

CHAPTER 3

Target Project 1

In the previous chapter, you have learned about the Twitter Bootstrap grid system. In this chapter, you will develop a web page that has a navigation bar fixed at the top as seen in Figure 3-1.

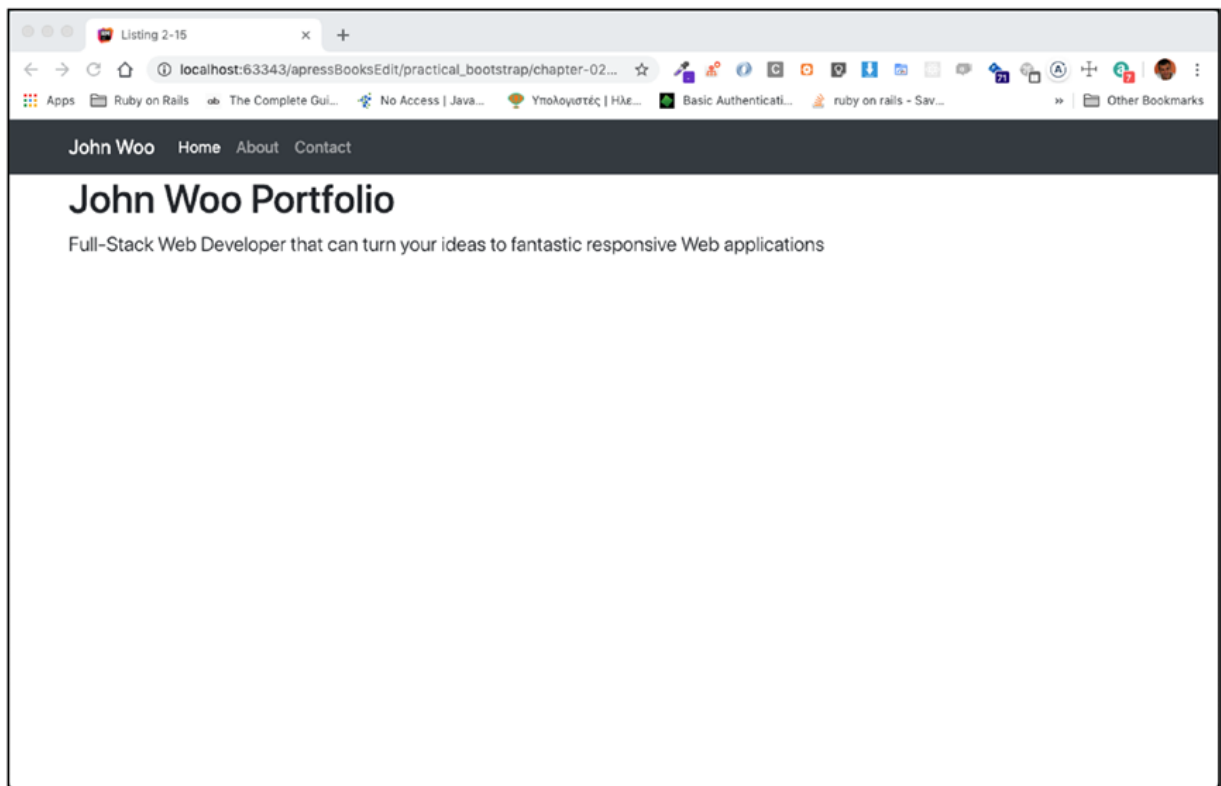


Figure 3-1. *Basic Template with a Top Nav Bar*

This navigation bar is responsive, which means it can be equally displayed on small-display devices. In those cases, the menu is hidden behind a hamburger icon (Figure 3-2).

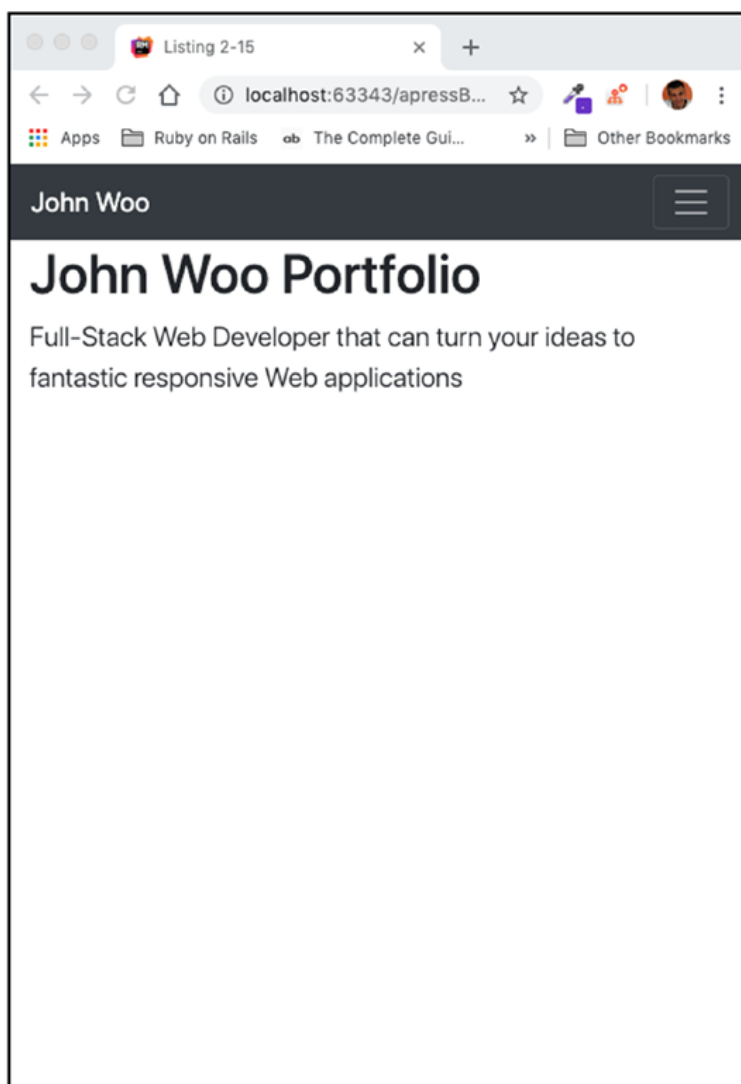


Figure 3-2. *This Is How a Site Looks on Mobile Devices*

When the user clicks the hamburger icon, the menu is unfolded, like the following (Figure 3-3).

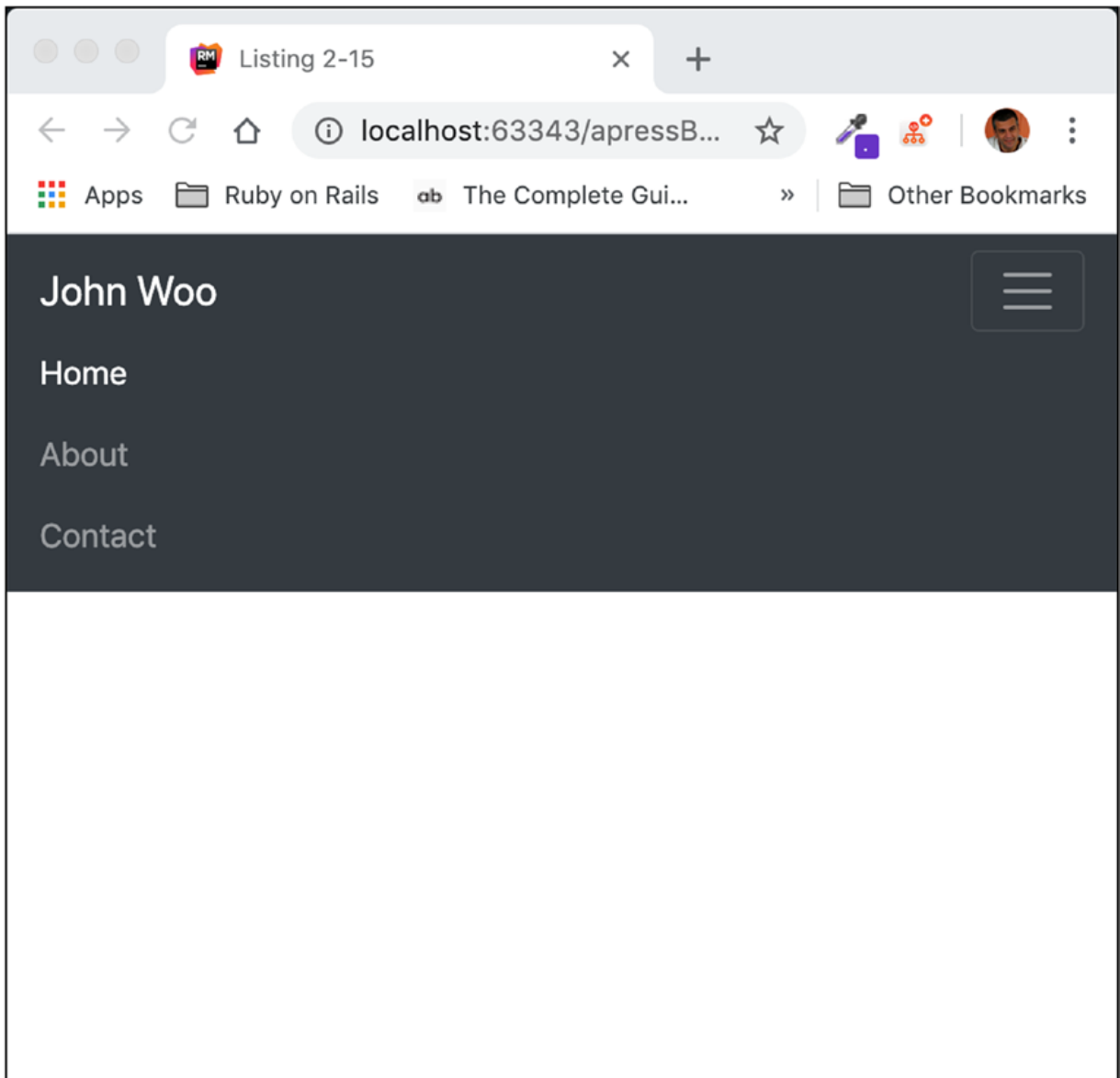


Figure 3-3. *Menu Options Appear on Hamburger Icon Click*

You will develop this web page, but at the same time, you will understand how the Twitter Bootstrap classes work.

Learning Goals

1. Learn to build a navigation bar for your site, step by step.
2. Make the navigation bar responsive.
3. Learn how to create a button with the shape of a hamburger and how to make it unfold the menu options.
4. Learn how to make the content of a paragraph stand out.
5. Learn about how to make your main content area not be hidden by the navigation bar.

Introduction

This and the following chapters will be teaching you the framework concepts while trying to create a project at the same time. Let's start, then, with our first project.

Assume that we want to build the following website in Figure 3-4.

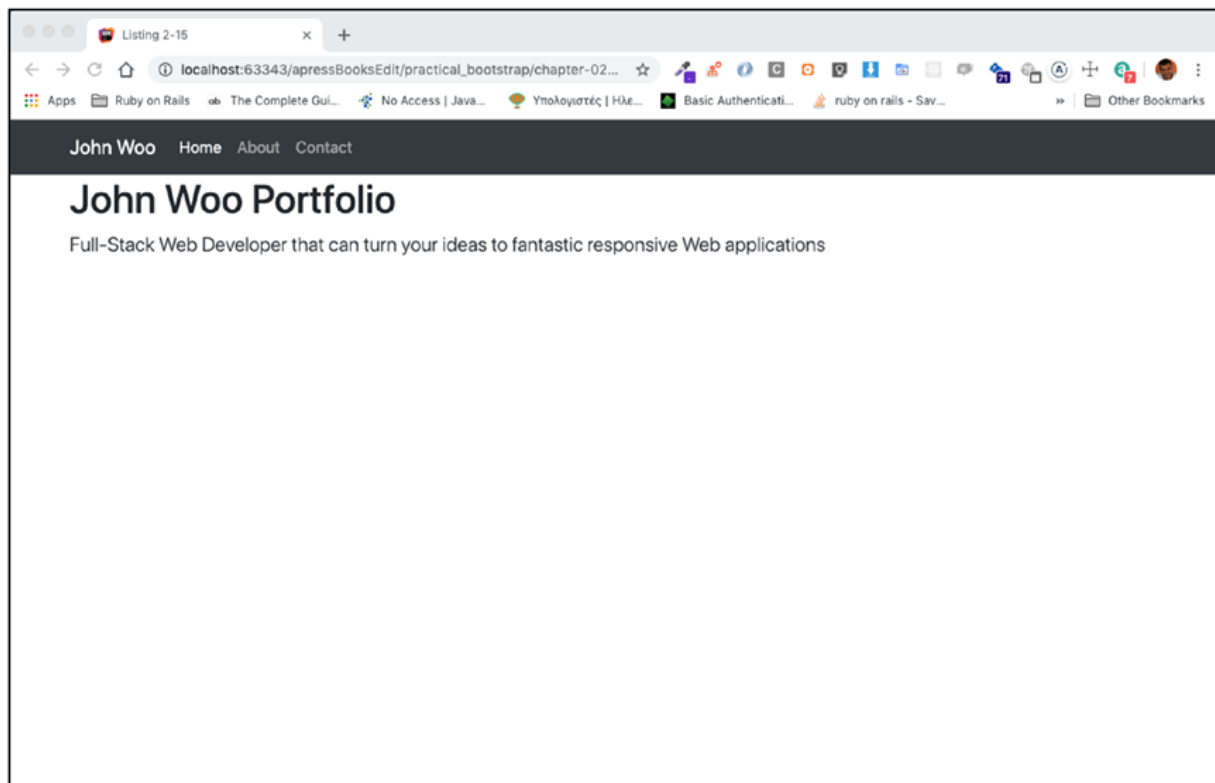


Figure 3-4. Target Project 1—Basic Template with a Top Nav Bar

This project has the following characteristics:

1. It has a navigation bar. This navigation bar is responsive. This means that when you resize your browser to mimic the size of a mobile phone, you will see something like Figure 3-5.

