## Context: You are an accessibility expert reviewing HTML for accessibility issues. Analyze the following code in the context of WCAG (Web Content Accessibility Guidelines) 2.2, listing any issues that may violate these guidelines. You will only be analyzing the HTML, ignoring any included or missing styles and scripts.

Code:

```<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<title>Bookstore</title>

<link rel="stylesheet" href="styles.css" />

</head>

<body>

<!-- Navigation Bar -->

<nav>

<ul>

<li><a href="#home">Home</a></li>

<li><a href="#books">Books</a></li>

<li><a href="#about">About</a></li>

<li><a href="#contact">Contact</a></li>

</ul>

</nav>

<!-- Header with Background Image -->

<header style="background-image: url('header-image.jpg')">

<h1>Welcome to our Bookstore</h1>

<p>Explore a vast collection of books.</p>

</header>

<!-- Search Bar and Filters -->

<section id="filters">

<label for="search">Search:</label>

<input

type="text"

id="search"

placeholder="Search..."

aria-label="Search"

/>

<select id="filter">

<option value="genre">Genre</option>

<option value="author">Author</option>

<!-- Add more filter options -->

</select>

<button>Apply Filters</button>

</section>

<!-- Book Display Section -->

<section id="books">

<!-- Display books dynamically from database -->

<!-- Example book -->

<div class="book">

<img src="book-cover.jpg" />

<h2>Book Title</h2>

<p>Author: Author Name</p>

<font size="5">Description of the book...</font>

<button>Read More</button>

</div>

<!-- Add more books dynamically -->

</section>

<!-- Video about the Bookstore -->

<section id="video">

<h2><b>Discover Our Bookstore</b></h2>

<video autoplay>

<source src="bookstore-video.mp4" type="video/mp4" />

Your browser does not support the video tag.

</video>

</section>

<!-- Contact Form -->

<section id="contact">

<h2>Contact Us</h2>

<form action="submit.php" method="POST">

<label for="name">Your Name:</label>

<input type="text" id="name" name="name" placeholder="Your Name" />

<label for="email">Your Email:</label>

<input type="email" id="email" name="email" placeholder="Your Email" />

<label for="message">Your Message:</label>

<textarea

id="message"

name="message"

placeholder="Your Message"

></textarea>

<button type="submit">Send</button>

</form>

</section>

<!-- Footer -->

<footer>

<p>&copy; 2023 Bookstore. All rights reserved.</p>

</footer>

<!-- JavaScript for interactivity -->

<script src="scripts.js"></script>

</body>

</html>

```

Response format:

Snippet with error: …

Location of error: …

WCAG Guideline violated: …

Detection Algorithm: …

Explanation of Error: …

### Snippet with error:

```html

<font size="5">Description of the book...</font>

```

### Location of error:

In the "Book Display Section" within the `<div class="book">` element.

### WCAG Guideline violated:

\*\*2.4.10\*\*: Focus Visible (A).

### Detection Algorithm:

Automatic detection through semantic analysis and manual review.

### Explanation of Error:

The `<font>` element is deprecated and should not be used in modern web development. It was used for styling text but has been replaced by CSS. Using deprecated elements like `<font>` can lead to accessibility issues because assistive technologies might not recognize or properly interpret its styling properties. This can affect users who rely on screen readers or other assistive devices to navigate and understand content.

To fix this issue, replace the `<font>` element with a `<span>` or another suitable container element and apply the desired font size using CSS. Here's how you can correct the code:

```html

<span style="font-size: 20px;">Description of the book...</span>

```

Or, if you prefer to separate concerns and keep styling in a stylesheet, add a class to the `<span>` and define the font size in your CSS file (`styles.css`):

```html

<span class="book-description">Description of the book...</span>

```

And in `styles.css`:

```css

.book-description {

font-size: 20px;

}

```

This approach ensures that the website remains accessible and adheres to current web standards.

