

PennyOnePage/penny.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # penny.py
5: # Author: Bob Dondero
6: #-----
7:
8: import time
9: import flask
10: import database
11:
12: #-----
13:
14: app = flask.Flask(__name__, template_folder='.')
15:
16: #-----
17:
18: def get_ampm():
19:     if time.strftime('%p') == "AM":
20:         return 'morning'
21:     return 'afternoon'
22:
23: def get_current_time():
24:     return time.asctime(time.localtime())
25:
26: #-----
27:
28: @app.route('/', methods=['GET'])
29: @app.route('/index', methods=['GET'])
30: def index():
31:
32:     author = flask.request.args.get('author')
33:     if author is None:
34:         author = ''
35:     author = author.strip()
36:
37:     if author == '':
38:         books = []
39:     else:
40:         books = database.get_books(author) # Exception handling omitted
41:
42:     html = flask.render_template('index.html',
43:                                 ampm=get_ampm(),
44:                                 current_time=get_current_time(),
45:                                 author=author,
46:                                 books=books)
47:     response = flask.make_response(html)
48:     return response

```

PennyOnePage/index.html (Page 1 of 1)

```

1: <!DOCTYPE html>
2: <html>
3:
4:     <head>
5:         <title>Penny.com</title>
6:     </head>
7:
8:     <body>
9:         <hr>
10:        Good {{ampm}} and welcome to <strong>Penny.com</strong>
11:        <hr>
12:
13:        <h1>Author Search</h1>
14:        <form action="/index" method="get">
15:            Please enter an author name:
16:            <input type="text" name="author" value="{{author}}" autoFocus>
17:            <input type="submit" value="Go">
18:        </form>
19:        <hr>
20:        {% for book in books: %}
21:            {{book['isbn']}}:
22:            <strong>{{book['author']}}:</strong>
23:            {{book['title']}}<br>
24:        {% endfor %}
25:
26:        <hr>
27:        Date and time: {{current_time}}.<br>
28:        Created by <a href="https://www.cs.princeton.edu/~rdondero">
29:        Bob Dondero</a>
30:        <hr>
31:    </body>
32:
33: </html>

```

PennyJavaScript/penny.py (Page 1 of 1)

```

1: #!/usr/bin/env python
2:
3: #-----
4: # penny.py
5: # Author: Bob Dondero
6: #-----
7:
8: import flask
9: import database
10:
11: #-----
12:
13: app = flask.Flask(__name__, template_folder='.')
14:
15: #-----
16:
17: @app.route('/', methods=['GET'])
18: @app.route('/index', methods=['GET'])
19: def index():
20:
21:     author = flask.request.args.get('author')
22:     if author is None:
23:         author = ''
24:     author = author.strip()
25:
26:     if author == '':
27:         books = []
28:     else:
29:         books = database.get_books(author) # Exception handling omitted
30:
31:     html_code = flask.render_template('index.html',
32:                                     author=author,
33:                                     books=books)
34:     response = flask.make_response(html_code)
35:     return response

```

PennyJavaScript/index.html (Page 1 of 1)

