

## book.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # book.py
5: # Author: Bob Dondero
6: #-----
7:
8: class Book:
9:
10:     def __init__(self, title, price):
11:         self._title = title
12:         self._price = price
13:
14:     def __str__(self):
15:         return self._title + ', ' + str(self._price)
16:
17:     def set_title(self, title):
18:         self._title = title
19:
20:     def get_title(self):
21:         return self._title
22:
23:     def set_price(self, price):
24:         self._price = price
25:
26:     def get_price(self):
27:         return self._price

```

## course.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # course.py
5: # Author: Bob Dondero
6: #-----
7:
8: class Course:
9:
10:     def __init__(self, name, book):
11:         self._name = name
12:         self._book = book
13:
14:     def __str__(self):
15:         return self._name + '\n' + str(self._book) + '\n'
16:
17:     def set_name(self, name):
18:         self._name = name
19:
20:     def get_name(self):
21:         return self._name
22:
23:     def set_book(self, book):
24:         self._book = book
25:
26:     def get_book(self):
27:         return self._book

```

## courseclient1.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # courseclient1.py
5: # Author: Bob Dondero
6: #-----
7:
8: from sys import exit, argv, stderr
9: from socket import socket
10: from course import Course
11: from book import Book
12:
13: #-----
14:
15: def main():
16:
17:     if len(argv) != 3:
18:         print('Usage: python %s host port' % argv[0])
19:         exit(1)
20:
21:     try:
22:         host = argv[1]
23:         port = int(argv[2])
24:
25:         with socket() as sock:
26:             sock.connect((host, port))
27:             flo = sock.makefile(mode='r', encoding='utf-8')
28:
29:             courses = []
30:
31:             for _ in range(2):
32:                 course_name = flo.readline()
33:                 if course_name == '':
34:                     raise Exception('Ran out of input')
35:                 course_name = course_name.rstrip()
36:
37:                 book_title = flo.readline()
38:                 if book_title == '':
39:                     raise Exception('Ran out of input')
40:                 book_title = book_title.rstrip()
41:
42:                 book_price = flo.readline()
43:                 if book_price == '':
44:                     raise Exception('Ran out of input')
45:                 book_price = book_price.rstrip()
46:
47:                 book = Book(book_title, float(book_price))
48:                 course = Course(course_name, book)
49:                 courses.append(course)
50:
51:             for course in courses:
52:                 print(course)
53:
54:     except Exception as ex:
55:         print(ex, file=stderr)
56:         exit(1)
57:
58: #-----
59:
60: if __name__ == '__main__':
61:     main()

```

## coursesserver1.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # coursesserver1.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 book import Book
12: from course import Course
13:
14: #-----
15:
16: def handle_client(sock):
17:
18:     book1 = Book('C Programming', 88.55)
19:     book2 = Book('The Practice of Programming', 35.14)
20:     course1 = Course('COS 217', book1)
21:     course2 = Course('COS 333', book2)
22:     courses = [course1, course2]
23:
24:     flo = sock.makefile(mode='w', encoding='utf-8')
25:     for course in courses:
26:         flo.write(course.get_name() + '\n')
27:         book = course.get_book()
28:         flo.write(book.get_title() + '\n')
29:         flo.write(str(book.get_price()) + '\n')
30:     flo.flush()
31:     print('Wrote courses to client')
32:
33: #-----
34:
35: def main():
36:
37:     if len(argv) != 2:
38:         print('Usage: python %s port' % argv[0])
39:         exit(1)
40:
41:     try:
42:         port = int(argv[1])
43:         server_sock = socket()
44:         print('Opened server socket')
45:         if name != 'nt':
46:             server_sock.setsockopt(SOL_SOCKET, SO_REUSEADDR, 1)
47:         server_sock.bind(('', port))
48:         print('Bound server socket to port')
49:         server_sock.listen()
50:         print('Listening')
51:         while True:
52:             try:
53:                 sock, client_addr = server_sock.accept()
54:                 with sock:
55:                     print('Accepted connection')
56:                     print('Opened socket')
57:                     print('Server IP addr and port:',
58:                           sock.getsockname())
59:                     print('Client IP addr and port:', client_addr)
60:                     handle_client(sock)
61:             except Exception as ex:
62:                 print(ex, file=stderr)
63:         except Exception as ex:
64:             print(ex, file=stderr)
65:             exit(1)
66:
67: #-----
68:
69: if __name__ == '__main__':
70:     main()