Citations:

## Are there any more issues? List all that you find.

Based on the provided code and the WCAG 2.2 AA checklist, here are additional issues identified:

### Issue 1: Focus Indicator for Interactive Elements

\*\*WCAG Guideline Violated:\*\* 2.4.11: Make sure focus is never completely hidden (e.g., by a sticky footer).

\*\*Detection Algorithm:\*\* Manual review and automatic detection through accessibility tools.

\*\*Explanation of Error:\*\* The provided code snippet does not include interactive elements that receive focus, such as links (`<a>` tags) or form controls (`<input>`, `<button>`). However, if interactive elements were present, it's crucial to ensure that when they receive focus, either through keyboard navigation or programmatically, they remain visible and not obscured by other content. This is essential for users relying on keyboard navigation or screen readers.

### Issue 2: Lack of Alternative Text for Non-text Content

\*\*WCAG Guideline Violated:\*\* 1.1.1: Provide alternative text for non-text content that is presented to the user.

\*\*Detection Algorithm:\*\* Manual review and automatic detection through accessibility tools.

\*\*Explanation of Error:\*\* The code includes an image (`<img src="book-cover.jpg" />`) within the "Book Display Section" but lacks an `alt` attribute providing a textual description of the image. This can hinder users who rely on screen readers or have low vision and cannot see images.

To resolve this issue, add an `alt` attribute to the `<img>` tag with a descriptive text:

```html

<img src="book-cover.jpg" alt="Cover of the book titled 'Book Title' by Author Name" />

```

### Issue 3: Missing ARIA Roles for Semantic Structure

\*\*WCAG Guideline Violated:\*\* 1.3.1: Information, structure, and behavior conveyed through presentation must be available in text.

\*\*Detection Algorithm:\*\* Manual review and automatic detection through accessibility tools.

\*\*Explanation of Error:\*\* While the code uses semantic HTML elements like `<nav>`, `<header>`, `<section>`, and `<footer>`, it does not explicitly include ARIA roles where they could enhance the semantics further. For instance, the `<nav>` element is correctly used, but adding `role="navigation"` could clarify its purpose to assistive technologies.

To address this, consider adding ARIA roles where applicable:

```html

<nav role="navigation">

...

</nav>

```

### Issue 4: No Keyboard Navigable Main Content

\*\*WCAG Guideline Violated:\*\* 2.4.3: For each section of content that is marked up with a heading, ensure that the section contains at least one H1 heading.

\*\*Detection Algorithm:\*\* Manual review and automatic detection through accessibility tools.

\*\*Explanation of Error:\*\* The code does not explicitly show the use of headings (`<h1>`, `<h2>`, etc.) to structure the main content. While headings are present, their usage and hierarchy should be consistent and logical to aid in navigation, especially for screen reader users.

Ensure that the main content starts with an `<h1>` heading and subsequent sections use `<h2>`, `<h3>`, etc., appropriately:

```html

<h1>Welcome to our Bookstore</h1>

...

<section id="books">

<h2>Featured Books</h2>

...

</section>

```

These corrections and considerations will help improve the accessibility of the webpage according to WCAG 2.2 AA standards.

Citations:

[1] https://adabook.medium.com/wcag-2-2-aa-guide-checklist-for-2021-web-accessibility-66c6fdaea034

[2] https://www.levelaccess.com/blog/wcag-2-2-aa-summary-and-checklist-for-website-owners/

[3] https://www.wuhcag.com/wcag-checklist/

[4] https://www.w3.org/TR/WCAG22/

[5] https://www.digitala11y.com/wcag-checklist/

[6] https://webaim.org/standards/wcag/checklist

[7] https://wcag.com/blog/wcag-2-2-aa-summary-and-checklist-for-website-owners/

[8] https://adasitecompliance.com/wcag-2-2-checklist/

[9] https://www.boia.org/wcag-2.2aa-checklist

[10] https://media.dequeuniversity.com/en/docs/web-accessibility-checklist-wcag-2.2.pdf

## Are there any more issues? List all that you find.

Reviewing the provided code against the WCAG 2.2 AA checklist and considering the additional insights from the sources, here are more potential issues:

### Issue 5: Insufficient Contrast for Text

\*\*WCAG Guideline Violated:\*\* 1.4.11: Ensure that all non-text content (e.g., buttons, form fields, icons) has a minimum contrast ratio of 3:1.

