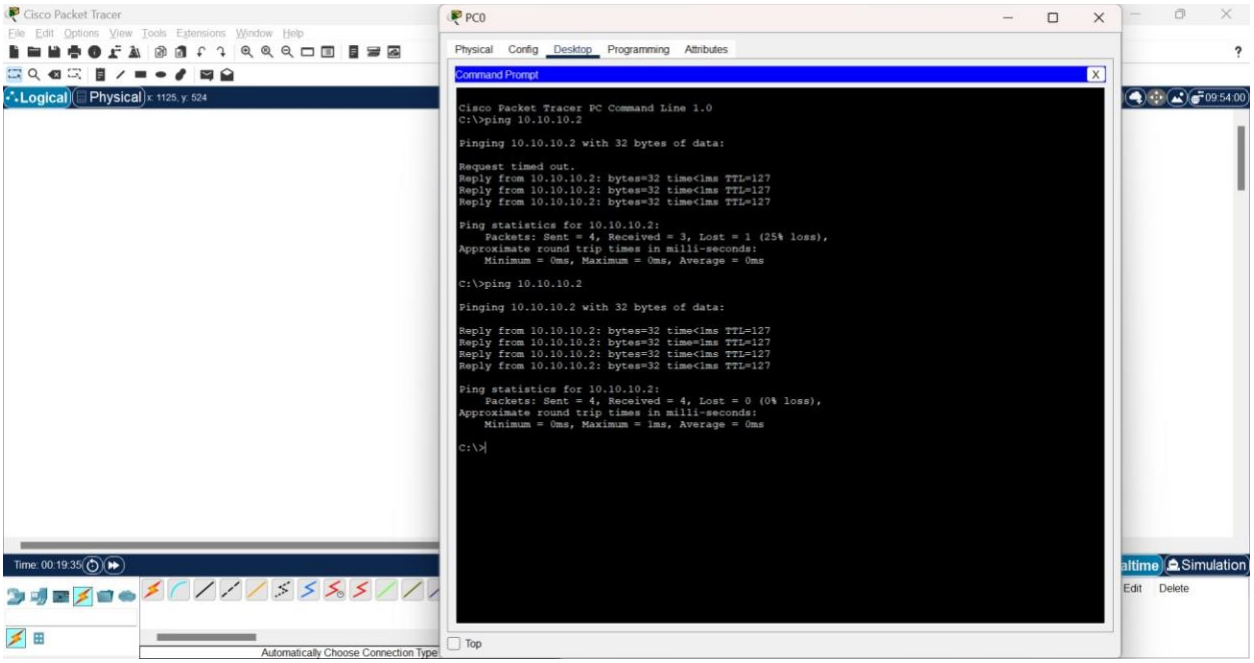
Rollno-33

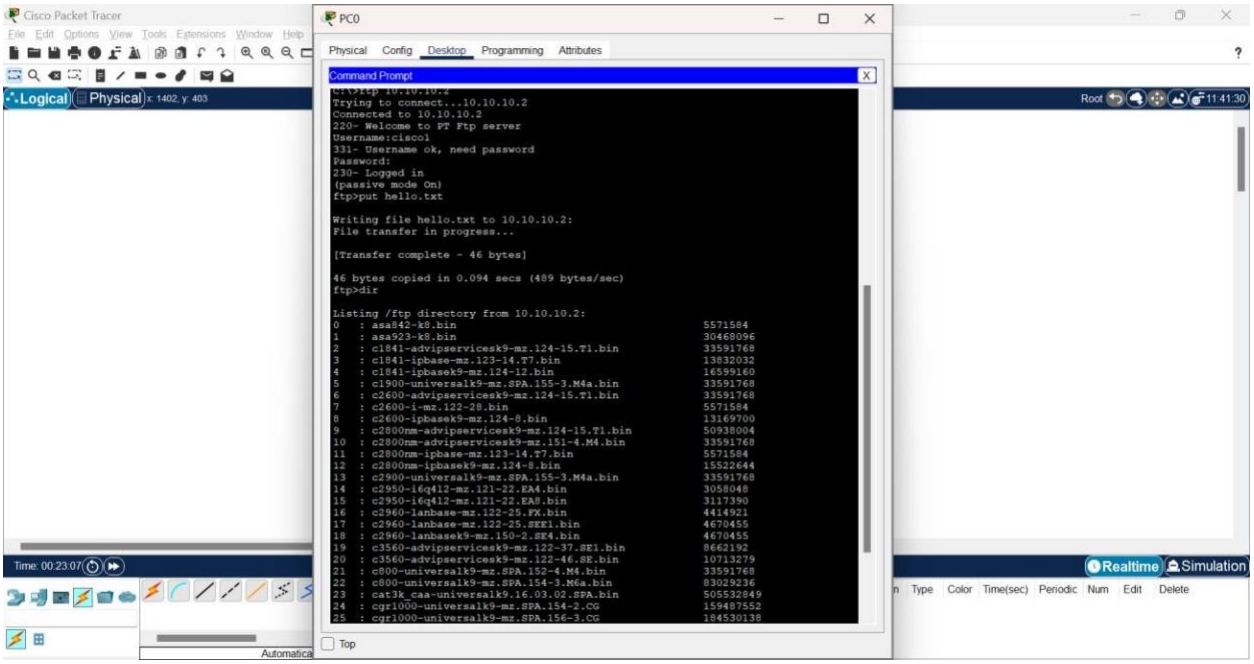


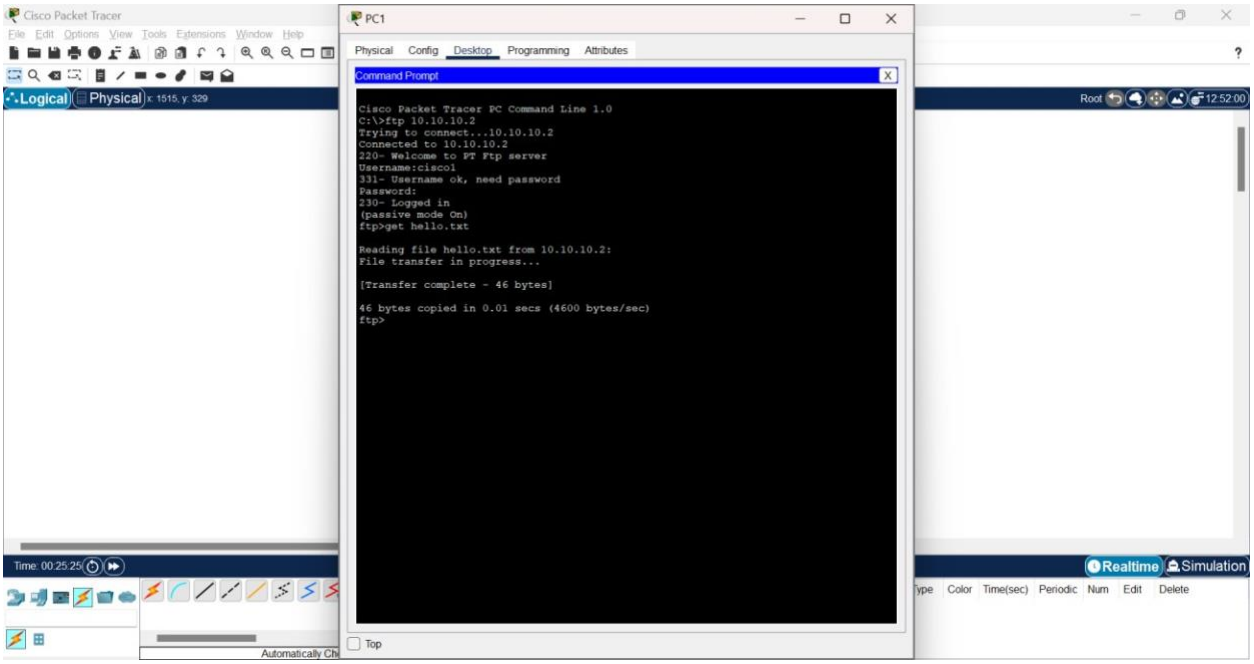


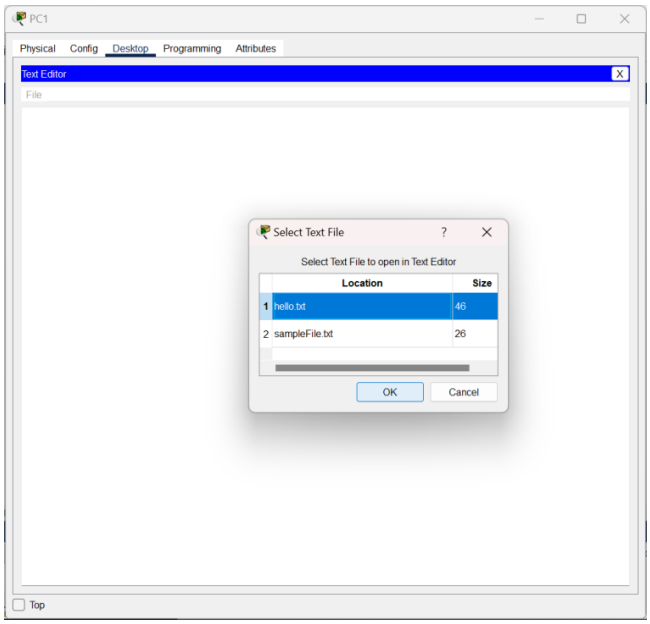

Cisco Packet Tracer
FTP-Server
Physical Config Services Desktop Programming Attributes
SERVICES
HTTP
DHCP
DHCPv6
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