```

1: <!DOCTYPE html>
2: <html>
3:   <head>
4:     <title>Penny.com</title>
5:   </head>
6:
7:   <body>
8:
9:     <hr>
10:    Good <span id="ampmSpan"></span> and welcome to
11:    <strong>Penny.com</strong>
12:    <hr>
13:
14:    <h1>Author Search</h1>
15:    <form action="/index" method="get">
16:      Please enter an author name:
17:      <input type="text" name="author" value="{{author}}" autoFocus>
18:      <input type="submit" value="Go">
19:    </form>
20:    <hr>
21:    {% for book in books: %}
22:      {{book['isbn']}}:
23:      <strong>
24:        {{book['author']}}:
25:      </strong>
26:      {{book['title']}}<br>
27:    {% endfor %}
28:
29:    <hr>
30:    Date and time: <span id="datetimeSpan"></span><br>
31:    Created by <a href="https://www.cs.princeton.edu/~rdondero">
32:    Bob Dondero</a>
33:    <hr>
34:
35:    <script>
36:
37:      'use strict';
38:
39:      function getAmPm() {
40:        let dateTime = new Date();
41:        let hours = dateTime.getHours();
42:        let amPm = (hours < 12) ? 'morning': 'afternoon';
43:        let ampmSpan = document.getElementById('ampmSpan');
44:        ampmSpan.innerHTML = amPm;
45:      }
46:
47:      function getDateTime() {
48:        let dateTime = new Date();
49:        let datetimeSpan = document.getElementById('datetimeSpan');
50:        datetimeSpan.innerHTML = dateTime.toLocaleString();
51:      }
52:
53:      function setup() {
54:        getAmPm();
55:        window.setInterval(getAmPm, 1000);
56:        getDateTime();
57:        window.setInterval(getDateTime, 1000);
58:      }
59:
60:      document.addEventListener('DOMContentLoaded', setup);
61:
62:    </script>
63:  </body>
64: </html>

```

PennyAjax1/penny.py (Page 1 of 1)

```
1: #!/usr/bin/env python
2:
3: #-----
4: # penny.py
5: # Author: Bob Dondero
6: #-----
7:
8: import json
9: import flask
10: import database
11:
12: #-----
13:
14: app = flask.Flask(__name__)
15:
16: #-----
17:
18: @app.route('/', methods=['GET'])
19: @app.route('/index', methods=['GET'])
20: def index():
21:
22:     return flask.send_file('index.html')
23:
24: #-----
25:
26: @app.route('/searchresults', methods=['GET'])
27: def search_results():
28:
29:     author = flask.request.args.get('author')
30:     if author is None:
31:         author = ''
32:     author = author.strip()
33:
34:     if author == '':
35:         books = []
36:     else:
37:         books = database.get_books(author) # Exception handling omitted
38:
39:     json_doc = json.dumps(books)
40:     response = flask.make_response(json_doc)
41:     response.headers['Content-Type'] = 'application/json'
42:     return response
```

blank (Page 1 of 1)

1: This page is intentionally blank.

PennyAjax1/index.html (Page 1 of 2)

```

1: <!DOCTYPE html>
2: <html>
3:   <head>
4:     <title>Penny.com</title>
5:   </head>
6:
7:   <body>
8:
9:     <hr>
10:    Good <span id="ampmSpan"></span> and welcome to
11:    <strong>Penny.com</strong>
12:    <hr>
13:
14:    <h1>Author Search</h1>
15:    Please enter an author name:
16:    <input type="text" id="authorInput" autoFocus>
17:    <hr>
18:    <div id="resultsDiv"></div>
19:
20:    <hr>
21:    Date and time: <span id="datetimeSpan"></span><br>
22:    Created by <a href="https://www.cs.princeton.edu/~rdonero">
23:    Bob Dondero</a>
24:    <hr>
25:
26:    <script>
27:
28:      'use strict';
29:
30:      function getAmPm() {
31:        let dateTime = new Date();
32:        let hours = dateTime.getHours();
33:        let amPm = (hours < 12) ? 'morning' : 'afternoon';
34:        let ampmSpan = document.getElementById('ampmSpan');
35:        ampmSpan.innerHTML = amPm;
36:      }
37:
38:      function getDateime() {
39:        let dateTime = new Date();
40:        let datetimeSpan =
41:        document.getElementById('datetimeSpan');
42:        datetimeSpan.innerHTML = dateTime.toLocaleString();
43:      }
44:
45:      function escape(s) {
46:        s = s.replace('&', '&amp;');
47:        s = s.replace('<', '&lt;');
48:        s = s.replace('>', '&gt;');
49:        s = s.replace('"', '&quot;');
50:        s = s.replace("'", '&apos;');
51:        return s;
52:      }
53:
54:      function convertToHtml(books) {
55:        let html = '';
56:        for (let book of books) {
57:          html += escape(book.isbn) + ': ';
58:          html += '<strong>';
59:          html += escape(book.author);
60:          html += '</strong>: ';
61:          html += escape(book.title) + '<br>';
62:        }
63:        return html;
64:      }
65:

```

PennyAjax1/index.html (Page 2 of 2)

