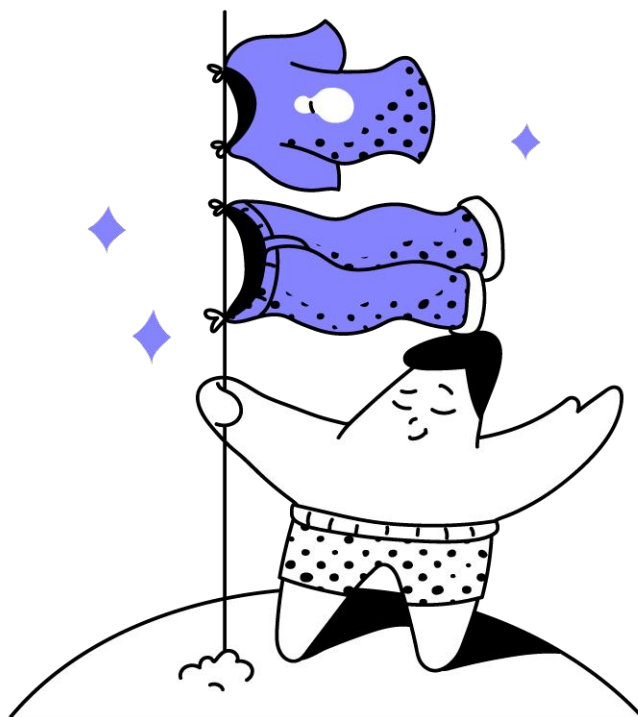


СДЕЛАТЬ ВЕБ ДОСТУПНЫМ ДЛЯ ВСЕХ



Веб-доступность — это возможность для людей с инвалидностью воспринимать, понимать, перемещаться и взаимодействовать с веб-сайтами и веб-инструментами

ОТСУТСТВИЕ ИЛИ НЕИНФОРМАТИВНЫЙ АЛЬТЕРНАТИВНЫЙ ТЕКСТ (ALT TEXT) ДЛЯ ИЗОБРАЖЕНИЙ

Скринридеры (программы для чтения с экрана, которыми пользуются незрячие и слабовидящие люди) читают альтернативный текст, чтобы описать изображение. Если его нет или он не несет смысла ("изображение", "фото"), пользователь не поймет содержание картинки.

HTML

```



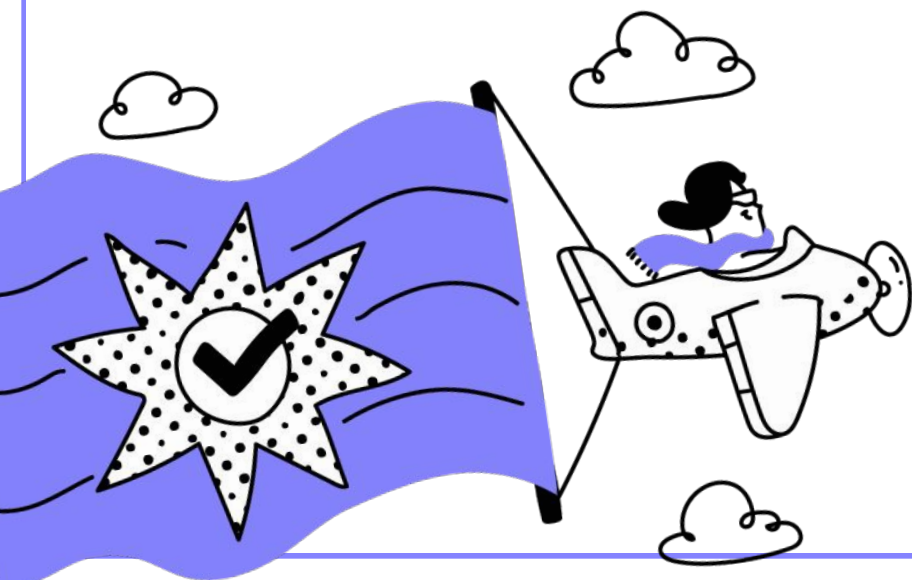






```

Всегда добавляйте осмысленный alt текст к тегу `<img>`. Он должен кратко и точно описывать содержание изображения и его функцию в контексте страницы. Если изображение чисто декоративное, используйте пустой `alt=""`.



