Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show ☑ source ☐ outline ☐ image report [Options...]

Check by [text input ∨]

[Enter the URL for an HTML, CSS, or SVG document]

[Check]

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

[Message Filtering]

1. [Warning] **Empty heading.**

   From line 69, column 17; to line 69, column 63

   `<h5 class="modal-title" id="errorMessageTitle"></h5>↵`

Document checking completed.

Source

```
 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">↵
 7.  ↵
 8.      <!-- CSRF Token -->↵
 9.      <meta name="csrf-token"
     content="hOivfx5fgpnG96DOSSKvfEDOOQUOypbCPgjJxkof">↵
10.  ↵
11.      <title>reFurniture</title>↵
12.  ↵
13.      <link rel="preconnect" href="https://fonts.googleapis.com">↵
14.      <link rel="preconnect" href="https://fonts.gstatic.com"
     crossorigin>↵
15.      <link↵
16.        href="https://fonts.googleapis.com
     /css2?family=DM+Serif+Display:ital@0;1&family=DM+Serif+Text:ital@0;
     1&display=swap"↵
17.        rel="stylesheet">↵
18.      <link href="https://fonts.googleapis.com/css2?family=Alata&
     display=swap" rel="stylesheet">↵
19.      <link href="http://localhost:8000/css/product.css"
     rel="stylesheet">↵
20.  ↵
21.  ↵
22.  ↵
23.      <!-- Styles -->↵
```

```
24.    <link href="http://localhost:8000/css/app.css" rel="stylesheet">↵
25.    <link href="http://localhost:8000/css/error.css"
   rel="stylesheet">↵
26.    <link href="http://localhost:8000/css/utility.css"
   rel="stylesheet">↵
27.    <link href="http://localhost:8000/css/notifications.css"
   rel="stylesheet">↵
28.    <link href="http://localhost:8000/css/tabs.css" rel="stylesheet">↵
29.    <link href="http://localhost:8000/css/login.css"
   rel="stylesheet">↵
30.    <link href="http://localhost:8000/css/checkout.css"
   rel="stylesheet">↵
31.    <link href="http://localhost:8000/css/number.selector.css"
   rel="stylesheet">↵
32.    <link href="http://localhost:8000/css/select.css"
   rel="stylesheet">↵
33.    <link href="http://localhost:8000/css/cart.css" rel="stylesheet">↵
34.    <link href="http://localhost:8000/css/address.css"
   rel="stylesheet">↵
35.    <link href="http://localhost:8000/css/product.css"
   rel="stylesheet">↵
36.    <link href="http://localhost:8000/css/auxiliar.css"
   rel="stylesheet">↵
37.    <link href="http://localhost:8000/css/home.css" rel="stylesheet">↵
38.    <script src="https://kit.fontawesome.com/1c937c97ed.js"
   crossorigin="anonymous"></script>↵
39.    <script>↵
40.       // Fix for Firefox autofocus CSS bug↵
41.       // See: http://stackoverflow.com/questions/18943276/html-
   5-autofocus-messes-up-css-loading/18945951#18945951↵
42.    </script>↵
43.    <script>↵
44.       window.__theme = 'bs5';↵
45.    </script>↵
46.    <script src=http://localhost:8000/js/app.js></script>↵
47.    <script src="https://js.pusher.com/7.0/pusher.min.js" defer>
   </script>↵
48.    <script src=http://localhost:8000/js/lib/uniqueCheckbox.js defer>
   </script>↵
49.    <script src=http://localhost:8000/js/lib/select.js defer>
   </script>↵
50.    <link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist
   /css/select2.min.css" rel="stylesheet" />↵
51.    <script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist
   /js/select2.min.js" defer></script>↵
52.    <link rel="stylesheet" href="http://localhost:8000/css/select2-
   bootstrap-5-theme.min.css">↵
53. ↵
54.    <script src=http://localhost:8000/js/ajax.js defer></script>↵
55.    <script src=http://localhost:8000/js/notifications.js defer>
   </script>↵
56.    <script src=http://localhost:8000/js/axios.js defer></script>↵
57.    <script src=http://localhost:8000/js/numberSelector.js defer>
   </script>↵
58.    <script src=http://localhost:8000/js/cart.js defer></script>↵
59.    <script src=http://localhost:8000/js/search.js defer></script>↵
60.    <script src=http://localhost:8000/js/userSearch.js defer>
   </script>↵
61.    <script src=http://localhost:8000/js/address.js defer></script>↵
62.    <script src=http://localhost:8000/js/addProduct.js defer>
   </script>↵
63. </head>↵
64.   <body>↵
65.    <div class="modal fade" id="errorMessage" tabindex="-1" aria-