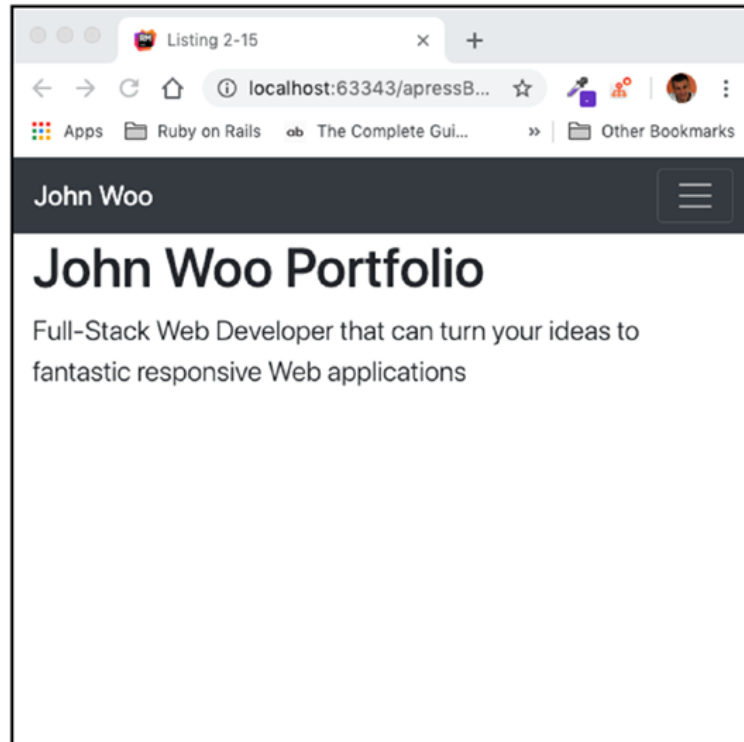


Figure 3-5. *This Is How a Site Looks Like on Mobile Devices*

As you can see, the menu options have been collapsed and turned to a hamburger icon. If the user wants to get the list of menu options, they need to click this icon (Figure 3-6).

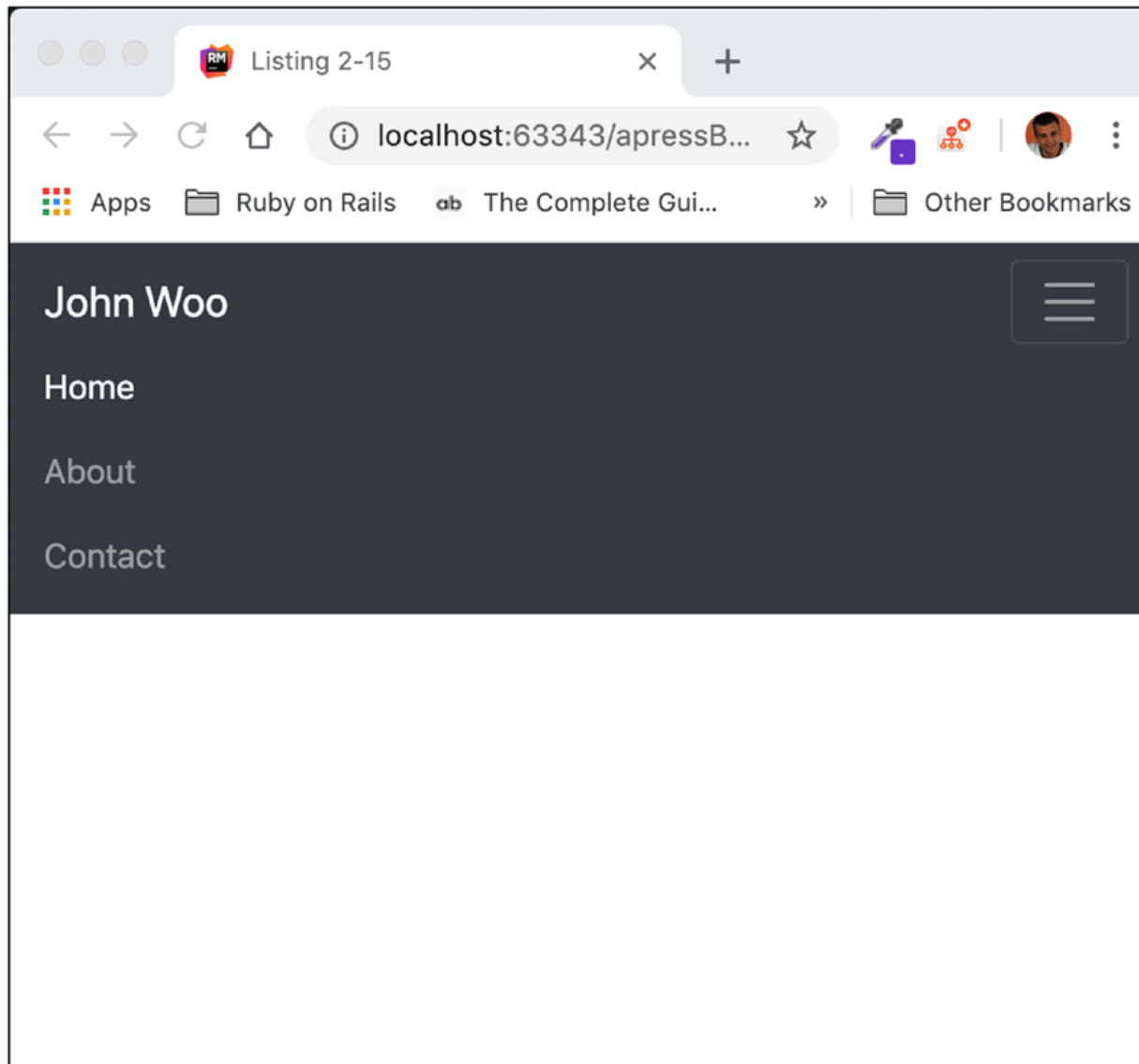


Figure 3-6. *Menu Options Appear on Hamburger Icon Click*

There are other features on this website that you take for free from Twitter Bootstrap. For example, when you mouse over the options, you will see that the options change color and intensity.

Let's try to build this website from scratch, using Twitter Bootstrap.

Start with HTML

Let's start with the basic HTML that includes a nav bar and a list of items that will be the menu options, like the following (Listing 3-1).

Listing 3-1. Starting with a nav Element and List of Menu Items

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
  shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.
  com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
  Vkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
  crossorigin="anonymous">

  <!-- Custom CSS -->
  <link rel="stylesheet" href="stylesheets/main.css" type="text/css">

  <title>Listing 3-1</title>
</head>
<body>
  <nav>

    <ul>
      <li><a href="#">Home</a></li>
      <li><a href="#about">About</a></li>
      <li><a href="#contact">Contact</a></li>
    </ul>

  </nav>
```

```

<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoR5JoZ+n" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

Save the preceding code into the file `index.html` and load the page on your browser. Note that the referenced `stylesheets/main.css` does not contain anything, for the time being. But create this file so that your page can find it.

You will see the following (Figure 3-7).

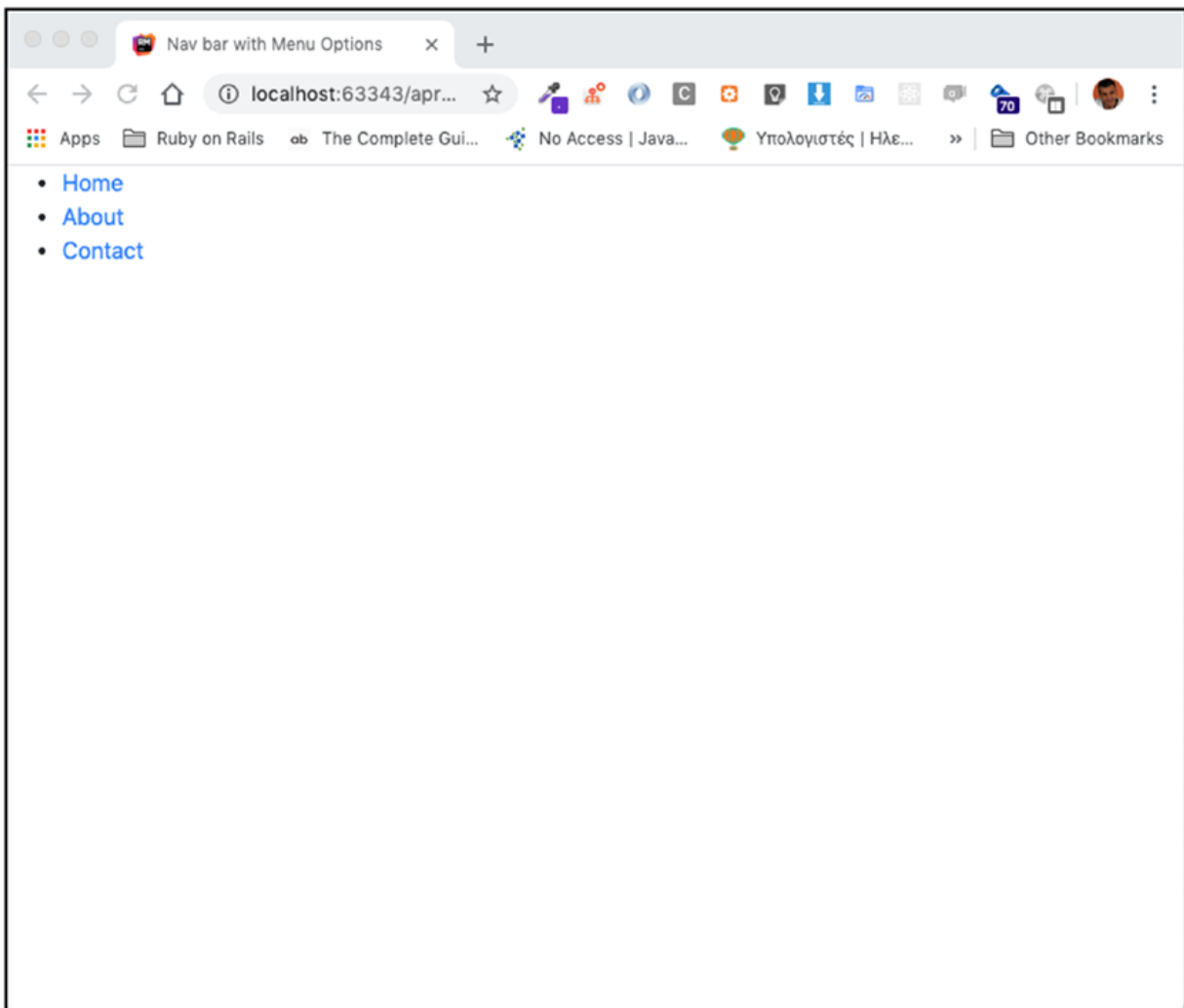


Figure 3-7. *Page with a Nav Bar and Menu Items but Not Styled*

I guess that this was expected. The body element contained nothing but a block element `nav` and a `ul` with the menu options. Note that we are using the semantic HTML5 element `nav`, instead of just a `div`.

The Class `nav`

You will first add the class `nav` on the `ul` element that holds the menu options. This will remove the bullets from the list. Do the following change. Instead of ``, write `<ul class="nav">`. Save the file and reload the page. You will see the following (Figure 3-8).

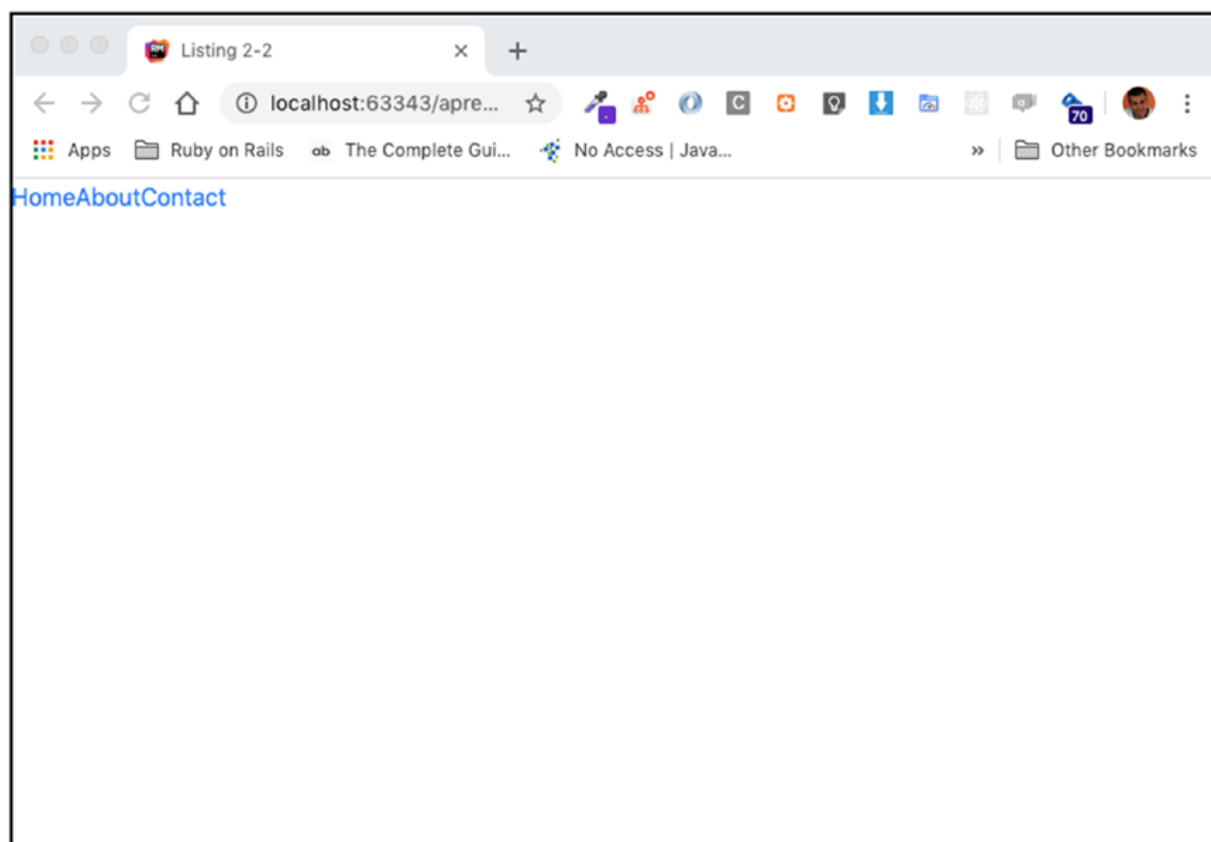


Figure 3-8. Added the Class *nav* on the *ul* Element

The Navigation Items and Navigation Links

Let's now make the menu items have some space around them, because, now, they are one exactly next to the other.

In order to achieve this, you will add the class *nav-item* on the *li* elements and the class *nav-link* on the anchor *a* elements. Read Listing 3-2 to see how the current state of your HTML page should be.