НИЗКИЙ ЦВЕТОВОЙ КОНТРАСТ ТЕКСТА

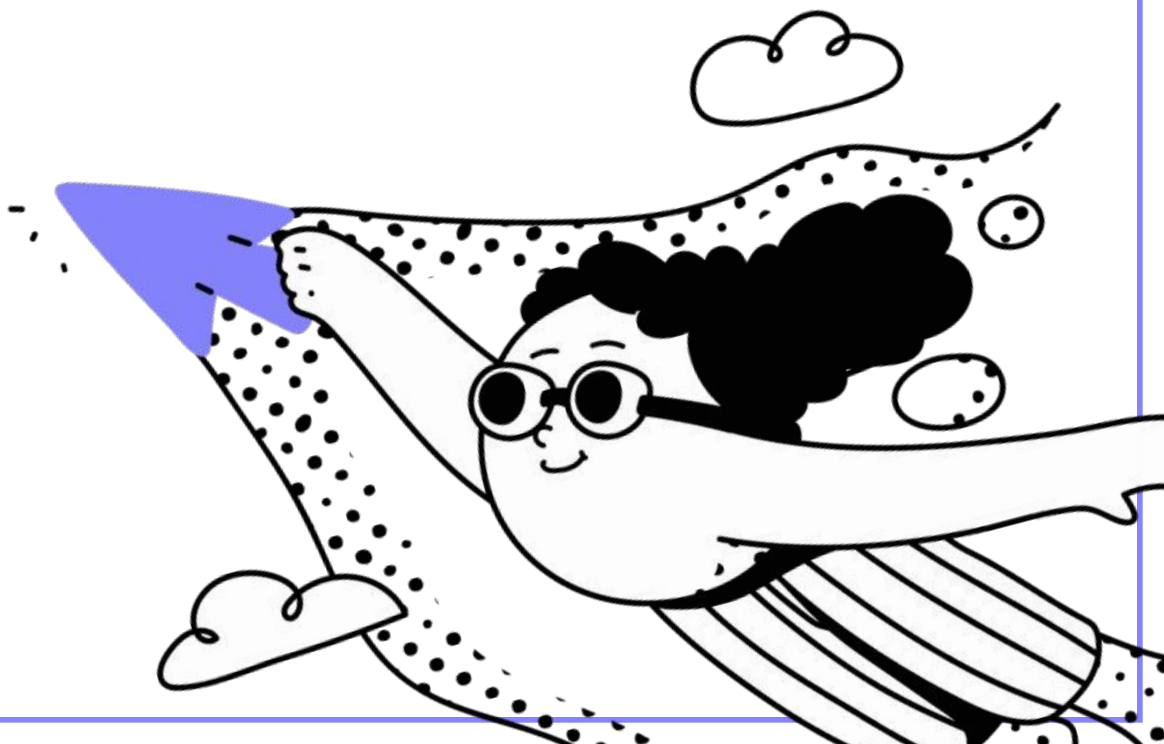
Недостаточный контраст между цветом текста и цветом фона делает текст трудночитаемым для людей с ослабленным зрением или дальтонизмом.

CSS

```
/* Пример плохого контраста */
.bad-contrast {
  color: #a3a3a3; /* Светло-серый */
  background-color: #f2f2f2; /* Очень светло-серый */
}

/* Пример хорошего контраста (соответствует WCAG AA) */
.good-contrast {
  color: #333333; /* Темно-серый */
  background-color: #ffffff; /* Белый */
}
```

Убедитесь, что контрастность текста соответствует стандартам WCAG (минимум **4.5:1** для обычного текста и **3:1** для крупного текста). Используйте онлайн-