   labelledby="errorMessageTitle" aria-hidden="true">↵
66.    <div class="modal-dialog modal-dialog-centered">↵
67.       <div class="modal-content" id="errorMessageContent">↵
68.          <div class="modal-header">↵
69.             <h5 class="modal-title" id="errorMessageTitle"></h5>↵
70.             <button type="button" class="btn-close" data-bs-
   dismiss="modal" aria-label="Close">↵
71.             </button>↵
72.          </div>↵
73.          <div class="modal-body" id="errorMessageBody">↵
74. ↵
```

```
 75.            </div>↵
 76.          </div>↵
 77.        </div>↵
 78. </div>↵
 79. ↵
 80. <script src="http://localhost:8000/js/errormodal.js" defer></script>↵
 81.      <main>↵
 82.        <header>↵
 83.          <!-- Navbar -->↵
 84. <div id="alert-placeholder">↵
 85. </div>↵
 86. ↵
 87. ↵
 88. ↵
 89. <link href="http://localhost:8000/css/alert.css" rel="stylesheet">↵
 90. <script src="http://localhost:8000/js/alert.js" defer></script>↵
 91. <nav>↵
 92.   <div class="p-3 text-center text-white">↵
 93.     <div class="container">↵
 94.       <div class="row d-flex align-items-center justify-content-
     between">↵
 95.         <div class="col-md-4 d-flex justify-content-center justify-
     content-md-start mb-3 mb-md-0">↵
 96.           <a href=http://localhost:8000/products class="ms-md-2">↵
 97.             <img src="/img/refurniture.svg" alt="" width="200"
     height="65" />↵
 98.           </a>↵
 99.         </div>↵
100. ↵
101.         <div class="col-md-4">↵
102.           <form id="search-products-form" class="d-flex input-group
     w-auto my-auto mb-3 mb-md-0">↵
103.             <input id="search-products-input" autocomplete="off"
     type="search" class="form-control rounded" placeholder="Search" />↵
104.             <span class="input-group-text border-0 d-none d-lg-
     flex"><i class="fas fa-search"></i></span>↵
105.           </form>↵
106.         </div>↵
107. ↵
108.         <div class="col-md-4 d-flex justify-content-center justify-
     content-md-end align-items-center">↵
109.           <div class="d-flex align-items-center justify-content-
     between" id="header-icons">↵
110.                             <div>↵
111.     <a id="cart-dropdown" class="text-reset hidden-arrow" data-bs-
     toggle="dropdown" aria-expanded="false" href="#">↵
112.         <i class="bi bi-cart-fill" style="color: #000000; font-
     size:1.5em;"></i>↵
113.     </a>↵
114.     <ul id="cart-dropdown-menu" class="dropdown-menu dropdown-menu-
     end" aria-labelledby="cart-dropdown">↵
115.         <li id="cart-dropdown-menu-content">↵
116.         </li>↵
117.     </ul>↵
118. </div>↵
119.                   <div>↵
120.                     <a href=http://localhost:8000/users/wishlist>↵
121.                       <i class="bi bi-bookmark-heart-fill" style="color:
     #000000; font-size:1.5em;"></i>↵
122.                     </a>↵
123.                   </div>↵
124.                   <!-- Notification -->↵
125.                   <div class="dropdown">↵
126.                     <a class="text-reset me-1 hidden-arrow" href="#"
     id="notification-dropdown"↵
127.                       data-bs-toggle="dropdown" aria-expanded="false">↵
128.                       <i class="bi bi-bell-fill" style="color:
     #000000; font-size:1.5em;"></i>↵
129.                     </a>↵
130.                     <ul id="notification-content" class="dropdown-menu
     dropdown-menu-end" aria-labelledby="notification-dropdown">↵
131.                     </ul>↵
132.                   </div>↵
133. ↵
134.                   <div class="notification-number">↵
```

```
135.                        <div class="new-notif"></div>↵
136.                    </div>↵
137.                            <!-- User -->↵
138.                    <div class="dropdown">↵
139.                    <a class="text-reset d-flex align-items-center hidden-
        arrow" href="#"↵
140.                        id="user-drodown" data-bs-toggle="dropdown" aria-
        expanded="false">↵
141.                            <img id="header-user-img"
        src=http://localhost:8000/images/user/person5.jpg class="rounded-
        circle" height="25" width="25" alt="" loading="lazy" />↵
