

A screenshot of a dark-themed code editor, likely Visual Studio Code, displaying a portion of an HTML file named `cb.html`. The file contains a table structure with 6 rows and 7 columns. The columns are defined by `<th>` elements numbered 1 through 6. The rows are defined by `<tr>` elements numbered 1 through 6. Each cell in the table is defined by a `<td>` element. The cells are styled with classes "light" and "dark". The "light" class is applied to cells at odd indices (1, 3, 5, 7, 9, 11) and the "dark" class is applied to cells at even indices (2, 4, 6, 8, 10). The code editor interface includes a top navigation bar with File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. On the left, there's a sidebar with icons for RUN AND DEBUG, RUN, and BROWSER BREAKPOINTS. The main area shows the file content with line numbers from 51 to 101.

```
C: > Users > bhava > Documents > Assignment 1 > cb.html > html
51 <th>6</th>
52 <td class="light"></td>
53 <td class="dark"></td>
54 <td class="light"></td>
55 <td class="dark"></td>
56 <td class="light"></td>
57 <td class="dark"></td>
58 <td class="light"></td>
59 <td class="dark"></td>
60 </tr>
61 <tr>
62 <th>5</th>
63 <td class="dark"></td>
64 <td class="light"></td>
65 <td class="dark"></td>
66 <td class="light"></td>
67 <td class="dark"></td>
68 <td class="light"></td>
69 <td class="dark"></td>
70 <td class="light"></td>
71 </tr>
72 <tr>
73 <th>4</th>
74 <td class="light"></td>
75 <td class="dark"></td>
76 <td class="light"></td>
77 <td class="dark"></td>
78 <td class="light"></td>
79 <td class="dark"></td>
80 <td class="light"></td>
81 <td class="dark"></td>
82 </tr>
83 <tr>
84 <th>3</th>
85 <td class="dark"></td>
86 <td class="light"></td>
87 <td class="dark"></td>
88 <td class="light"></td>
89 <td class="dark"></td>
90 <td class="light"></td>
91 <td class="dark"></td>
92 <td class="light"></td>
93 </tr>
94 <tr>
95 <th>2</th>
96 <td class="light"></td>
97 <td class="dark"></td>
98 <td class="light"></td>
99 <td class="dark"></td>
100 <td class="light"></td>
101 <td class="dark"></td>
```

A screenshot of a code editor interface, likely Visual Studio Code, displaying a chessboard HTML template. The editor has a dark theme with light-colored syntax highlighting. The top navigation bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The left sidebar features icons for RUN AND DEBUG, RUN, and other development tools. The main workspace shows the file path: C:\Users\bhuva\Documents\Assignment 1\cb.html. The code itself is a template for a chessboard table:

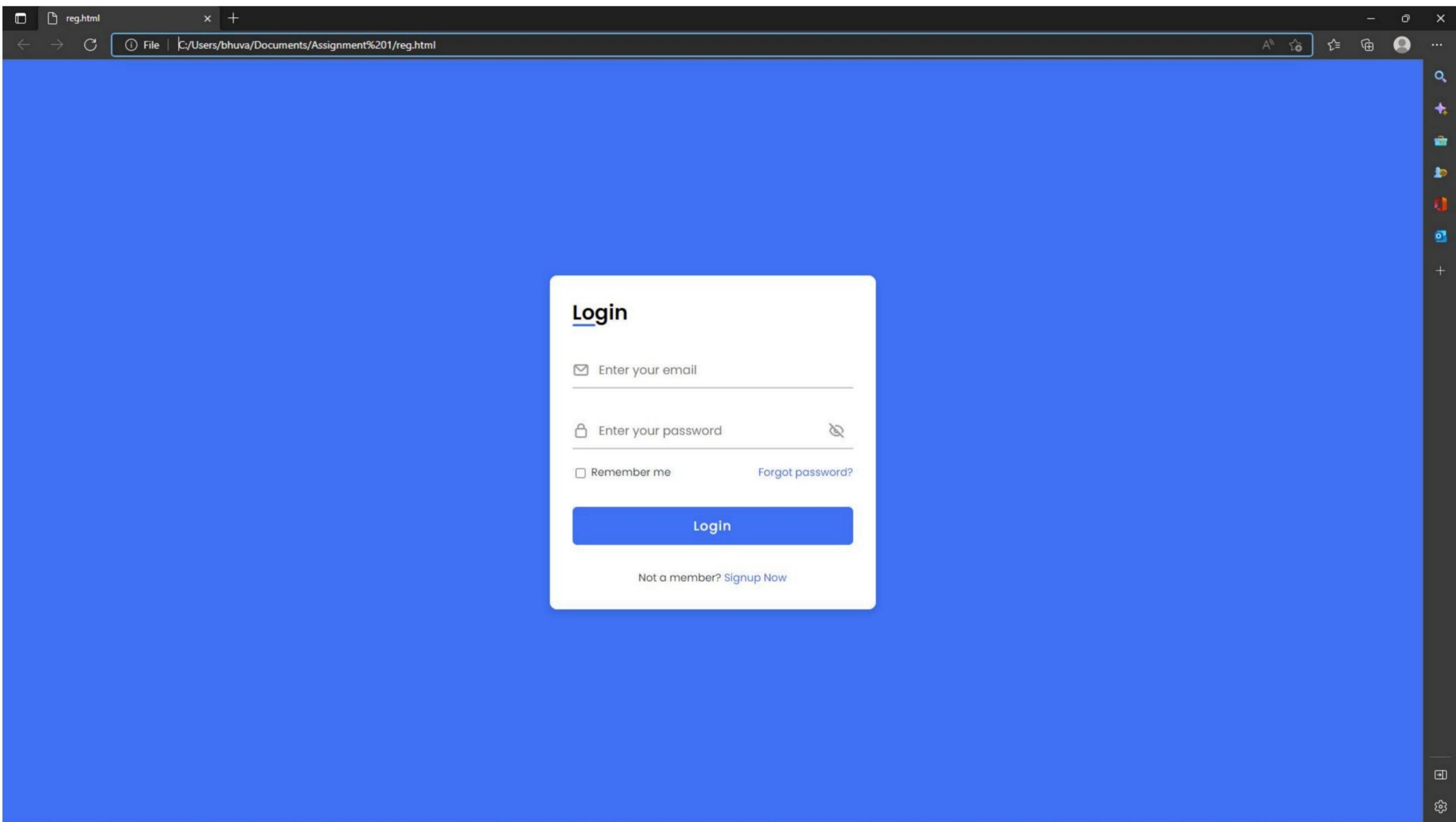
```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title></title>
5          <meta charset="UTF-8">
6          <style>
7              .chess-board { border-spacing: 0; border-collapse: collapse; }
8              .chess-board th { padding: .5em; }
9              .chess-board td { border: 1px solid; width: 2em; height: 2em; }
10             .chess-board .light { background: #eee; }
11             .chess-board .dark { background: #000; }
12         </style>
13     </head>
14     <body>
15         <table class="chess-board">
16             <tbody>
17                 <tr>
18                     <th></th>
19                     <th>a</th>
20                     <th>b</th>
21                     <th>c</th>
22                     <th>d</th>
23                     <th>e</th>
24                     <th>f</th>
25                     <th>g</th>
26                     <th>h</th>
27                 </tr>
28                 <tr>
29                     <th>8</th>
30                     <td class="light"></td>
31                     <td class="dark"></td>
32                     <td class="light"></td>
33                     <td class="dark"></td>
34                     <td class="light"></td>
35                     <td class="dark"></td>
36                     <td class="light"></td>
37                     <td class="dark"></td>
38                 </tr>
39                 <tr>
40                     <th>7</th>
41                     <td class="dark"></td>
42                     <td class="light"></td>
43                     <td class="dark"></td>
44                     <td class="light"></td>
45                     <td class="dark"></td>
46                     <td class="light"></td>
47                     <td class="dark"></td>
48                     <td class="light"></td>
49                 </tr>
50                 <tr>
51                     <th>6</th>
```

The bottom status bar indicates the code is 119 lines long, 8 columns wide, in UTF-8 encoding, and is an HTML file.

The screenshot shows a dark-themed code editor interface. The main area displays a CSS file named `reg.css` with the following content:

```
C: > Users > bhava > Documents > Assignment 1 > # reg.css > .container.forms
140     accent-color: #4070f4;
141 }
142
143 .form .text{
144     color: #333;
145     font-size: 14px;
146 }
147
148 .form a.text{
149     color: #4070f4;
150     text-decoration: none;
151 }
152 .form a:hover{
153     text-decoration: underline;
154 }
155
156 .form .button{
157     margin-top: 35px;
158 }
159
160 .form .button input{
161     border: none;
162     color: #fff;
163     font-size: 17px;
164     font-weight: 500;
165     letter-spacing: 1px;
166     border-radius: 6px;
167     background-color: #4070f4;
168     cursor: pointer;
169     transition: all 0.3s ease;
170 }
171
172 .button input:hover{
173     background-color: #265df2;
174 }
175
176 .form .login-signup{
177     margin-top: 30px;
178     text-align: center;
179 }
```

The sidebar on the left includes icons for file operations, search, and browser breakpoints. The status bar at the bottom shows file statistics: Line 32, Column 19, Spaces: 4, UTF-8, CRLF, CSS, and a refresh icon.