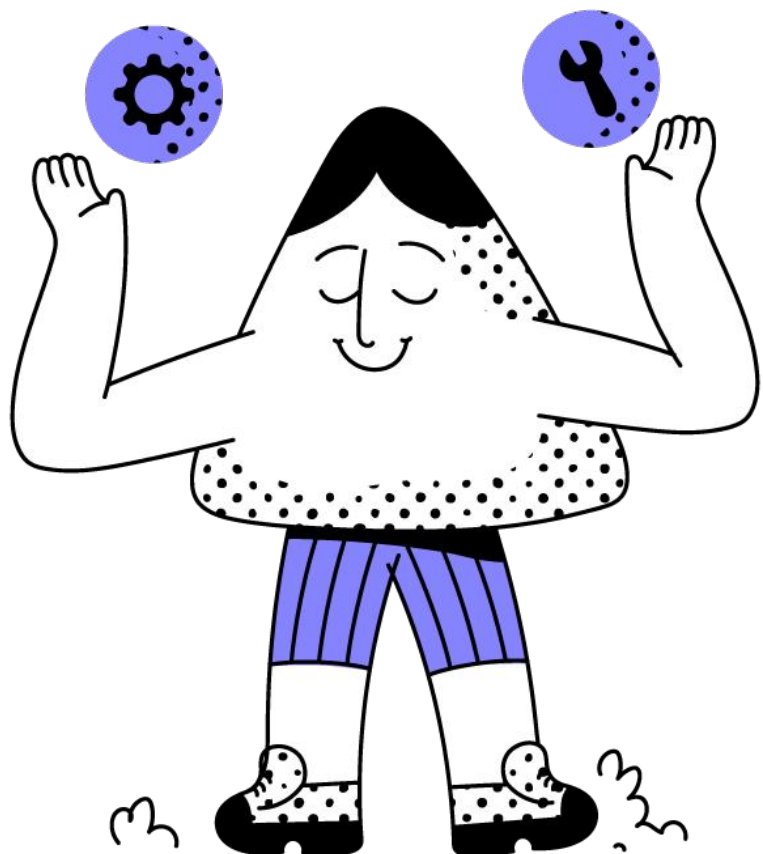
ОТСУТСТВИЕ ИЛИ НЕКОРРЕКТНЫЕ ПОДПИСИ У ПОЛЕЙ ФОРМ

Пользователи скринридеров или те, кто пользуется клавиатурной навигацией, могут испытывать трудности с определением назначения полей ввода, если они не связаны с видимыми текстовыми метками.

HTML

```
<input type="text" placeholder="Ваше имя">  
  
<label for="name">Ваше имя:</label>  
<input type="text" id="name">
```

Используйте тег `<label>` для каждого поля формы (`<input>`, `<textarea>`, `<select>`). Свяжите label с полем ввода с помощью атрибута `for`, значение которого совпадает со значением атрибута `id` поля ввода.



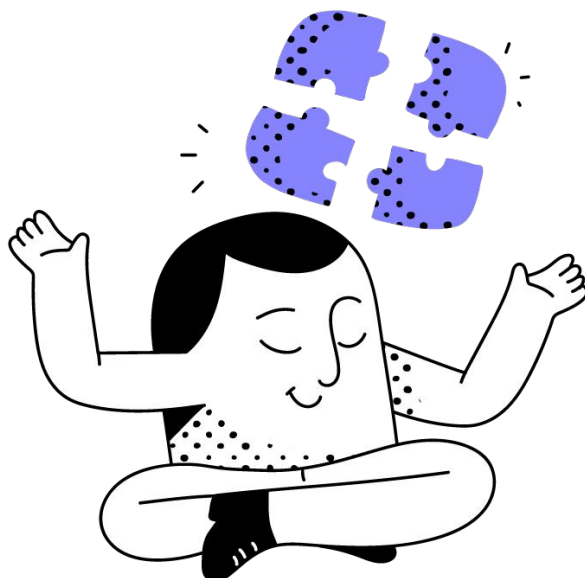
НЕДОСТУПНАЯ НАВИГАЦИЯ С КЛАВИАТУРЫ

Некоторые пользователи (например, с моторными нарушениями) полностью или частично используют только клавиатуру для навигации по сайту. Если интерактивные элементы (ссылки, кнопки, поля форм) недоступны для навигации с помощью клавиши Tab или не имеют видимого индикатора фокуса, сайт становится непригодным для использования.

CSS

```
button:focus, a:focus {  
  outline: 2px solid blue; /* Пример: синяя рамка */  
  /* Или другие стили, делающие элемент заметным */  
}
```

Убедитесь, что все интерактивные элементы доступны с клавиатуры в логичном порядке. Визуальный индикатор фокуса (обычно синяя рамка вокруг элемента) должен быть всегда виден. Избегайте использования CSS свойства `outline: none;` без предоставления альтернативного индикатора фокуса.