142.                            <h5 id="header-user-name"
        class="px-3 mt-1" style="color: #000000;">↵
143.                            Terrence↵
144.                        </h5>↵
145.                    </a>↵
146.                    <ul class="dropdown-menu" aria-labelledby="user-
        drodown">↵
147.                            <li><a class="dropdown-item"
        href=http://localhost:8000/users/6>My profile</a></li>↵
148.                            <li><a class="dropdown-item"
        href=http://localhost:8000/users/6/private>Settings</a></li>↵
149.                        <li><form method="POST" class="col"
        action="http://localhost:8000/logout">↵
150.                            <input type="hidden" name="_token"
        value="hOivfx5fgpnG96DOSSKvfEDOOQUOypbCPgjJxkof">
        <button type="submit" class="dropdown-item form-control"> Logout
        </button>↵
151.                        </form></li>↵
152.                    </ul>↵
153.                    </div>↵
154.                        </div>↵
155.            </div>↵
156.            </div>↵
157.        </div>↵
158.    </div>↵
159. ↵
160. </nav>↵
161. ↵
162. ↵
163. <script>↵
164.     window.addEventListener("load", () => {↵
165.         const notification = document.querySelector("#notification-
        dropdown i");↵
166.         const notificationNumber =
        document.querySelector(".notification-number");↵
167.         const notificationContent =
        document.getElementById("notification-content");↵
168. ↵
169.         let skip = 0;↵
170. ↵
171.         const removeButton = () => {↵
172.             const prev = document.getElementById("next-page-btn");↵
173.             if (prev) {↵
174.                 prev.remove();↵
175.             }↵
176.         }↵
177. ↵
178.         const createButton = () => {↵
179.             const button = buildNextNotificationButton();↵
180.             button.addEventListener('click', (e) => {↵
181.                 get(`/api/users/6/notifications?skip=${skip}`)↵
182.                 .then(data => {↵
183.                     handleNextRequest(data.data);↵
184.                 });↵
185.                 e.stopPropagation();↵
186.             });↵
187.             notificationContent.appendChild(button);↵
188.         }↵
189. ↵
190.         const handleNextRequest = (data) => {↵
191.             const notifications = data.notifications;↵
192.             notifications.forEach(noti => {↵
193.                 const notif = parseNotification(noti);↵
194.                 notificationContent.appendChild(notif);↵
```

```
195.                notificationContent.appendChild(getDivider());
196.            });
197.
198.            if (data.new_nots > 0) {
199.                notificationNumber.style.visibility = "visible";
200.                console.log('new notif');
201.            }
202.
203.            skip += notifications.length;
204.
205.            removeButton();
206.
207.            if (skip < data.total) {
208.                createButton();
209.            }
210.        }
211.
212.        const handleNewRequest = (data) => {
213.            const notifications = data.notifications;
214.            notifications.forEach((noti, idx) => {
215.                const item = parseNotification(noti);
216.                if (item) {
217.                    notificationContent.appendChild(item);
218.                    notificationContent.appendChild(getDivider());
219.                }
220.            });
221.
222.            skip += notifications.length;
223.            removeButton();
224.
225.            if (skip < data.total) {
226.                createButton();
227.            }
228.        }
229.
230.
231.        get(`/api/users/6/notifications`)
232.            .then(data => {
233.                handleNewRequest(data.data)
234.
235.                if (data.data.new_nots > 0) {
236.                  notificationNumber.style.visibility = "visible";
237.                }
238.            });
239.
240.
241.        // Enable pusher logging - don't include this in production
242.        Pusher.logToConsole = true;
243.
244.        const pusher = new Pusher('4c7db76f6f7fd6381f0e', {
245.            cluster: 'eu'
