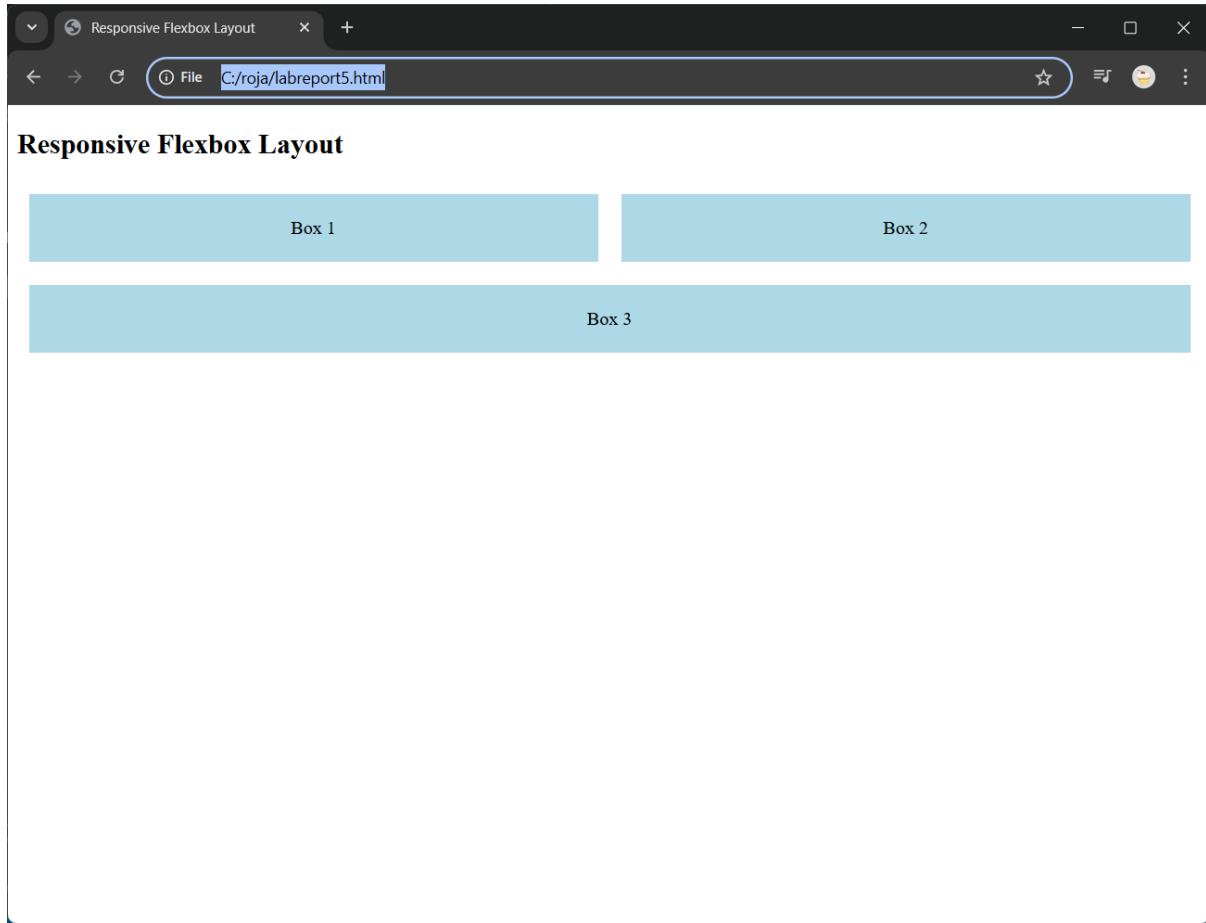


STUDENT NAME	R.ROJA	
STUDENT REGISTRATION NUMBER	251U1R2064	CLASS: CSE(AIML)
PROGRAM	UG	YEAR and TERM: 1 st year & 1 st term
SUBJECT NAME	HTML	
NAME OF THE ASSESSMENT	Reflective lab journal-5	
DATE OF SUBMISSION	25.10.25	