A screenshot of a code editor interface, likely Visual Studio Code, displaying an HTML file named `reg.html`. The code editor has a dark theme with syntax highlighting for HTML, CSS, and JavaScript.

The `reg.html` file contains the following HTML code:

```
C: > Users > bhava > Documents > Assignment 1 > reg.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7
8      <link rel="stylesheet" href="https://unicons.iconscout.com/release/v4.0.0/css/line.css">
9
10     <link rel="stylesheet" href="regc.css">
11
12 </head>
13 <body>
14
15     <div class="container">
16         <div class="forms">
17             <div class="form login">
18                 <span class="title">Login</span>
19
20                 <form action="#">
21                     <div class="input-field">
22                         <input type="text" placeholder="Enter your email" required>
23                         <i class="uil uil-envelope icon"></i>
24                     </div>
25                     <div class="input-field">
26                         <input type="password" class="password" placeholder="Enter your password" required>
27                         <i class="uil uil-lock icon"></i>
28                         <i class="uil uil-eye-slash showHidePw"></i>
29                     </div>
30
31                     <div class="checkbox-text">
32                         <div class="checkbox-content">
33                             <input type="checkbox" id="logCheck">
34                             <label for="logCheck" class="text">Remember me</label>
35                         </div>
36                     </div>
37             </div>
38         </div>
39     </div>
40
41 </body>
42
43 </html>
```

The code editor interface includes a navigation bar with tabs for `kav.java 2`, `reg.html`, `reg.js`, and `reg.css`. The left sidebar shows a tree view of the project structure under the `RUN` section, with a button for `Run and Debug`. The bottom status bar shows file statistics: `Ln 99, Col 8`, `Spaces: 4`, `UTF-8`, `CRLF`, `HTML`.

The screenshot shows a dark-themed code editor interface, likely Visual Studio Code, displaying an HTML file named 'reg.html'. The code is for a registration form, including fields for name, email, password, and password confirmation, along with login and forgot password links.

```
C: > Users > bhava > Documents > Assignment 1 > reg.html > html
      35     <label for="logCheck" class="text">Remember me</label>
      36     </div>
      37
      38     <a href="#" class="text">Forgot password?</a>
      39     </div>
      40
      41     <div class="input-field button">
      42         <input type="button" value="Login">
      43     </div>
      44 </form>
      45
      46     <div class="login-signup">
      47         <span class="text">Not a member?
      48             <a href="#" class="text signup-link">Signup Now</a>
      49         </span>
      50     </div>
      51 </div>
      52
      53     <!-- Registration Form -->
      54     <div class="form signup">
      55         <span class="title">Registration</span>
      56
      57         <form action="#">
      58             <div class="input-field">
      59                 <input type="text" placeholder="Enter your name" required>
      60                 <i class="uil uil-user"></i>
      61             </div>
      62             <div class="input-field">
      63                 <input type="text" placeholder="Enter your email" required>
      64                 <i class="uil uil-envelope icon"></i>
      65             </div>
      66             <div class="input-field">
      67                 <input type="password" class="password" placeholder="Create a password" required>
      68                 <i class="uil uil-lock icon"></i>
      69             </div>
      70             <div class="input-field">
      71                 <input type="password" class="password" placeholder="Confirm a password" required>
      72                 <i class="uil uil-lock icon"></i>
```

The interface includes a sidebar with icons for File, Edit, Selection, View, Go, Run, Terminal, Help, RUN AND DEBUG, and RUN. The RUN AND DEBUG section is expanded, showing 'Run and Debug' selected. A message indicates that to customize run and debug configurations, a launch.json file should be created. The bottom navigation bar includes PROBLEMS, OUTPUT, DEBUG CONSOLE (which is selected), and TERMINAL. The status bar at the bottom right shows 'Ln 99, Col 8 Spaces: 4 UTF-8 CRLF HTML'.

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The title bar shows tabs for Get Started, kav.java 2, reg.html, regj.js, and reg.css. The left sidebar has sections for RUN AND DEBUG and RUN, with a "Run and Debug" button highlighted. A message in the RUN section says: "To customize Run and Debug, open a folder and create a launch.json file." Another message says: "Show all automatic debug configurations." The main editor area displays the following HTML code:

```
C: > Users > bhava > Documents > Assignment 1 > reg.html > html
  ...
  70      </div>
  71      <div class="input-field">
  72          <input type="password" class="password" placeholder="Confirm a password" required>
  73          <i class="uil uil-lock icon"></i>
  74          <i class="uil uil-eye-slash showHidePw"></i>
  75      </div>
  76
  77      <div class="checkbox-text">
  78          <div class="checkbox-content">
  79              <input type="checkbox" id="termCon">
  80              <label for="termCon" class="text">I accepted all terms and conditions</label>
  81          </div>
  82      </div>
  83
  84      <div class="input-field button">
  85          <input type="button" value="Signup">
  86      </div>
  87
  88      <div class="login-signup">
  89          <span class="text">Already a member?
  90          <a href="#" class="text login-link">Login Now</a>
  91      </span>
  92  </div>
  93 </div>
  94 </div>
  95 </div>
  96
  97      <script src="regj.js"></script>
  98 </body>
  99 </html>
```

The bottom navigation bar includes PROBLEMS (2), OUTPUT, DEBUG CONSOLE (selected), TERMINAL, and a Filter input field. The bottom status bar shows BROWSER BREAKPOINTS (1), Line 99, Column 8, Spaces: 4, UTF-8, CRLF, HTML, and a refresh icon.

The screenshot shows the Visual Studio Code (VS Code) interface with a dark theme. The main area displays a JavaScript file named `reg.js`. The code is as follows:

```
C: > Users > bhava > Documents > Assignment 1 > JS reg.js > ...
1 const container = document.querySelector(".container"),
2 pwShowHide = document.querySelectorAll(".showHidePw"),
3 pwFields = document.querySelectorAll(".password"),
4 signUp = document.querySelector(".signup-link"),
5 login = document.querySelector(".login-link");
6
7 pwShowHide.forEach(eyeIcon =>{
8     eyeIcon.addEventListener("click", ()=>{
9         pwFields.forEach(pwField =>{
10             if(pwField.type ==="password"){
11                 pwField.type = "text";
12
13                 pwShowHide.forEach(icon =>{
14                     icon.classList.replace("uil-eye-slash", "uil-eye");
15                 })
16             }else{
17                 pwField.type = "password";
18
19                 pwShowHide.forEach(icon =>{
20                     icon.classList.replace("uil-eye", "uil-eye-slash");
21                 })
22             }
23         })
24     })
25
26 signUp.addEventListener("click", ( )=>{
27     container.classList.add("active");
28 });
29 login.addEventListener("click", ( )=>{
30     container.classList.remove("active");
31 });
32 );
```

The left sidebar contains the "RUN AND DEBUG" section, which includes "Run and Debug" (selected), "Show all automatic debug configurations", "JavaScript Debug Terminal" (selected), and "Debug URL". The bottom left also shows "BROWSER BREAKPOINTS" with 1 breakpoint.

The bottom status bar indicates the file is 32 lines long, 8 columns wide, in UTF-8 encoding, and is a JavaScript file. It also shows the current position at Ln 32, Col 8.

The screenshot shows a dark-themed code editor interface. At the top, there's a navigation bar with File, Edit, Selection, View, Go, Run, Terminal, and Help. To the right of the navigation bar is a search bar labeled "Search". The main area displays a CSS file named "reg.css" with the following content:

```
C: > Users > bhava > Documents > Assignment 1 > # reg.css > .container.forms
1 @import url('https://fonts.googleapis.com/css2?family=Poppins:wght@300;400;500;600;700&display=swap');
2
3 *{
4   margin: 0;
5   padding: 0;
6   box-sizing: border-box;
7   font-family: 'Poppins', sans-serif;
8 }
9
10 body{
11   height: 100vh;
12   display: flex;
13   align-items: center;
14   justify-content: center;
15   background-color: #4070f4;
16 }
17
18 .container{
19   position: relative;
20   max-width: 430px;
21   width: 100%;
22   background: #fff;
23   border-radius: 10px;
24   box-shadow: 0 5px 10px rgba(0, 0, 0, 0.1);
25   overflow: hidden;
26   margin: 0 20px;
27 }
28
29 .container .forms{
30   display: flex;
31   align-items: center;
32   height: 440px;
33   width: 200%;
34   transition: height 0.2s ease;
35 }
36
37
38 .container .form{
39   width: 50%;
40   padding: 30px;
41   background-color: #fff;
42   transition: margin-left 0.18s ease;
43 }
44
45 .container.active .login{
46   margin-left: -50%;
47   opacity: 0;
48   transition: margin-left 0.18s ease, opacity 0.15s ease;
49 }
50
51 .container .signup{
```

The sidebar on the left contains icons for RUN AND DEBUG, RUN, and other development tools. The RUN section is currently active, showing "Run and Debug" and instructions to customize the run configuration. The bottom status bar shows "Ln 32, Col 19" and "Spaces: 4" along with other file information.