```

## courseclient2.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # courseclient2.py
5: # Author: Bob Dondero
6: #-----
7:
8: from sys import exit, argv, stderr
9: from socket import socket
10: from pickle import load
11: from course import Course
12: from book import Book
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 = 'rb')
29:
30:             courses = []
31:
32:             for _ in range(2):
33:                 course_name = load(flo)
34:                 book_title = load(flo)
35:                 book_price = load(flo)
36:                 book = Book(book_title, book_price)
37:                 course = Course(course_name, book)
38:                 courses.append(course)
39:
40:             for course in courses:
41:                 print(course)
42:
43:     except Exception as ex:
44:         print(ex, file=stderr)
45:         exit(1)
46:
47: #-----
48:
49: if __name__ == '__main__':
50:     main()

```

## courseserver2.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # courseserver2.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 pickle import dump
12: from book import Book
13: from course import Course
14:
15: #-----
16:
17: def handle_client(sock):
18:
19:     book1 = Book('C Programming', 88.55)
20:     book2 = Book('The Practice of Programming', 35.14)
21:     course1 = Course('COS 217', book1)
22:     course2 = Course('COS 333', book2)
23:     courses = [course1, course2]
24:
25:     flo = sock.makefile(mode='wb')
26:     for course in courses:
27:         dump(course.get_name(), flo)
28:         book = course.get_book()
29:         dump(book.get_title(), flo)
30:         dump(book.get_price(), flo)
31:     flo.flush()
32:     print('Wrote courses to client')
33:
34: #-----
35:
36: def main():
37:
38:     if len(argv) != 2:
39:         print('Usage: python %s port' % argv[0])
40:         exit(1)
41:
42:     try:
43:         port = int(argv[1])
44:         server_sock = socket()
45:         print('Opened server socket')
46:         if name != 'nt':
47:             server_sock.setsockopt(SOL_SOCKET, SO_REUSEADDR, 1)
48:         server_sock.bind(('', port))
49:         print('Bound server socket to port')
50:         server_sock.listen()
51:         print('Listening')
52:         while True:
53:             try:
54:                 sock, client_addr = server_sock.accept()
55:                 with sock:
56:                     print('Accepted connection')
57:                     print('Opened socket')
58:                     print('Server IP addr and port:',
59:                           sock.getsockname())
60:                     print('Client IP addr and port:', client_addr)
61:                     handle_client(sock)
62:             except Exception as ex:
63:                 print(ex)
64:         except Exception as ex:
65:             print(ex, file=stderr)
66:             exit(1)
67:
68: #-----
69:
70: if __name__ == '__main__':
71:     main()

```

## courseclient3.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # courseclient3.py
5: # Author: Bob Dondero
6: #-----
7:
8: from sys import exit, argv, stderr
9: from socket import socket
10: from pickle import load
11:
12: #-----
13:
14: def main():
15:
16:     if len(argv) != 3:
17:         print('Usage: python %s host port' % argv[0])
18:         exit(1)
19:
20:     try:
21:         host = argv[1]
22:         port = int(argv[2])
23:
24:         with socket() as sock:
25:             sock.connect((host, port))
26:             flo = sock.makefile(mode='rb')
27:
28:             courses = []
29:             for _ in range(2):
30:                 course = load(flo)
31:                 courses.append(course)
32:
33:             for course in courses:
34:                 print(course)
35:
36:     except Exception as ex:
37:         print(ex, file=stderr)
38:         exit(1)
39:
40: #-----
41:
42: if __name__ == '__main__':
43:     main()

```

## coursesserver3.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # coursesserver3.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 pickle import dump
12: from book import Book
13: from course import Course
14:
15: #-----
16:
17: def handle_client(sock):
18:
19:     book1 = Book('C Programming', 88.55)
20:     book2 = Book('The Practice of Programming', 35.14)
21:     course1 = Course('COS 217', book1)
22:     course2 = Course('COS 333', book2)
23:     courses = [course1, course2]
24:
25:     flo = sock.makefile(mode='wb')
26:     for course in courses:
27:         dump(course, flo)
28:     flo.flush()
29:     print('Wrote courses to client')
30:
31: #-----
32:
33: def main():
34:
35:     if len(argv) != 2:
36:         print('Usage: python %s port' % argv[0])
37:         exit(1)
38:
39:     try:
40:         port = int(argv[1])
41:         server_sock = socket()
42:         print('Opened server socket')
43:         if name != 'nt':
44:             server_sock.setsockopt(SOL_SOCKET, SO_REUSEADDR, 1)
45:         server_sock.bind(('', port))
46:         print('Bound server socket to port')
47:         server_sock.listen()
48:         print('Listening')
49:         while True:
50:             try:
51:                 sock, client_addr = server_sock.accept()
52:                 with sock:
53:                     print('Accepted connection')
54:                     print('Opened socket')
55:                     print('Server IP addr and port:',
56:                           sock.getsockname())
57:                     print('Client IP addr and port:', client_addr)
58:                     handle_client(sock)
59:             except Exception as ex:
60:                 print(ex, file=stderr)
61:         except Exception as ex:
62:             print(ex, file=stderr)
63:             exit(1)
64:
65: #-----
66:
67: if __name__ == '__main__':
68:     main()