1.write a program on Responsive web design:

```
html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Responsive Flexbox Layout</title>
<style>
.container {
  display: flex;
  flex-wrap: wrap;
}
.box {
  flex: 1 1 300px; /* Shrinks or grows as needed */
  margin: 10px;
  padding: 20px;
  background-color: lightblue;
  text-align: center;
}
</style>
</head>
<body>
<h2>Responsive Flexbox Layout</h2>
<div class="container">
  <div class="box">Box 1</div>
  <div class="box">Box 2</div>
  <div class="box">Box 3</div>
</div>
</body>
</html>
```

Output:



Explanation:

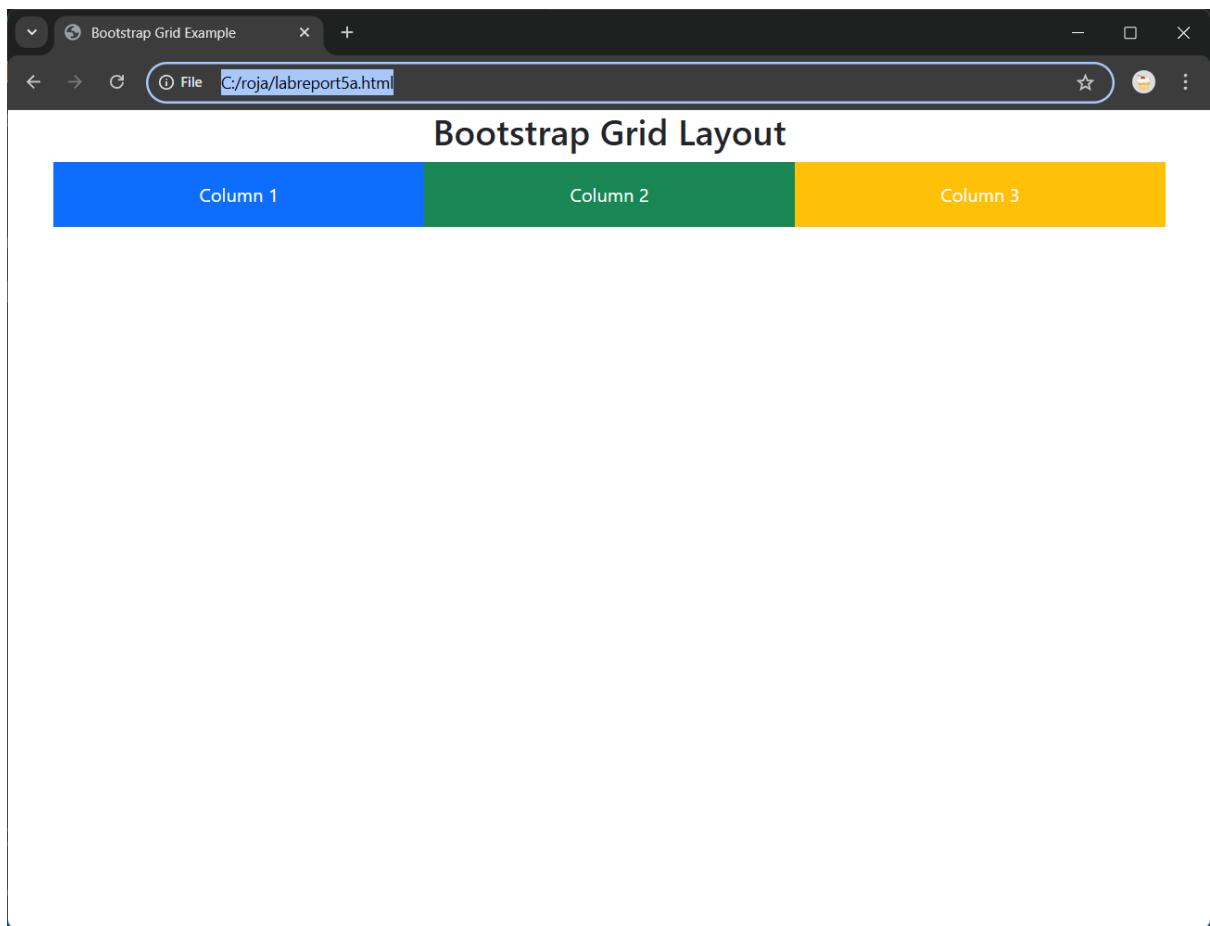
- The `flex-wrap` property allows boxes to wrap onto new lines on smaller screens.
- Each box adjusts its size automatically (`flex: 1 1 300px`).
- The `<!DOCTYPE html>` declaration defines the document as an HTML5 webpage.
- The `<meta name="viewport" content="width=device-width, initial-scale=1.0">` tag ensures the page scales properly on mobile devices.
- The `.container` class is styled with `display: flex`, which turns it into a **flex container**, allowing flexible layout of its child elements.
- The `flex-wrap: wrap` property makes the boxes move to the next line if there isn't enough space horizontally.
- Each `.box` inside the container acts as a **flex item**, which can grow or shrink depending on the available space.
- The `flex: 1 1 300px` rule means each box will take up equal space but will not shrink smaller than **300px** in width.
- The `margin: 10px` adds spacing around each box to prevent them from touching.
- The `padding: 20px` adds inner spacing, making the box content look neat and readable.
- The `background-color: lightblue` visually distinguishes each box and makes them easier to see.