```

66:      function handleResponse() {
67:        if (this.status !== 200) {
68:          alert('Error: Failed to fetch data from server');
69:          return;
70:        }
71:        let books = JSON.parse(this.response);
72:        let html = convertToHtml(books);
73:        let resultsDiv = document.getElementById('resultsDiv');
74:        resultsDiv.innerHTML = html;
75:      }
76:
77:      function handleError() {
78:        alert('Error: Failed to fetch data from server');
79:      }
80:
81:      function getResults() {
82:        let authorInput = document.getElementById('authorInput');
83:        let author = authorInput.value;
84:        let encodedAuthor = encodeURIComponent(author);
85:        let url = '/searchresults?author=' + encodedAuthor;
86:        let request = new XMLHttpRequest();
87:        request.onload = handleResponse;
88:        request.onerror = handleError;
89:        request.open('GET', url);
90:        request.send();
91:      }
92:
93:      function setup() {
94:        getAmPm();
95:        window.setInterval(getAmPm, 1000);
96:        getDateime();
97:        window.setInterval(getDateime, 1000);
98:        let authorInput = document.getElementById('authorInput');
99:        authorInput.addEventListener('input', getResults);
100:      }
101:
102:      document.addEventListener('DOMContentLoaded', setup);
103:
104:    </script>
105:  </body>
106: </html>

```

PennyAjax2/index.html (Page 1 of 2)

```

1: <!DOCTYPE html>
2: <html>
3:   <head>
4:     <title>Penny.com</title>
5:   </head>
6:
7:   <body>
8:
9:     <hr>
10:    Good <span id="ampmSpan"></span> and welcome to
11:    <strong>Penny.com</strong>
12:    <hr>
13:
14:    <h1>Author Search</h1>
15:    Please enter an author name:
16:    <input type="text" id="authorInput" autoFocus>
17:    <hr>
18:    <div id="resultsDiv"></div>
19:
20:    <hr>
21:    Date and time: <span id="datetimeSpan"></span><br>
22:    Created by <a href="https://www.cs.princeton.edu/~rdondero">
23:    Bob Dondero</a>
24:    <hr>
25:
26:    <script src=
27:      "https://cdn.jsdelivr.net/npm/mustache@4.2.0/mustache.min.js">
28:    </script>
29:
30:    <!-- <script src="/static/mustache.min.js"></script> -->
31:
32:    <script>
33:
34:      'use strict';
35:
36:      function getAmPm() {
37:        let dateTime = new Date();
38:        let hours = dateTime.getHours();
39:        let amPm = (hours < 12) ? 'morning' : 'afternoon';
40:        let ampMspan = document.getElementById('ampmSpan');
41:        ampMspan.innerHTML = amPm;
42:      }
43:
44:      function getDateTime() {
45:        let dateTime = new Date();
46:        let datetimeSpan =
47:          document.getElementById('datetimeSpan');
48:        datetimeSpan.innerHTML = dateTime.toLocaleString();
49:      }
50:
51:      function convertToHtml(books) {
52:        let template = `
53:          {{#books}}
54:            {{isbn}}:
55:            <strong>{{author}}</strong>:
56:            {{title}}
57:            <br>
58:          {{/books}}
59:        `;
60:        let map = {books: books};
61:        let html = Mustache.render(template, map);
62:        return html;
63:      }
64:
65:      function handleResponse() {

```

PennyAjax2/index.html (Page 2 of 2)