Listing 3-2. Menu Items on the Same Line, One Separate from the Other

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
  shrink-to-fit=no">
```

```

<!-- Bootstrap CSS -->
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.
com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
Vtkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
crossorigin="anonymous">

<!-- Custom CSS -->
<link rel="stylesheet" href="stylesheets/main.css" type="text/css">

<title>Listing 3-2</title>
</head>
<body>
  <nav>

    <ul class="nav">
      <li class="nav-item">
        <a class="nav-link" href="#">Home</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#about">About</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#contact">Contact</a>
      </li>
    </ul>

  </nav>

  <!-- Optional JavaScript -->
  <!-- jQuery first, then Popper.js, then Bootstrap JS -->
  <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoRSJoZ+n" crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>

```

```
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>
```

If you load the preceding page on your browser, you will see the following (Figure 3-9).

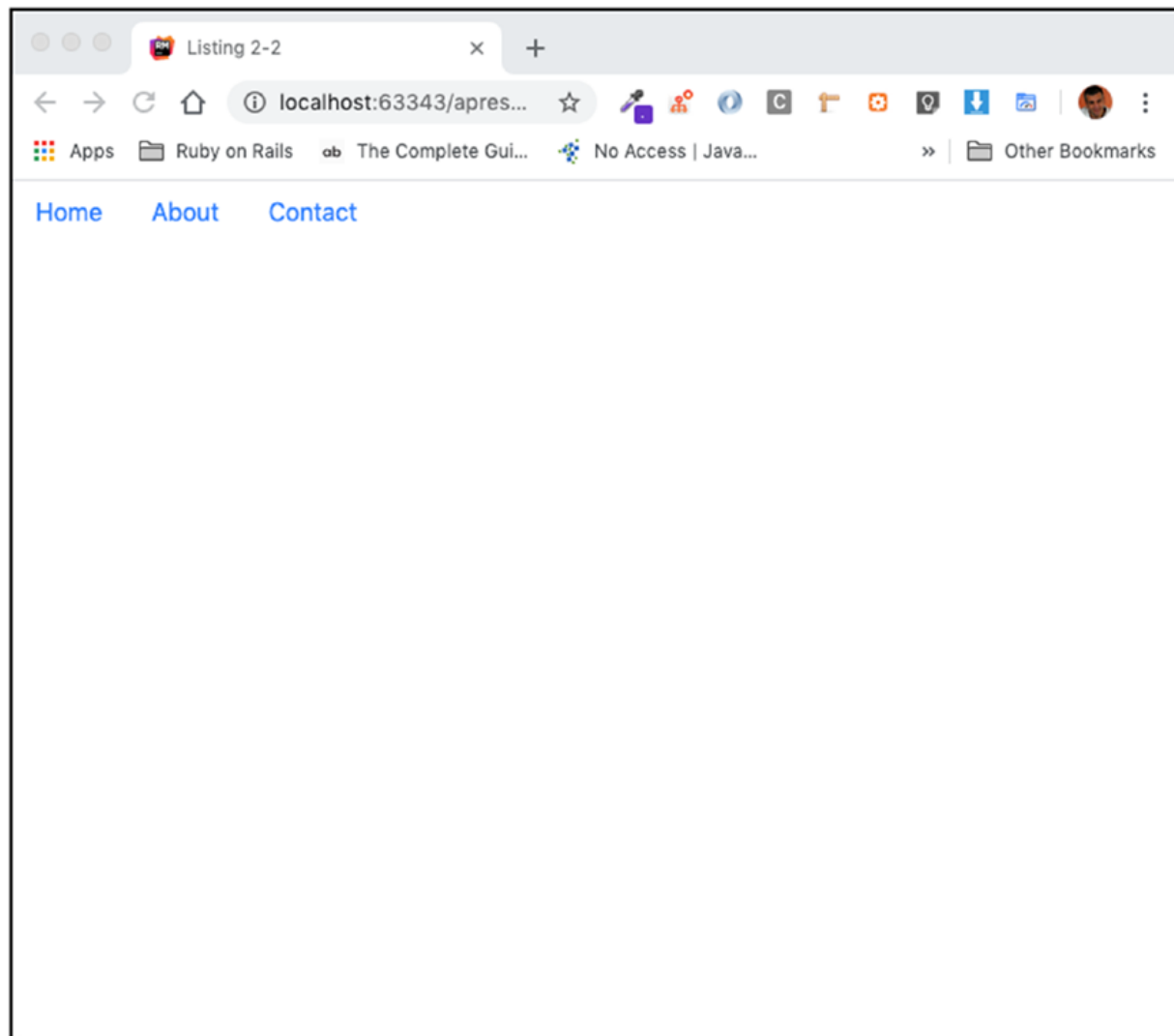


Figure 3-9. *Menu Items Separated*

Collapse and Expand

In order for the list of menu items to be collapsible, that is, to be replaced with a hamburger icon on small devices, which, when clicked, would enlist the menu options, you need to add a special class on the nav element, and you need to wrap the collapsible/expandable content into a div with special classes, like in Listing 3-3.

Listing 3-3. Make Menu Options Collapse and Expand

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
  shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.
  com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
  Vkoo8x4CGs03+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
  crossorigin="anonymous">

  <!-- Custom CSS -->
  <link rel="stylesheet" href="stylesheets/main.css" type="text/css">

  <title>Listing 3-3</title>
</head>
<body>
  <nav class="navbar-expand-lg">
    <div class="collapse navbar-collapse">
      <ul class="nav">
        <li class="nav-item">
          <a class="nav-link" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#about">About</a>
```

```

        </li>
        <li class="nav-item">
            <a class="nav-link" href="#contact">Contact</a>
        </li>
    </ul>
</div>

</nav>
<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoRStJoZ+n" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1iqlfktj0Uod8GCExl30G8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

What you have done in order to make the menu options collapse and expand is the following:

1. You have added the class `navbar-expand-lg` on the `nav` element. This instructs the browser to expand the content at the `lg` breakpoint, that is, for displays greater than or equal to 992px. Apparently, you could have used another breakpoint, for example, the `md` breakpoint, if you wanted the menu to expand for displays with width ≥ 768 px.
2. You have wrapped the `ul` element into a `div` element with classes `collapse` and `navbar-collapse`. This makes the browser know which content it needs to collapse and expand.

If you load the new HTML content into your browser and set the pixel width to 991px, then you will see a blank page like the following (Figure 3-10).

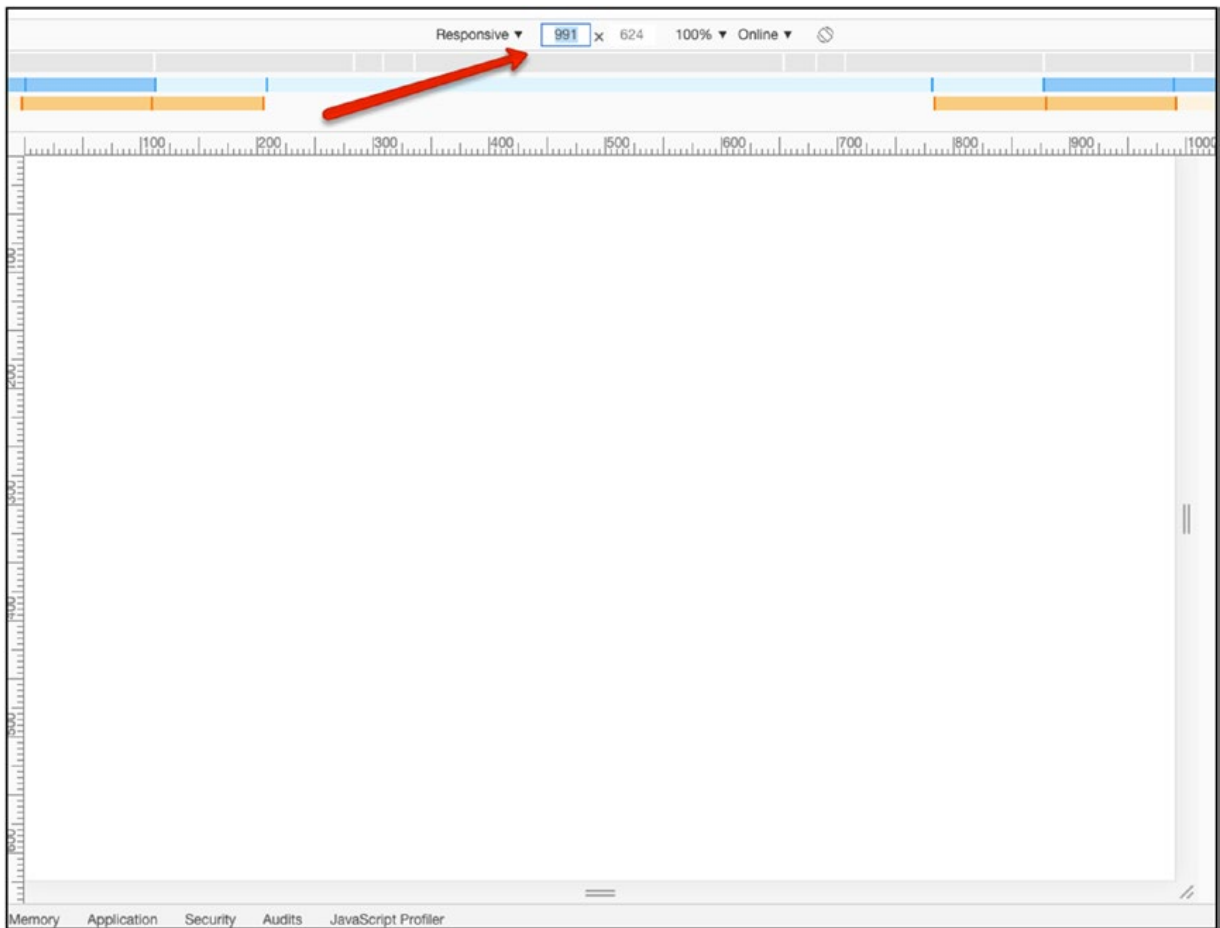


Figure 3-10. *A Blank Page on Width Less Than 992px*

But, if you expand to 992px, then you will see the menu options appearing back again (Figure 3-11).

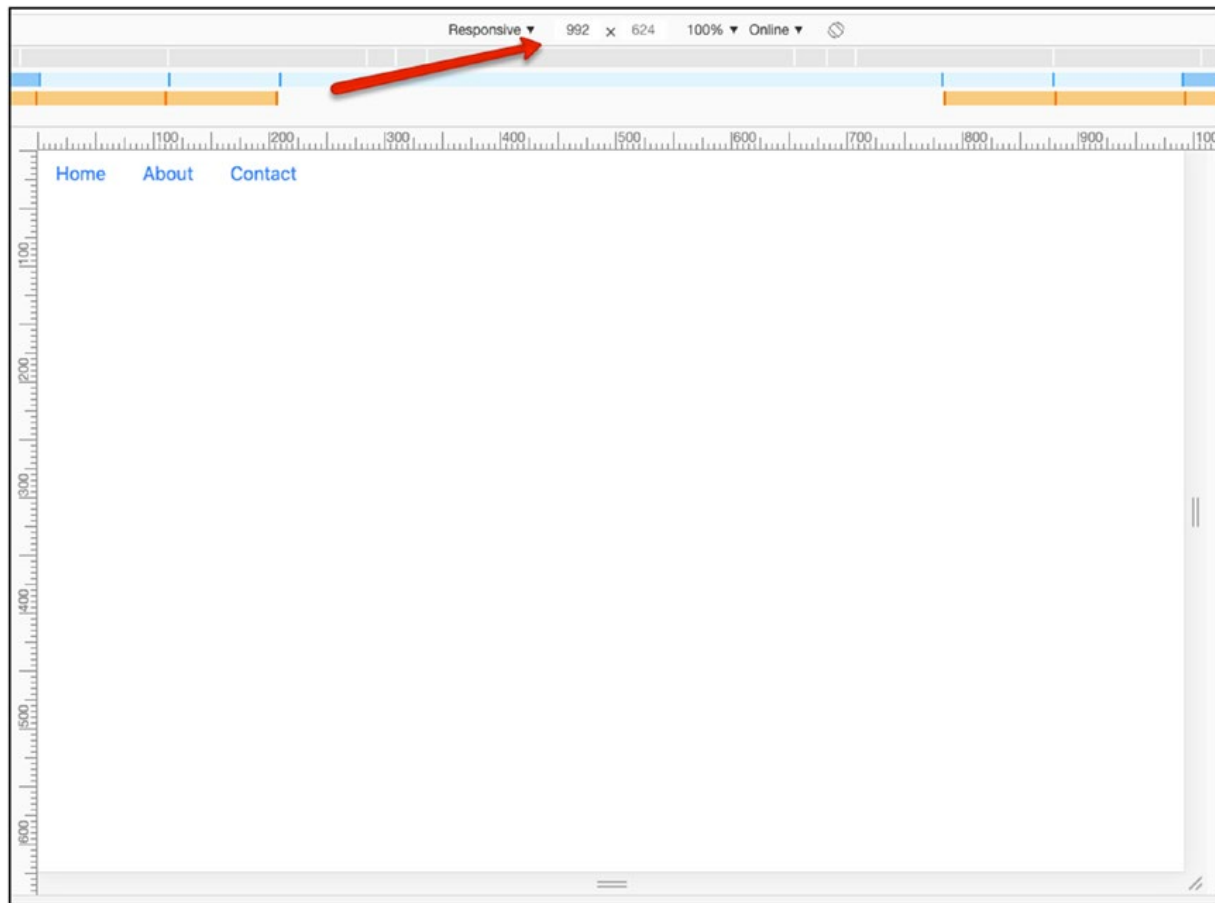


Figure 3-11. Menu Options Appear on 992px

The Brand link

If you go back and look again at the final result, you will see that there is the link with text John Woo, which is not hidden when the page shrinks to mobile size. It is always visible, and it is supposed to be a link that would take the user to the home page of the site. It is not part of the list of menu items.

Let's put that above the collapsible div, but still inside the nav element (Listing 3-4).

Listing 3-4. Add the Link That Would Take the User to the Home Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
```



```

<meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no">

<!-- Bootstrap CSS -->
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.
com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
Vkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
crossorigin="anonymous">

<!-- Custom CSS -->
<link rel="stylesheet" href="stylesheets/main.css" type="text/css">

<title>Listing 3-4</title>
</head>
<body>
  <nav class="navbar-expand-lg">

    <a href="#">John Woo</a>

    <div class="collapse navbar-collapse">
      <ul class="nav">
        <li class="nav-item">
          <a class="nav-link" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#about">About</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#contact">Contact</a>
        </li>
      </ul>
    </div>

  </nav>
  <!-- Optional JavaScript -->
  <!-- jQuery first, then Popper.js, then Bootstrap JS -->

