

ipaddress.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # ipaddress.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) != 2:
16:         print(f'usage: python {sys.argv[0]} domainname',
17:               file=sys.stderr)
18:         sys.exit(1)
19:
20:     domain_name = sys.argv[1]
21:
22:     try:
23:         host_name, aliases, ip_addresses = (
24:             socket.gethostbyname_ex(domain_name))
25:         print('Host name:', host_name)
26:         for alias in aliases:
27:             print('Alias:', alias)
28:         for ip_address in ip_addresses:
29:             print('IP address:', ip_address)
30:
31:     except Exception as ex:
32:         print(f'{sys.argv[0]}: {ex}', file=sys.stderr)
33:         sys.exit(1)
34:
35: #-----
36:
37: if __name__ == '__main__':
38:     main()

```

blank (Page 1 of 1)

1: This page is intentionally blank.

daytime/client.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # client.py
5: # Author: Bob Dondero
6: #-----
7: # Try this:
8: # python client.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(f'usage: python {sys.argv[0]} host port', file=sys.stderr)
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:             with sock.makefile(mode='r', encoding='ascii') as flo:
29:                 for line in flo:
30:                     print(line, end='')
31:
32:     except Exception as ex:
33:         print(f'{sys.argv[0]}: {ex}', file=sys.stderr)
34:         sys.exit(1)
35:
36: #-----
37:
38: if __name__ == '__main__':
39:     main()

```

daytime/server.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # server.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:     with sock.makefile(mode='w', encoding='ascii') as flo:
19:         flo.write(datetime + '\n')
20:
21: #-----
22:
23: def main():
24:
25:     if len(sys.argv) != 2:
26:         print(f'usage: python {sys.argv[0]} port', file=sys.stderr)
27:         sys.exit(1)
28:
29:     try:
30:         port = int(sys.argv[1])
31:
32:         server_sock = socket.socket()
33:         print('Opened server socket')
34:         if os.name != 'nt':
35:             server_sock.setsockopt(
36:                 socket.SOL_SOCKET, 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(f'{sys.argv[0]}: {ex}', file=sys.stderr)
54:
55:     except Exception as ex:
56:         print(f'{sys.argv[0]}: {ex}', file=sys.stderr)
57:         sys.exit(1)
58:
59: #-----
60:
61: if __name__ == '__main__':
62:     main()

```

daytimealt/client.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # client.py
5: # Author: Bob Dondero
6: #-----
7: # Try this:
8: # python client.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(f'usage: python {sys.argv[0]} host port', file=sys.stderr)
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:             with sock.makefile(mode='r', encoding='ascii') as flo:
29:                 datetime = flo.read()
30:                 print(datetime)
31:
32:     except Exception as ex:
33:         print(f'{sys.argv[0]}: {ex}', file=sys.stderr)
34:         sys.exit(1)
35:
36: #-----
37:
38: if __name__ == '__main__':
39:     main()

```

daytimealt/server.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # server.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:     with sock.makefile(mode='w', encoding='ascii') as flo:
19:         flo.write(datetime)
20:
21: #-----
22:
23: def main():
24:
25:     if len(sys.argv) != 2:
26:         print(f'usage: python {sys.argv[0]} port', file=sys.stderr)
27:         sys.exit(1)
28:
29:     try:
30:         port = int(sys.argv[1])
31:
32:         server_sock = socket.socket()
33:         print('Opened server socket')
34:         if os.name != 'nt':
35:             server_sock.setsockopt(
36:                 socket.SOL_SOCKET, 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(f'{sys.argv[0]}: {ex}', file=sys.stderr)
54:
55:     except Exception as ex:
56:         print(f'{sys.argv[0]}: {ex}', file=sys.stderr)
57:         sys.exit(1)
58:
59: #-----
60:
61: if __name__ == '__main__':
62:     main()

```

echo/client.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # client.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(f'usage: python {sys.argv[0]} host port', file=sys.stderr)
17:         sys.exit(1)
18:
19:     try:
20:         host = sys.argv[1]
21:         port = int(sys.argv[2])
22:
23:         chars = input()
24:         line = chars + '\n'
25:
26:         with socket.socket() as sock:
27:             sock.connect((host, port))
28:
29:             with sock.makefile(mode='w', encoding='utf-8') as out_flo:
30:                 out_flo.write(line)
31:
32:             with sock.makefile(mode='r', encoding='utf-8') as in_flo:
33:                 echoed_line = in_flo.readline()
34:
35:             if echoed_line == '':
36:                 print('The echo server crashed', file=sys.stderr)
37:             else:
38:                 print(echoed_line, end='')
39:
40:     except Exception as ex:
41:         print(f'{sys.argv[0]}: {ex}', file=sys.stderr)
42:         sys.exit(1)
43:
44: #-----
45:
46: if __name__ == '__main__':
47:     main()

```

echo/server.py (Page 1 of 1)

```

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

```

echoaltdbad/client.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # client.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(f'usage: python {sys.argv[0]} host port', file=sys.stderr)
17:         sys.exit(1)
18:
19:     try:
20:         host = sys.argv[1]
21:         port = int(sys.argv[2])
22:
23:         chars = input()
24:
25:         with socket.socket() as sock:
26:             sock.connect((host, port))
27:
28:             with sock.makefile(mode='w', encoding='utf-8') as out_flo:
29:                 out_flo.write(chars)
30:
31:             with sock.makefile(mode='r', encoding='utf-8') as in_flo:
32:                 echoed_chars = in_flo.read()
33:
34:             if echoed_chars == '':
35:                 print('The echo server crashed', file=sys.stderr)
36:             else:
37:                 print(echoed_chars)
38:
39:     except Exception as ex:
40:         print(f'{sys.argv[0]}: {ex}', file=sys.stderr)
41:         sys.exit(1)
42:
43: #-----
44:
45: if __name__ == '__main__':
46:     main()

```

echoaltdbad/server.py (Page 1 of 1)

```

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

```