

# **WEB700 Web Programming**

## **Assignment # 4**

Student ID: 112867247

Student Name: Kulsum Timol

E-mail: katimol@myseneca.ca

Semester:2

Professor name: Pritesh Patel

## About.html File

The screenshot displays a VS Code editor window with the following details:

- File Explorer (Top):** Shows a project structure with files: `about.html`, `theme.css`, `server.js`, and `legoSets.js`.
- Editor Content:** The active file is `about.html`. It contains an HTML document with a Bootstrap navbar. The navbar includes a toggle button and a list of links: `Home`, `About`, and a link to `about.html` which is currently active.
- Editor Interface:** The editor has a dark theme. The left sidebar contains icons for Explorer, Search, Source Control, and Run and Debug. The top bar shows the file explorer with the selected file `about.html`. The bottom status bar indicates the current position is `Ln 30, Col 1`.

The screenshot shows a VS Code editor with the following content in 'about.html':

```

<html>
<body>
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <div class="container">
      <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="navbar-nav me-auto">
          <li class="nav-item"><a class="nav-link" href="/">Home</a></li>
          <li class="nav-item">
            <a class="nav-link active" href="/about">About</a>
          </li>
        </ul>
        <ul class="navbar-nav ms-auto">
          <li class="nav-item">
            <a class="nav-link" href="/lego/sets/0013-1">Set</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="/lego/sets?theme=tech">Theme</a>
          </li>
        </ul>
      </div>
    </nav>

    <br />

    <div class="container">
      <div class="row">
        <main class="col-md-12">
          <section class="bg-light text-center py-5">
            <div class="container">
              <h1 class="display-5 fw-bold">About</h1>
              <p class="lead text-muted">
                Everything you want to know about me - the designer & developer.
              </p>
              <p>
                I'm a student at Seneca Polytechnic, currently enrolled in the