```

```

<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoR5JoZ+n" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1i9fktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

If you save the preceding code and reload your page, you will see the John Woo link on the left, which, if you shrink the page, remains visible and does not disappear like the menu options do (Figure 3-12).

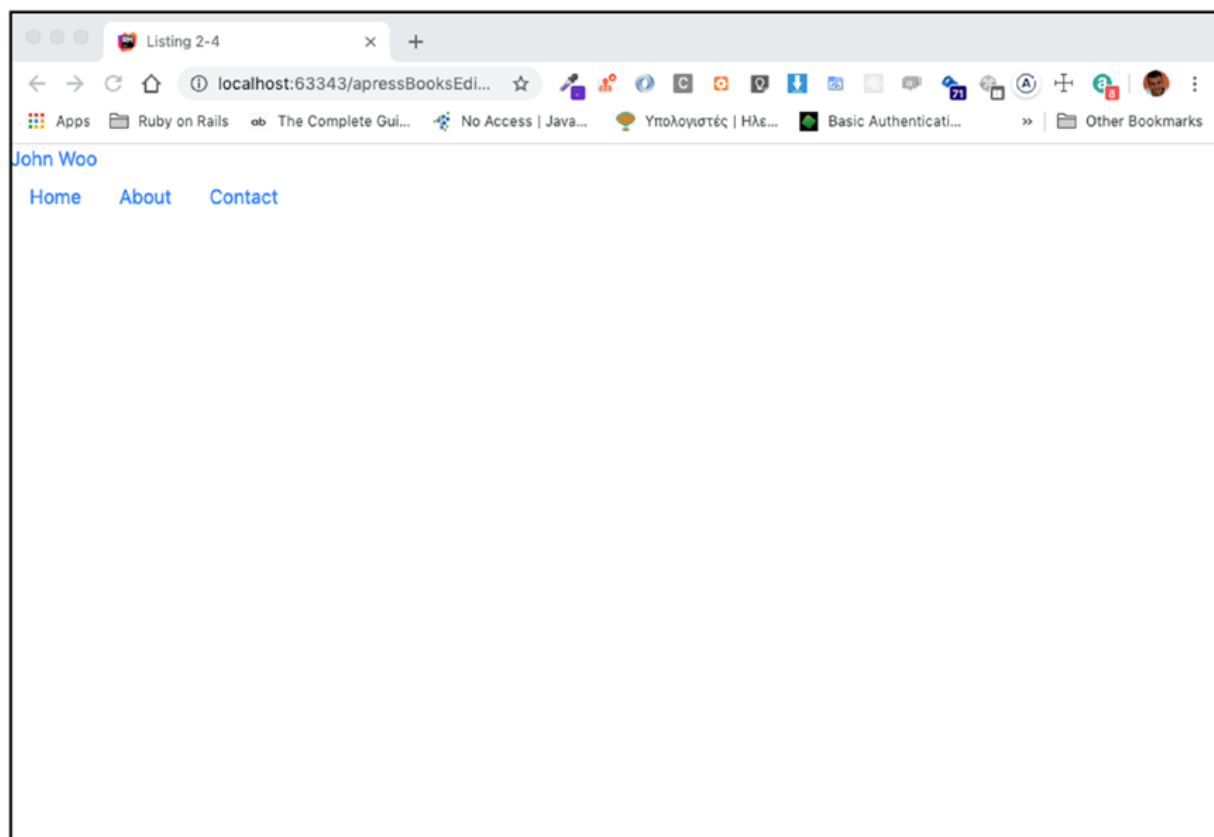


Figure 3-12. Brand Link Is Separate from Menu Options

Make the Brand Link Stand Out

The brand link, on the top left, needs to be on the same level as the menu; and it needs to stand out a little bit. Let's do that by

1. Adding the class navbar to the nav wrapping element
2. Adding the class navbar-brand to the anchor element

Caution The navbar class on the nav wrapping element is crucial for the whole styling of the navigation bar, not only for the brand link.

Listing 3-5 is what you need to have in your HTML page in order to make this work.

Listing 3-5. Brand Link on the Same Line and Standing Out

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
  shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.
  com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
  Vkoo8x4CGs03+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
  crossorigin="anonymous">

  <!-- Custom CSS -->
  <link rel="stylesheet" href="stylesheets/main.css" type="text/css">

  <title>Listing 3-5</title>
</head>
```

```

<body>
  <nav class="navbar navbar-expand-lg">

    <a class="navbar-brand" href="#">John Woo</a>

    <div class="collapse navbar-collapse">
      <ul class="nav">
        <li class="nav-item">
          <a class="nav-link" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#about">About</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#contact">Contact</a>
        </li>
      </ul>
    </div>

  </nav>
  <!-- Optional JavaScript -->
  <!-- jQuery first, then Popper.js, then Bootstrap JS -->
  <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoR5JoZ+n" crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

If you load the preceding HTML page on your browser, you will see what is depicted in Figure 3-13.

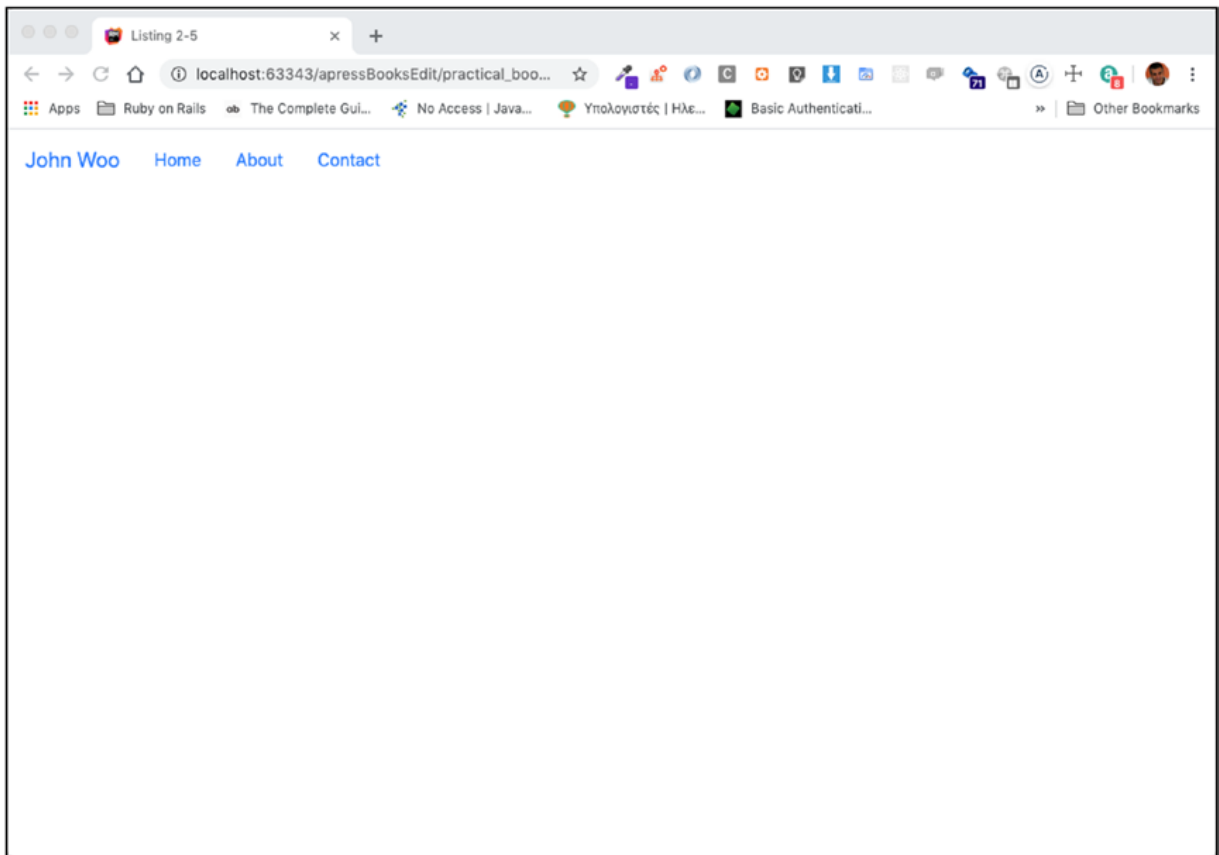


Figure 3-13. Brand Link Standing Out on the Left

Add Some Color to Your Nav Bar

Let's add some color to your nav bar. In order to do that, you have to

1. Use one of the navbar-dark or navbar-light classes at the nav element.
2. Combine it with a background color utility class.

The former is basically dealing with the font colors, whereas the latter is dealing with the background colors. Usually you would like to combine the *-dark class with the dark background colors and the *-light class with the light background colors.

The background color utilities are listed in [Figure 3-14](#).

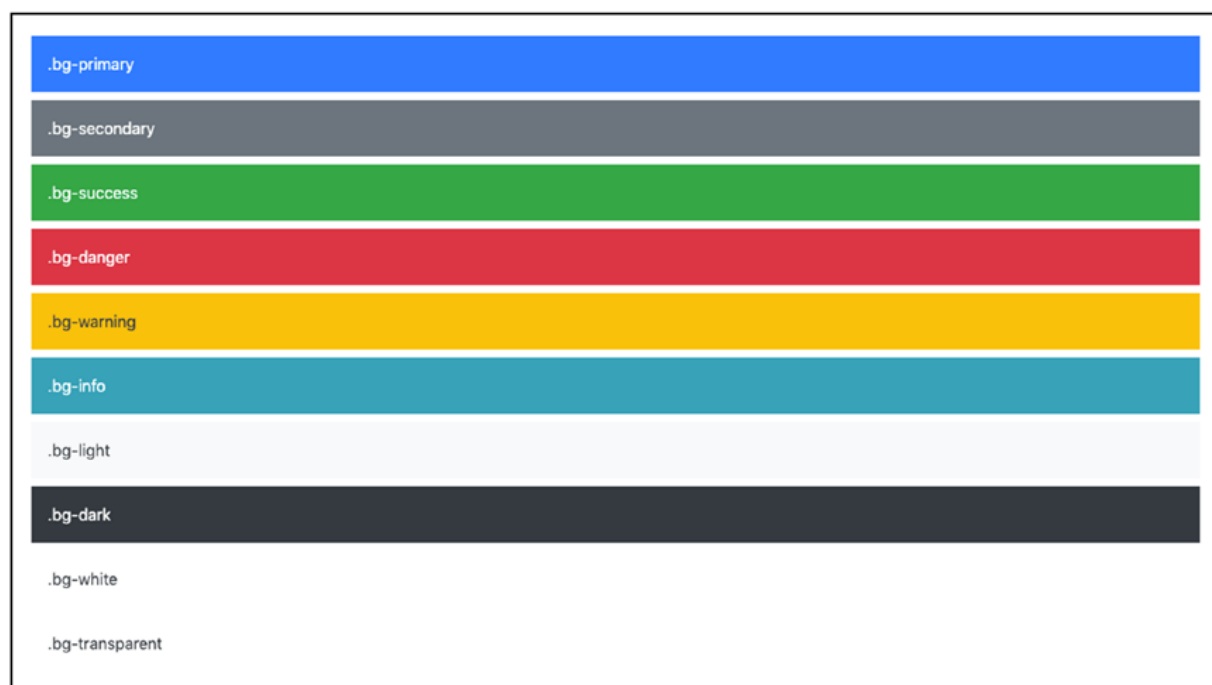


Figure 3-14. Background Color Utility Classes

Caution The color classes, `navbar-dark` and `navbar-light`, have been designed to expect the `ul` element to have the class `navbar-nav`. So we are going to add that too.

In Listing 3-6, we are applying the `bg-dark` background color utility class, combined with the `navbar-dark` color scheme for the navigation bar. We also add the class `navbar-nav` to the `ul` element.

Listing 3-6. Coloring the Navigation Bar

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
    shrink-to-fit=no">
```

```

<!-- Bootstrap CSS -->
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.
com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
Vkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
crossorigin="anonymous">

<!-- Custom CSS -->
<link rel="stylesheet" href="stylesheets/main.css" type="text/css">

<title>Listing 3-6</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">

    <a class="navbar-brand" href="#">John Woo</a>

    <div class="collapse navbar-collapse">
      <ul class="navbar-nav nav">
        <li class="nav-item">
          <a class="nav-link" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#about">About</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#contact">Contact</a>
        </li>
      </ul>
    </div>

  </nav>
  <!-- Optional JavaScript -->
  <!-- jQuery first, then Popper.js, then Bootstrap JS -->
  <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoRSJoZ+n" crossorigin="anonymous"></script>