- This design is **responsive** because when the screen is resized, the boxes automatically rearrange to fit the available width, maintaining a clean and flexible layout across devices.

2. Write a program on bootstrap grid:

```
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Bootstrap Grid Example</title>
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
<div class="container text-center">
<h2>Bootstrap Grid Layout</h2>
<div class="row">
<div class="col-md-4 bg-primary text-white p-3">Column 1</div>
<div class="col-md-4 bg-success text-white p-3">Column 2</div>
<div class="col-md-4 bg-warning text-white p-3">Column 3</div>
</div>
</div>
</body>
</html>
```

Output:



Explanation:

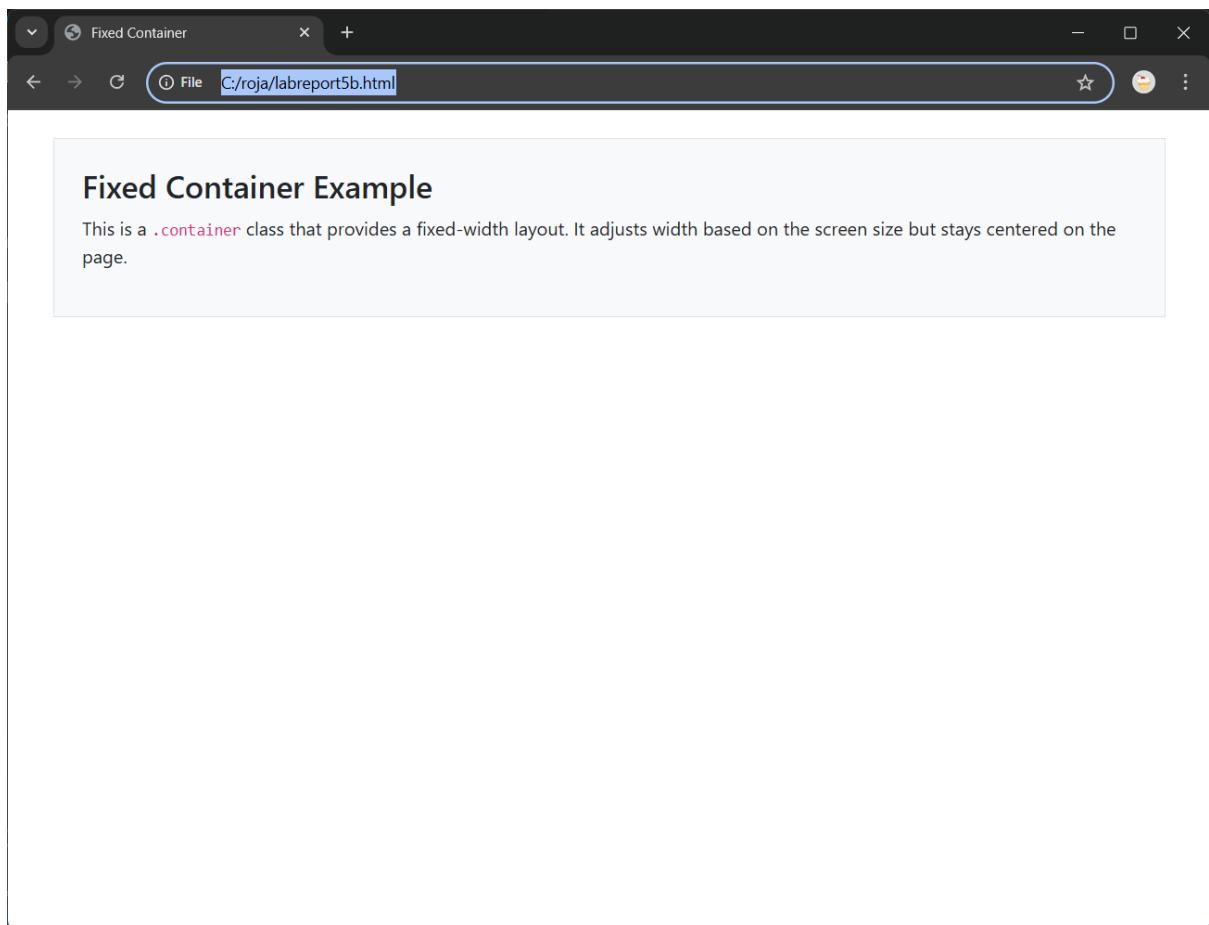
1. The `<!DOCTYPE html>` declaration defines the page as an HTML5 document.
2. The `<meta name="viewport" content="width=device-width, initial-scale=1">` tag ensures the layout adapts to different screen sizes and devices.
3. The Bootstrap CSS file is linked via CDN to use its built-in responsive classes and components.
4. The `<div class="container">` class provides a fixed-width, responsive container that centers the content on the page.
5. Inside the container, the `<div class="row">` acts as a horizontal wrapper that holds the columns.
6. Each `<div class="col-md-4">` defines one column that occupies 4 out of 12 grid spaces on medium and larger screens.
7. Because $4 + 4 + 4 = 12$, all three columns fit perfectly in one row.
8. The color classes like `bg-primary`, `bg-success`, and `bg-warning` apply Bootstrap's predefined background colors.
9. The `text-white` class makes the text color white for better contrast against the background.