A screenshot of the Visual Studio Code (VS Code) interface. The window title is "kav.java 2". The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The search bar at the top right contains the placeholder "Search". The left sidebar features icons for File Explorer, Search, Problems, and others. A "RUN AND DEBUG" section is expanded, showing a "Run and Debug" button which is highlighted in blue. Below it, there's a note: "To customize Run and Debug, open a folder and create a launch.json file." Another note says "Show all automatic debug configurations." The main editor area displays a CSS file named "reg.css" with the following content:

```
C: > Users > bhava > Documents > Assignment 1 > # reg.css > .container.forms
51 .container.signup{
52   opacity: 0;
53   transition: opacity 0.09s ease;
54 }
55 .container.active.signup{
56   opacity: 1;
57   transition: opacity 0.2s ease;
58 }
59
60 .container.active.forms{
61   height: 600px;
62 }
63 .container.form.title{
64   position: relative;
65   font-size: 27px;
66   font-weight: 600;
67 }
68
69 .form.title::before{
70   content: '';
71   position: absolute;
72   left: 0;
73   bottom: 0;
74   height: 3px;
75   width: 30px;
76   background-color: #4070f4;
77   border-radius: 25px;
78 }
79
80 .form.input-field{
81   position: relative;
82   height: 50px;
83   width: 100%;
84   margin-top: 30px;
85 }
86
87 .input-field input{
88   position: absolute;
89   height: 100%;
90   width: 100%;
91   padding: 0 35px;
92   border: none;
93   outline: none;
94   font-size: 16px;
95   border-bottom: 2px solid #ccc;
96   border-top: 2px solid transparent;
97   transition: all 0.2s ease;
98 }
99
100 .input-field input:focus, :valid{
101   border-bottom-color: #4070f4;
```

The status bar at the bottom shows "Ln 32, Col 19" and "Spaces: 4" along with other standard file information.

The screenshot shows a code editor window with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Search Bar:** Search field with a magnifying glass icon.
- Toolbar:** RUN AND DEBUG, Get Started, kav.java 2, reg.html, reg.js, reg.css, X.
- Sidebar:** Icons for file operations (New, Open, Save, Find, Replace, Copy, Paste, Delete, Undo, Redo), Run and Debug (selected), Show all automatic debug configurations, and Browser Breakpoints (with 1 breakpoint).
- Status Bar:** BROWSE BREAKPOINTS > 0 △ 2 ➡, Ln 32, Col 19, Spaces: 4, UTF-8, CRLF, CSS, ⚙️, 🗑️.

The main content area displays the `reg.css` file:

```
C: > Users > bhuvan > Documents > Assignment 1 > # reg.css > .container.forms
101     border-bottom-color: #4070f4;
102 }
103
104 .input-field i{
105     position: absolute;
106     top: 50%;
107     transform: translateY(-50%);
108     color: #999;
109     font-size: 23px;
110     transition: all 0.2s ease;
111 }
112
113 .input-field input:focus, :valid ~ i{
114     color: #4070f4;
115 }
116
117 .input-field i.icon{
118     left: 0;
119 }
120 .input-field i.showHidePw{
121     right: 0;
122     cursor: pointer;
123     padding: 10px;
124 }
125
126 .form .checkbox-text{
127     display: flex;
128     align-items: center;
129     justify-content: space-between;
130     margin-top: 20px;
131 }
132
133 .checkbox-text .checkbox-content{
134     display: flex;
135     align-items: center;
136 }
137
138 .checkbox-content input{
139     margin: 0 8px -2px 4px;
140     accent-color: #4070f4;
141 }
142
143 .form .text{
144     color: #333;
145     font-size: 14px;
146 }
147
148 .form a.text{
149     color: #4070f4;
150     text-decoration: none;
151 }
```

TIME TABLE

Day/Period	I 9:30-10:20	II 10:20-11:10	III 11:10-12:00	12:00-12:40	IV 12:40-1:30	V 1:30-2:20	VI 2:20-3:10	VII 3:10-4:00
Monday	Eng	Mat	Che			LAB		Phy
Tuesday			LAB		Eng	Che	Mat	SPORTS
Wednesday	Mat	phy	Eng		Che			LIBRARY
Thursday	Phy	Eng	Che			LAB		Mat
Friday		LAB			Mat	Che	Eng	Phy
Saturday	Eng	Che	Mat		SEMINAR			SPORTS

The screenshot shows a browser developer tools interface with the following panels:

- File Edit Selection View Go Run Terminal Help**
- Search**
- RUN AND DEBUG**
- VARIABLES**
- WATCH**
- CALL STACK**
 - Open tt.html: tt.html RUNNING
- LOADED SCRIPTS**
- BREAKPOINTS**
 - Caught Exceptions
 - Uncought Exceptions
- ROWSER BREAKPOINTS**

The main content area displays the following HTML code:

```
C: > Users > bhuva > Documents > Assignment 1 > tt.html > html
1  <!DOCTYPE html>
2  <html>
3
4  <body>
5      <h1>TIME TABLE</h1>
6      <table border="5" cellspacing="0" align="center">
7          <!--<caption>Timetable</caption>-->
8          <tr>
9              <td align="center" height="50"
10                 width="100"><br>
11                 <b>Day/Period</b></br>
12             </td>
13             <td align="center" height="50"
14                 width="100">
15                 <b>I<br>9:30-10:20</b>
16             </td>
17             <td align="center" height="50"
18                 width="100">
19                 <b>II<br>10:20-11:10</b>
20             </td>
21             <td align="center" height="50"
22                 width="100">
23                 <b>III<br>11:10-12:00</b>
24             </td>
25             <td align="center" height="50"
26                 width="100">
27                 <b>12:00-12:40</b>
28             </td>
29             <td align="center" height="50"
30                 width="100">
31                 <b>IV<br>12:40-1:30</b>
32             </td>
33             <td align="center" height="50"
34                 width="100">
35                 <b>V<br>1:30-2:20</b>
36             </td>
37             <td align="center" height="50"
38                 width="100">
39                 <b>VI<br>2:20-3:10</b>
40             </td>
41             <td align="center" height="50"
42                 width="100">
43                 <b>VII<br>3:10-4:00</b>
44             </td>
45         </tr>
46         <tr>
47             <td align="center" height="50">
48                 <b>Monday</b></td>
49             <td align="center" height="50">Eng</td>
50             <td align="center" height="50">Mat</td>
51             <td align="center" height="50">Che</td>
```

The screenshot shows a browser-based debugger interface with the following sections:

- RUN AND DEBUG**: Shows tabs for "Get Started", "kav.java 2", "reg.html", "reg.js", "# reg.js", and "tt.html".
- VARIABLES**: Displays the source code of "tt.html" with line numbers 51 to 101. The code defines a table with rows for days of the week, each containing cells for subjects like Che, LAB, Phy, etc., and a "LIBRARY" row.
- WATCH**: Shows variables being monitored.
- CALL STACK**: Shows the current call stack with "Open tt.html: tt.html" at the top and "RUNNING" status.
- LOADED SCRIPTS**: Shows loaded scripts.
- BREAKPOINTS**: Shows caught and uncaught exceptions.
- BROWSER BREAKPOINTS**: Shows browser breakpoints.

At the bottom, the status bar indicates "Ln 122, Col 8" and "Spaces: 4" and "UTF-8".