НЕПРАВИЛЬНОЕ ИСПОЛЬЗОВАНИЕ ЗАГОЛОВКОВ (HEADING HIERARCHY)

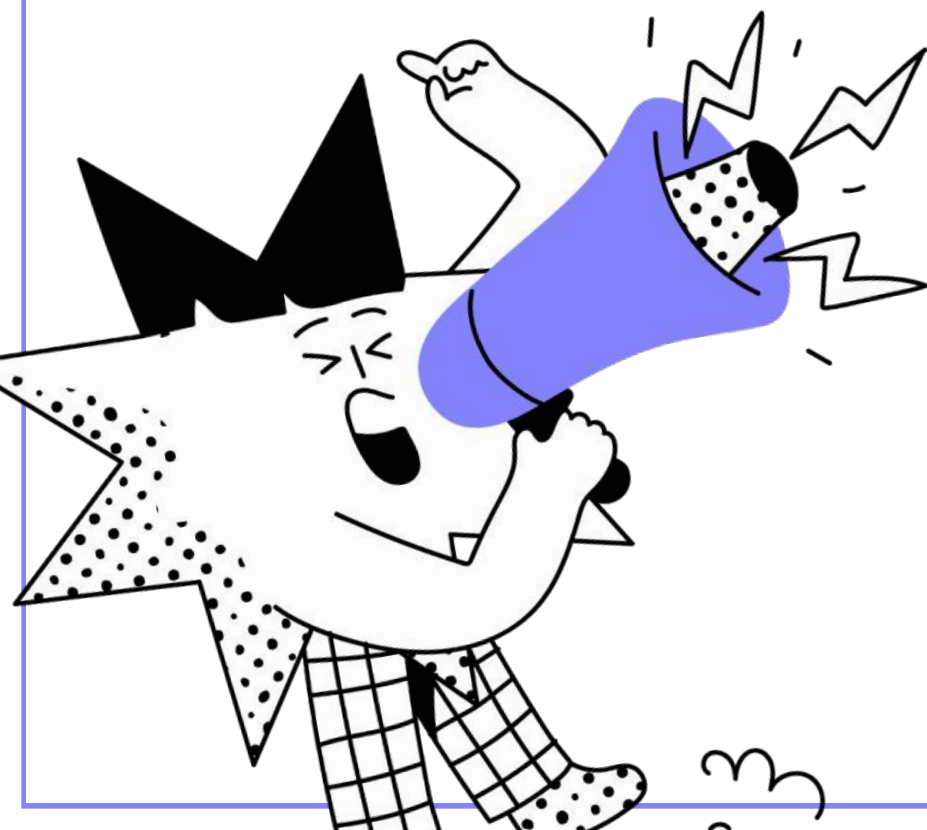
Заголовки (`<h1>`, `<h2>`, и т.д.) используются скринридерами для создания структуры страницы и быстрой навигации. Пропуск уровней заголовков (например, после `<h1>` сразу идет `<h3>`) или использование заголовков для стилизации текста нарушает эту структуру.

HTML

```
<h2>Название статьи</h2>
<h4>Подзаголовок</h4>

<h1>Название статьи</h1>
<h2>Первый раздел</h2>
<h3>Подраздел первого раздела</h3>
<h2>Второй раздел</h2>
```

Используйте заголовки последовательно, отражая логическую структуру контента. На странице должен быть только один `<h1>` (обычно заголовок страницы/статьи).



