

## 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: from sys import exit, argv, stderr
12: from socket import socket
13:
14: #-----
15:
16: def main():
17:
18:     if len(argv) != 3:
19:         print('Usage: python %s host port' % argv[0])
20:         exit(1)
21:
22:     try:
23:         host = argv[1]
24:         port = int(argv[2])
25:
26:         with 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=stderr)
34:         exit(1)
35:
36: #-----
37:
38: if __name__ == '__main__':
39:     main()

```

## daytimeserver.py (Page 1 of 1)

```

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

```

## echoclient.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # echoclient.py
5: # Author: Bob Dondero
6: #-----
7:
8: from sys import exit, argv, stderr
9: from socket import socket
10:
11: #-----
12:
13: def main():
14:
15:     if len(argv) != 3:
16:         print('Usage: python %s host port' % argv[0])
17:         exit(1)
18:
19:     try:
20:         host = argv[1]
21:         port = int(argv[2])
22:
23:         line = input()
24:         if line is None:
25:             return
26:
27:         with 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=stderr)
39:             else:
40:                 print(echoed_line, end='')
41:
42:     except Exception as ex:
43:         print(ex, file=stderr)
44:         exit(1)
45:
46: #-----
47:
48: if __name__ == '__main__':
49:     main()

```

## echoserver.py (Page 1 of 1)

```

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

```