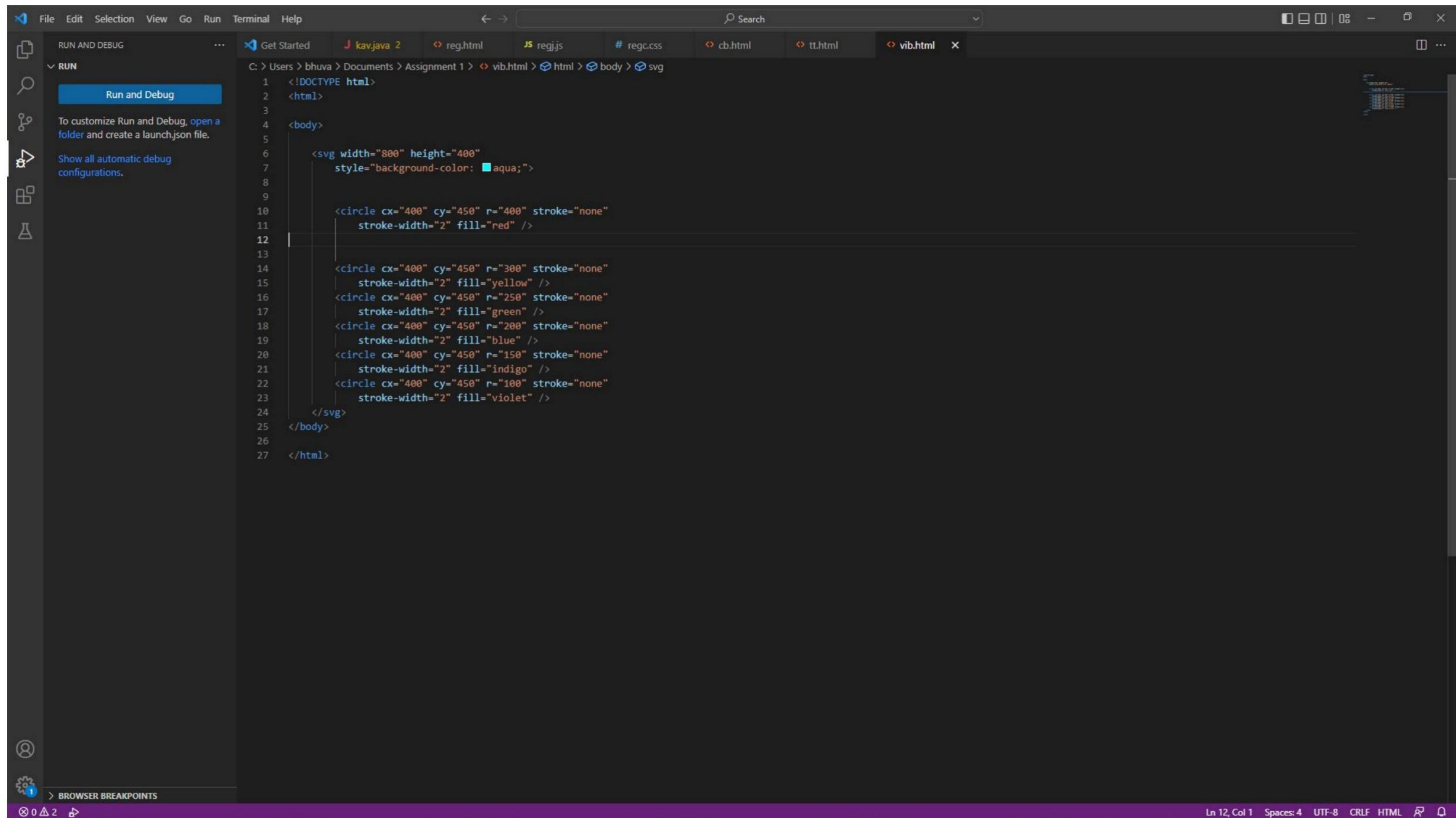
```
C: > Users > bhava > Documents > Assignment 1 > tt.html > html
51      <td align="center" height="50">Che</td>
52      <td rowspan="6" align="center" height="50">
53          <h2>L<br>U<br>N<br>C<br>H</h2>
54      </td>
55      <td colspan="3" align="center"
56          height="50">LAB</td>
57      <td align="center" height="50">Phy</td>
58  </tr>
59  <tr>
60      <td align="center" height="50">
61          <b>Tuesday</b>
62      </td>
63      <td colspan="3" align="center"
64          height="50">LAB
65      </td>
66      <td align="center" height="50">Eng</td>
67      <td align="center" height="50">Che</td>
68      <td align="center" height="50">Mat</td>
69      <td align="center" height="50">SPORTS</td>
70  </tr>
71  <tr>
72      <td align="center" height="50">
73          <b>Wednesday</b>
74      </td>
75      <td align="center" height="50">Mat</td>
76      <td align="center" height="50">phy</td>
77      <td align="center" height="50">Eng</td>
78      <td align="center" height="50">Che</td>
79      <td colspan="3" align="center"
80          height="50">LIBRARY
81      </td>
82  </tr>
83  <tr>
84      <td align="center" height="50">
85          <b>Thursday</b>
86      </td>
87      <td align="center" height="50">Phy</td>
88      <td align="center" height="50">Eng</td>
89      <td align="center" height="50">Che</td>
90      <td colspan="3" align="center"
91          height="50">LAB
92      </td>
93      <td align="center" height="50">Mat</td>
94  </tr>
95  <tr>
96      <td align="center" height="50">
97          <b>Friday</b>
98      </td>
99      <td colspan="3" align="center"
100         height="50">LAB
101     </td>
```

The screenshot shows a code editor interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Search Bar:** Search.
- Toolbars:** RUN AND DEBUG, Get Started, kav.java 2, reg.html, reg.js, # reg, html.
- Left Sidebar:** VARIABLES, WATCH, CALL STACK (with Open tt.html: tt.html RUNNING).
- Right Sidebar:** A large sidebar area showing a tree view of loaded scripts and breakpoints.
- Bottom Status Bar:** Line 122, Column 8, Spaces: 4, UTF-8, CRLF, HTML.

The main code editor area displays the following HTML code:

```
C:\> Users > bhava > Documents > Assignment 1 > tt.html > html
  <tr>
    <td align="center" height="50">LAB</td>
  </tr>
  <tr>
    <td align="center" height="50">Mat</td>
  </tr>
  <tr>
    <td align="center" height="50">
      <b>Friday</b>
    </td>
    <td colspan="3" align="center" height="50">LAB</td>
  </tr>
  <tr>
    <td align="center" height="50">Mat</td>
    <td align="center" height="50">Che</td>
    <td align="center" height="50">Eng</td>
    <td align="center" height="50">Phy</td>
  </tr>
  <tr>
    <td align="center" height="50">
      <b>Saturday</b>
    </td>
    <td align="center" height="50">Eng</td>
    <td align="center" height="50">Che</td>
    <td align="center" height="50">Mat</td>
  </tr>
  <tr>
    <td colspan="3" align="center" height="50">SEMINAR</td>
    <td align="center" height="50">SPORTS</td>
  </tr>
</table>
</body>
</html>
```



The screenshot shows the Microsoft Edge DevTools interface with the DOM tree open for the file `vib.html`. The tree is collapsed, showing the main structure of the document.

The DOM tree structure is as follows:

- `<!DOCTYPE html>`
- `<html>`
- `<body>`
- `<svg width="800" height="400" style="background-color: aqua;">`
- `<circle cx="400" cy="450" r="400" stroke="none" stroke-width="2" fill="red" />`
- `<circle cx="400" cy="450" r="300" stroke="none" stroke-width="2" fill="yellow" />`
- `<circle cx="400" cy="450" r="250" stroke="none" stroke-width="2" fill="green" />`
- `<circle cx="400" cy="450" r="200" stroke="none" stroke-width="2" fill="blue" />`
- `<circle cx="400" cy="450" r="150" stroke="none" stroke-width="2" fill="indigo" />`
- `<circle cx="400" cy="450" r="100" stroke="none" stroke-width="2" fill="violet" />`
- `</svg>`
- `</body>`
- `</html>`

The code editor on the left shows the corresponding HTML code with line numbers 1 through 27. The status bar at the bottom indicates the file is 12 lines long, 4 spaces wide, and uses CRLF line endings.

tic.html

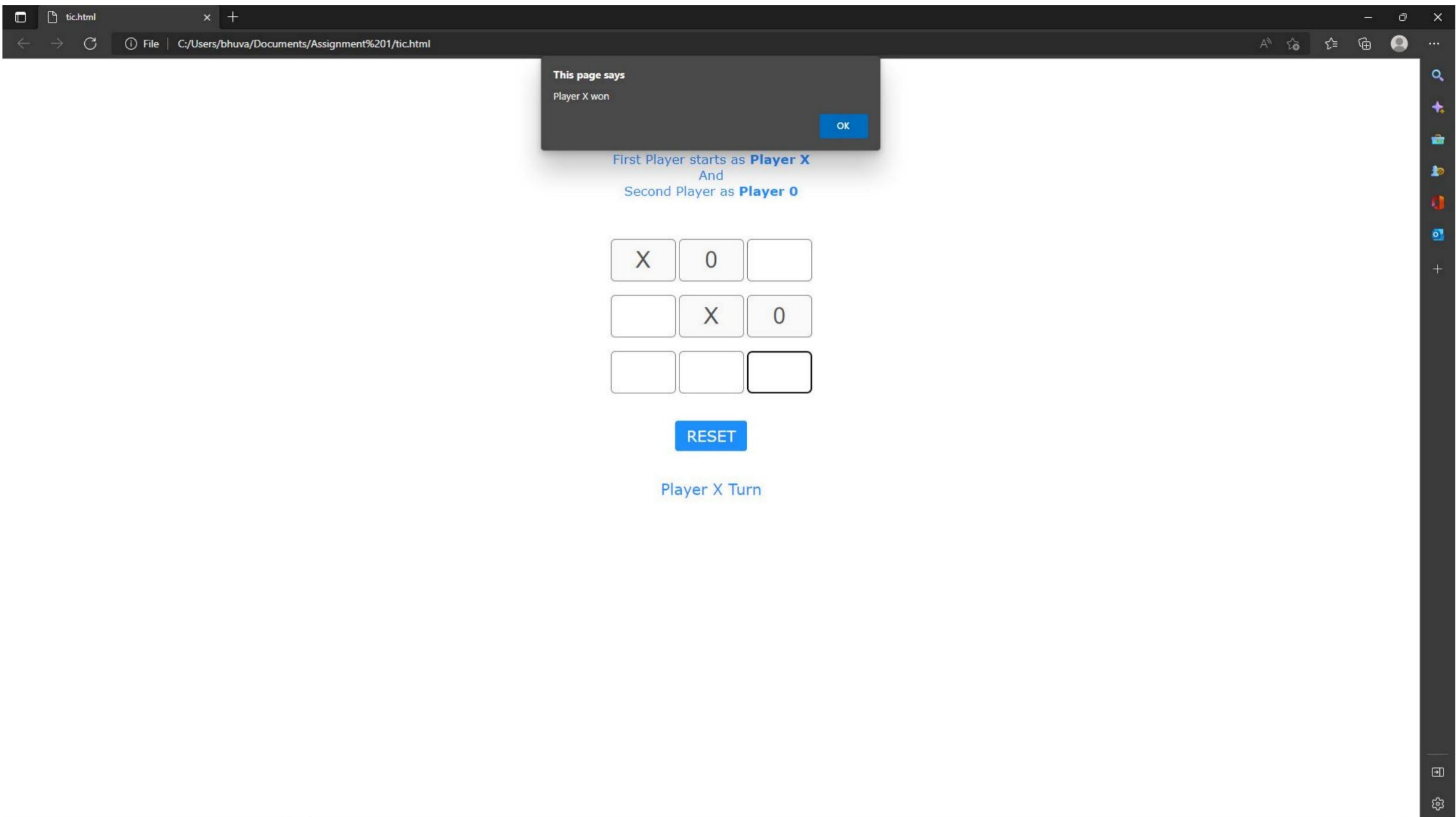
File | C:/Users/bhuva/Documents/Assignment%201/tic.html