```

## courseclient4.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # courseclient4.py
5: # Author: Bob Dondero
6: #-----
7:
8: from sys import exit, argv, stderr
9: from socket import socket
10: from pickle import load
11:
12: #-----
13:
14: def main():
15:
16:     if len(argv) != 3:
17:         print('Usage: python %s host port' % argv[0])
18:         exit(1)
19:
20:     try:
21:         host = argv[1]
22:         port = int(argv[2])
23:
24:         with socket() as sock:
25:             sock.connect((host, port))
26:             flo = sock.makefile(mode='rb')
27:             courses = load(flo)
28:
29:             for course in courses:
30:                 print(course)
31:
32:     except Exception as ex:
33:         print(ex, file=stderr)
34:         exit(1)
35:
36: #-----
37:
38: if __name__ == '__main__':
39:     main()

```

## courseserver4.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # courseserver4.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 pickle import dump
12: from book import Book
13: from course import Course
14:
15: #-----
16:
17: def handle_client(sock):
18:
19:     book1 = Book('C Programming', 88.55)
20:     book2 = Book('The Practice of Programming', 35.14)
21:     course1 = Course('COS 217', book1)
22:     course2 = Course('COS 333', book2)
23:     courses = [course1, course2]
24:
25:     flo = sock.makefile(mode='wb')
26:     dump(courses, flo)
27:     flo.flush()
28:     print('Wrote courses to client')
29:
30: #-----
31:
32: def main():
33:
34:     if len(argv) != 2:
35:         print('Usage: python %s port' % argv[0])
36:         exit(1)
37:
38:     try:
39:         port = int(argv[1])
40:         server_sock = socket()
41:         print('Opened server socket')
42:         if name != 'nt':
43:             server_sock.setsockopt(SOL_SOCKET, SO_REUSEADDR, 1)
44:         server_sock.bind('', port)
45:         print('Bound server socket to port')
46:         server_sock.listen()
47:         print('Listening')
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(ex, file=stderr)
60:         except Exception as ex:
61:             print(ex, file=stderr)
62:             exit(1)
63:
64: #-----
65:
66: if __name__ == '__main__':
67:     main()

```

## courseclient5.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # courseclient5.py
5: # Author: Bob Dondero
6: #-----
7:
8: from sys import exit, argv, stderr
9: from socket import socket
10: from pickle import load
11:
12: #-----
13:
14: def main():
15:
16:     if len(argv) != 3:
17:         print('Usage: python %s host port' % argv[0])
18:         exit(1)
19:
20:     try:
21:         host = argv[1]
22:         port = int(argv[2])
23:
24:         with socket() as sock:
25:             sock.connect((host, port))
26:             flo = sock.makefile(mode = 'rb')
27:             courses = []
28:             for _ in range(2):
29:                 course = load(flo)
30:                 courses.append(course)
31:
32:             for course in courses:
33:                 print(course)
34:
35:             courses[0].get_book().set_price(99.99)
36:
37:             for course in courses:
38:                 print(course)
39:
40:     except Exception as ex:
41:         print(ex, file=stderr)
42:         exit(1)
43:
44: #-----
45:
46: if __name__ == '__main__':
47:     main()

```

## coursesserver5.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # coursesserver5.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 pickle import dump
12: from book import Book
13: from course import Course
14:
15: #-----
16:
17: def handle_client(sock):
18:
19:     # book1 = Book('C Programming', 88.55) # Unused
20:     book2 = Book('The Practice of Programming', 35.14)
21:     course1 = Course('COS 217', book2) # Shared book
22:     course2 = Course('COS 333', book2) # Shared book
23:     courses = [course1, course2]
24:
25:     flo = sock.makefile(mode='wb')
26:     for course in courses:
27:         dump(course, flo)
28:     flo.flush()
29:     print('Wrote courses to client')
30:
31: #-----
32:
33: def main():
34:
35:     if len(argv) != 2:
36:         print('Usage: python %s port' % argv[0])
37:         exit(1)
38:
39:     try:
40:         port = int(argv[1])
41:         server_sock = socket()
42:         print('Opened server socket')
43:         if name != 'nt':
44:             server_sock.setsockopt(SOL_SOCKET, SO_REUSEADDR, 1)
45:         server_sock.bind(('', port))
46:         print('Bound server socket to port')
47:         server_sock.listen()
48:         print('Listening')
49:         while True:
50:             try:
51:                 sock, client_addr = server_sock.accept()
52:                 with sock:
53:                     print('Accepted connection')
54:                     print('Opened socket')
55:                     print('Server IP addr and port:',
56:                           sock.getsockname())
57:                     print('Client IP addr and port:', client_addr)
58:                     handle_client(sock)
59:             except Exception as ex:
60:                 print(ex, file=stderr)
61:         except Exception as ex:
62:             print(ex, file=stderr)
63:             exit(1)
64:
65: #-----
66:
67: if __name__ == '__main__':
68:     main()