246.        });
247.
248.        const handlePusherNotification = () => {
249.            skip++;
250.            notificationNumber.style.visibility = "visible";
251.            try{
252.              const audio = new Audio('/sounds/notif.wav');
253.              audio.play();
254.            } catch (e) {
255.              console.log(e);
256.            }
257.
258.        }
259.
260.        const channelProfileEdited = pusher.subscribe("profile-
     edited");
261.        channelProfileEdited.bind("profile-edited-6", function(data)
     {
262.            handlePusherNotification();
263.            const notif = buildEditedNotifcation(data);
264.            notificationContent.prepend(getDivider());
265.            notificationContent.prepend(notif);
266.        });
267.
```

```
268.            const channelOrderStatus = pusher.subscribe("order-status");
269.            channelOrderStatus.bind("order-status-6", function(data) {
270.                handlePusherNotification();
271.                const notif = buildOrderNotification(data);
272.                notificationContent.prepend(getDivider());
273.                notificationContent.prepend(notif);
274.            });
275.
276.            const cartItemUpdated = pusher.subscribe("cart-item");
277.            cartItemUpdated.bind("cart-item-6", function(data) {
278.                handlePusherNotification();
279.                const notif = buildCartNotification(data);
280.                notificationContent.prepend(getDivider());
281.                notificationContent.prepend(notif);
282.            });
283.
284.            const wishlistItemUpdated = pusher.subscribe("wishlist-
     item");
285.            wishlistItemUpdated.bind("wishlist-item-6", function(data) {
286.                handlePusherNotification();
287.                const notif = buildWishlistNotification(data);
288.                notificationContent.prepend(getDivider());
289.                notificationContent.prepend(notif);
290.            });
291.
292.            notification.addEventListener("click", () => {
293.                formDataPost(`/api/users/6/notifications/`)
294.                    .then(data => {
295.                        console.log(data);
296.                        notificationNumber.style.visibility = "hidden";
297.                    });
298.            });
299.
300.        });
301.
302. </script>
303.
304.        </header>
305.        <section id="content">
306.          <div class="container h-100 my-3">
307.      <div class="row">
308.          <div class="col-md-3 col-sm-12 container">
309.              <div class="d-flex justify-content-center align-items-
     center flex-column">
310.                  <div class="w-50">
311.                                                              <img
     id="user-img" src=http://localhost:8000/images/user/person5.jpg
     class="img-fluid" alt="" loading="lazy" />
312.                                      </div>
313.                  <div class="w-100">
314.                                          <h3 id="name-profile"
     class="text-center" style="overflow-wrap: break-word;">Terrence
     Arpino</h3>
315.                                      </div>
316.              </div>
317.              
318.              <div class="my-3 mx-2">
319.                  <div class="accordion accordion-flush"
     id="accordionFlushExample">
320.                      <div class="accordion-item">
321.                        <h2 class="accordion-header menu-collapse-btn"
     id="flush-headingOne">
322.                            <button class="accordion-button collapsed bg-
     primary text-white" type="button" data-bs-toggle="collapse" data-bs-
     target="#flush-collapseOne" aria-expanded="false" aria-
     controls="flush-collapseOne">
323.                                Menu
324.                            </button>
325.                        </h2>
326.                        <div id="flush-collapseOne" class="accordion-
     collapse collapse dont-collapse-sm" aria-labelledby="flush-headingOne"
     data-bs-parent="#accordionFlushExample">
327.                            <div class="accordion-body">