IoT
VM Management
Radius EAP
FTP
Service On Off
User Setup
Username Password
Write Read Delete Rename List
Username Password Permission
1 cisco cisco RWDNL
2 cisco1 test RWDNL
Add
Save
Remove
File
1 asa842-k8.bin
2 asa923-k8.bin
3 c1841-advipservicesk9-mz.124-15.T1.bin
4 c1841-ipbase-mz.123-14.T7.bin
5 c1841-ipbasek9-mz.124-12.bin
6 c1900-universalk9-mz.SPA.155-3.M4a.bin
Remove
Top
Logical Physical
Realtime Simulation

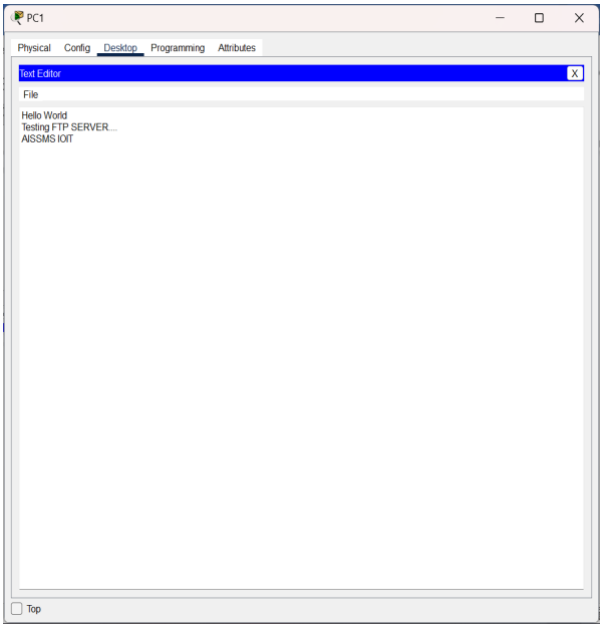

Cisco Packet Tracer
PC0
Physical Config Desktop Programming Attributes
Text Editor
File
Hello World
Testing FTP SERVER...
AISSMS IOIT
Logical Physical
Realtime Simulation