```

The status bar at the bottom indicates the file is on the 'master\*' branch and is 'Not Committed Yet'. The cursor is at line 47, column 13.

```
File Edit Selection View Go Run ... assignment4
about.html M X # theme.css U JS server.js M JS legoSets.js M
views > about.html > html > body > nav.navbar.navbar-expand-lg.navbar-light.bg-light > div.container
2 <html>
15 <body>
52 <div class="container">
53 <div class="row">
54 <main class="col-md-12">
55 <section class="bg-light text-center py-5">
56 <div class="container">
57 <h1 class="display-5 fw-bold">About</h1>
58 <p class="lead text-muted">
59 Everything you want to know about me - the designer & developer.
60 </p>
61 <p>
62 I'm a student at Seneca Polytechnic, currently enrolled in the
63 BTT program. I'm passionate about coding, love working on
64 creative web projects, and enjoy Reading and music in my spare
65 time.
66 </p>
67 </div>
68 </section>
69 <section class="py-5">
70 <div class="container">
71 <h2>Theme Demo</h2>
72 <p>
73 This demo shows how different HTML elements appear using my
74 custom site theme.
75 </p>
76
77 <!-- Tables Section -->
78 <h3>Tables</h3>
79 <table class="table table-bordered">
80 <caption>
81 Programming Skills
82 </caption>
83 <thead>
84 <tr>
85 <th>Language</th>
86 <th>Experience</th>
```

```
File Edit Selection View Go Run ... assignment4
about.html M X # theme.css U JS server.js M JS legoSets.js M
views > about.html > html > body > nav.navbar.navbar-expand-lg.navbar-light.bg-light > div.container
2 <html>
15 <body>
52 <div class="container">
53 <div class="row">
54 <main class="col-md-12">
77 <!-- Tables Section -->
78 <h3>Tables</h3>
79 <table class="table table-bordered">
80 <caption>
81 Programming Skills
82 </caption>
83 <thead>
84 <tr>
85 <th>Language</th>
86 <th>Experience</th>
87 <th>Proficiency</th>
88 </tr>
89 </thead>
90 <tbody>
91 <tr>
92 <td>JavaScript</td>
93 <td>2 Years</td>
94 <td rowspan="2">Intermediate</td>
95 </tr>
96 <tr>
97 <td>Python</td>
98 <td>3 Years</td>
99 </tr>
100 <tr>
101 <td colspan="3">Learning More Every Day!</td>
102 </tr>
103 </tbody>
104 </table>
```

```
File Edit Selection View Go Run ... assignment4
about.html M X # theme.css U X JS server.js M JS legoSets.js M
views > > about.html > html > body > nav.navbar.navbar-expand-lg.navbar-light.bg-light > div.container
2 <html>
15 <body>
52 <div class="container">
53 <div class="row">
54 <main class="col-md-12">
105 <tr>
106 <td colspan="3">Updated July 2025</td>
107 </tr>
108 </tfoot>
109 </table>
110
111 <table class="table table-bordered">
112 <caption>
113 Product Information
114 </caption>
115 <thead>
116 <tr>
117 <th>Name</th>
118 <th>Price</th>
119 <th>Store Location</th>
120 </tr>
121 </thead>
122 <tbody>
123 <tr>
124 <td>Wireless Mouse</td>
125 <td>$25.99</td>
126 <td>New York</td>
127 </tr>
128 <tr>
129 <td>Bluetooth Headphones</td>
130 <td>$59.99</td>
131 <td colspan="2">New York</td>
132 </tr>
133
134 <tr>
135 <td>USB-C Charger</td>
136 <td>$49.99</td>
137 <td>Toronto</td>
138 </tr>
139 <tr>
140 <td>Laptop Stand</td>
141 <td>$45.99</td>
142 <td>Toronto</td>
143 </tr>
144 </tbody>
145 <tfoot>
146 <tr>
147 <td colspan="3">End of Product List</td>
148 </tr>
149 </tfoot>
150 </table>
151
152 <p>Product Price and Store Location.</p>
153
154 <!-- Lists Section -->
155 <h3>Lists</h3>
156 <ol>
157 <li>
158 Frontend
159 <ul>
```

```
File Edit Selection View Go Run ... assignment4
about.html M X # theme.css U X JS server.js M JS legoSets.js M
views > > about.html > html > body > nav.navbar.navbar-expand-lg.navbar-light.bg-light > div.container
2 <html>
15 <body>
52 <div class="container">
53 <div class="row">
54 <main class="col-md-12">
131 <td colspan="2">New York</td>
132 </tr>
133 <tr>
134 <td>USB-C Charger</td>
135 <td>$49.99</td>
136 <td>Toronto</td>
137 </tr>
138 <tr>
139 <td>Laptop Stand</td>
140 <td>$45.99</td>
141 <td>Toronto</td>
142 </tr>
143 </tbody>
144 <tfoot>
145 <tr>
146 <td colspan="3">End of Product List</td>
147 </tr>
148 </tfoot>
149 </table>
150
151 <p>Product Price and Store Location.</p>
152
153 <!-- Lists Section -->
154 <h3>Lists</h3>
155 <ol>
156 <li>
157 Frontend
158 <ul>
```

```
<? about.html M X # theme.css U X JS server.js M JS legoSets.js M
views > <? about.html > <? html > <? body > <? nav.navbar.navbar-expand-lg.navbar-light.bg-light > <? div.container
2 <html>
15 <body>
52 <div class="container">
53 <div class="row">
54 <main class="col-md-12">
145 <tr>
146 <td colspan="3">End of Product List</td>
147 </tr>
148 </tfoot>
149 </table>
150
151 <p>Product Price and Store Location.</p>
152
153 <!-- Lists Section -->
154 <h3>Lists</h3>
155 <ol>
156 <li>
157 Frontend
158 <ul>
159 <li>HTML</li>
160 <li>CSS</li>
161 <li>JavaScript</li>
162 </ul>
163 </li>
164 <li>Backend</li>
165 <li>Database</li>
166 <li>Cloud</li>
167 <li>DevOps</li>
168 </ol>
169 <p>
170 The list shows my technical interests, with a nested list under
171 frontend.
172 </p>
```

```
<? about.html M X # theme.css U X JS server.js M JS legoSets.js M
views > <? about.html > <? html > <? body > <? nav.navbar.navbar-expand-lg.navbar-light.bg-light > <? div.container
2 <html>
15 <body>
52 <div class="container">
53 <div class="row">
54 <main class="col-md-12">
173 <!-- Images Section -->
174 <h3>Images</h3>
175 
180 
183 
188 <p>
189 These images showcase different Lego products, like bricks and
190 figures.
191 </p>
192
193 <!-- Audio Section -->
194 <h3>Audio</h3>
195 <audio controls>
196 <source
197 src="https://pat-crawford-sdds.netlify.app/shared/winter-2025/web700/AA/sample-audio.mp3"
198 type="audio/mp3"
199 />
200 <source
201 src="https://pat-crawford-sdds.netlify.app/shared/winter-2025/web700/AA/sample-audio.ogg"
202 type="audio/ogg"
203 />
204 </audio>
205 <p>
206 This audio sample demonstrates how sound media is integrated on
207 the site.
208 </p>
209
210 <!-- Video Section -->
211 <h3>Video</h3>
212 <video width="480" height="320" controls>
213 <source
214 src="https://pat-crawford-sdds.netlify.app/shared/winter-2025/web700/AA/sample-video.mp4"
215 type="video/mp4"
216 />
217 <source
218 src="https://pat-crawford-sdds.netlify.app/shared/winter-2025/web700/AA/sample-video.webm"
219 type="video/webm"
220 />
221 </video>
222 <p>
223 This video section allows us to display rich multimedia content
224 using HTML5.
225 </p>
226 </div>
227 </section>
228 </main>
229 </div>
```