TIC TAC TOE

Game starts by just Tap on box

First Player starts as **Player X**
And
Second Player as **Player O**

RESET



The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The title bar shows the current file is 'tic.html'. The left sidebar has sections for RUN AND DEBUG and RUN, with 'Run and Debug' being the active tab. It contains instructions to customize the run configuration by creating a launch.json file and shows a list of automatic debug configurations. The main editor area displays the HTML code for a Tic Tac Toe game. The code includes a head section with meta tags for viewport and CSS link, and a body section with a main div containing a h1 header and game instructions. Below the instructions is a 3x3 grid of input fields (b1 to b7) each with an onclick event that calls myfunc_1 through myfunc_8 functions. The bottom status bar shows browser breakpoints information and file details like Ln 5, Col 49, Spaces: 4, UTF-8, CRLF, HTML.

```
C: > Users > bhuvu > Documents > Assignment 1 > tic.html > head > meta
1  <!DOCTYPE html>
2
3  <head>
4      <meta name="viewport" content=
5          "width=device-width, initial-scale=1.0">
6
7      <!-- CSS file Included -->
8      <link rel="stylesheet"
9          type="text/css" href="tic.css">
10
11     <!-- JavaScript file included -->
12     <script src="tic.js"></script>
13 </head>
14
15 <body>
16     <div id="main">
17         <h1>TIC TAC TOE</h1>
18
19         <!-- Game Instructions -->
20         <p id="ins">Game starts by just Tap on
21             box<br><br>First Player starts as
22             <b>Player X</b><br>And<br>Second
23             Player as <b>Player 0</b>
24         </p>
25
26
27
28
29         <br><br>
30         <!-- 3*3 grid of Boxes -->
31         <input type="text" id="b1" onclick=
32             "myfunc_1(); myfunc();" readonly>
33
34         <input type="text" id="b2" onclick=
35             "myfunc_2(); myfunc();" readonly>
36
37         <input type="text" id="b3" onclick=
38             "myfunc_3(); myfunc();" readonly>
39         <br><br>
40
41         <input type="text" id="b4" onclick=
42             "myfunc_4(); myfunc();" readonly>
43
44         <input type="text" id="b5" onclick=
45             "myfunc_5(); myfunc();" readonly>
46
47         <input type="text" id="b6" onclick=
48             "myfunc_6(); myfunc();" readonly>
49         <br><br>
50
51         <input type="text" id="b7" onclick=
```

The screenshot shows the Microsoft Edge DevTools interface with the DOM tab selected. The left sidebar has 'RUN AND DEBUG' and 'RUN' sections. The main area displays the following HTML code:

```
C: > Users > bhuva > Documents > Assignment 1 > tic.html > head > meta

38     "myfunc_5(); myfunc();" readonly>
39     <br><br>
40
41     <input type="text" id="b4" onclick=
42         "myfunc_6(); myfunc();" readonly>
43
44     <input type="text" id="b5" onclick=
45         "myfunc_7(); myfunc();" readonly>
46
47     <input type="text" id="b6" onclick=
48         "myfunc_8(); myfunc();" readonly>
49     <br><br>
50
51     <input type="text" id="b7" onclick=
52         "myfunc_9(); myfunc();" readonly>
53
54     <input type="text" id="b8" onclick=
55         "myfunc_10();myfunc();" readonly>
56
57     <input type="text" id="b9" onclick=
58         "myfunc_11();myfunc();" readonly>
59
60     <!-- Grid end here -->
61     <br><br><br>
62     <!-- Button to reset game -->
63     <button id="but" onclick="myfunc_2()">
64         RESET
65     </button>
66
67     <br><br>
68     <!-- Space to show player turn -->
69     <p id="print"></p>
70
71
72
73
74     </div>
75 </body>
76
77 </html>
78
```

The status bar at the bottom shows: Ln 5, Col 49 Spaces: 4 UTF-8 CRLF HTML ⚡ Q.

The screenshot shows the Microsoft Edge DevTools interface with the CSS tab selected. The left sidebar has 'Run and Debug' selected under the 'RUN' section. The main pane displays the contents of 'tic.css'. The code defines styles for several elements:

```
C: > Users > bhava > Documents > Assignment 1 > # tic.css > ...
1  <style>
2      /* CSS Code */
3      /* Heading */
4      h1 {
5          color: orange;
6          font-size: 45px;
7      }
8
9      /* 3*3 Grid */
10     #b1, #b2, #b3, #b4, #b5,
11     #b6, #b7, #b8, #b9 {
12         width: 80px;
13         height: 52px;
14         margin: auto;
15         border: 1px solid gray;
16         border-radius: 6px;
17         font-size: 30px;
18         text-align: center;
19     }
20
21     /* Reset Button */
22     #but {
23         box-sizing: border-box;
24         width: 95px;
25         height: 40px;
26         border: 1px solid dodgerblue;
27         margin: auto;
28         border-radius: 4px;
29         font-family: Verdana,
30             Geneva, Tahoma, sans-serif;
31
32         background-color: dodgerblue;
33         color: white;
34         font-size: 20px;
35         cursor: pointer;
36     }
37
38     /* Player turn space */
39     #print {
40         font-family: Verdana,
41             Geneva, Tahoma, sans-serif;
42
43         color: dodgerblue;
44         font-size: 20px;
45     }
46
47     /* Main Container */
48     #main {
49         text-align: center;
50     }
51
```

The status bar at the bottom indicates: Ln 61, Col 1 Spaces:4 UTF-8 CRLF CSS.

The screenshot shows a dark-themed code editor interface, likely Visual Studio Code, displaying a CSS file named `tic.css`. The file path is shown as `C: > Users > bhava > Documents > Assignment 1 > # tic.css > ...`. The code in the editor is as follows:

```
27 margin: auto;
28 border-radius: 4px;
29 font-family: Verdana,
30     Geneva, Tahoma, sans-serif;
31
32 background-color: #dodgerblue;
33 color: white;
34 font-size: 20px;
35 cursor: pointer;
36 }
37
38 /* Player turn space */
39 #print {
40     font-family: Verdana,
41         Geneva, Tahoma, sans-serif;
42
43     color: #dodgerblue;
44     font-size: 20px;
45 }
46
47 /* Main Container */
48 #main {
49     text-align: center;
50 }
51
52 /* Game Instruction Text */
53 #ins {
54     font-family: Verdana,
55         Geneva, Tahoma, sans-serif;
56
57     color: #dodgerblue;
58     font-size: 17px;
59 }
60
61 </style>
```

The left sidebar contains icons for File, Edit, Selection, View, Go, Run, Terminal, Help, RUN AND DEBUG, and RUN. The RUN AND DEBUG section has a "Run and Debug" button highlighted in blue. Below it, there is a note: "To customize Run and Debug, open a folder and create a launch.json file." and a link to "Show all automatic debug configurations". The bottom left also shows a user icon and a "BROWSER BREAKPOINTS" section with a gear icon and the number 1.

The screenshot shows a code editor window with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The title bar shows the current file path: C:\> Users > bhava > Documents > Assignment 1 > JS ticjs > ... The status bar at the bottom indicates Line 376, Column 1, Spaces: 4, UTF-8, CRLF, and a JavaScript file type.

