

daytime/daytimeclient.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # daytimeclient.py
5: # Author: Bob Dondero
6: #-----
7: # Try this:
8: # daytimeclient.py time-a.nist.gov 13
9: #-----
10:
11: import sys
12: import socket
13:
14: #-----
15:
16: def main():
17:
18:     if len(sys.argv) != 3:
19:         print('Usage: python %s host port' % sys.argv[0])
20:         sys.exit(1)
21:
22:     try:
23:         host = sys.argv[1]
24:         port = int(sys.argv[2])
25:
26:         with socket.socket() as sock:
27:             sock.connect((host, port))
28:             flo = sock.makefile(mode='r', encoding='ascii')
29:             for line in flo:
30:                 print(line, end='')
31:
32:     except Exception as ex:
33:         print(ex, file=sys.stderr)
34:         sys.exit(1)
35:
36: #-----
37:
38: if __name__ == '__main__':
39:     main()

```

daytime/daytimeserver.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # daytimeserver.py
5: # Author: Bob Dondero
6: #-----
7:
8: import os
9: import sys
10: import socket
11: import time
12:
13: #-----
14:
15: def handle_client(sock):
16:
17:     datetime = time.asctime(time.localtime())
18:     flo = sock.makefile(mode='w', encoding='ascii')
19:     flo.write(datetime + '\n')
20:     flo.flush()
21:
22: #-----
23:
24: def main():
25:
26:     if len(sys.argv) != 2:
27:         print('Usage: python %s port' % sys.argv[0])
28:         sys.exit(1)
29:
30:     try:
31:         port = int(sys.argv[1])
32:
33:         server_sock = socket.socket()
34:         print('Opened server socket')
35:         if os.name != 'nt':
36:             server_sock.setsockopt(
37:                 socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
38:         server_sock.bind(('', port))
39:         print('Bound server socket to port')
40:         server_sock.listen()
41:         print('Listening')
42:
43:         while True:
44:             try:
45:                 sock, client_addr = server_sock.accept()
46:                 with sock:
47:                     print('Accepted connection')
48:                     print('Opened socket')
49:                     print('Server IP addr and port:',
50:                           sock.getsockname())
51:                     print('Client IP addr and port:', client_addr)
52:                     handle_client(sock)
53:             except Exception as ex:
54:                 print(ex, file=sys.stderr)
55:
56:     except Exception as ex:
57:         print(ex, file=sys.stderr)
58:         sys.exit(1)
59:
60: #-----
61:
62: if __name__ == '__main__':
63:     main()

```

echo/echoclient.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # echoclient.py
5: # Author: Bob Dondero
6: #-----
7:
8: import sys
9: import socket
10:
11: #-----
12:
13: def main():
14:
15:     if len(sys.argv) != 3:
16:         print('Usage: python %s host port' % sys.argv[0])
17:         sys.exit(1)
18:
19:     try:
20:         host = sys.argv[1]
21:         port = int(sys.argv[2])
22:
23:         line = input()
24:         if line is None:
25:             return
26:
27:         with socket.socket() as sock:
28:             sock.connect((host, port))
29:
30:             out_flo = sock.makefile(mode='w', encoding='utf-8')
31:             out_flo.write(line + '\n')
32:             out_flo.flush()
33:
34:             in_flo = sock.makefile(mode='r', encoding='utf-8')
35:             echoed_line = in_flo.readline()
36:
37:             if echoed_line == '':
38:                 print('The echo server crashed', file=sys.stderr)
39:             else:
40:                 print(echoed_line, end='')
41:
42:     except Exception as ex:
43:         print(ex, file=sys.stderr)
44:         sys.exit(1)
45:
46: #-----
47:
48: if __name__ == '__main__':
49:     main()

```

echo/echoserver.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # echoserver.py
5: # Author: Bob Dondero
6: #-----
7:
8: import os
9: import sys
10: import socket
11:
12: #-----
13: def handle_client(sock):
14:
15:     in_flo = sock.makefile(mode='r', encoding='utf-8')
16:     line = in_flo.readline()
17:     if line == '':
18:         print('The echo client crashed')
19:         return
20:     print('Read from client: ' + line, end='')
21:
22:     out_flo = sock.makefile(mode='w', encoding='utf-8')
23:     out_flo.write(line)
24:     out_flo.flush()
25:     print('Wrote to client: ' + line, end='')
26:
27: #-----
28: def main():
29:
30:     if len(sys.argv) != 2:
31:         print('Usage: python %s port' % sys.argv[0])
32:         sys.exit(1)
33:
34:     try:
35:         port = int(sys.argv[1])
36:
37:         server_sock = socket.socket()
38:         print('Opened server socket')
39:         if os.name != 'nt':
40:             server_sock.setsockopt(
41:                 socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
42:         server_sock.bind(('', port))
43:         print('Bound server socket to port')
44:         server_sock.listen()
45:         print('Listening')
46:
47:         while True:
48:             try:
49:                 sock, client_addr = server_sock.accept()
50:                 with sock:
51:                     print('Accepted connection')
52:                     print('Opened socket')
53:                     print('Server IP addr and port:',
54:                           sock.getsockname())
55:                     print('Client IP addr and port:', client_addr)
56:                     handle_client(sock)
57:             except Exception as ex:
58:                 print(ex, file=sys.stderr)
59:
60:     except Exception as ex:
61:         print(ex, file=sys.stderr)
62:         sys.exit(1)
63:
64: #-----
65: if __name__ == '__main__':
66:     main()

```