## Step 1: HTML Elements Demo (about.html)

To begin, I First wrote a short paragraph describing Myself.

Then I created a relevant **header** and **paragraph** to describe the content of the next section – AS the "Theme Demo".

Then I added Following Sections in my about.html page:

## Tables Section

I added a section titled "**Tables**" that contains two HTML tables. Each table:

- Has at least 3 rows and 3 columns
- Uses <caption>, <thead>, <tbody>, and <tfoot>
- Includes rowspan and colspan for merged cells
- Contains meaningful content such as product listings and schedules

This demonstrates how my table styles render under the new theme.

## Lists Section

Next, I added a section titled "**Lists**" containing one ordered list (<ol>) with at least 5 items. One of the items includes a nested unordered list (<ul>) with 3 additional items. This showcases how list styling and spacing are handled.

## Images Section

This section includes 3 images sourced from Wikimedia Commons, each with appropriate alt text and external URLs. At the bottom, I added a paragraph explaining the content of the images. These images display how the theme handles media.

## Audio Section

I embedded an HTML5 audio player using the sample .mp3 and .ogg files provided in the assignment. A short paragraph explains what the audio player demonstrates.

## Video Section

Lastly, I added an HTML5 video player with the provided .mp4 and .webm formats. This shows how video components are styled within the theme.