ДОСТУПНОСТЬ НА МОБИЛЬНЫХ УСТРОЙСТВАХ

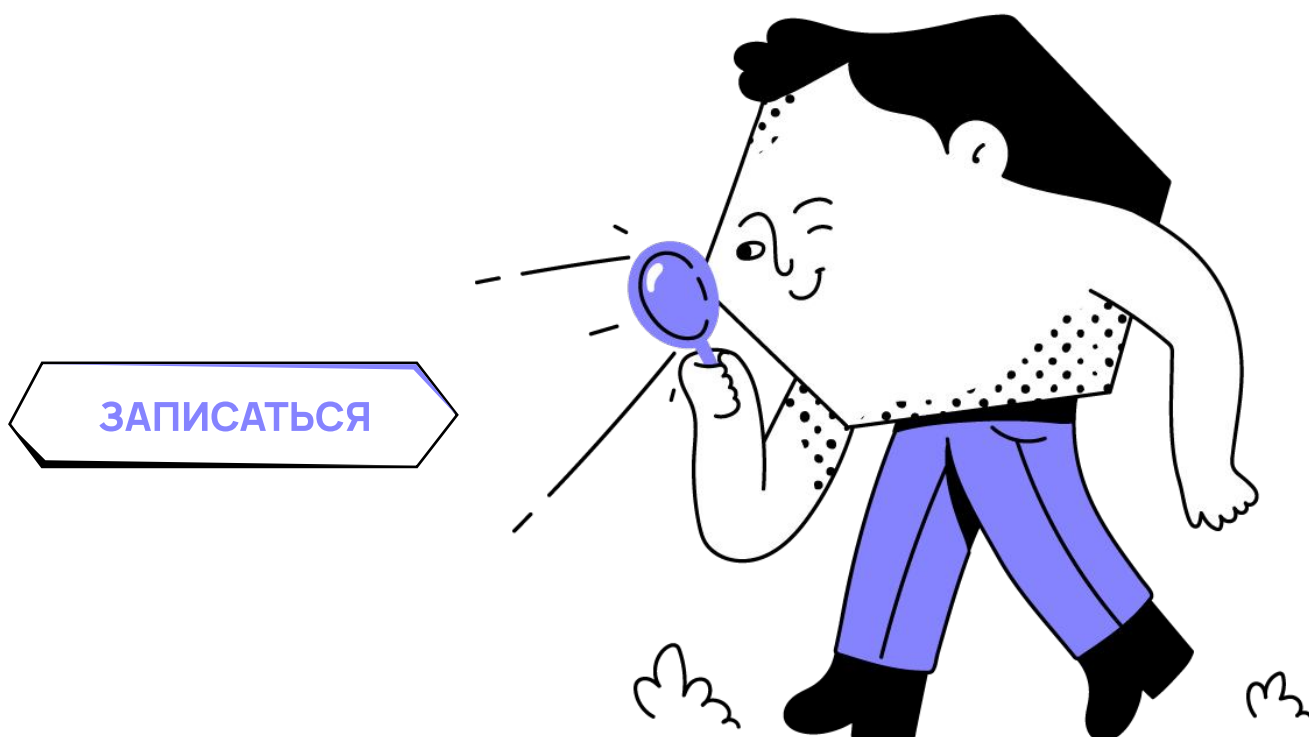
Доступность на мобильных устройствах имеет свои особенности и критически важна, учитывая рост мобильного трафика.

Адаптивный дизайн: Сайт должен корректно отображаться и быть удобным для использования на экранах любого размера. Используйте гибкие сетки, изображения и медиа-запросы CSS.

Размер интерактивных элементов: Кнопки и ссылки должны быть достаточно большими для удобного нажатия пальцем (рекомендуемый минимальный размер тач-таргета - 44x44 пикселя)

Интуитивная навигация: Мобильная навигация должна быть простой и понятной. Используйте стандартные паттерны (например, меню-бургер).
Читаемость текста: Размер шрифта должен быть достаточно крупным (минимум 16px для основного текста) и легко масштабироваться пользователем.

Пространство: Убедитесь, что между интерактивными элементами достаточно пространства, чтобы избежать случайных нажатий



ДОСТУПНОСТЬ НА МОБИЛЬНЫХ УСТРОЙСТВАХ

Маленькие кнопки меню, расположенные слишком близко друг к другу на мобильном экране.

CSS

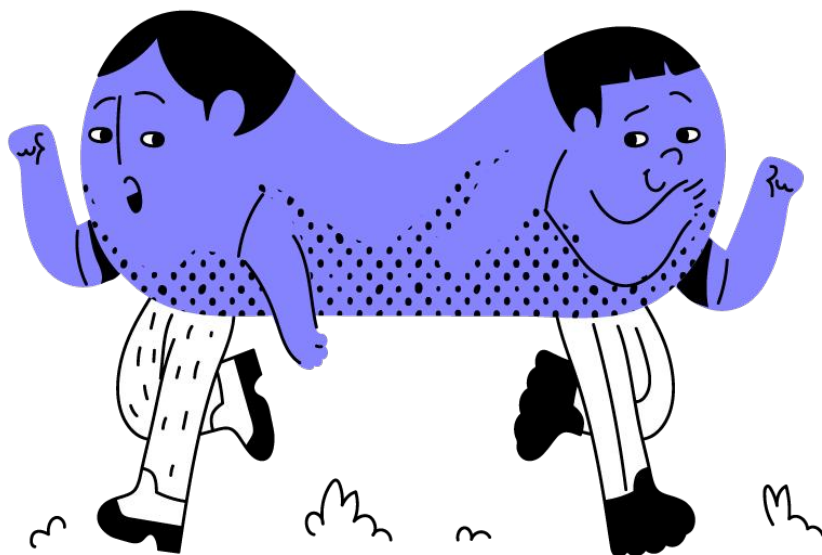
```
/* Плохо (слишком маленькие и близко расположены) */
```

```
.mobile-nav button {  
  width: 30px;  
  height: 30px;  
  margin: 2px;  
}
```

```
/* Хорошо (больше размер и отступы) */
```

```
.mobile-nav button {  
  width: 44px;  
  height: 44px;  
  margin: 8px; /* Или padding */  
}
```