\*\*Detection Algorithm:\*\* Visual inspection and automated tools.

\*\*Explanation of Error:\*\* The code does not specify colors for text or background elements. Without explicit color values, it's impossible to determine if the contrast meets the WCAG requirements. Even though the code snippet does not directly show text or background colors, it's crucial to ensure that any text used in the final implementation meets the contrast ratio requirement.

### Issue 6: Lack of Keyboard Accessibility for Interactive Elements

\*\*WCAG Guideline Violated:\*\* 2.1.4: If a keyboard shortcut is implemented, ensure that it can be turned off, remapped, or is only active when focused on a specific component.

\*\*Detection Algorithm:\*\* Review of interactive elements and their associated attributes.

\*\*Explanation of Error:\*\* The code includes interactive elements such as buttons and links, but it does not specify whether any keyboard shortcuts are implemented. If shortcuts are used, they must adhere to the guideline to ensure accessibility for users who rely on keyboards.

### Issue 7: No Alternative Means for Complex Actions

\*\*WCAG Guideline Violated:\*\* 2.5.7: For actions that require a dragging movement, provide an alternative means of performing the action, such as tapping or clicking.

\*\*Detection Algorithm:\*\* Analysis of interactive elements that might require complex gestures.

\*\*Explanation of Error:\*\* The code does not include elements that require complex interactions like dragging. However, if such elements were introduced (e.g., sliders, draggable items), they must comply with this guideline to ensure accessibility.

### Issue 8: Target Size for Interactive Elements

\*\*WCAG Guideline Violated:\*\* 2.5.8: Ensure that all interactive elements are at least 24x24 CSS pixels in size, including white space around the target.

\*\*Detection Algorithm:\*\* Visual inspection of interactive elements.

\*\*Explanation of Error:\*\* The code snippet does not detail the dimensions of interactive elements. It's important to ensure that buttons, links, and other interactive elements meet the minimum size requirement to be easily clickable or tappable, especially for users with motor control limitations.

### Issue 9: Consistency of Help Options

\*\*WCAG Guideline Violated:\*\* 3.2.6: Help options, such as support links or contact information, should be consistently placed across different pages.

\*\*Detection Algorithm:\*\* Review of the presence and placement of help options.

\*\*Explanation of Error:\*\* The code does not include help options. If such options were present, they should be consistently placed and labeled across all pages to aid users in finding assistance.

### Issue 10: Authentication Methods

\*\*WCAG Guideline Violated:\*\* 3.3.7: Offer alternative authentication methods that do not rely on cognitive tests, such as remembering usernames or solving puzzles.

\*\*Detection Algorithm:\*\* Examination of authentication mechanisms.

\*\*Explanation of Error:\*\* The code mentions a form submission (`action="submit.php" method="POST"`) but does not specify the authentication process. If the site requires users to log in, it should offer alternative authentication methods, such as email verification links, to accommodate users who may struggle with traditional username/password memorization.

Addressing these issues will further enhance the accessibility of the webpage, aligning it more closely with WCAG 2.2 AA standards.

Citations:

[1] https://www.w3.org/TR/WCAG22/

[2] https://adabook.medium.com/wcag-2-2-aa-guide-checklist-for-2021-web-accessibility-66c6fdaea034

[3] https://www.digitala11y.com/wcag-checklist/

[4] https://media.dequeuniversity.com/en/docs/web-accessibility-checklist-wcag-2.2.pdf

[5] https://accessible.org/wcag/

[6] https://webaim.org/standards/wcag/checklist

[7] https://adasitecompliance.com/wcag-2-2-checklist/

[8] https://beaccessible.com/post/wcag-2-2-checklist/

[9] https://www.levelaccess.com/blog/wcag-2-2-aa-summary-and-checklist-for-website-owners/

[10] https://www.w3.org/WAI/WCAG22/quickref/