* <link rel="shortcut icon" href="./assets/nficon2023.ico" type="image/x-icon">
* In head
  + <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">
* In body
  + <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz" crossorigin="anonymous"></script>

The order of including CSS files in your HTML matters because styles are applied in a cascading manner, with later styles overriding earlier ones when conflicts arise.

**General Rule:**

* **Include your custom CSS file after Bootstrap's CSS file.**

**Why?**

1. **Bootstrap First**:
   * Bootstrap provides a foundation of default styles.
   * These styles can be overridden by your custom styles if your CSS is included **after** Bootstrap.
2. **Your CSS Last**:
   * Custom CSS is often used to tweak or override the default styles provided by frameworks like Bootstrap.
   * Placing it after Bootstrap ensures your custom rules have higher precedence.

**Correct Example:**

html

Copy code

<!-- Include Bootstrap CSS -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">

<!-- Include Custom CSS -->

<link rel="stylesheet" href="custom-styles.css">

**Incorrect Example:**

html

Copy code

<!-- Include Custom CSS -->

<link rel="stylesheet" href="custom-styles.css">

<!-- Include Bootstrap CSS -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">

**Exception:**

If your custom CSS is designed to **extend** Bootstrap without overriding its styles (e.g., by adding new classes), you can include it **before** Bootstrap. However, this is less common.

**Debugging Tip:**

* Use browser developer tools to inspect elements and check the order of styles applied. The last rule in the cascade is the one that takes effect.
* <body style="background-color: #1e1e1e;">

The following code is an HTML snippet that creates a collapsible toggle button commonly used in navigation bars (navbars) for responsive web designs. This functionality is often associated with frameworks like Bootstrap, where the data-bs-\* attributes define behavior and interactivity.

**Explanation of Each Part:**

**<button class="navbar-toggler" ...>**

* **<button>**: This is an HTML button element that serves as the toggle control for expanding or collapsing the navigation menu.
* **class="navbar-toggler"**: This class is part of Bootstrap's predefined styles for navbar togglers. It provides default styling for the button.

**type="button"**

* Indicates the button's type. Here, it's specified as "button", meaning it will not submit a form when clicked (unlike "submit").

**data-bs-toggle="collapse"**

* **data-bs-toggle**: A Bootstrap attribute that specifies the behavior of the button.
* **"collapse"**: Indicates that this button will toggle the visibility of a collapsible element.

**data-bs-target="#navbarNav"**

* **data-bs-target**: Specifies the target element that the button controls.
* **"#navbarNav"**: Refers to the element with the ID navbarNav. When the button is clicked, this element will be expanded or collapsed.

**aria-controls="navbarNav"**

* **aria-controls**: An accessibility attribute that links the button to the element it controls (in this case, navbarNav). This helps assistive technologies understand the relationship between the two elements.

**aria-expanded="false"**

* **aria-expanded**: Another accessibility attribute. It indicates whether the collapsible content is currently expanded (true) or collapsed (false). Initially, it’s set to false.

**aria-label="Toggle navigation"**

* **aria-label**: Provides an accessible name for the button, explaining its function. Screen readers use this to describe the button to visually impaired users.

**<span class="navbar-toggler-icon"></span>**

* **<span>**: A span element used to visually represent the button.
* **class="navbar-toggler-icon"**: A Bootstrap class that applies an icon (usually three horizontal lines, also called a "hamburger icon") to indicate the button's purpose.

**How It Works**

1. The button is part of a responsive navigation bar.
2. When clicked, it triggers the data-bs-toggle functionality (collapse) to show or hide the menu specified by data-bs-target (#navbarNav).
3. Accessibility attributes like aria-controls, aria-expanded, and aria-label ensure the toggle is usable by screen readers, improving the experience for users with disabilities.

**Example Usage with Associated HTML**

Here’s a fuller example to show how this button interacts with the collapsible navigation menu:

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<div class="container-fluid">

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav"

aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNav">

<ul class="navbar-nav">

<li class="nav-item">

<a class="nav-link active" aria-current="page" href="#">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">Features</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">Pricing</a>

</li>

</ul>

</div>

</div>

</nav>

In this example:

* The button collapses or expands the menu inside the <div id="navbarNav">.
* The responsive behavior makes the button visible on smaller screens, ensuring usability across devices.  
    
  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  + The CSS property width: 70vh; means that the width of an element is set to **70% of the viewport's height**.
* <form class="main-search" role="search">
* <input class="main-search-sub" type="search" placeholder="Email address"
* aria-label="search"/>
* <button class="main-btn" type="submit">Get started ></button>
* </form>

This code is for a simple search form styled using **Bootstrap** classes along with custom classes for styling. Let's break down each part:

**1. <form class="main-search" role="search">**

* **<form>**: This is the form element that wraps around the search input field and the submit button.
* **class="main-search"**: This is a custom class, possibly defined in the external CSS, to style the form. It's not a Bootstrap class, so it will have specific styles applied according to the designer's preferences.
* **role="search"**: This attribute is used for accessibility purposes. It specifies that the form is meant for searching, which helps screen readers understand the form's purpose.

**2. <input class="main-search-sub" type="search" placeholder="Email address" aria-label="search"/>**

* **<input>**: This is the input field where the user types the search query.
* **class="main-search-sub"**: This is a custom class (again, not Bootstrap). It's likely defined in the website's stylesheet for styling purposes, such as the width, padding, border, etc.
* **type="search"**: This input type specifies that the field is intended for search queries. It often provides browser-specific enhancements like a "clear" button in some browsers.
* **placeholder="Email address"**: The placeholder text that appears inside the input field before the user types anything. In this case, it prompts the user to enter an email address.
* **aria-label="search"**: This attribute provides a label for the input element for users relying on assistive technologies like screen readers. It ensures that the purpose of the input field is clearly communicated to these users.

**3. <button class="main-btn" type="submit">Get started ></button>**

* **<button>**: This is the button that will submit the form when clicked.
* **class="main-btn"**: This is a custom class for styling the button. It might control the button's color, font, borders, or other styles, but it's not a default Bootstrap class.
* **type="submit"**: This defines the button's behavior. When clicked, it will submit the form.
* **Get started >**: The button's text, which in this case invites the user to start something, possibly initiating a sign-up or registration process. The ">" symbol gives it a dynamic, forward-moving feel.

**Summary:**

* The form is designed to be accessible and user-friendly.
* It uses custom classes (main-search, main-search-sub, and main-btn) to apply styling, but also utilizes the type="search" input field, which is common for search forms.
* The button submits the form with the text "Get started >", implying some kind of action or engagement.

You would need additional CSS (for the custom classes) or Bootstrap utility classes to style and align elements properly, as well as any backend logic to process the form submission.

* .main{
* width: 100%;
* height:100vh;
* background-image:url("assets/Background.jpeg");
* }