Увеличьте размер кнопок и добавьте отступы (**padding** или **margin**) между ними.



ДОСТУПНОСТЬ PDF-ДОКУМЕНТОВ

PDF (Portable Document Format) является распространенным форматом для обмена документами. Однако, как и веб-страницы, PDF-файлы могут быть недоступными для людей, использующих вспомогательные технологии, если они не созданы с учетом доступности.

Основной принцип доступности PDF — это создание тегированного PDF (Tagged PDF). Теги PDF аналогичны HTML-тегам; они определяют структуру документа (заголовки, параграфы, списки, таблицы, изображения) и порядок чтения.

Вот что необходимо для доступного PDF:

1. Тегированный PDF: Документ должен содержать корректные теги, определяющие его логическую структуру и элементы
2. Правильный порядок чтения: Порядок тегов должен соответствовать визуальному порядку контента на странице.
3. Альтернативный текст (Alt Text) для изображений: Все значимые изображения должны иметь осмысленный альтернативный текст. Декоративные изображения должны быть помечены как таковые (с помощью соответствующих тегов).
4. Заголовки (Headings): Используйте правильную иерархию заголовков (H1, H2, H3 и т.д.) для структурирования контента.
5. Списки: Списки (маркированные и нумерованные) должны быть правильно тегированы как списки.
6. Таблицы: Сложные таблицы должны иметь корректные теги заголовков строк и столбцов, чтобы вспомогательные технологии могли правильно интерпретировать связи между ячейками.

7. Ссылки: Ссылки должны быть интерактивными и иметь понятный, описывающий их назначение текст.
8. Язык документа: Основной язык документа должен быть указан.
9. Закладки (Bookmarks): Для длинных документов полезно добавить закладки, основанные на структуре заголовков, для удобной навигации.
10. Цветовой контраст: Как и на веб-страницах, текст и важные графические элементы должны иметь достаточный цветовой контраст с фоном.
11. Доступные формы: Поля форм должны иметь метки, подсказки и корректный порядок табуляции.
12. Безопасность: Настройки безопасности не должны препятствовать использованию вспомогательных технологий (например, запрещать копирование текста).

Лучший способ получить доступный PDF — начать с доступного исходного документа (например, в Microsoft Word, Adobe InDesign) и экспортировать его в PDF с сохранением тегов и структуры. Простое сохранение или печать в PDF часто не сохраняет необходимую информацию о доступности.



ЗАКЛЮЧЕНИЕ

Создание доступных веб-сайтов — это не дополнительная функция, а фундаментальный аспект качественной веб-разработки. Применяя эти простые рекомендации на этапе проектирования и разработки, вы делаете Интернет лучше для всех.

Полезные ресурсы для проверки и понимания веб-доступности и валидности кода:

WCAG (Web Content Accessibility Guidelines): Официальные руководства по доступности веб-контента от Консорциума Всемирной паутины (W3C). Это основной стандарт, на который стоит ориентироваться.

Lighthouse: Инструмент с открытым исходным кодом от Google для автоматизированной проверки качества веб-страниц, включая доступность, производительность, SEO и лучшие практики. Встроен в Chrome DevTools.

W3C Markup Validation Service: Инструмент для проверки синтаксиса HTML, XHTML, SMIL и других языков разметки веб-страниц. Чистый и валидный код является хорошей основой для доступности, хотя сам по себе не гарантирует ее полностью.

Axe (Deque Systems): Браузерное расширение axe DevTools очень удобно для быстрой автоматизированной проверки доступности непосредственно в браузере.

