PennyWsgi/runserver.py (Page 1 of 1)

```
#!/usr/bin/env python

#-----------------------------------------------------------------------
# runserver.py
# Author: Bob Dondero
#-----------------------------------------------------------------------

import sys
import wsgiref.simple_server
import penny

def main():

    if len(sys.argv) != 2:
        print(f'usage: {sys.argv[0]} port', file=sys.stderr)
        sys.exit(1)

    try:
        port = int(sys.argv[1])
    except Exception:
        print(f'{sys.argv[0]}: Port must be an integer.',
            file=sys.stderr)
        sys.exit(1)

    try:
        httpd = wsgiref.simple_server.make_server(
            '0.0.0.0', port, penny.app)
        print('Listening on port ' + str(port))
        httpd.serve_forever()
    except Exception as ex:
        print(f'{sys.argv[0]}: {ex}', file=sys.stderr)
        sys.exit(1)

if __name__ == '__main__':
    main()
```

PennyWsgi/common.py (Page 1 of 1)

```
#!/usr/bin/env python

#-----------------------------------------------------------------------
# common.py
# Author: Bob Dondero
#-----------------------------------------------------------------------

import time

#-----------------------------------------------------------------------

def get_header():

    if time.strftime('%p') == 'AM':
        ampm = 'morning'
    else:
        ampm = 'afternoon'

    html_str = f'''
        <hr>
        Good {ampm} and welcome to <strong>Penny.com</strong>
        <hr>
        '''

    return html_str

#-----------------------------------------------------------------------

def get_footer():

    current_time = time.asctime(time.localtime())

    html_str = f'''
        <hr>
        Today is {current_time}<br>
        Created by <a href="https://www.cs.princeton.edu/~rdondero">
        Bob Dondero</a>
        <hr>
        '''

    return html_str

#-----------------------------------------------------------------------

# For testing:

def _test():

    print(get_header())
    print()
    print()
    print(get_footer())

if __name__ == '__main__':
    _test()
```

PennyWsgi/penny.py (Page 1 of 3)

```python
#!/usr/bin/env python

#-----------------------------------------------------------------------
# penny.py
# Author: Bob Dondero
#-----------------------------------------------------------------------

import http.cookies
import html
import common
import parseargs
import database

#-----------------------------------------------------------------------

def index(environ, start_response):

    html_code = f'''
        <!DOCTYPE html>
        <html>
            <head>
                <title>Penny.com</title>
            </head>
            <body>
                {common.get_header()}
                <br>
                Click to <a href="/searchform">begin</a>.<br>
                <br>
                {common.get_footer()}
            </body>
        </html>
        '''

    content_header = ('content-type', 'text/html; charset=utf-8')
    headers = [content_header]
    start_response('200 OK', headers)
    return [html_code.encode('utf-8')]

#-----------------------------------------------------------------------

def search_form(environ, start_response):

    prev_author = '(None)'
    if 'HTTP_COOKIE' in environ:
        cookie = http.cookies.SimpleCookie(environ['HTTP_COOKIE'])
        if 'prev_author' in cookie:
            prev_author_morsel = cookie['prev_author']
            if prev_author_morsel:
                prev_author = prev_author_morsel.value

    html_code = f'''
        <!DOCTYPE html>
        <html>
            <head>
                <title>Penny.com</title>
            </head>
            <body>
                {common.get_header()}
                <h1>Author Search</h1>
                <form action="/searchresults" method="get">
                    Please enter an author name:
                    <input type="text" name="author" autofocus>
                    <input type="submit" value="Go">
                </form>
                <br>
```

PennyWsgi/penny.py (Page 2 of 3)

```python
                <br>
                <strong>Previous author search:</strong>
                {html.escape(prev_author)}
                <br>
                <br>
                {common.get_footer()}
            </body>
        </html>
        '''

    content_header = ('content-type', 'text/html; charset=utf-8')
    headers = [content_header]
    start_response('200 OK', headers)
    return [html_code.encode('utf-8')]

#-----------------------------------------------------------------------

def convert_to_html(books):

    if len(books) == 0:
        return '(None)'
    html_code = ''
    for book in books:
        html_code += f'''
            <strong>{html.escape(book['isbn'])}</strong>:
            {html.escape(book['author'])}:
            {html.escape(book['title'])}<br>
            '''
    return html_code

#-----------------------------------------------------------------------

def search_results(environ, start_response):

    args_str = environ.get('QUERY_STRING', '')
    args = parseargs.parse(args_str)
    author = args.get('author', '')

    author = author.strip()

    if author == '':
        prev_author = '(None)'
        books = []
    else:
        prev_author = author
        books = database.get_books(author) # Exception handling omitted

    html_code = f'''
        <!DOCTYPE html>
        <html>
            <head>
                <title>Penny.com</title>
            </head>
            <body>
                {common.get_header()}
                <h1>Author Search Results</h1>
                <h2>Books by {html.escape(prev_author)}:</h2>
                {convert_to_html(books)}
                <br>
                <br>
                Click here to do another
                <a href="/searchform">author search</a>.
                <br>
                <br>
                {common.get_footer()}
```

PennyWsgi/penny.py (Page 3 of 3)

```
            </body>
        </html>
        '''

    content_header = ('content-type', 'text/html; charset=utf-8')
    cookie = http.cookies.SimpleCookie()
    cookie['prev_author'] = prev_author
    cookie_header = ('Set-Cookie', cookie['prev_author'].OutputString())
    headers = [content_header, cookie_header]
    start_response('200 OK', headers)
    return [html_code.encode('utf-8')]

#-----------------------------------------------------------------------

def not_found(environ, start_response):

    html_code = '''
        <!DOCTYPE html>
        <html>
            <head>
                <title>404 Not Found</title>
            </head>
            <body>
                <h1>Not Found</h1>
                <p>The requested URL was not found on the server.
                If you entered the URL manually please check your
                spelling and try again.</p>
            </body>
        </html>
        '''

    content_header = ('content-type', 'text/html; charset=utf-8')
    headers = [content_header]
    start_response('404 Not Found', headers)
    return [html_code.encode('utf-8')]

#-----------------------------------------------------------------------

def app(environ, start_response):

    path = environ.get('PATH_INFO', '').strip('/')

    if path in ('', 'index'):
        return index(environ, start_response)
    if path == 'searchform':
        return search_form(environ, start_response)
    if path == 'searchresults':
        return search_results(environ, start_response)

    return not_found(environ, start_response)
```