```

66:        if (this.status !== 200) {
67:          alert('Error: Failed to fetch data from server');
68:          return;
69:        }
70:        let books = JSON.parse(this.response);
71:        let html = convertToHtml(books);
72:        let resultsDiv = document.getElementById('resultsDiv');
73:        resultsDiv.innerHTML = html;
74:      }
75:
76:      function handleError() {
77:        alert('Error: Failed to fetch data from server');
78:      }
79:
80:      function getResults() {
81:        let authorInput = document.getElementById('authorInput');
82:        let author = authorInput.value;
83:        let encodedAuthor = encodeURIComponent(author);
84:        let url = '/searchresults?author=' + encodedAuthor;
85:        let request = new XMLHttpRequest();
86:        request.onload = handleResponse;
87:        request.onerror = handleError;
88:        request.open('GET', url);
89:        request.send();
90:      }
91:
92:      function setup() {
93:        getAmPm();
94:        window.setInterval(getAmPm, 1000);
95:        getDateTime();
96:        window.setInterval(getDateTime, 1000);
97:        let authorInput = document.getElementById('authorInput');
98:        authorInput.addEventListener('input', getResults);
99:      }
100:
101:      document.addEventListener('DOMContentLoaded', setup);
102:
103:    </script>
104:  </body>
105: </html>

```

PennyAjax3/index.html (Page 1 of 2)

```

1: <!DOCTYPE html>
2: <html>
3:   <head>
4:     <title>Penny.com</title>
5:   </head>
6:   <body>
7:
8:
9:     <hr>
10:    Good <span id="ampmSpan"></span> and welcome to
11:    <strong>Penny.com</strong>
12:    <hr>
13:
14:    <h1>Author Search</h1>
15:    Please enter an author name:
16:    <input type="text" id="authorInput" autoFocus>
17:    <hr>
18:    <div id="resultsDiv"></div>
19:
20:    <hr>
21:    Date and time: <span id="datetimeSpan"></span><br>
22:    Created by <a href="https://www.cs.princeton.edu/~rdondero">
23:    Bob Dondero</a>
24:    <hr>
25:
26:    <script src=
27:    "https://cdn.jsdelivr.net/npm/mustache@4.2.0/mustache.min.js">
28:    </script>
29:
30:    <script>
31:
32:      'use strict';
33:
34:      function getAmPm() {
35:        let dateTime = new Date();
36:        let hours = dateTime.getHours();
37:        let amPm = (hours < 12) ? 'morning' : 'afternoon';
38:        let ampMspan = document.getElementById('ampmSpan');
39:        ampMspan.innerHTML = amPm;
40:      }
41:
42:      function getDateTime() {
43:        let dateTime = new Date();
44:        let datetimeSpan =
45:        document.getElementById('datetimeSpan');
46:        datetimeSpan.innerHTML = dateTime.toLocaleString();
47:      }
48:
49:      function convertToHtml(books) {
50:        let template = `
51:          {{#books}}
52:            {{isbn}}:
53:            <strong>{{author}}</strong>:
54:            {{title}}
55:            <br>
56:          {{/books}}
57:        `;
58:        let map = {books: books};
59:        let html = Mustache.render(template, map);
60:        return html;
61:      }
62:
63:      function handleResponse() {
64:        if (this.status !== 200) {
65:          alert('Error: Failed to fetch data from server');

```

PennyAjax3/index.html (Page 2 of 2)