RUN AND DEBUG

RUN

- Run and Debug
- To customize Run and Debug, open a folder and create a launch.json file.
- Show all automatic debug configurations.
- JavaScript Debug Terminal
- You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line.
- Debug URL

Get Started

tic.js

```
C:\> Users > bhava > Documents > Assignment 1 > JS ticjs > ...
1 // Function called whenever user tab on any box
2 function myfunc() {
3
4     // Setting DOM to all boxes or input field
5     var b1, b2, b3, b4, b5, b6, b7, b8, b9;
6     b1 = document.getElementById("b1").value;
7     b2 = document.getElementById("b2").value;
8     b3 = document.getElementById("b3").value;
9     b4 = document.getElementById("b4").value;
10    b5 = document.getElementById("b5").value;
11    b6 = document.getElementById("b6").value;
12    b7 = document.getElementById("b7").value;
13    b8 = document.getElementById("b8").value;
14    b9 = document.getElementById("b9").value;
15
16    // Checking if Player X won or not and after
17    // that disabled all the other fields
18    if ((b1 == 'x' || b1 == 'X') && (b2 == 'x' ||
19        b2 == 'X') && (b3 == 'x' || b3 == 'X')) {
20        document.getElementById('print')
21            .innerHTML = "Player X won";
22        document.getElementById("b4").disabled = true;
23        document.getElementById("b5").disabled = true;
24        document.getElementById("b6").disabled = true;
25        document.getElementById("b7").disabled = true;
26        document.getElementById("b8").disabled = true;
27        document.getElementById("b9").disabled = true;
28        window.alert('Player X won');
29    }
30    else if ((b1 == 'x' || b1 == 'X') && (b4 == 'x' ||
31        b4 == 'X') && (b7 == 'x' || b7 == 'X')) {
32        document.getElementById('print')
33            .innerHTML = "Player X won";
34        document.getElementById("b2").disabled = true;
35        document.getElementById("b3").disabled = true;
36        document.getElementById("b5").disabled = true;
37        document.getElementById("b6").disabled = true;
38        document.getElementById("b8").disabled = true;
39        document.getElementById("b9").disabled = true;
40
41        window.alert('Player X won');
42    }
43    else if ((b7 == 'x' || b7 == 'X') && (b8 == 'x' ||
44        b8 == 'X') && (b9 == 'x' || b9 == 'X')) {
45        document.getElementById('print')
46            .innerHTML = "Player X won";
47        document.getElementById("b1").disabled = true;
48        document.getElementById("b2").disabled = true;
49        document.getElementById("b3").disabled = true;
50        document.getElementById("b4").disabled = true;
51        document.getElementById("b5").disabled = true;
```

BROWSER BREAKPOINTS

The screenshot shows the Microsoft Edge DevTools interface, specifically the JavaScript Debugger. The main area displays a block of JavaScript code for a Tic-Tac-Toe game. The code uses document.getElementById to select elements and manipulate their disabled status and innerHTML. The code editor has syntax highlighting and line numbers. A toolbar at the top includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. On the left, there's a sidebar with sections for RUN AND DEBUG, RUN, JavaScript Debug Terminal, and Debug URL. The bottom status bar shows file information like 'Ln 376, Col 1' and encoding 'UTF-8'.

```
51     document.getElementById("b5").disabled = true;
52     document.getElementById("b6").disabled = true;
53     window.alert('Player X won');
54 }
55 else if ((b3 == 'x' || b3 == 'X') && (b6 == 'x' ||
56 b6 == 'X') && (b9 == 'x' || b9 == 'X')) {
57     document.getElementById('print')
58     .innerHTML = "Player X won";
59     document.getElementById("b1").disabled = true;
60     document.getElementById("b2").disabled = true;
61     document.getElementById("b4").disabled = true;
62     document.getElementById("b5").disabled = true;
63     document.getElementById("b7").disabled = true;
64     document.getElementById("b8").disabled = true;
65     window.alert('Player X won');
66 }
67 else if ((b1 == 'x' || b1 == 'X') && (b5 == 'x' ||
68 b5 == 'X') && (b9 == 'x' || b9 == 'X')) {
69     document.getElementById('print')
70     .innerHTML = "Player X won";
71     document.getElementById("b2").disabled = true;
72     document.getElementById("b3").disabled = true;
73     document.getElementById("b4").disabled = true;
74     document.getElementById("b6").disabled = true;
75     document.getElementById("b7").disabled = true;
76     document.getElementById("b8").disabled = true;
77     window.alert('Player X won');
78 }
79 else if ((b3 == 'x' || b3 == 'X') && (b5 == 'x' ||
80 b5 == 'X') && (b7 == 'x' || b7 == 'X')) {
81     document.getElementById('print')
82     .innerHTML = "Player X won";
83     document.getElementById("b1").disabled = true;
84     document.getElementById("b2").disabled = true;
85     document.getElementById("b4").disabled = true;
86     document.getElementById("b6").disabled = true;
87     document.getElementById("b8").disabled = true;
88     document.getElementById("b9").disabled = true;
89     window.alert('Player X won');
90 }
91 else if ((b2 == 'x' || b2 == 'X') && (b5 == 'x' ||
92 b5 == 'X') && (b8 == 'x' || b8 == 'X')) {
93     document.getElementById('print')
94     .innerHTML = "Player X won";
95     document.getElementById("b1").disabled = true;
96     document.getElementById("b3").disabled = true;
97     document.getElementById("b4").disabled = true;
98     document.getElementById("b6").disabled = true;
99     document.getElementById("b7").disabled = true;
100    document.getElementById("b9").disabled = true;
101    window.alert('Player X won');
```

File Edit Selection View Go Run Terminal Help

RUN AND DEBUG ... Get Started kav.java 2 reg.html reg.js reg.css cb.html tt.html vib.html tic.html # tic.css JS tic.js X

RUN

Run and Debug To customize Run and Debug, open a folder and create a launch.json file.

Show all automatic debug configurations.

JavaScript Debug Terminal You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line.

Debug URL

```
C: > Users > bhava > Documents > Assignment 1 > JS ticjs > ...
101     window.alert('Player X won');
102 }
103 else if ((b4 == 'x' || b4 == 'X') && (b5 == 'x' ||
104         b5 == 'X') && (b6 == 'x' || b6 == 'X')) {
105     document.getElementById('print')
106     .innerHTML = "Player X won";
107     document.getElementById("b1").disabled = true;
108     document.getElementById("b2").disabled = true;
109     document.getElementById("b3").disabled = true;
110     document.getElementById("b7").disabled = true;
111     document.getElementById("b8").disabled = true;
112     document.getElementById("b9").disabled = true;
113     window.alert('Player X won');
114 }
115
116 // Checking of Player X finish
117 // Checking for Player 0 starts, Is player 0 won or
118 // not and after that disabled all the other fields
119 else if ((b1 == '0' || b1 == '0') && (b2 == '0' ||
120         b2 == '0') && (b3 == '0' || b3 == '0')) {
121     document.getElementById('print')
122     .innerHTML = "Player 0 won";
123     document.getElementById("b4").disabled = true;
124     document.getElementById("b5").disabled = true;
125     document.getElementById("b6").disabled = true;
126     document.getElementById("b7").disabled = true;
127     document.getElementById("b8").disabled = true;
128     document.getElementById("b9").disabled = true;
129     window.alert('Player 0 won');
130 }
131 else if ((b1 == '0' || b1 == '0') && (b4 == '0' ||
132         b4 == '0') && (b7 == '0' || b7 == '0')) {
133     document.getElementById('print')
134     .innerHTML = "Player 0 won";
135     document.getElementById("b2").disabled = true;
136     document.getElementById("b3").disabled = true;
137     document.getElementById("b5").disabled = true;
138     document.getElementById("b6").disabled = true;
139     document.getElementById("b8").disabled = true;
140     document.getElementById("b9").disabled = true;
141     window.alert('Player 0 won');
142 }
143 else if ((b7 == '0' || b7 == '0') && (b8 == '0' ||
144         b8 == '0') && (b9 == '0' || b9 == '0')) {
145     document.getElementById('print')
146     .innerHTML = "Player 0 won";
147     document.getElementById("b1").disabled = true;
148     document.getElementById("b2").disabled = true;
149     document.getElementById("b3").disabled = true;
150     document.getElementById("b4").disabled = true;
151     document.getElementById("b5").disabled = true;
```

BROWSER BREAKPOINTS

Ln 376, Col 1 Spaces:4 UTF-8 CRLF {} JavaScript

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The title bar shows the current file path: C:\> Users\> bhava\> Documents\> Assignment 1\> JS\tic.js. A search bar is also present.

The left sidebar contains several icons and sections:

- RUN AND DEBUG**
- RUN**: Contains "Run and Debug" (selected), "To customize Run and Debug, open a folder and create a launch.json file.", "Show all automatic debug configurations.", "JavaScript Debug Terminal" (selected), and "You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line."
- BROWSE BREAKPOINTS**