```

```

<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

If you load the preceding HTML page on your browser, you will see what is depicted in Figure 3-15.

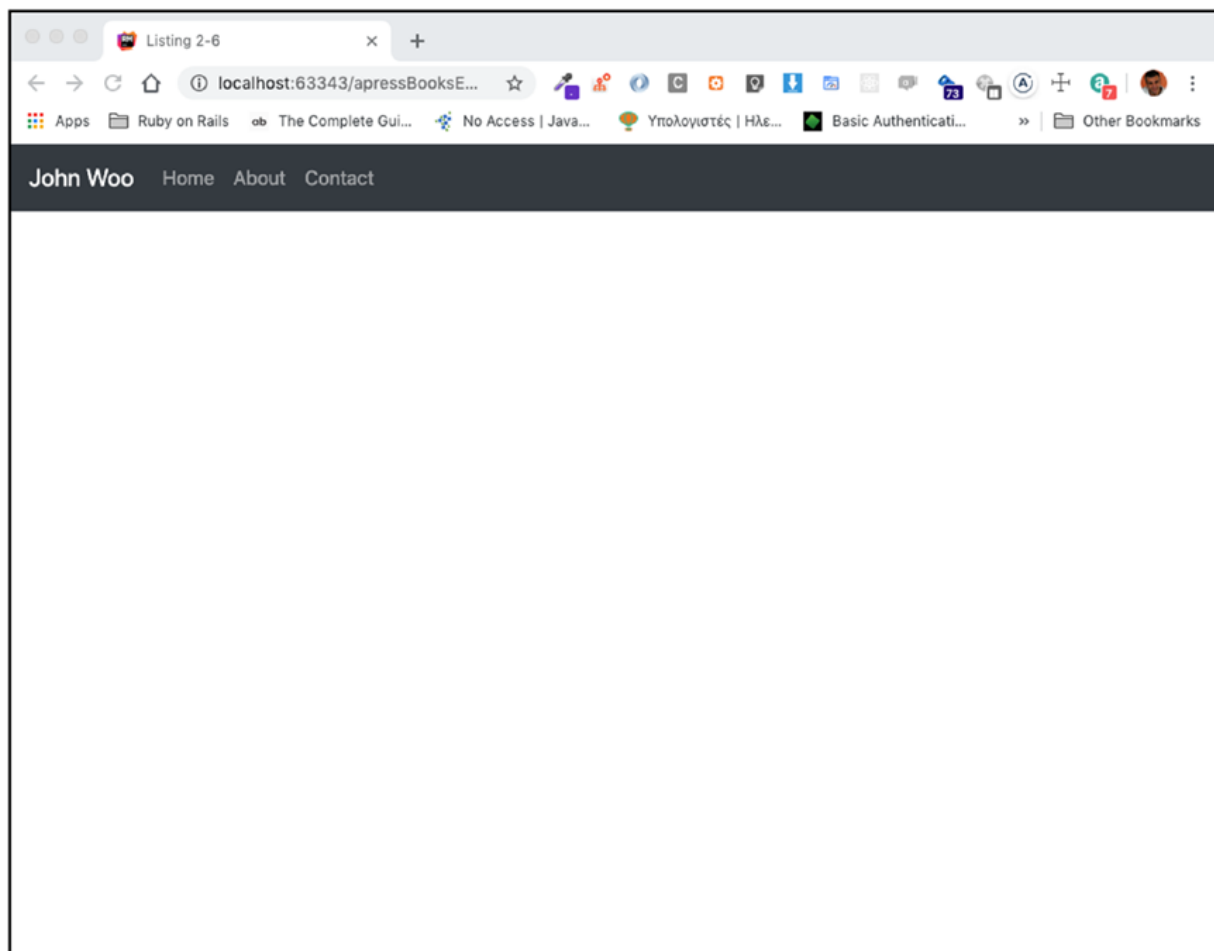


Figure 3-15. *Coloring the Navigation Bar*

Also, hover your mouse pointer over the menu options. You will see how the links are getting lighter.

Adding the Button to Unfold Menu Options on Small Devices

Let's add a button. This will be clicked to unfold the menu options on small devices. See Listing 3-7.

Listing 3-7. Add a Button That Will Toggle the Menu On and Off on Small Devices

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
  shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.
  com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
  Vkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
  crossorigin="anonymous">

  <!-- Custom CSS -->
  <link rel="stylesheet" href="stylesheets/main.css" type="text/css">

  <title>Listing 3-7</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">

    <a class="navbar-brand" href="#">John Woo</a>

    <button type="button">
</button>
```

```

    <div class="collapse navbar-collapse">
      <ul class="nav navbar-nav">
        <li class="nav-item">
          <a class="nav-link" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#about">About</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#contact">Contact</a>
        </li>
      </ul>
    </div>

  </nav>
  <!-- Optional JavaScript -->
  <!-- jQuery first, then Popper.js, then Bootstrap JS -->
  <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
  integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
  yYfoR5SJoZ+n" crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
  popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
  zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
  bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
  aYd1iqfktj0Uod8GCExl30G8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

You have added the button as a sibling of the brand anchor, but after it. If you load this HTML page on your browser, you will see the following (Figure 3-16).

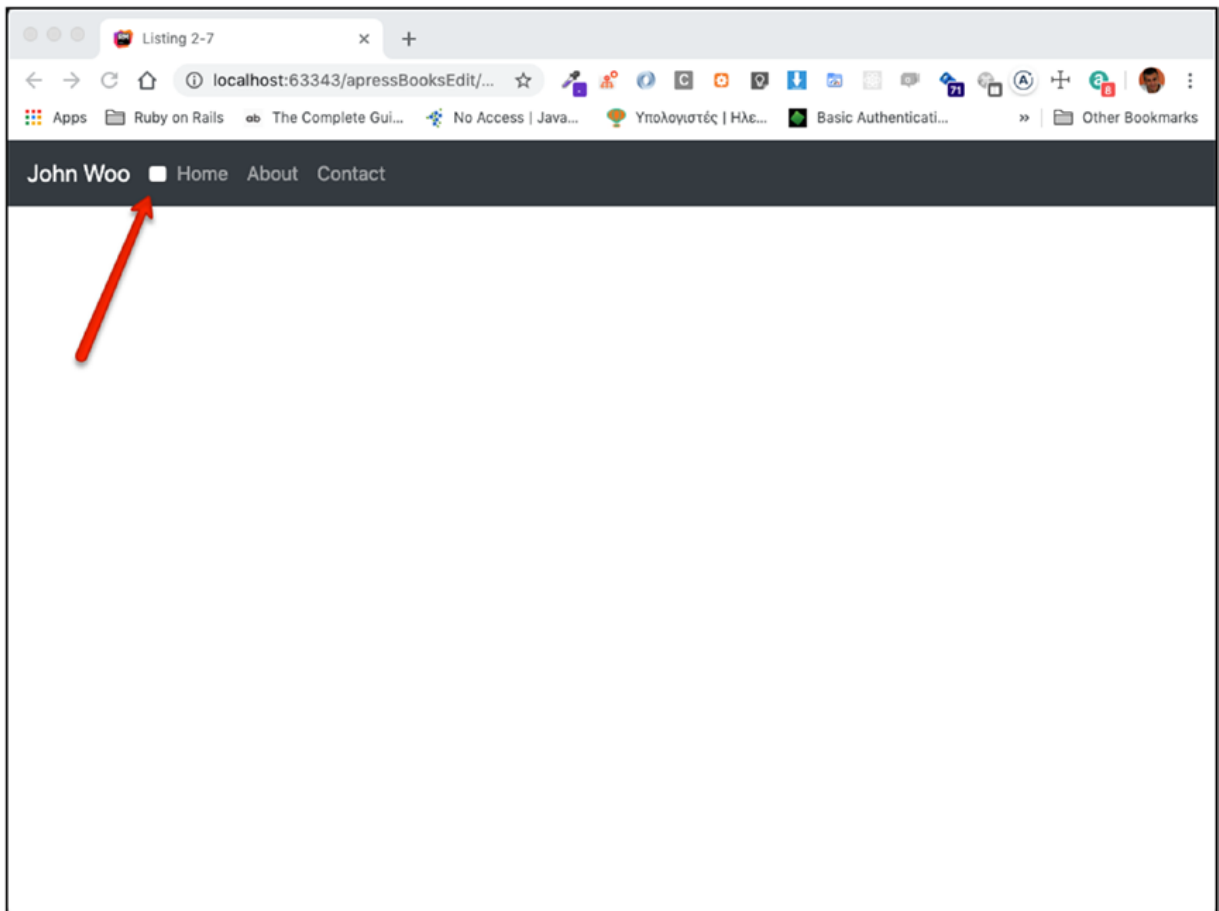


Figure 3-16. *Unstyled Button to Toggle the Menu On and Off*

This is pretty bad. Indeed. But it's a start.

Making the Button Appear Only on Small Devices

The problem you have with the button, at the moment, is that it always appears. What you want is that it appears only on devices with width below the expanded breakpoint that you have set on the nav element. The breakpoint that you have set is `lg`, which means that you want the button to appear for displays with width `<992px`.

In order to do that, you can add the class `navbar-toggler`. Let's do that. See Listing 3-8.

Listing 3-8. Make the Button Appear Only on Widths Below the Expanded Breakpoint

```

<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
  shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.
  com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
  Vkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
  crossorigin="anonymous">

  <!-- Custom CSS -->
  <link rel="stylesheet" href="stylesheets/main.css" type="text/css">

  <title>Listing 3-8</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">

    <a class="navbar-brand" href="#">John Woo</a>

    <button type="button" class="navbar-toggler">
</button>

    <div class="collapse navbar-collapse">
      <ul class="nav navbar-nav">
        <li class="nav-item">
          <a class="nav-link" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#about">About</a>
        </li>
        <li class="nav-item">

```

```

        <a class="nav-link" href="#contact">Contact</a>
    </li>
</ul>
</div>

</nav>
<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoRSJoZ+n" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1i9fktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

Now, when you have a display $\geq 992\text{px}$, the button does not appear (Figure 3-17).

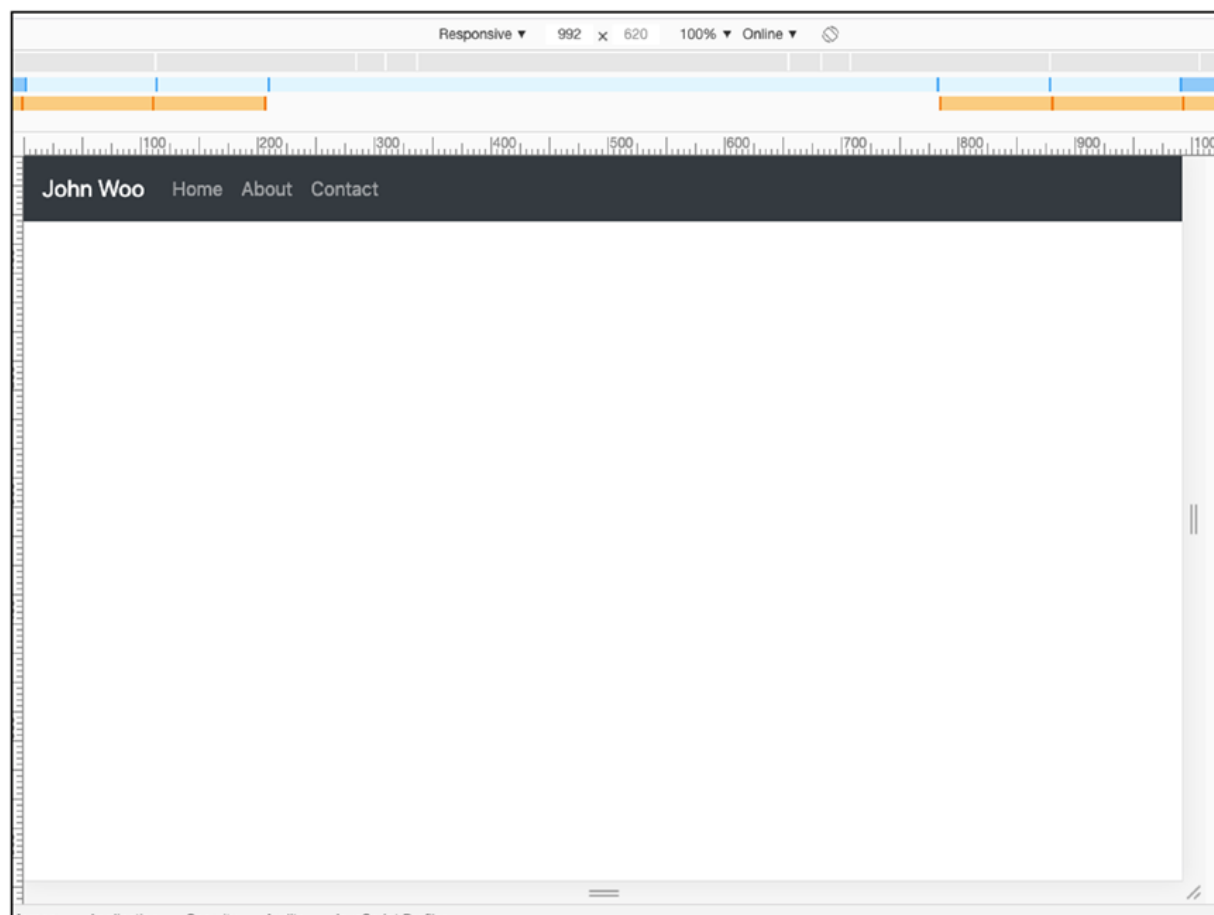


Figure 3-17. *Menu Is Visible; Button to Toggle Is Not Visible ($\geq 992\text{px}$)*

When you shrink by 1 pixel, to 991px, then you will see the menu disappearing and the button appearing (Figure 3-18).

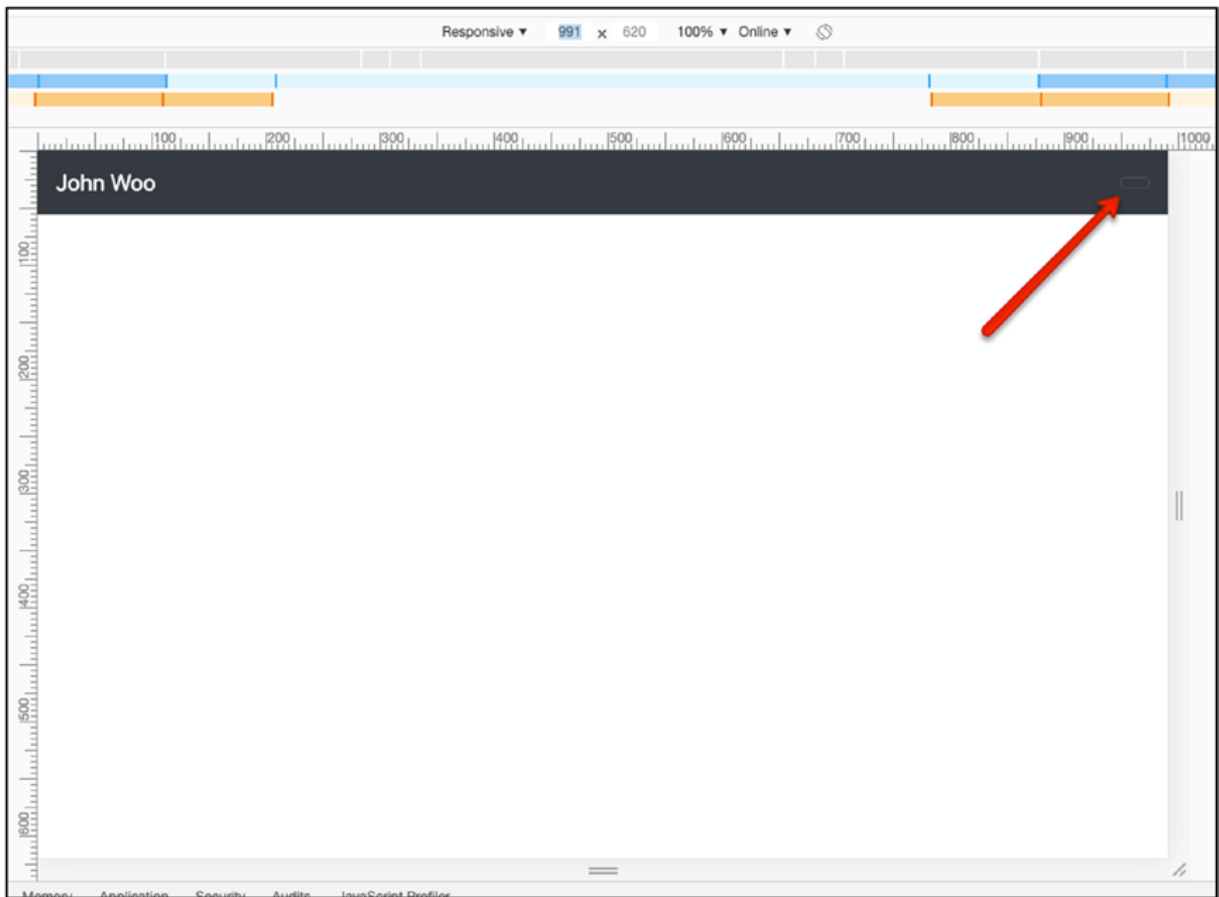


Figure 3-18. Menu Hidden, but Button Present (<992px)

The button appears in a way that is very hard to see. This is fixed next.

Creating the Hamburger Icon

The button does not look good on small devices, and this is because it does not have actual content inside. In order to create the visual icon of the hamburger, Twitter Bootstrap gives you the class `navbar-toggler-icon`. You need to apply that on a `span` element inside the button element. See Listing 3-9.

Listing 3-9. Creating the Hamburger Icon

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
```

```

<meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no">

<!-- Bootstrap CSS -->
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.
com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
Vko08x4CGs03+Hhvxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
crossorigin="anonymous">

<!-- Custom CSS -->
<link rel="stylesheet" href="stylesheets/main.css" type="text/css">

<title>Listing 3-9</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">

    <a class="navbar-brand" href="#">John Woo</a>

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

    <div class="collapse navbar-collapse">
      <ul class="nav navbar-nav">
        <li class="nav-item">
          <a class="nav-link" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#about">About</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#contact">Contact</a>
        </li>
      </ul>
    </div>

  </nav>

  <!-- Optional JavaScript -->
  <!-- jQuery first, then Popper.js, then Bootstrap JS -->

```



```

<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoRSJoZ+n" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

If you load the preceding HTML page on a display with 991px width, you will see the following (Figure 3-19).