## Theme.css File

```
<> about.html M # theme.css U X JS server.js M JS legoSets.js M
public > css > # theme.css > ...
1  /* Import Google Font */
2  @import url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;600&display=swap');
3  @import url('https://fonts.googleapis.com/css2?family=Pacifico&display=swap');
4  @import url('https://fonts.googleapis.com/css2?family=Cookie&display=swap');
5
6
7  /* Set app-wide font and gradient background */
8  body, html {
9      font-family: 'Poppins', sans-serif;
10     background: linear-gradient(to bottom, #c7c6f6, #e9e9f9);
11     overflow: auto;
12     margin: 0;
13     padding: 0;
14     color: #333;
15 }
16
17 /* -----
18 | Navigation Bar & Hero
19 | ----- */
20 .navbar.bg-light {
21     background: linear-gradient(to right, #b6b0de) !important;
22     box-shadow: 0 4px 8px rgba(0,0,0,0.2);
23 }
24
25 /* Navigation Brand */
26 .navbar-light .navbar-brand {
27     font-family: 'Pacifico', sans-serif;
28     font-size: 1.6rem;
29     font-weight: 600;
30     color: #000000 !important;
31     text-shadow: 1px 1px 3px rgba(0, 0, 0, 0.5);
32 }
33
34 /* Navigation Links */
35 .nav-link {
36     font-size: 1rem;
37     color: #000000 !important;
38     transition: color 0.3s ease;
39 }
40
41 .nav-link:hover {
42     color: #000000 !important;
43     text-decoration: underline;
44 }
45
46 /* Headings */
47 h1, h2, h3, h4, h5, h6 {
48     font-family: 'Pacifico', sans-serif;
49     font-weight: 600;
50     color: #000000;
51     text-shadow: 1px 1px 0 #ccc;
52 }
53
54 /* Tables */
55 .table {
56     border-collapse: collapse;
57     width: 100%;
58     margin: 1rem 0;
59 }
60
61 .table caption {
62     caption-side: top;
63     font-weight: bold;
64     padding: 0.5rem;
65     color: #6a1b9a;
66 }
67
68 .table th, .table td {
69     border: 1px solid #ccc;
70     padding: 10px;
71     text-align: left;
72 }
73
74 .table tr:nth-child(odd) td {
75     background: linear-gradient(to right, #f2efff, #ffffff);
76 }
```

```
<> about.html M # theme.css U X JS server.js M JS legoSets.js M
public > css > # theme.css > ...
32 }
33
34 /* Navigation Links */
35 .nav-link {
36     font-size: 1rem;
37     color: #000000 !important;
38     transition: color 0.3s ease;
39 }
40
41 .nav-link:hover {
42     color: #000000 !important;
43     text-decoration: underline;
44 }
45
46 /* Headings */
47 h1, h2, h3, h4, h5, h6 {
48     font-family: 'Pacifico', sans-serif;
49     font-weight: 600;
50     color: #000000;
51     text-shadow: 1px 1px 0 #ccc;
52 }
53
54 /* Tables */
55 .table {
56     border-collapse: collapse;
57     width: 100%;
58     margin: 1rem 0;
59 }
60
61 .table caption {
62     caption-side: top;
63     font-weight: bold;
64     padding: 0.5rem;
65     color: #6a1b9a;
66 }
67
68 .table th, .table td {
69     border: 1px solid #ccc;
70     padding: 10px;
71     text-align: left;
72 }
73
74 .table tr:nth-child(odd) td {
75     background: linear-gradient(to right, #f2efff, #ffffff);
76 }
```

```
<> about.html M # theme.css U X JS server.js M JS legoSets.js M
public > css > # theme.css > img
78  /* Anchor Tags (All Links) */
79  a {
80    color: #4b0082;
81    text-decoration: none;
82  }
83
84  a:hover {
85    color: #2c1a5b;
86    text-decoration: underline;
87  }
88
89  /* Lists */
90  ul, ol {
91    margin-left: 2rem;
92    padding-bottom: 1rem;
93  }
94
95  li {
96    margin-bottom: 0.5rem;
97  }
98
99  /* Media Elements (Images, Audio, Video) */
100 img, video, audio {
101   display: block;
102   margin: 1rem auto;
103   max-width: 100%;
104   border-radius: 12px;
105   box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);
106 }
107
108 /* Section */
109 section {
110   background-color: #dedcfc;
111   padding: 2rem 1.5rem;
112   margin-bottom: 2rem;
113   border-radius: 16px;
114   box-shadow: 0 6px 14px rgba(0, 0, 0, 0.1);
115 }
116
117
master* 0 0 0 Ln 106, Col 2 Spaces: 2 UTF-8 CRLF CSS Go Live
```

## Step 2: CSS for custom "Theme" (theme.css)

### Explanation:

I imported three different Google Fonts to give my website a custom and polished look:

- **Poppins** is used for general body content because it's clean and highly readable.
- **Pacifico** is used for headings and the navbar brand to give a friendly, handwritten feel.
- **Cookie** is included but not used; I initially considered it for headers but chose Pacifico instead

```
body, html {
```

```
  font-family: 'Poppins', sans-serif;
```

```
  background: linear-gradient(to bottom, #c7c6f6, #e9e9f9);
```

```
  overflow: auto;
```

```
  margin: 0;
```

```
  padding: 0;
```

```
  color: #333;
```

```
}
```

This sets the **default font** across the entire site to **Poppins** and applies a soft **purple gradient background**, as shown in the assignment's "After" screenshot. `overflow: auto` ensures scrollbars appear only when needed. The color sets text to dark gray for better contrast.



I customized the Bootstrap .bg-light navbar to have a soft purple gradient and a subtle shadow underneath to create depth. I used !important to override Bootstrap's default light gray background.

the **navbar brand** to use the **Pacifico font**, increasing its size and applying a soft black **text shadow**.

All navbar links are styled with black text and smooth hover transitions. On hover, they remain black but become underlined. I used !important to override Bootstrap's link coloring.

Headings use **Pacifico** for a consistent handwritten look. I applied a soft gray text shadow to give the text a slightly raised appearance — especially visible on the hero "About" header section.

Tables: tables follow a clean, readable style. I use alternating row colors using :nth-child(odd). Captions are bold and purple. Borders and padding create good spacing between cells.

Links: links a custom **purple** color to match the rest of the theme. When hovered, links become darker purple and are underlined.

Lists: List spacing is adjusted to make **bullets and numbers clearly readable**. The margin-left indents nested lists nicely. These styles make both ordered and unordered lists fit visually with the rest of the layout.

Media Elements (Images, Audio, Video): This styles images and media elements with:

- Center alignment
- Rounded corners
- Drop shadows

These changes make media **blend smoothly into the layout**.