```

66:          return;
67:        }
68:        let books = JSON.parse(this.response);
69:        let html = convertToHtml(books);
70:        let resultsDiv = document.getElementById('resultsDiv');
71:        resultsDiv.innerHTML = html;
72:      }
73:
74:      function handleError() {
75:        alert('Error: Failed to fetch data from server');
76:      }
77:
78:      let request = null;
79:
80:      function getResults() {
81:        let authorInput = document.getElementById('authorInput');
82:        let author = authorInput.value;
83:        let encodedAuthor = encodeURIComponent(author);
84:        let url = '/searchresults?author=' + encodedAuthor;
85:        if (request !== null)
86:          request.abort();
87:        request = new XMLHttpRequest();
88:        request.onload = handleResponse;
89:        request.onerror = handleError;
90:        request.open('GET', url);
91:        request.send();
92:      }
93:
94:      function setup() {
95:        getAmPm();
96:        window.setInterval(getAmPm, 1000);
97:        getDateTime();
98:        window.setInterval(getDateTime, 1000);
99:        let authorInput = document.getElementById('authorInput');
100:        authorInput.addEventListener('input', getResults);
101:      }
102:
103:      document.addEventListener('DOMContentLoaded', setup);
104:
105:    </script>
106:  </body>
107: </html>

```

PennyAjax4/index.html (Page 1 of 2)

```

1: <!DOCTYPE html>
2: <html>
3:   <head>
4:     <title>Penny.com</title>
5:   </head>
6:   <body>
7:
8:
9:     <hr>
10:    Good <span id="ampmSpan"></span> and welcome to
11:    <strong>Penny.com</strong>
12:    <hr>
13:
14:    <h1>Author Search</h1>
15:    Please enter an author name:
16:    <input type="text" id="authorInput" autoFocus>
17:    <hr>
18:    <div id="resultsDiv"></div>
19:
20:    <hr>
21:    Date and time: <span id="datetimeSpan"></span><br>
22:    Created by <a href="https://www.cs.princeton.edu/~rdondero">
23:    Bob Dondero</a>
24:    <hr>
25:
26:    <script src=
27:    "https://cdn.jsdelivr.net/npm/mustache@4.2.0/mustache.min.js">
28:    </script>
29:
30:    <script>
31:
32:      'use strict';
33:
34:      function getAmPm() {
35:        let dateTime = new Date();
36:        let hours = dateTime.getHours();
37:        let amPm = (hours < 12) ? 'morning' : 'afternoon';
38:        let ampMspan = document.getElementById('ampmSpan');
39:        ampMspan.innerHTML = amPm;
40:      }
41:
42:      function getDateime() {
43:        let dateTime = new Date();
44:        let datetimeSpan =
45:        document.getElementById('datetimeSpan');
46:        datetimeSpan.innerHTML = dateTime.toLocaleString();
47:      }
48:
49:      function convertToHtml(books) {
50:        let template = `
51:          {{#books}}
52:            {{isbn}}:
53:            <strong>{{author}}</strong>:
54:            {{title}}
55:            <br>
56:          {{/books}}
57:        `;
58:        let map = {books: books};
59:        let html = Mustache.render(template, map);
60:        return html;
61:      }
62:
63:      function handleResponse() {
64:        if (this.status !== 200) {
65:          alert('Error: Failed to fetch data from server');

```

PennyAjax4/index.html (Page 2 of 2)

```

66:          return;
67:        }
68:        let books = JSON.parse(this.response);
69:        let html = convertToHtml(books);
70:        let resultsDiv = document.getElementById('resultsDiv');
71:        resultsDiv.innerHTML = html;
72:      }
73:
74:      function handleError() {
75:        alert('Error: Failed to fetch data from server');
76:      }
77:
78:      let request = null;
79:
80:      function getResults() {
81:        let authorInput = document.getElementById('authorInput');
82:        let author = authorInput.value;
83:        let encodedAuthor = encodeURIComponent(author);
84:        let url = '/searchresults?author=' + encodedAuthor;
85:        if (request !== null)
86:          request.abort();
87:        request = new XMLHttpRequest();
88:        request.onload = handleResponse;
89:        request.onerror = handleError;
90:        request.open('GET', url);
91:        request.send();
92:      }
93:
94:      let timer = null;
95:
96:      function debouncedGetResults() {
97:        clearTimeout(timer);
98:        timer = setTimeout(getResults, 500);
99:      }
100:
101:      function setup() {
102:        getAmPm();
103:        window.setInterval(getAmPm, 1000);
104:        getDateime();
105:        window.setInterval(getDateime, 1000);
106:        let authorInput = document.getElementById('authorInput');
107:        authorInput.addEventListener('input', debouncedGetResults);
108:      }
109:
110:      document.addEventListener('DOMContentLoaded', setup);
111:
112:    </script>
113:  </body>
114: </html>

```

PennyAjaxFetch1/index.html (Page 1 of 2)