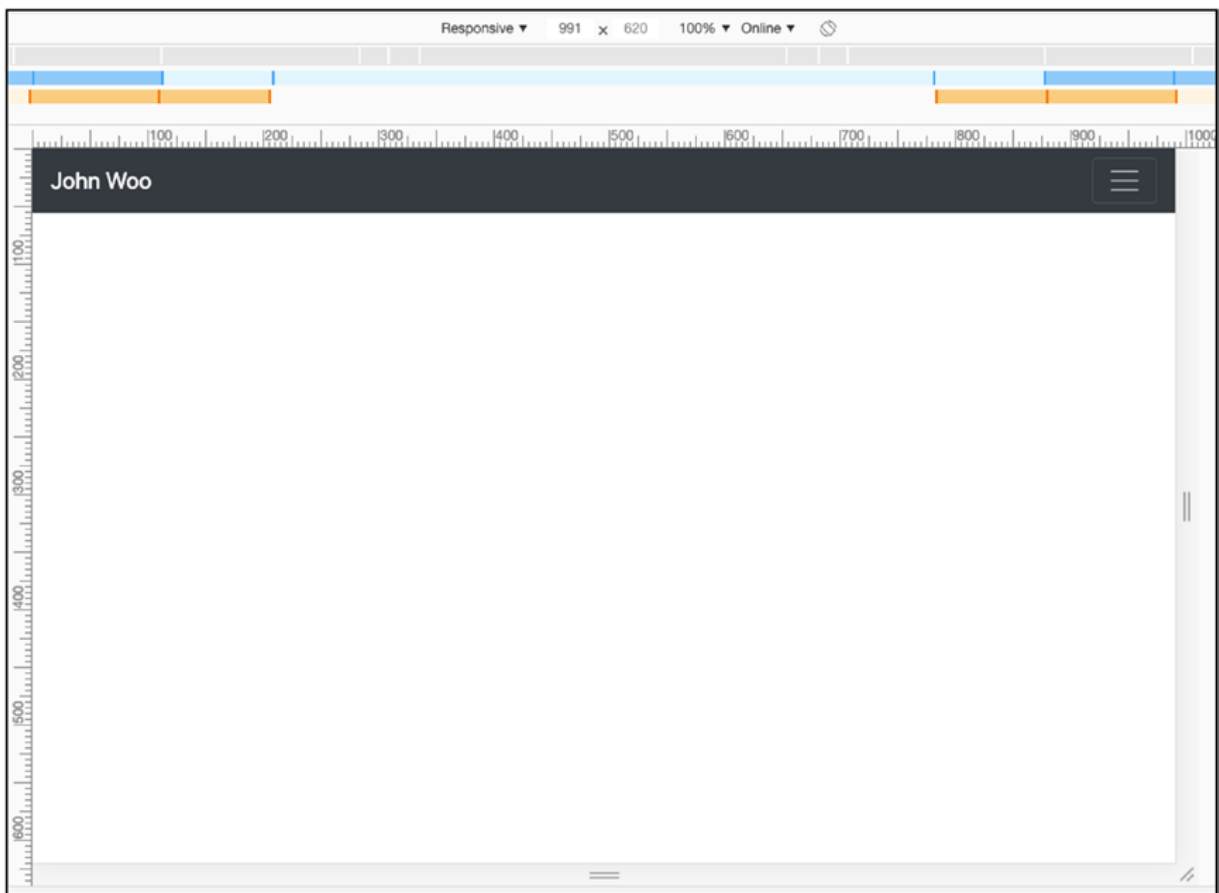


Figure 3-19. *Hamburger Icon Now Present*

Adding Behavior to the Button

If you try to click the hamburger button, nothing happens. You now need to add behavior to the button. When the button is clicked, the menu options unfold. You will do that by adding two `data-*` attributes to the button element:

- `data-toggle="collapse"`, which tells Twitter Bootstrap that this button is used to collapse/unfold a specific area of the HTML document
- `data-target="#navbar"`, which tells Twitter Bootstrap that the specific area that needs to be collapsed/unfolded is the one that is selected with the CSS selector `#navbar`. Of course, since there is no such element (with id `navbar`) in your HTML document, but the element that you want to fold/unfold is the container that holds the menu items list, you will add the `id` attribute with value `navbar` to this particular div. Let's do that (Listing 3-10).

Listing 3-10. Make the Button Toggle the Menu

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
  shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.
  com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
  Vkoo8x4CGs03+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
  crossorigin="anonymous">

  <!-- Custom CSS -->
  <link rel="stylesheet" href="stylesheets/main.css" type="text/css">

  <title>Listing 3-10</title>
</head>
```

```

<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">

    <a class="navbar-brand" href="#">John Woo</a>

    <button type="button" class="navbar-toggler" data-toggle="collapse"
      data-target="#navbar">
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbar">
      <ul class="nav navbar-nav">
        <li class="nav-item">
          <a class="nav-link" href="#">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#about">About</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#contact">Contact</a>
        </li>
      </ul>
    </div>

  </nav>

  <!-- Optional JavaScript -->
  <!-- jQuery first, then Popper.js, then Bootstrap JS -->
  <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
    integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
    yYfoRSJoZ+n" crossorigin="anonymous"></script>
  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
    popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
    zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
    bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
    aYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

If you save the preceding content and load the page on your browser, you will now see that the button toggles the menu on and off. In Figure 3-20, I display how it looks like with the menu expanded.

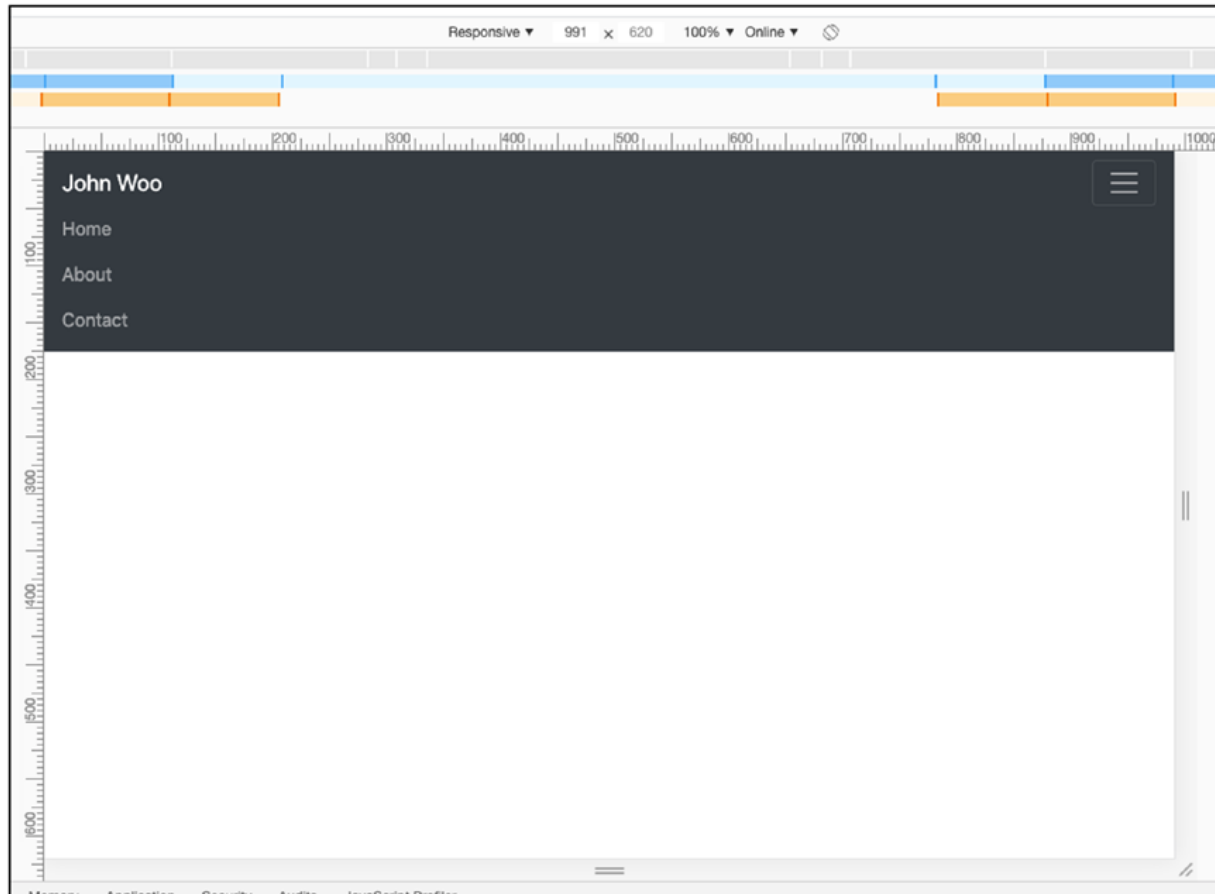


Figure 3-20. *Menu Expanded After Clicking the Button*

Final Touches to Our Navigation Bar

- fixed-top on nav element will fix the position of the navigation bar at the top of the page.
- You will also add some elements and attributes to make the whole navigation bar ready to be used with assistive technologies, for example, with screen readers.
- You will also add the class `active` to the first menu item, to indicate that you are on the page that this link corresponds to.

Here is the final version of your page with the navigation bar ready (Listing 3-11).

Listing 3-11. Final HTML Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
  shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.
  com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
  Vkoo8x4CGs03+Hh xv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
  crossorigin="anonymous">

  <!-- Custom CSS -->
  <link rel="stylesheet" href="stylesheets/main.css" type="text/css">

  <title>Listing 3-11</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark fixed-top">
    <a class="navbar-brand" href="#">John Woo</a>

    <button type="button" class="navbar-toggler" data-toggle="collapse"
    data-target="#navbar"
      aria-controls="navbar"
      aria-expanded="false"
      aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbar">
      <ul class="nav navbar-nav">
        <li class="nav-item">
```

```

        <a class="nav-link active" href="#">Home <span
        class="sr-only">(current)</span></a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="#about">About</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="#contact">Contact</a>
    </li>
</ul>
</div>
</nav>

<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoR5JoZ+n" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

If you save the preceding content and load the page on a display of 992px width, you will see Figure 3-21.

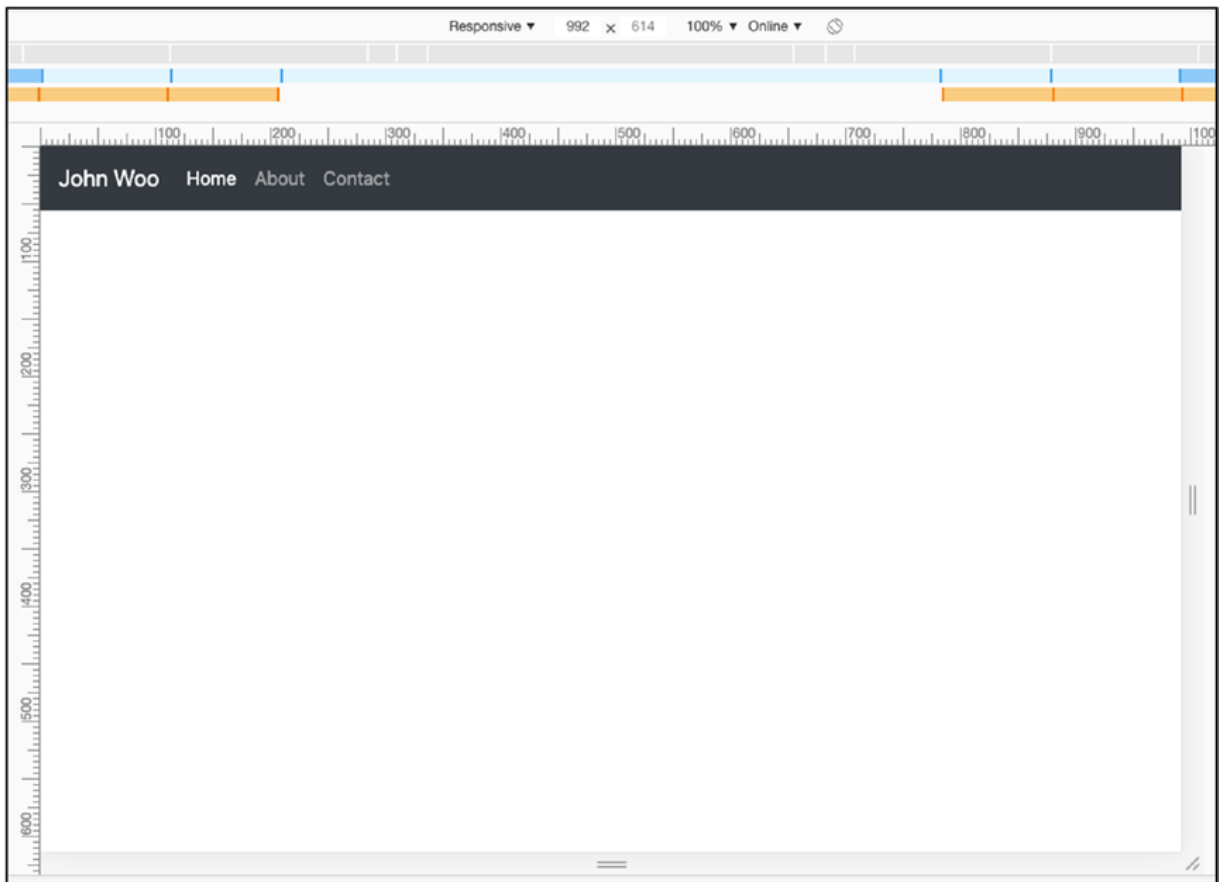


Figure 3-21. Final Page with a Navigation Bar on 992px

Your navigation bar works perfectly, and you have not written a single line of CSS or JavaScript code. Isn't it amazing? All thanks to Twitter Bootstrap.

Adding the Main Content

You have added/created the navigation bar. Let's now add the main content of your page (Listing 3-12).

Listing 3-12. We Add the Main Content

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
```

```

<meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no">

<!-- Bootstrap CSS -->
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.
com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
Vkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
crossorigin="anonymous">

<!-- Custom CSS -->
<link rel="stylesheet" href="stylesheets/main.css" type="text/css">

<title>Listing 3-12</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark fixed-top">
    <a class="navbar-brand" href="#">John Woo</a>

    <button type="button" class="navbar-toggler" data-toggle="collapse"
data-target="#navbar"
      aria-controls="navbar"
      aria-expanded="false"
      aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbar">
      <ul class="nav navbar-nav">
        <li class="nav-item">
          <a class="nav-link active" href="#">Home <span
class="sr-only">(current)</span></a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#about">About</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#contact">Contact</a>
        </li>
      </ul>
    </div>
  </nav>

```



```

        </ul>
    </div>
</nav>

<div class="container">

    <h1>John Woo Portfolio</h1>
    <p>Full-Stack Web Developer that can turn your ideas to fantastic
        responsive Web applications</p>

</div>

<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoR5JoZ+n" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1i9fktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

You have added a `div` with the class `container` below the `nav` element. Inside this new `div`, you have added an `h1` and a `p` element with some content inside them.

