

TECHNICAL DOCUMENTATION PAGE - CERTIFICATION

PROJECT

Objective: Build an app that is functionally similar to <https://technical-documentation-page.freecodecamp.rocks>. Do not copy this demo project.

User Stories:

1. You can see a `main` element with a corresponding `id="main-doc"`, which contains the page's main content (technical documentation)
2. Within the `#main-doc` element, you can see several `section` elements, each with a class of `main-section`. There should be a minimum of five
3. The first element within each `.main-section` should be a `header` element, which contains text that describes the topic of that section.
4. Each `section` element with the class of `main-section` should also have an `id` that corresponds with the text of each `header` contained within it. Any spaces should be replaced with underscores (e.g. The section that contains the header "JavaScript and Java" should have a corresponding `id="JavaScript_and_Java"`)
5. The `.main-section` elements should contain at least ten `p` elements total (not each)
6. The `.main-section` elements should contain at least five `code` elements total (not each)
7. The `.main-section` elements should contain at least five `li` items total (not each)

8. You can see a `nav` element with a corresponding `id="navbar"`
9. The navbar element should contain one `header` element which contains text that describes the topic of the technical documentation
10. Additionally, the navbar should contain link (`a`) elements with the class of `nav-link`. There should be one for every element with the class `main-section`
11. The `header` element in the `#navbar` must come before any link (`a`) elements in the navbar
12. Each element with the class of `nav-link` should contain text that corresponds to the `header` text within each `section` (e.g. if you have a "Hello world" section/header, your navbar should have an element which contains the text "Hello world")
13. When you click on a navbar element, the page should navigate to the corresponding section of the `#main-doc` element (e.g. If you click on a `.nav-link` element that contains the text "Hello world", the page navigates to a `section` element with that id, and contains the corresponding header)
14. On regular sized devices (laptops, desktops), the element with `id="navbar"` should be shown on the left side of the screen and should always be visible to the user
15. Your technical documentation should use at least one media query

Fulfill the user stories and pass all the tests below to complete this project. Give it your own personal style. Happy Coding!

Note: Be sure to add `<link rel="stylesheet" href="styles.css">` in your HTML to link your stylesheet and apply your CSS

TESTS

1. You should have a `main` element with an `id` of `main-doc`.
2. You should have at least five `section` elements with a class of `main-section`.
3. All of your `.main-section` elements should be `section` elements.
4. You should have at least five `.main-section` elements that are descendants of `#main-doc`.
5. The first child of each `.main-section` should be a `header` element.
6. None of your `header` elements should be empty.
7. All of your `.main-section` elements should have an `id`.
8. Each `.main-section` should have an `id` that matches the text of its first child, having any spaces in the child's text replaced with underscores (`_`) for the id's.
9. You should have at least 10 `p` elements (total) within your `.main-section` elements.
10. You should have at least five `code` elements that are descendants of `.main-section` elements.
11. You should have at least five `li` elements that are descendants of `.main-section` elements.
12. You should have a `nav` element with an `id` of `navbar`.
13. Your `#navbar` should have exactly one `header` element within it.
14. You should have at least one `a` element with a class of `nav-link`.
15. All of your `.nav-link` elements should be anchor (`a`) elements.
16. All of your `.nav-link` elements should be in the `#navbar`.
17. You should have the same number of `.nav-link` and `.main-section` elements.
18. The `header` element in the `#navbar` should come before any link (`a`) elements also in the `#navbar`.

19. Each `.nav-link` should have text that corresponds to the `header` text of its related `section` (e.g. if you have a "Hello world" section/header, your `#navbar` should have a `.nav-link` which has the text "Hello world")
20. Each `.nav-link` should have an `href` attribute that links to its corresponding `.main-section` (e.g. If you click on a `.nav-link` element that contains the text "Hello world", the page navigates to a `section` element with that id).
21. Your `#navbar` should always be on the left edge of the window.
22. Your Technical Documentation project should use at least one media query.