328.                                <ul class="nav nav-tabs v-nav-tabs flex-
     column mb-5">
```

```
329.        <li class="nav-item">↵
330.            <a class="nav-link active" href=http://localhost:8000
     /users/6> About Me </a>↵
331.        </li>↵
332.        <li class="nav-item">↵
333.            <a class="nav-link" href=http://localhost:8000/users
     /6/private>  Edit Personal Data </a>↵
334.        </li>↵
335.        <li class="nav-item">↵
336.            <a class="nav-link" href=http://localhost:8000/users
     /6/private/addresses>  Manage Addresses </a>↵
337.        </li>↵
338.        <li class="nav-item">↵
339.            <a class="nav-link" href=http://localhost:8000/users
     /orders>  Purchase History </a>↵
340.        </li>↵
341.        <li class="nav-item">↵
342.            <a class="nav-link" href=http://localhost:8000/users
     /wishlist> Wishlist </a>↵
343.        </li>↵
344.     </ul>↵
345. ↵
346.        ↵
347.     <form  method="POST" action="http://localhost:8000/logout">↵
348.         <input type="hidden" name="_token"
     value="hOivfx5fgpnG96DOSSKvfEDOOQUOypbCPgjJxkof">          <button
     type="submit" class="my-2 btn btn-primary form-control"> Logout
     </button>↵
349.     </form>↵
350.     <a id="show-delete-confirm" class="my-2 btn btn-danger w-100">
     Delete Account </a>↵
351. ↵
352. ↵
353. <div class="modal fade" id="confirm" tabindex="-1" aria-
     labelledby="confirmTitle" aria-hidden="true">↵
354.     <div class="modal-dialog modal-dialog-centered">↵
355.        <div class="modal-content" id="confirmContent">↵
356.            <div class="modal-header">↵
357.                <h5 class="modal-title" id="confirmTitle">Confirm
     delete account</h5>↵
358.                <button type="button" class="btn-close" data-bs-
     dismiss="modal" aria-label="Close">↵
359.                </button>↵
360.            </div>↵
361.            <div class="modal-body" id="confirmBody">↵
362.                This is an irreversible action, please take caution↵
363.                <form  method="POST" action="http://localhost:8000
     /users/6/delete">↵
364.                    <input type="hidden" name="_token"
     value="hOivfx5fgpnG96DOSSKvfEDOOQUOypbCPgjJxkof">
     <div class="form-group my-4">↵
365.                        <label for="cur-password"><b>Confirm your
     password before applying changes:</b></label>↵
366.                        <input autocomplete="on" required id="cur-
     password" class="form-control" type="password" name="cur-password">↵
367.                        <span class="error form-text text-danger"
     id="cur-password-error"></span>↵
368.                    </div>↵
369.                    <button type="submit" class="my-2 btn btn-danger
     w-100">↵
370.                        Delete Account↵
371.                    </button>↵
372.                </form>↵
373.            </div>↵
374.        </div>↵
375.     </div>↵
376. </div>↵
377. ↵
378. <script>↵
379.     window.addEventListener('load', () => {↵
380.        let deleteModal = new
     bootstrap.Modal(document.getElementById('confirm'));↵
381. ↵
382.        document.getElementById("show-delete-
     confirm").addEventListener("click", () => {↵
```

```
383.                    deleteModal.show();↵
384.            });↵
385.        })↵
386.    ↵
387. </script>↵
388.                            </div>↵
389.                        </div>↵
390.                    </div>↵
391.                </div>↵
392.            </div>↵
393.        </div>↵
394.        <div class="col-md-9 col-sm-12 container">↵
395.            <div>↵
396.    <h3>About Terrence Arpino</h3>↵
397.        <p style="white-space: break-spaces;">Vestibulum ac est lacinia
     nisi venenatis tristique. Fusce congue, diam id ornare imperdiet,
     sapien urna pretium nisl, ut volutpat sapien arcu sed augue. Aliquam
     erat volutpat.</p>↵
398. </div>↵
399. <div>↵
400.    <h3>Placeholder for Reviews</h3>↵
401. </div>↵
402.            </div>↵
403.        </div>↵
404. </div>      </section>↵
405.    </main>↵
406. </body>↵
407.    ↵
408. </html>
```

Used the HTML parser.

Total execution time 28 milliseconds.

About this checker • Report an issue • Version: 22.1.9