Let's save the content and load the page on the browser. You will see the following (Figure 3-22).

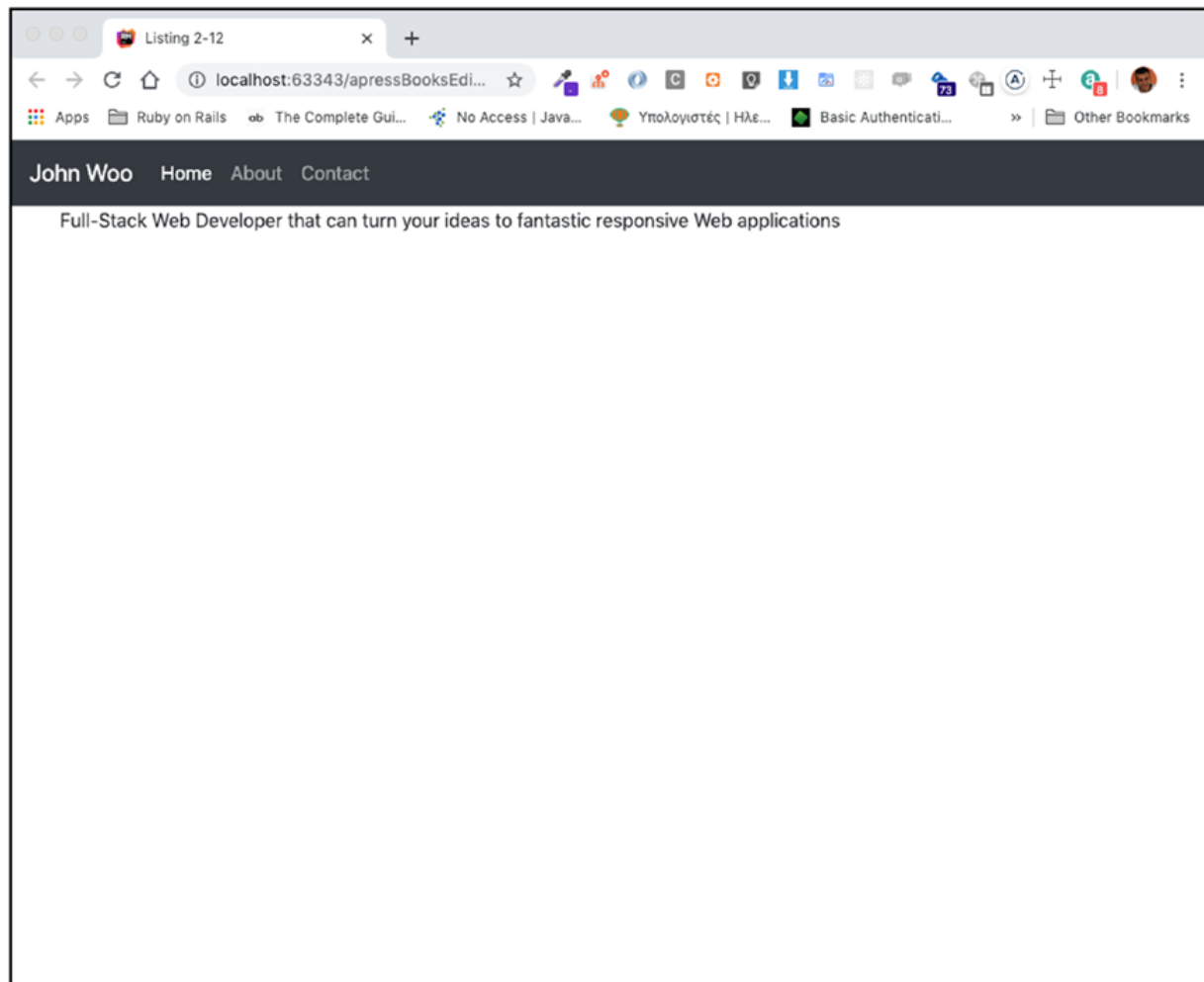


Figure 3-22. *Main Content Hidden Behind the Top Nav Bar*

You can see the problem, I guess. The h1 header is hidden behind the top navigation bar.

In order to resolve this, we will have to add some custom CSS. You will have to ask the body to position its actual content so many pixels below the top, as the height of the navigation bar. Figure 3-23 is depicting how you can find out the height of the top navigation bar, with the help of the developer tools.

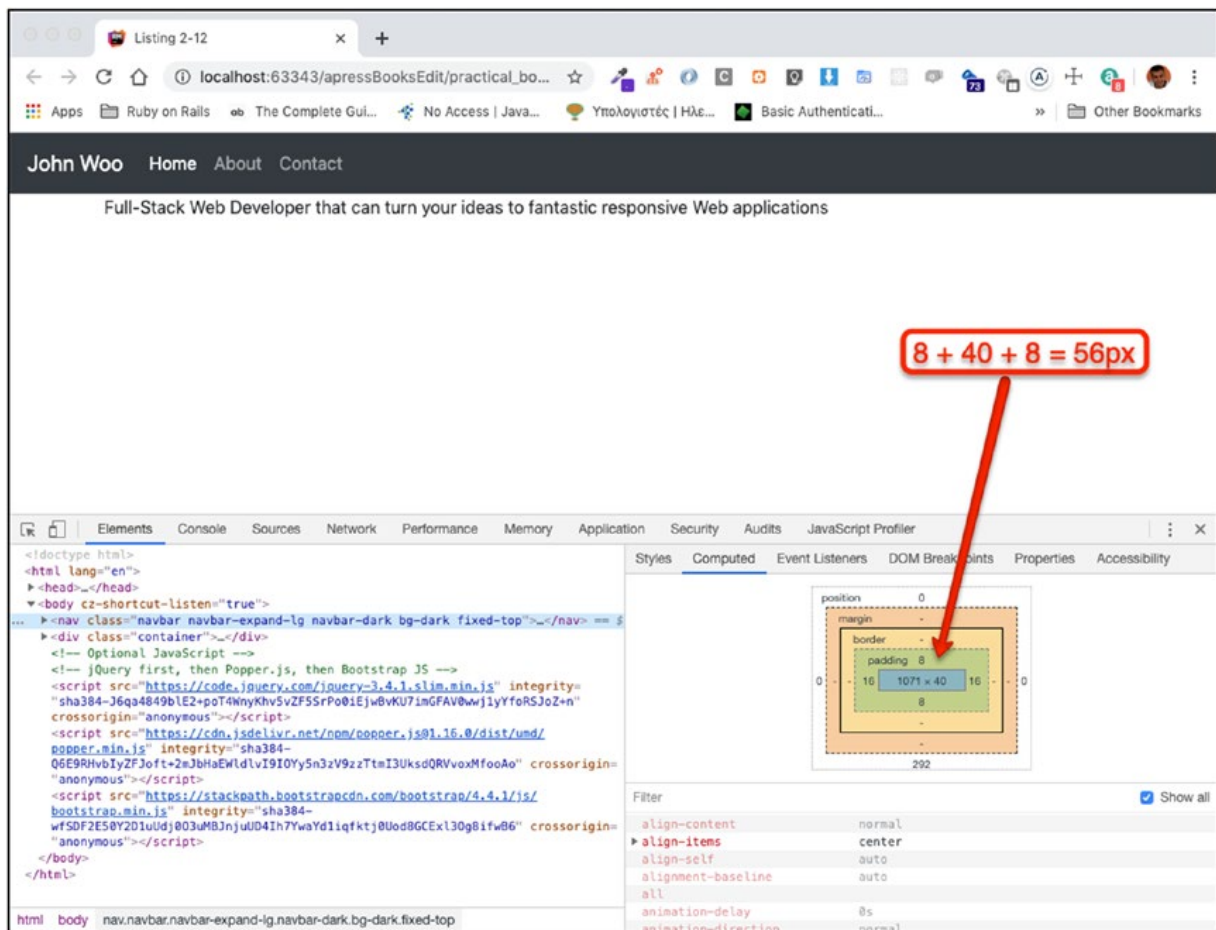


Figure 3-23. Calculate the Height of the Top Navigation Bar

As you can see from the developer tools, the height of the top navigation bar is 56px, calculated by adding the top and bottom paddings and content height.

With 56px at hand, let's add some content inside the stylesheets/main.css file (Listing 3-13).

Listing 3-13. Content of the main.css File

```
body {
  padding-top: 56px;
}
```

Now, if you load your page again (make sure that it references the stylesheets/main.css), it will look like the following (Figure 3-24).

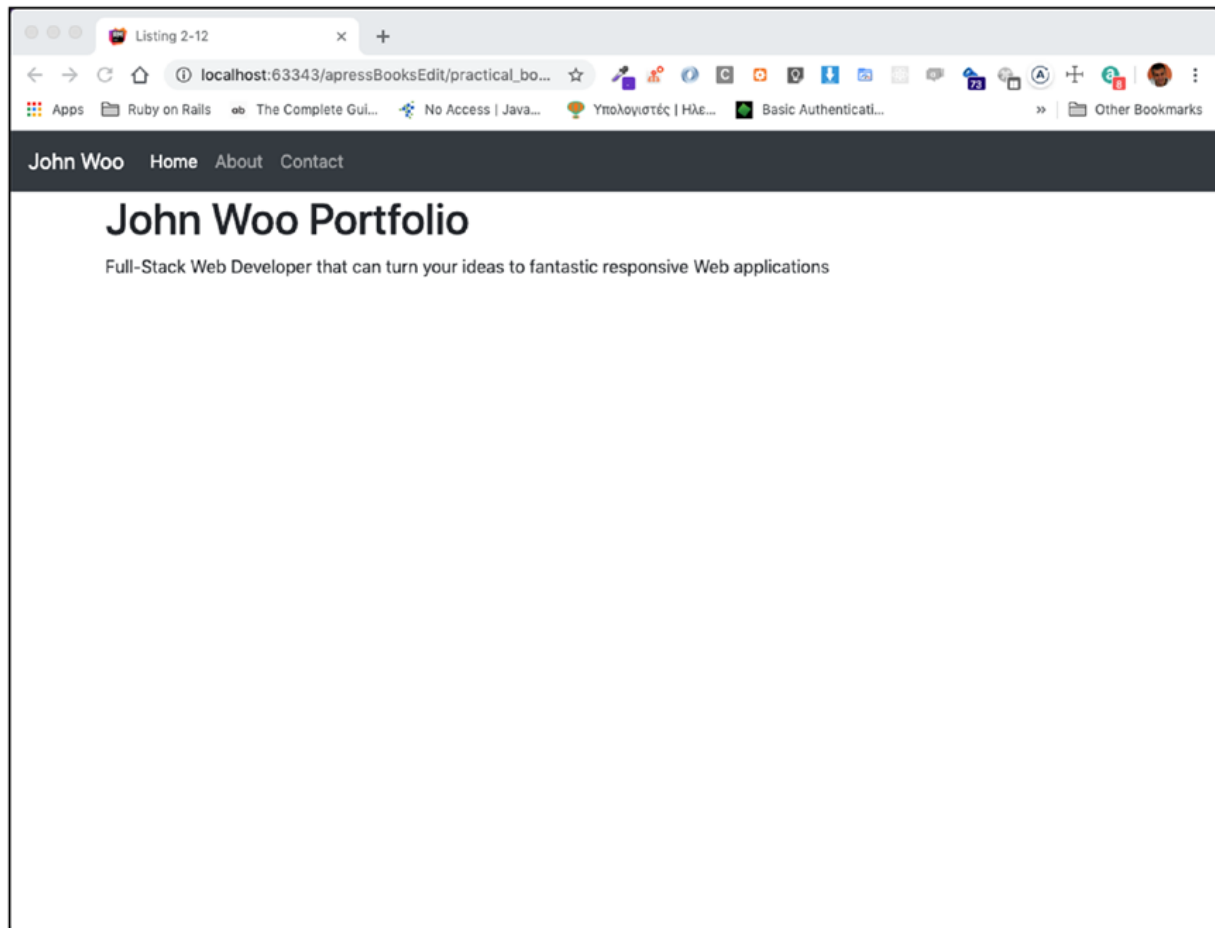


Figure 3-24. *h1 Is Now Visible*

Making the Paragraph Content Stand Out

There are some small differences to the original site. One is that the paragraph content is plain, normal-size text. You want to make it stand out. You will use the class `lead` for that (again provided to us by Twitter Bootstrap).

Make the `p` element have that class (Listing 3-14).

Listing 3-14. Make the Paragraph Stand Out

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
```

```

<meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no">

<!-- Bootstrap CSS -->
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.
com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
Vkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
crossorigin="anonymous">

<!-- Custom CSS -->
<link rel="stylesheet" href="stylesheets/main.css" type="text/css">

<title>Listing 3-14</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark fixed-top">
    <a class="navbar-brand" href="#">John Woo</a>

    <button type="button" class="navbar-toggler" data-toggle="collapse"
data-target="#navbar"
      aria-controls="navbar"
      aria-expanded="false"
      aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbar">
      <ul class="nav navbar-nav">
        <li class="nav-item">
          <a class="nav-link active" href="#">Home <span
class="sr-only">(current)</span></a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#about">About</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#contact">Contact</a>
        </li>

```

```

        </ul>
    </div>
</nav>

<div class="container">

    <h1>John Woo Portfolio</h1>
    <p class="lead">Full-Stack Web Developer that can turn your ideas
    to fantastic responsive Web applications</p>

</div>

<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoRSJoZ+n" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1iqfktj0Uod8GCExl30g8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

If you save and load your page again, you will see [Figure 3-25](#).