```

1: <!DOCTYPE html>
2: <html>
3:   <head>
4:     <title>Penny.com</title>
5:   </head>
6:
7:   <body>
8:
9:     <hr>
10:    Good <span id="ampmSpan"></span> and welcome to
11:    <strong>Penny.com</strong>
12:    <hr>
13:
14:    <h1>Author Search</h1>
15:    Please enter an author name:
16:    <input type="text" id="authorInput" autoFocus>
17:    <hr>
18:    <div id="resultsDiv"></div>
19:
20:    <hr>
21:    Date and time: <span id="datetimeSpan"></span><br>
22:    Created by <a href="https://www.cs.princeton.edu/~rdondero">
23:    Bob Dondero</a>
24:    <hr>
25:
26:    <script src=
27:    "https://cdn.jsdelivr.net/npm/mustache@4.2.0/mustache.min.js">
28:    </script>
29:
30:    <script>
31:
32:      'use strict';
33:
34:      function getAmPm() {
35:        let dateTime = new Date();
36:        let hours = dateTime.getHours();
37:        let amPm = (hours < 12) ? 'morning' : 'afternoon';
38:        let ampmSpan = document.getElementById('ampmSpan');
39:        ampmSpan.innerHTML = amPm;
40:      }
41:
42:      function getDateTime() {
43:        let dateTime = new Date();
44:        let datetimeSpan =
45:        document.getElementById('datetimeSpan');
46:        datetimeSpan.innerHTML = dateTime.toLocaleString();
47:      }
48:
49:      function usingResponseGetText(response) {
50:        if (! response.ok)
51:          throw new Error();
52:        return response.text();
53:      }
54:
55:      function convertToHtml(books) {
56:        let template = `
57:        {{#books}}
58:          {{isbn}}:
59:          <strong>{{author}}</strong>:
60:          {{title}}
61:          <br>
62:        {{/books}}
63:        `;
64:        let map = {books: books};
65:        let html = Mustache.render(template, map);

```

PennyAjaxFetch1/index.html (Page 2 of 2)

```

66:          return html;
67:        }
68:
69:        function usingTextUpdateResultsDiv(text) {
70:          let books = JSON.parse(text);
71:          let html = convertToHtml(books);
72:          let resultsDiv = document.getElementById('resultsDiv');
73:          resultsDiv.innerHTML = html;
74:        }
75:
76:        function handleError(err) {
77:          if (err.name !== 'AbortError')
78:            alert('Error: Failed to fetch data from server');
79:        }
80:
81:        let controller = null;
82:
83:        function getResults() {
84:          let authorInput = document.getElementById('authorInput');
85:          let author = authorInput.value;
86:          let encodedAuthor = encodeURIComponent(author);
87:          let url = '/searchresults?author=' + encodedAuthor;
88:          if (controller !== null)
89:            controller.abort();
90:          controller = new AbortController();
91:          fetch(url, {signal: controller.signal})
92:            .then(usingResponseGetText)
93:            .then(usingTextUpdateResultsDiv)
94:            .catch(handleError);
95:        }
96:
97:        let timer = null;
98:
99:        function debouncedGetResults() {
100:          clearTimeout(timer);
101:          timer = setTimeout(getResults, 500);
102:        }
103:
104:        function setup() {
105:          getAmPm();
106:          window.setInterval(getAmPm, 1000);
107:          getDateTime();
108:          window.setInterval(getDateTime, 1000);
109:          let authorInput = document.getElementById('authorInput');
110:          authorInput.addEventListener('input', debouncedGetResults);
111:        }
112:
113:        document.addEventListener('DOMContentLoaded', setup);
114:
115:      </script>
116:    </body>
117:  </html>

```


PennyAjaxFetch2/index.html (Page 1 of 2)