The main editor area displays a JavaScript file (tic.js) with line numbers from 151 to 201. The code implements logic for a Tic-Tac-Toe game, specifically checking for a win condition where three cells in a row (horizontal, vertical, or diagonal) have the same value ('X' or 'O'). If a win is detected, it disables all buttons and displays an alert message indicating the player has won.

```
151     document.getElementById("b5").disabled = true;
152     document.getElementById("b6").disabled = true;
153     window.alert('Player O won');
154 }
155 else if ((b3 == '0' || b3 == '0') && (b6 == '0' ||
156 b6 == '0') && (b9 == '0' || b9 == '0')) {
157     document.getElementById('print')
158     .innerHTML = "Player O won";
159     document.getElementById("b1").disabled = true;
160     document.getElementById("b2").disabled = true;
161     document.getElementById("b4").disabled = true;
162     document.getElementById("b5").disabled = true;
163     document.getElementById("b7").disabled = true;
164     document.getElementById("b8").disabled = true;
165     window.alert('Player O won');
166 }
167 else if ((b1 == '0' || b1 == '0') && (b5 == '0' ||
168 b5 == '0') && (b9 == '0' || b9 == '0')) {
169     document.getElementById('print')
170     .innerHTML = "Player O won";
171     document.getElementById("b2").disabled = true;
172     document.getElementById("b3").disabled = true;
173     document.getElementById("b4").disabled = true;
174     document.getElementById("b6").disabled = true;
175     document.getElementById("b7").disabled = true;
176     document.getElementById("b8").disabled = true;
177     window.alert('Player O won');
178 }
179 else if ((b3 == '0' || b3 == '0') && (b5 == '0' ||
180 b5 == '0') && (b7 == '0' || b7 == '0')) {
181     document.getElementById('print')
182     .innerHTML = "Player O won";
183     document.getElementById("b1").disabled = true;
184     document.getElementById("b2").disabled = true;
185     document.getElementById("b4").disabled = true;
186     document.getElementById("b6").disabled = true;
187     document.getElementById("b8").disabled = true;
188     document.getElementById("b9").disabled = true;
189     window.alert('Player O won');
190 }
191 else if ((b2 == '0' || b2 == '0') && (b5 == '0' ||
192 b5 == '0') && (b8 == '0' || b8 == '0')) {
193     document.getElementById('print')
194     .innerHTML = "Player O won";
195     document.getElementById("b1").disabled = true;
196     document.getElementById("b3").disabled = true;
197     document.getElementById("b4").disabled = true;
198     document.getElementById("b6").disabled = true;
199     document.getElementById("b7").disabled = true;
200     document.getElementById("b9").disabled = true;
201     window.alert('Player O won');
```

The status bar at the bottom indicates: Ln 376, Col 1, Spaces: 4, UTF-8, CRLF, {} JavaScript.

File Edit Selection View Go Run Terminal Help

RUN AND DEBUG ... Get Started kav.java 2 reg.html reg.js reg.css cb.html tt.html vib.html tic.html # tic.css JS tic.js X

RUN

Run and Debug To customize Run and Debug, open a folder and create a launch.json file.

Show all automatic debug configurations.

JavaScript Debug Terminal You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line.

Debug URL

```
C: > Users > bhuva > Documents > Assignment 1 > JS ticjs > ...
201     window.alert('Player 0 won');
202 }
203 else if ((b4 == '0' || b4 == '0') && (b5 == '0' ||
204         b5 == '0') && (b6 == '0' || b6 == '0')) {
205     document.getElementById('print')
206     .innerHTML = "Player 0 won";
207     document.getElementById("b1").disabled = true;
208     document.getElementById("b2").disabled = true;
209     document.getElementById("b3").disabled = true;
210     document.getElementById("b7").disabled = true;
211     document.getElementById("b8").disabled = true;
212     document.getElementById("b9").disabled = true;
213     window.alert('Player 0 won');
214 }
215
216 // Checking of Player 0 finish
217 // Here, Checking about Tie
218 else if ((b1 == 'X' || b1 == '0') && (b2 == 'X'
219         || b2 == '0') && (b3 == 'X' || b3 == '0') &&
220         (b4 == 'X' || b4 == '0') && (b5 == 'X' ||
221             b5 == '0') && (b6 == 'X' || b6 == '0') &&
222             (b7 == 'X' || b7 == '0') && (b8 == 'X' ||
223                 b8 == '0') && (b9 == 'X' || b9 == '0')) {
224     document.getElementById('print')
225     .innerHTML = "Match Tie";
226     window.alert('Match Tie');
227 }
228 else {
229
230     // Here, Printing Result
231     if (flag == 1) {
232         document.getElementById('print')
233         .innerHTML = "Player X Turn";
234     }
235     else {
236         document.getElementById('print')
237         .innerHTML = "Player 0 Turn";
238     }
239 }
240
241 // Function to reset game
242 function myfunc_2() {
243     location.reload();
244     document.getElementById('b1').value = '';
245     document.getElementById("b2").value = '';
246     document.getElementById("b3").value = '';
247     document.getElementById("b4").value = '';
248     document.getElementById("b5").value = '';
249     document.getElementById("b6").value = '';
250     document.getElementById("b7").value = '';
251 }
```

BROWSER BREAKPOINTS

0 △ 2 ↗

Ln 376, Col 1 Spaces: 4 UTF-8 CRLF {} JavaScript

File Edit Selection View Go Run Terminal Help

RUN AND DEBUG ... Get Started kav.java 2 reg.html reg.js reg.css cb.html tt.html vib.html tic.html # tic.css JS tic.js X

RUN

Run and Debug To customize Run and Debug, open a folder and create a launch.json file.

Show all automatic debug configurations.

JavaScript Debug Terminal You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line.

Debug URL

```
C: > Users > bhava > Documents > Assignment 1 > JS tic.js > ...
251     document.getElementById("b7").value = '';
252     document.getElementById("b8").value = '';
253     document.getElementById("b9").value = '';
254 }
255 // Here onwards, functions check turn of the player
256 // and put accordingly value X or 0
257 flag = 1;
258 function myfunc_3() {
259     if (flag == 1) {
260         document.getElementById("b1").value = "X";
261         document.getElementById("b1").disabled = true;
262         flag = 0;
263     } else {
264         document.getElementById("b1").value = "0";
265         document.getElementById("b1").disabled = true;
266         flag = 1;
267     }
268 }
269 function myfunc_4() {
270     if (flag == 1) {
271         document.getElementById("b2").value = "X";
272         document.getElementById("b2").disabled = true;
273         flag = 0;
274     } else {
275         document.getElementById("b2").value = "0";
276         document.getElementById("b2").disabled = true;
277         flag = 1;
278     }
279 }
280 function myfunc_5() {
281     if (flag == 1) {
282         document.getElementById("b3").value = "X";
283         document.getElementById("b3").disabled = true;
284         flag = 0;
285     } else {
286         document.getElementById("b3").value = "0";
287         document.getElementById("b3").disabled = true;
288         flag = 1;
289     }
290 }
291 function myfunc_6() {
292     if (flag == 1) {
293         document.getElementById("b4").value = "X";
294     }
295 }
```

BROWSER BREAKPOINTS

Ln 376, Col 1 Spaces:4 UTF-8 CRLF {} JavaScript

File Edit Selection View Go Run Terminal Help

RUN AND DEBUG ... Get Started kav.java 2 reg.html reg.js reg.css cb.html tt.html vib.html tic.html # tic.css JS tic.js X

RUN Run and Debug To customize Run and Debug, open a folder and create a launch.json file. Show all automatic debug configurations. JavaScript Debug Terminal You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line. Debug URL

```
C: > Users > bhava > Documents > Assignment 1 > JS tic.js > ...
301     document.getElementById("b4").value = "X";
302     document.getElementById("b4").disabled = true;
303     flag = 0;
304   }
305   else {
306     document.getElementById("b4").value = "0";
307     document.getElementById("b4").disabled = true;
308     flag = 1;
309   }
310 }
311 function myfunc_7() {
312   if (flag == 1) {
313     document.getElementById("b5").value = "X";
314     document.getElementById("b5").disabled = true;
315     flag = 0;
316   }
317   else {
318     document.getElementById("b5").value = "0";
319     document.getElementById("b5").disabled = true;
320     flag = 1;
321   }
322 }
323
324 function myfunc_8() {
325   if (flag == 1) {
326     document.getElementById("b6").value = "X";
327     document.getElementById("b6").disabled = true;
328     flag = 0;
329   }
330   else {
331     document.getElementById("b6").value = "0";
332     document.getElementById("b6").disabled = true;
333     flag = 1;
334   }
335 }
336
337 function myfunc_9() {
338   if (flag == 1) {
339     document.getElementById("b7").value = "X";
340     document.getElementById("b7").disabled = true;
341     flag = 0;
342   }
343   else {
344     document.getElementById("b7").value = "0";
345     document.getElementById("b7").disabled = true;
346     flag = 1;
347   }
348 }
349
350
351 function myfunc_10() {
```

BROWSER BREAKPOINTS 0 △ 2 ↻

Ln 376, Col 1 Spaces: 4 UTF-8 CRLF () JavaScript ⚙️

The screenshot shows the Microsoft Edge DevTools interface, specifically the JavaScript Debugger. The main area displays a block of JavaScript code with line numbers from 335 to 376. The code contains three functions: myfunc_9, myfunc_10, and myfunc_11. Each function toggles the value and disabled state of input elements based on a flag variable. The code uses document.getElementById to select elements by ID and manipulate their value and disabled properties.

```
335     }
336 }
337
338 function myfunc_9() {
339     if (flag == 1) {
340         document.getElementById("b7").value = "X";
341         document.getElementById("b7").disabled = true;
342         flag = 0;
343     }
344     else {
345         document.getElementById("b7").value = "0";
346         document.getElementById("b7").disabled = true;
347         flag = 1;
348     }
349 }
350
351 function myfunc_10() {
352     if (flag == 1) {
353         document.getElementById("b8").value = "X";
354         document.getElementById("b8").disabled = true;
355         flag = 0;
356     }
357     else {
358         document.getElementById("b8").value = "0";
359         document.getElementById("b8").disabled = true;
360         flag = 1;
361     }
362 }
363
364 function myfunc_11() {
365     if (flag == 1) {
366         document.getElementById("b9").value = "X";
367         document.getElementById("b9").disabled = true;
368         flag = 0;
369     }
370     else {
371         document.getElementById("b9").value = "0";
372         document.getElementById("b9").disabled = true;
373         flag = 1;
374     }
375 }
376 }
```

The left sidebar of the DevTools window contains several sections:

- RUN AND DEBUG**: A general overview of the run and debug configuration.
- RUN**: A section for running and debugging, with a note to customize it by creating a launch.json file.
- Run and Debug**: A button to start the debugger.
- To customize Run and Debug, open a folder and create a launch.json file.**
- Show all automatic debug configurations**: A link to view existing configurations.
- JavaScript Debug Terminal**: A section for using the terminal to debug Node.js processes.
- Debug URL**: A button to copy the URL for remote debugging.
- BROWSER BREAKPOINTS**: A section for managing breakpoints in the browser.