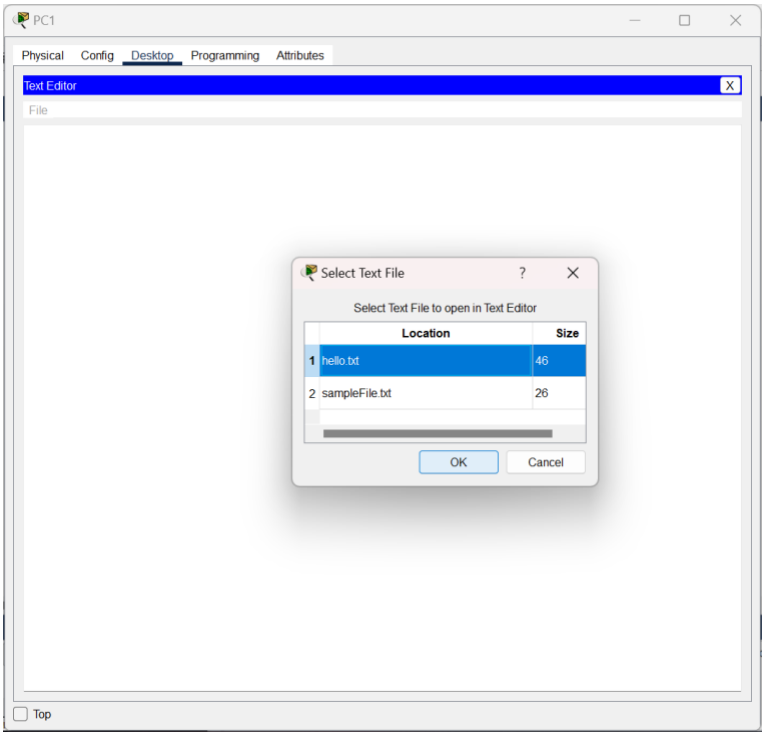

Cisco Packet Tracer
File Edit Options View Tools Extensions Window Help
Logical Physical x: 1125, y: 524
PC0
Physical Config Desktop Programming Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.10.10.2
Pinging 10.10.10.2 with 32 bytes of data:
Request timed out.
Reply from 10.10.10.2: bytes=32 time<1ms TTL=127
Reply from 10.10.10.2: bytes=32 time<1ms TTL=127
Reply from 10.10.10.2: bytes=32 time<1ms TTL=127
Ping statistics for 10.10.10.2:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 10.10.10.2
Pinging 10.10.10.2 with 32 bytes of data:
Reply from 10.10.10.2: bytes=32 time<1ms TTL=127
Reply from 10.10.10.2: bytes=32 time=1ms TTL=127
Reply from 10.10.10.2: bytes=32 time<1ms TTL=127
Reply from 10.10.10.2: bytes=32 time<1ms TTL=127
Ping statistics for 10.10.10.2:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>
Top
Time: 00:19:35
Simulation
Edit Delete
Automatically Choose Connection Type


Cisco Packet Tracer
File Edit Options View Tools Extensions Window Help
Logical Physical x: 1402, y: 403
PC0
Physical Config Desktop Programming Attributes
Command Prompt
Trying to connect...10.10.10.2
Connected to 10.10.10.2
220- Welcome to PT Ftp server
Username:cisco1
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>put hello.txt
Writing file hello.txt to 10.10.10.2:
File transfer in progress...
[Transfer complete - 46 bytes]
46 bytes copied in 0.094 secs (489 bytes/sec)
ftp>dir
Listing /ftp directory from 10.10.10.2:
0 : asa842-k8.bin 5571584
1 : asa923-k8.bin 30468096
2 : c1841-advipservicesk9-mz.124-15.T1.bin 33591768
3 : c1841-ipbase-mz.123-14.T7.bin 13832032
4 : c1841-ipbasek9-mz.124-12.bin 16599160
5 : c1900-universalk9-mz.SPA.155-3.M4a.bin 33591768
6 : c2600-advipservicesk9-mz.124-15.T1.bin 33591768
7 : c2600-i-mz.122-28.bin 5571584
8 : c2600-ipbasek9-mz.124-8.bin 13169700
9 : c2800nm-advipservicesk9-mz.124-15.T1.bin 50938004
10 : c2800nm-advipservicesk9-mz.151-4.M4.bin 33591768
11 : c2800nm-ipbase-mz.123-14.T7.bin 5571584
12 : c2800nm-ipbasek9-mz.124-8.bin 15522644
13 : c2900-universalk9-mz.SPA.155-3.M4a.bin 33591768
14 : c2950-i6q412-mz.121-22.EA4.bin 3058048
15 : c2950-i6q412-mz.121-22.EA8.bin 3117390
16 : c2960-lanbase-mz.122-25.FX.bin 4414921
17 : c2960-lanbase-mz.122-25.SEE1.bin 4670455
18 : c2960-lanbasek9-mz.150-2.SE4.bin 4670455
19 : c3560-advipservicesk9-mz.122-37.SE1.bin 8662192
20 : c3560-advipservicesk9-mz.122-46.SE.bin 10713279
21 : c800-universalk9-mz.SPA.152-4.M4.bin 33591768
22 : c800-universalk9-mz.SPA.154-3.M6a.bin 83029236
23 : cat3k_caa-universalk9.16.03.02.SPA.bin 505532849
24 : cgr1000-universalk9-mz.SPA.154-2.CG 159487552
25 : cgr1000-universalk9-mz.SPA.156-3.CG 184530138
Top
Time: 00:23:07
Realtime Simulation
Type Color Time(sec) Periodic Num Edit Delete


Cisco Packet Tracer
File Edit Options View Tools Extensions Window Help
Logical
Physical
x: 1515, y: 329
Root
12:52:00
Time: 00:25:25
Realtime
Simulation
Type Color Time(sec) Periodic Num Edit Delete
Automatically Ch
PC1
Physical Config Desktop Programming Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ftp 10.10.10.2
Trying to connect...10.10.10.2
Connected to 10.10.10.2
220- Welcome to PT Ftp server
Username:cisco1
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>get hello.txt
Reading file hello.txt from 10.10.10.2:
File transfer in progress...
[Transfer complete - 46 bytes]
46 bytes copied in 0.01 secs (4600 bytes/sec)
ftp>
Top


PC1
Physical Config Desktop Programming Attributes
Text Editor
File
Select Text File
Select Text File to open in Text Editor
Location
Size
1 hello.txt 46
2 sampleFile.txt 26
OK
Cancel
Top


PC1
Physical Config Desktop Programming Attributes
Text Editor
File
Hello World
Testing FTP SERVER...
AISSMS IOIT
Top


PC1
Physical
Config
Desktop
Programming
Attributes
Text Editor
File
Select Text File
Select Text File to open in Text Editor
Location
Size
1
hello.txt
46
2
sampleFile.txt
26
OK
Cancel
Top