```

1: <!DOCTYPE html>
2: <html>
3:   <head>
4:     <title>Penny.com</title>
5:   </head>
6:
7:   <body>
8:
9:     <hr>
10:    Good <span id="ampmSpan"></span> and welcome to
11:    <strong>Penny.com</strong>
12:    <hr>
13:
14:    <h1>Author Search</h1>
15:    Please enter an author name:
16:    <input type="text" id="authorInput" autoFocus>
17:    <hr>
18:    <div id="resultsDiv"></div>
19:
20:    <hr>
21:    Date and time: <span id="datetimeSpan"></span><br>
22:    Created by <a href="https://www.cs.princeton.edu/~rdondero">
23:    Bob Dondero</a>
24:    <hr>
25:
26:    <script src=
27:    "https://cdn.jsdelivr.net/npm/mustache@4.2.0/mustache.min.js">
28:    </script>
29:
30:    <script>
31:      'use strict';
32:
33:      function getAmPm() {
34:        let dateTime = new Date();
35:        let hours = dateTime.getHours();
36:        let amPm = (hours < 12) ? 'morning' : 'afternoon';
37:        let ampMspan = document.getElementById('ampmSpan');
38:        ampMspan.innerHTML = amPm;
39:      }
40:
41:      function getDateime() {
42:        let dateTime = new Date();
43:        let datetimeSpan =
44:        document.getElementById('datetimeSpan');
45:        datetimeSpan.innerHTML = dateTime.toLocaleString();
46:      }
47:
48:      function convertToHtml(books) {
49:        let template = `
50:          {{#books}}
51:            {{isbn}}:
52:            <strong>{{author}}</strong>:
53:            {{title}}
54:            <br>
55:          {{/books}}
56:        `;
57:        let map = {books: books};
58:        let html = Mustache.render(template, map);
59:        return html;
60:      }
61:
62:      async function fetchBooks(url) {
63:        try {
64:          let response =

```

PennyAjaxFetch2/index.html (Page 2 of 2)

```

66:          await fetch(url, {signal: controller.signal});
67:          if (!response.ok)
68:            throw new Error();
69:          let text = await response.text();
70:          let books = JSON.parse(text);
71:          let html = convertToHtml(books);
72:          let resultsDiv = document.getElementById('resultsDiv');
73:          resultsDiv.innerHTML = html;
74:        }
75:      } catch (err) {
76:        if (err.name !== 'AbortError')
77:          alert('Error: Failed to fetch data from server');
78:      }
79:    }
80:
81:    let controller = null;
82:
83:    function getResults() {
84:      let authorInput = document.getElementById('authorInput');
85:      let author = authorInput.value;
86:      let encodedAuthor = encodeURIComponent(author);
87:      let url = '/searchresults?author=' + encodedAuthor;
88:      if (controller !== null)
89:        controller.abort();
90:      controller = new AbortController();
91:      fetchBooks(url);
92:    }
93:
94:    let timer = null;
95:
96:    function debouncedGetResults() {
97:      clearTimeout(timer);
98:      timer = setTimeout(getResults, 500);
99:    }
100:
101:    function setup() {
102:      getAmPm();
103:      window.setInterval(getAmPm, 1000);
104:      getDateime();
105:      window.setInterval(getDateime, 1000);
106:      let authorInput = document.getElementById('authorInput');
107:      authorInput.addEventListener('input', debouncedGetResults);
108:    }
109:
110:    document.addEventListener('DOMContentLoaded', setup);
111:
112:  </script>
113: </body>
114: </html>

```