The bottom status bar shows the current file path (C:\Users\bhuva\Documents\Assignment 1\tic.js), line number (Ln 376), column number (Col 1), and encoding (UTF-8 CRLF). It also includes links for JavaScript and CSS files.

The screenshot shows the Microsoft Edge DevTools interface with the CSS tab selected. The left sidebar has 'Run and Debug' selected under the 'RUN' section. The main pane displays the contents of 'tic.css'. The code defines styles for several elements:

```
C: > Users > bhava > Documents > Assignment 1 > # tic.css > ...
1  <style>
2  /* CSS Code */
3  /* Heading */
4  h1 {
5      color: orange;
6      font-size: 45px;
7  }
8
9  /* 3*3 Grid */
10 #b1, #b2, #b3, #b4, #b5,
11 #b6, #b7, #b8, #b9 {
12     width: 80px;
13     height: 52px;
14     margin: auto;
15     border: 1px solid gray;
16     border-radius: 6px;
17     font-size: 30px;
18     text-align: center;
19 }
20
21 /* Reset Button */
22 #but {
23     box-sizing: border-box;
24     width: 95px;
25     height: 40px;
26     border: 1px solid dodgerblue;
27     margin: auto;
28     border-radius: 4px;
29     font-family: Verdana,
30                 Geneva, Tahoma, sans-serif;
31
32     background-color: dodgerblue;
33     color: white;
34     font-size: 20px;
35     cursor: pointer;
36 }
37
38 /* Player turn space */
39 #print {
40     font-family: Verdana,
41                 Geneva, Tahoma, sans-serif;
42
43     color: dodgerblue;
44     font-size: 20px;
45 }
46
47 /* Main Container */
48 #main {
49     text-align: center;
50 }
```

The status bar at the bottom shows: Ln 61, Col 1 Spaces:4 UTF-8 CRLF CSS. There are also icons for browser breakpoints and a refresh button.

The screenshot shows the Microsoft Edge DevTools interface with the CSS tab selected. The file 'tic.css' is open, displaying the following CSS code:

```
C: > Users > bhava > Documents > Assignment 1 > # tic.css > ...
19
20    /* Reset Button */
21    #but {
22        box-sizing: border-box;
23        width: 95px;
24        height: 40px;
25        border: 1px solid #dodgerblue;
26        margin: auto;
27        border-radius: 4px;
28        font-family: Verdana,
29                    Geneva, Tahoma, sans-serif;
30
31        background-color: #dodgerblue;
32        color: #white;
33        font-size: 20px;
34        cursor: pointer;
35    }
36
37
38    /* Player turn space */
39    #print {
40        font-family: Verdana,
41                    Geneva, Tahoma, sans-serif;
42
43        color: #dodgerblue;
44        font-size: 20px;
45    }
46
47    /* Main Container */
48    #main {
49        text-align: center;
50    }
51
52    /* Game Instruction Text */
53    #ins {
54        font-family: Verdana,
55                    Geneva, Tahoma, sans-serif;
56
57        color: #dodgerblue;
58        font-size: 17px;
59    }
60
61 </style>
```

The sidebar on the left contains icons for File, Edit, Selection, View, Go, Run, Terminal, Help, RUN AND DEBUG, and RUN. The RUN section includes a 'Run and Debug' button and instructions for creating a launch.json file. The bottom status bar shows 'Ln 61, Col 1' and other developer information.

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The title bar shows the current file is 'tic.html'. The left sidebar has sections for RUN AND DEBUG and RUN, with 'Run and Debug' being the active tab. It contains instructions to customize the run configuration by creating a launch.json file and shows a list of automatic debug configurations. The main editor area displays the HTML code for a Tic Tac Toe game. The code includes a head section with meta tags for viewport and CSS link, and a body section with a main div containing a h1 header and game instructions. Below the instructions is a 3x3 grid of input fields (b1 to b7) each with an onclick event that calls a function like myfunc_3() or myfunc_8(). The bottom status bar shows browser breakpoints, file statistics (Ln 5, Col 49, Spaces: 4), and encoding information (UTF-8, CRLF, HTML).

```
C: > Users > bhuvu > Documents > Assignment 1 > tic.html > head > meta
1  <!DOCTYPE html>
2
3  <head>
4      <meta name="viewport" content=
5          "width=device-width, initial-scale=1.0">
6
7      <!-- CSS file Included -->
8      <link rel="stylesheet"
9          type="text/css" href="tic.css">
10
11     <!-- JavaScript file included -->
12     <script src="tic.js"></script>
13 </head>
14
15 <body>
16     <div id="main">
17         <h1>TIC TAC TOE</h1>
18
19         <!-- Game Instructions -->
20         <p id="ins">Game starts by just Tap on
21             box<br><br>First Player starts as
22             <b>Player X</b><br>And<br>Second
23             Player as <b>Player 0</b>
24         </p>
25
26
27
28
29         <br><br>
30         <!-- 3*3 grid of Boxes -->
31         <input type="text" id="b1" onclick=
32             "myfunc_3(); myfunc();" readonly>
33
34         <input type="text" id="b2" onclick=
35             "myfunc_4(); myfunc();" readonly>
36
37         <input type="text" id="b3" onclick=
38             "myfunc_5(); myfunc();" readonly>
39         <br><br>
40
41         <input type="text" id="b4" onclick=
42             "myfunc_6(); myfunc();" readonly>
43
44         <input type="text" id="b5" onclick=
45             "myfunc_7(); myfunc();" readonly>
46
47         <input type="text" id="b6" onclick=
48             "myfunc_8(); myfunc();" readonly>
49         <br><br>
50
51         <input type="text" id="b7" onclick=
```

The screenshot shows the Microsoft Edge DevTools interface with the DOM tab selected. The left sidebar has 'RUN AND DEBUG' and 'RUN' sections. The main area displays the HTML code for a 3x3 grid of input fields and a reset button.

```
C: > Users > bhava > Documents > Assignment 1 > tic.html > head > meta
30     <!-- 3*3 grid of Boxes -->
31     <input type="text" id="b1" onclick=
32         "myfunc_3(); myfunc();" readonly>
33
34     <input type="text" id="b2" onclick=
35         "myfunc_4(); myfunc();" readonly>
36
37     <input type="text" id="b3" onclick=
38         "myfunc_5(); myfunc();" readonly>
39     <br><br>
40
41     <input type="text" id="b4" onclick=
42         "myfunc_6(); myfunc();" readonly>
43
44     <input type="text" id="b5" onclick=
45         "myfunc_7(); myfunc();" readonly>
46
47     <input type="text" id="b6" onclick=
48         "myfunc_8(); myfunc();" readonly>
49     <br><br>
50
51     <input type="text" id="b7" onclick=
52         "myfunc_9(); myfunc();" readonly>
53
54     <input type="text" id="b8" onclick=
55         "myfunc_10(); myfunc();" readonly>
56
57     <input type="text" id="b9" onclick=
58         "myfunc_11(); myfunc();" readonly>
59
60     <!-- Grid end here -->
61     <br><br><br>
62     <!-- Button to reset game -->
63     <button id="but" onclick="myfunc_2()">
64         RESET
65     </button>
66
67     <br><br>
68     <!-- Space to show player turn -->
69     <p id="print"></p>
70
71
72
73
74     </div>
75 </body>
76
77 </html>
78
```

BROWSE BREAKPOINTS

Ln 5, Col 49 Spaces: 4 UTF-8 CRLF HTML ⚙️ ⚙️

File Edit Selection View Go Run Terminal Help

RUN AND DEBUG

RUN

Run and Debug

To customize Run and Debug, open a folder and create a launch.json file.

Show all automatic debug configurations.

C: > Users > bhava > Documents > Assignment 1 > cb.html

```
98 <tr class="light"></tr>
99 <td class="dark"></td>
100 <td class="light"></td>
101 <td class="dark"></td>
102 <td class="light"></td>
103 <td class="dark"></td>
104 </tr>
105 <tr>
106 <th>1</th>
107 <td class="dark"></td>
108 <td class="light"></td>
109 <td class="dark"></td>
110 <td class="light"></td>
111 <td class="dark"></td>
112 <td class="light"></td>
113 <td class="dark"></td>
114 <td class="light"></td>
115 </tr>
116 </tbody>
117 </table>
118 </body>
119 </html>
```

BROWSER BREAKPOINTS

Ln 119, Col 8 Spaces: 4 UTF-8 CRLF HTML