Section: The hero section on the About page is styled with a **light purple background**, large padding, and a soft box shadow. This creates a **card-like layout** that separates content from the page background.

## legoSets.js File

```
modules > JS legoSets.js > LegoData > initialize > <function>
20  class LegoData {
70
71    // Returns sets that match a partial theme name (case-insensitive)
72    getSetsByTheme(theme) {
73      return new Promise((resolve, reject) => {
74        const themeLower = theme.toLowerCase(); // Convert search term to lowercase
75        const foundSets = this.sets.filter(set => set.theme.toLowerCase().includes(themeLower));
76        if (foundSets.length > 0) {
77          resolve(foundSets); // Return matching sets
78        } else {
79          reject(`Unable to find sets with theme: ${theme}`); // If none found
80        }
81      });
82    }
83    // Adds a new set if it doesn't already exist
84    addSet(newSet) {
85      return new Promise((resolve, reject) => {
86        // Check for duplicate set_num
87        const exists = this.sets.find(set => set.set_num === newSet.set_num);
88
89        if (exists) {
90          reject("Set already exists"); // Duplicate found
91        } else {
92          this.sets.push(newSet); // Add new set to the array
93          resolve(); // Resolve with no value
94        }
95      });
96    }
97  }
98  module.exports = LegoData;
99
100
```

## Step 3: Adding New Lego Sets

To extend the backend functionality, I updated my legoSets.js module (LegoData class) with a new method:

```
js
CopyEdit
addSet(newSet) {
  return new Promise((resolve, reject) => {
    const exists = this.sets.find(set => set.set_num === newSet.set_num);
    if (exists) {
      reject("Set already exists");
    } else {
      this.sets.push(newSet);
      resolve();
    }
  });
}
```

## Server.js File

```
<> about.html M    # theme.css U    JS server.js M X    JS legoSets.js M

JS server.js > ...
1  const LegoData = require("../modules/legoSets");
2  const legoData = new LegoData();
3
4
5  const path = require('path');
6  const express = require('express'); // "require" the Express module
7  const app = express(); // obtain the "app" object
8  const HTTP_PORT = process.env.PORT || 8081; // assign a port
9
10
11 // Serve static files
12 app.use(express.static(path.join(__dirname, 'public')));
13
14 // start the server on the port and output a confirmation to the console
15 app.listen(HTTP_PORT, () => console.log(`server listening on: ${HTTP_PORT}`));
16
17 app.get('/', (req, res) => {
18   res.sendFile(path.join(__dirname, '/views/home.html'));
19 });
20
21 app.get('/about', (req, res) => {
22   res.sendFile(path.join(__dirname, '/views/about.html'));
23 });
24
25
```

```
<> about.html M    # theme.css U    JS server.js M X    JS legoSets.js M

JS server.js > ...
45
46 app.get("/lego/add-test", async (req, res) => {
47   const testSet = {
48     set_num: "123",
49     name: "TestSet Name",
50     year: "2024",
51     theme_id: "366",
52     num_parts: "123",
53     img_url: "https://fakeimg.pl/375x375?text=[+Lego+]"
54   };
55
56   try {
57     await legoData.addSet(testSet);
58     res.redirect("/lego/sets");
59   } catch (err) {
60     res.status(422).send(err);
61   }
62 });
63
64 app.use((req, res) => {
65   res.status(404).sendFile(path.join(__dirname, "/views/404.html"));
66 });
67
68
69 legoData.initialize().then(() => {
70   app.listen(HTTP_PORT, () => {
71     console.log(`Server listening on: ${HTTP_PORT}`);
72   });
73 }).catch(err => {
74   console.log("Error initializing lego data:", err);
75 });
76
77
```

## Step 4: Updating Your Deployment

### Testing Route

To test this feature, I created the following route in `server.js`:

```
js
CopyEdit
app.get("/lego/add-test", async (req, res) => {
  const testSet = {
    set_num: "123",
    name: "Test Set",
    year: "2024",
    theme_id: "366",
    num_parts: "123",
    img_url: "https://fakeimg.pl/375x375?text=[+Lego+]"
  };

  try {
    await legoData.addSet(testSet);
    res.redirect("/lego/sets");
  } catch (err) {
    res.status(422).send(err);
  }
});
```

When I visit `/lego/add-test` the first time, the new set is added and I'm redirected to the sets list. On repeat visits, the app displays an error since the set already exists — confirming the method works correctly.