```

## courseclient6.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # courseclient6.py
5: # Author: Bob Dondero
6: #-----
7:
8: from sys import exit, argv, stderr
9: from socket import socket
10: from pickle import load
11:
12: #-----
13:
14: def main():
15:
16:     if len(argv) != 3:
17:         print('Usage: python %s host port' % argv[0])
18:         exit(1)
19:
20:     try:
21:         host = argv[1]
22:         port = int(argv[2])
23:
24:         with socket() as sock:
25:             sock.connect((host, port))
26:             flo = sock.makefile(mode='rb')
27:             courses = load(flo)
28:
29:             for course in courses:
30:                 print(course)
31:
32:             courses[0].get_book().set_price(99.99)
33:
34:             for course in courses:
35:                 print(course)
36:
37:     except Exception as ex:
38:         print(ex, file=stderr)
39:         exit(1)
40:
41: #-----
42:
43: if __name__ == '__main__':
44:     main()

```

## courseserver6.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # courseserver6.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 pickle import dump
12: from book import Book
13: from course import Course
14:
15: #-----
16:
17: def handle_client(sock):
18:
19:     # book1 = Book('C Programming', 88.55) # Unused
20:     book2 = Book('The Practice of Programming', 35.14)
21:     course1 = Course('COS 217', book2) # Shared book
22:     course2 = Course('COS 333', book2) # Shared book
23:     courses = [course1, course2]
24:
25:     flo = sock.makefile(mode='wb')
26:     dump(courses, flo)
27:     flo.flush()
28:     print('Wrote courses to client')
29:
30: #-----
31:
32: def main():
33:
34:     if len(argv) != 2:
35:         print('Usage: python %s port' % argv[0])
36:         exit(1)
37:
38:     try:
39:         port = int(argv[1])
40:         server_sock = socket()
41:         print('Opened server socket')
42:         if name != 'nt':
43:             server_sock.setsockopt(SOL_SOCKET, SO_REUSEADDR, 1)
44:         server_sock.bind(('', port))
45:         print('Bound server socket to port')
46:         server_sock.listen()
47:         print('Listening')
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(ex, file=stderr)
60:         except Exception as ex:
61:             print(ex, file=stderr)
62:             exit(1)
63:
64: #-----
65:
66: if __name__ == '__main__':
67:     main()

```

## ipaddress.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # ipaddress.py
5: # Author: Bob Dondero
6: #-----
7:
8: from sys import exit, argv, stderr
9: from socket import gethostbyname_ex
10:
11: #-----
12:
13: def main():
14:
15:     if len(argv) != 2:
16:         print('Usage: python %s domain_name' % argv[0], file=stderr)
17:         exit(1)
18:
19:     domain_name = argv[1]
20:
21:     try:
22:         host_name, aliases, ip_addresses = gethostbyname_ex(domain_name)
23:         print('Host name:', host_name)
24:         for alias in aliases:
25:             print('Alias:', alias)
26:         for ip_address in ip_addresses:
27:             print('IP address:', ip_address)
28:
29:     except Exception as ex:
30:         print(ex, file=stderr)
31:         exit(1)
32:
33: #-----
34:
35: if __name__ == '__main__':
36:     main()

```