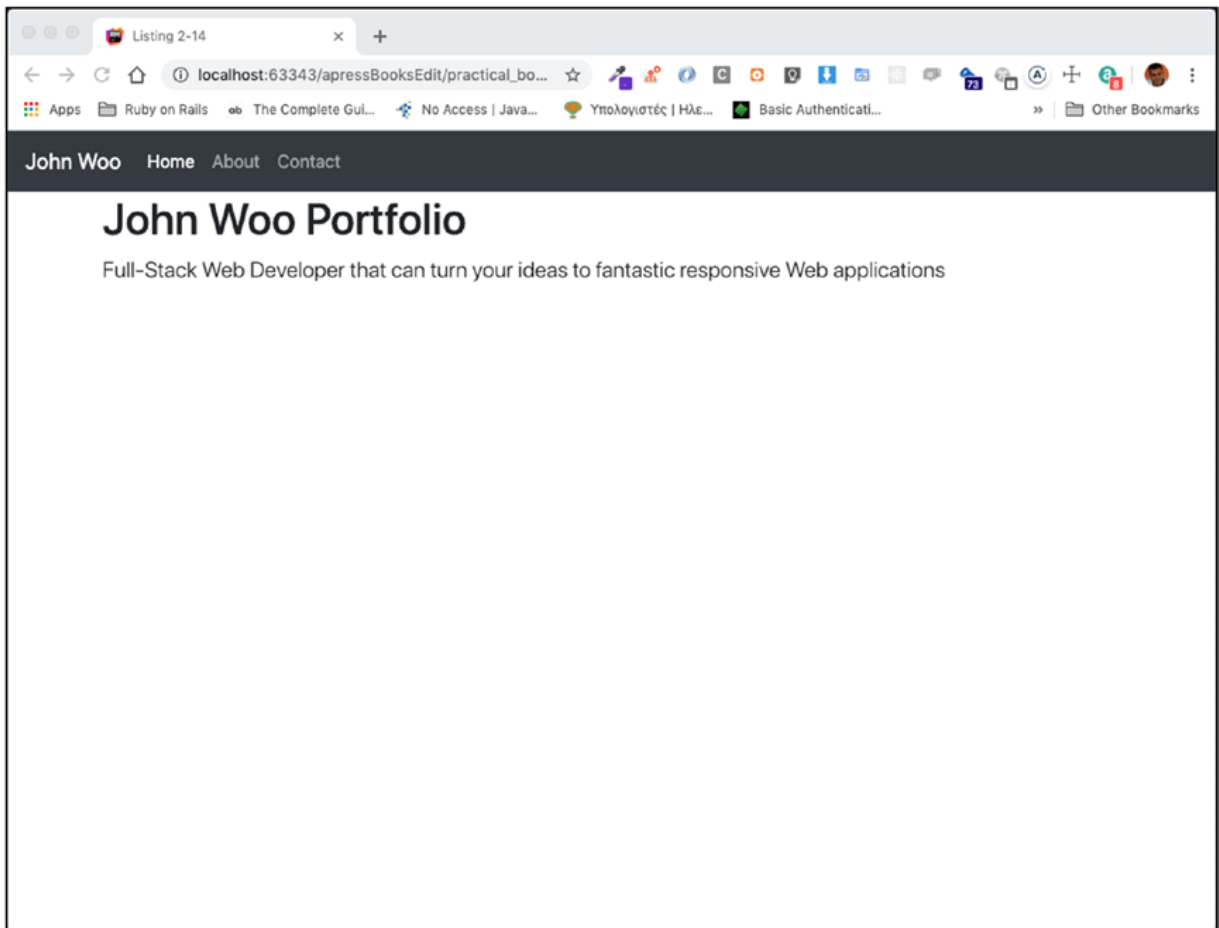


Figure 3-25. Paragraph Stands Out

Aligning Nav Bar Content with Page Main Content

So far, so good, but there is one more thing to fix. The content of the nav bar is not aligned with the main content of the page. See the problem in [Figure 3-26](#).

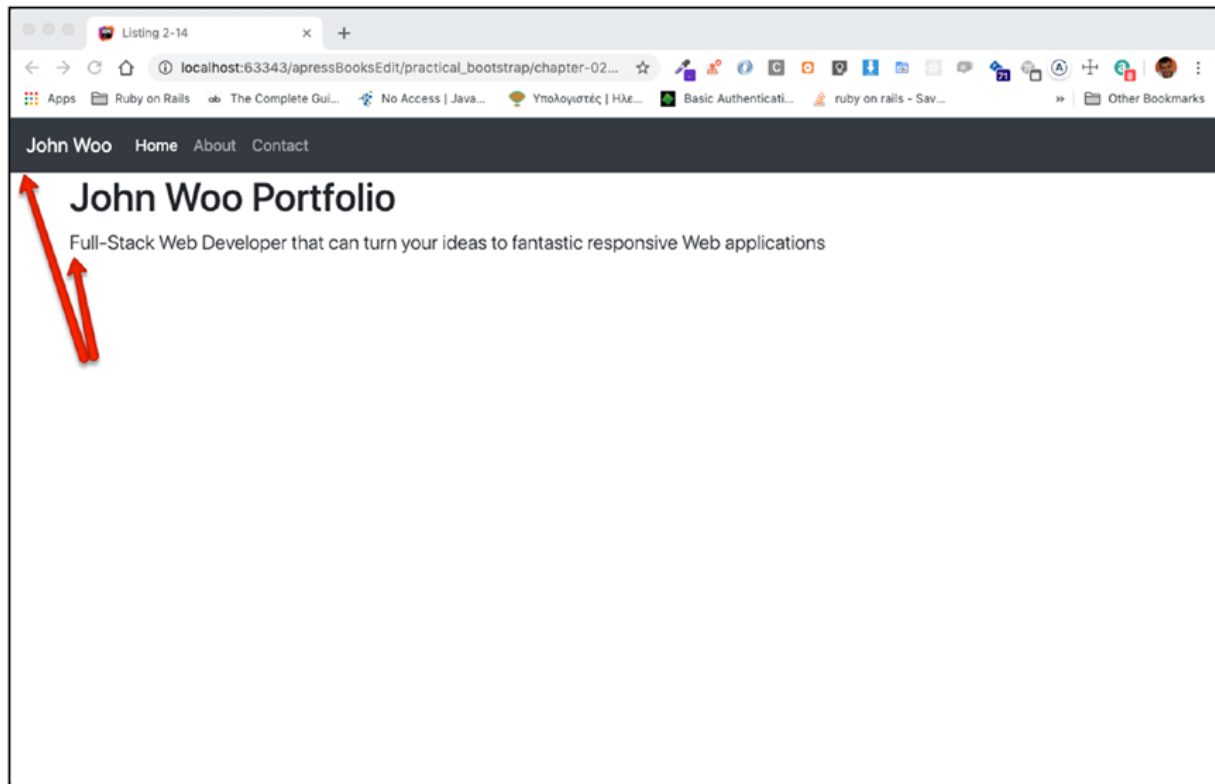


Figure 3-26. *Nav Bar Content Misaligned with Page Main Content*

You can fix this by wrapping the nav bar content into a div with class container, like you have done with the main content of the page (Listing 3-15).

Listing 3-15. Wrap Nav Bar Content into a Container

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1,
  shrink-to-fit=no">

  <!-- Bootstrap CSS -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.
  com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-
  Vkoo8x4CGs03+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
  crossorigin="anonymous">
```



```

<!-- Custom CSS -->
<link rel="stylesheet" href="stylesheets/main.css" type="text/css">

<title>Listing 3-15</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark fixed-top">
    <div class="container">
      <a class="navbar-brand" href="#">John Woo</a>

      <button type="button" class="navbar-toggler" data-
toggle="collapse" data-target="#navbar"
      aria-controls="navbar"
      aria-expanded="false"
      aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
      </button>

      <div class="collapse navbar-collapse" id="navbar">
        <ul class="nav navbar-nav">
          <li class="nav-item">
            <a class="nav-link active" href="#">Home <span
class="sr-only">(current)</span></a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="#about">About</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="#contact">Contact</a>
          </li>
        </ul>
      </div>
    </div>
  </nav>

```

```

<div class="container">

    <h1>John Woo Portfolio</h1>
    <p class="lead">Full-Stack Web Developer that can turn your ideas
    to fantastic responsive Web applications</p>

</div>

<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js"
integrity="sha384-J6qa4849b1E2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1
yYfoR5Joz+n" crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/
popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3
zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/
bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj003uMBJnjuUD4Ih7Yw
aYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
</body>
</html>

```

If you save and load this page on your browser, you will see the following (Figure 3-27).

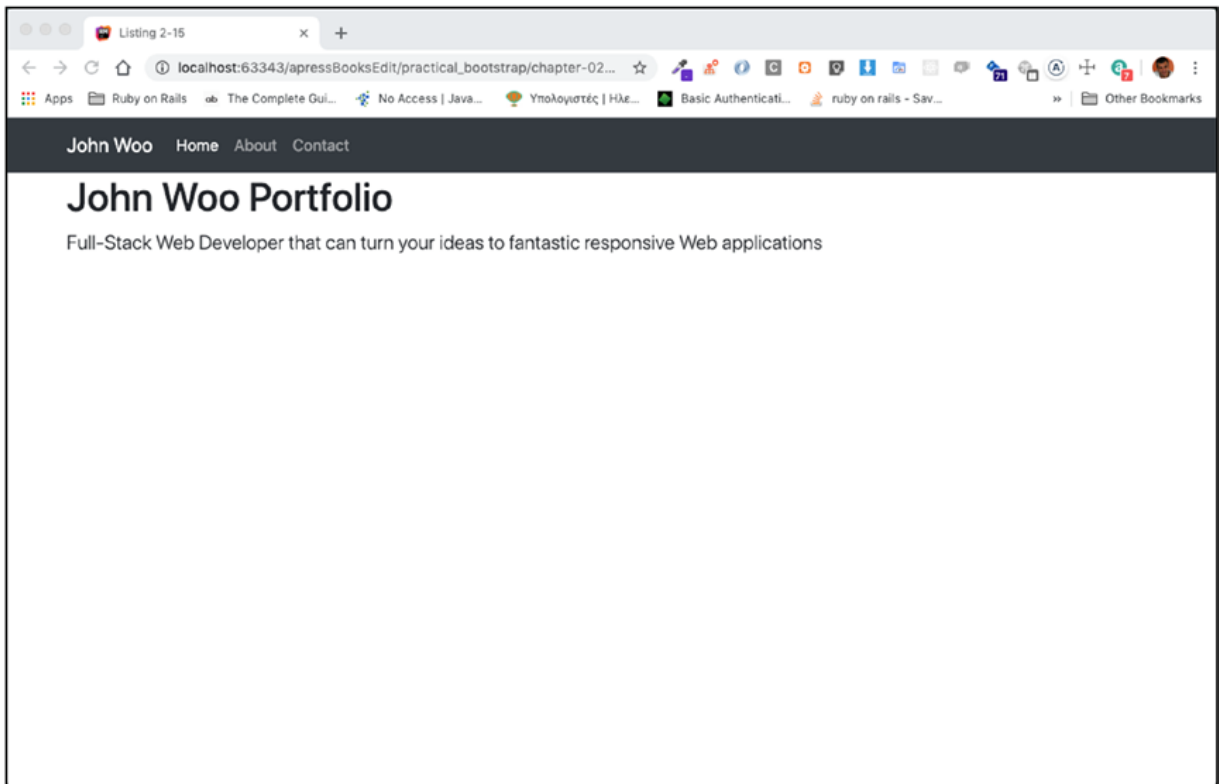


Figure 3-27. *Nav Bar Content Aligned with Page Main Content*

Well done! This matches the original page that you wanted to build.

Closing Notes

You have created a website with a responsive navigation bar, without writing any single piece of CSS code. Twitter Bootstrap has provided everything to you. You only had to use the correct classes (and in one case, two `data-*` attributes).

Tasks and Quizzes

TASK DETAILS

1. You need to implement a web page with a responsive navigation bar, like the following (Figure 3-28).

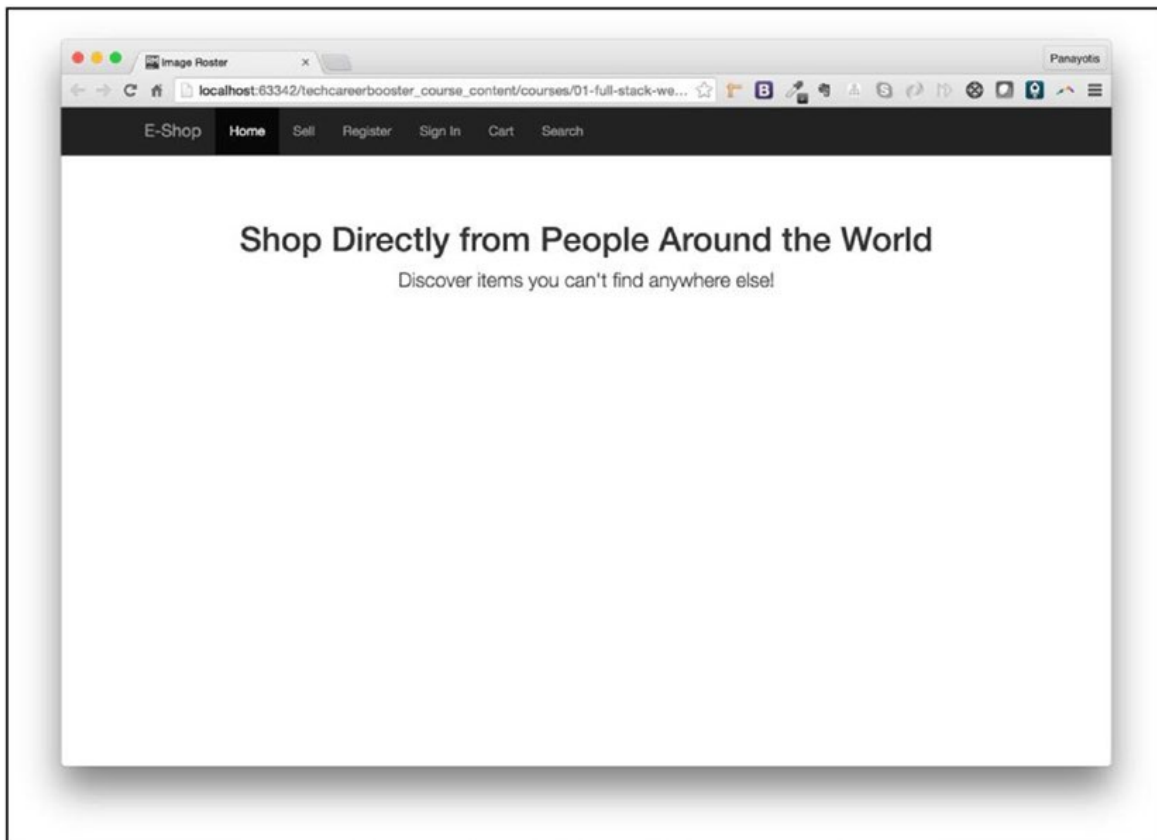


Figure 3-28. Task Project with a Navigation Bar

2. Watch out for the following:
 1. The main content should be centered.
 2. There is a lot of blank space above the main content and below the navigation bar. Make sure that you have the same on your page.

3. The navigation bar should be responsive, and when you load your page on small displays, the menu items should be hidden. In their place, a hamburger icon should be displayed. The icon, when clicked, will unfold the list with the menu options.

Good luck!

Key Takeaways

Congratulations! You now know how to implement a responsive navigation bar. Here is a list of the key things you learned:

- How to implement a navigation bar with menu options that are hidden on small devices and visible on large devices.
- For small devices, you have implemented a button that when clicked toggles on/off the menu with the options.
- You have also learned how to color the navigation bar and how to position it.
- You have learned how to put content on the main page without it being hidden by the top navigation bar.
- You have learned how to make a paragraph stand out.

In the following chapter, you are going to get acquainted with the most popular Twitter Bootstrap components, like buttons, badges, tabs, and so on.