10. The layout is **responsive**, so when viewed on smaller devices (below "md" size), the three columns automatically stack vertically for a clean mobile-friendly design.

3.write a program on container:

```
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Fixed Container</title>
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
<div class="container bg-light border mt-4 p-4">
<h3>Fixed Container Example</h3>
<p>This is a <code>.container</code> class that provides a fixed-width
layout. It adjusts width based on the screen size but stays centered on the
page.</p>
</div>
</body>
```

```
</html>
```

Output:



Explanation:

1. The `<!DOCTYPE html>` declaration defines the document as HTML5.
2. The Bootstrap stylesheet is linked using the official **CDN** to apply Bootstrap classes.
3. The `<div class="container">` creates a **fixed-width container** that adjusts based on screen size but has maximum width limits.
4. The `<div class="container-fluid">` creates a **full-width container** that always spans the entire viewport width.
5. The `<div class="container-sm">` is a **responsive container**, full-width until the screen reaches the small (sm) breakpoint.
6. Each container has a `bg-*` class (like `bg-light`, `bg-info`, `bg-warning`) to add background color.

7. The p-4 and my-4 classes add padding and margin spacing for better visual separation.
8. The border class outlines each container, making the boundaries visible.
9. The text classes like text-white improve readability on colored backgrounds.
10. This program shows how **Bootstrap containers** manage layout widths across different devices, forming the base structure for responsive web pages.

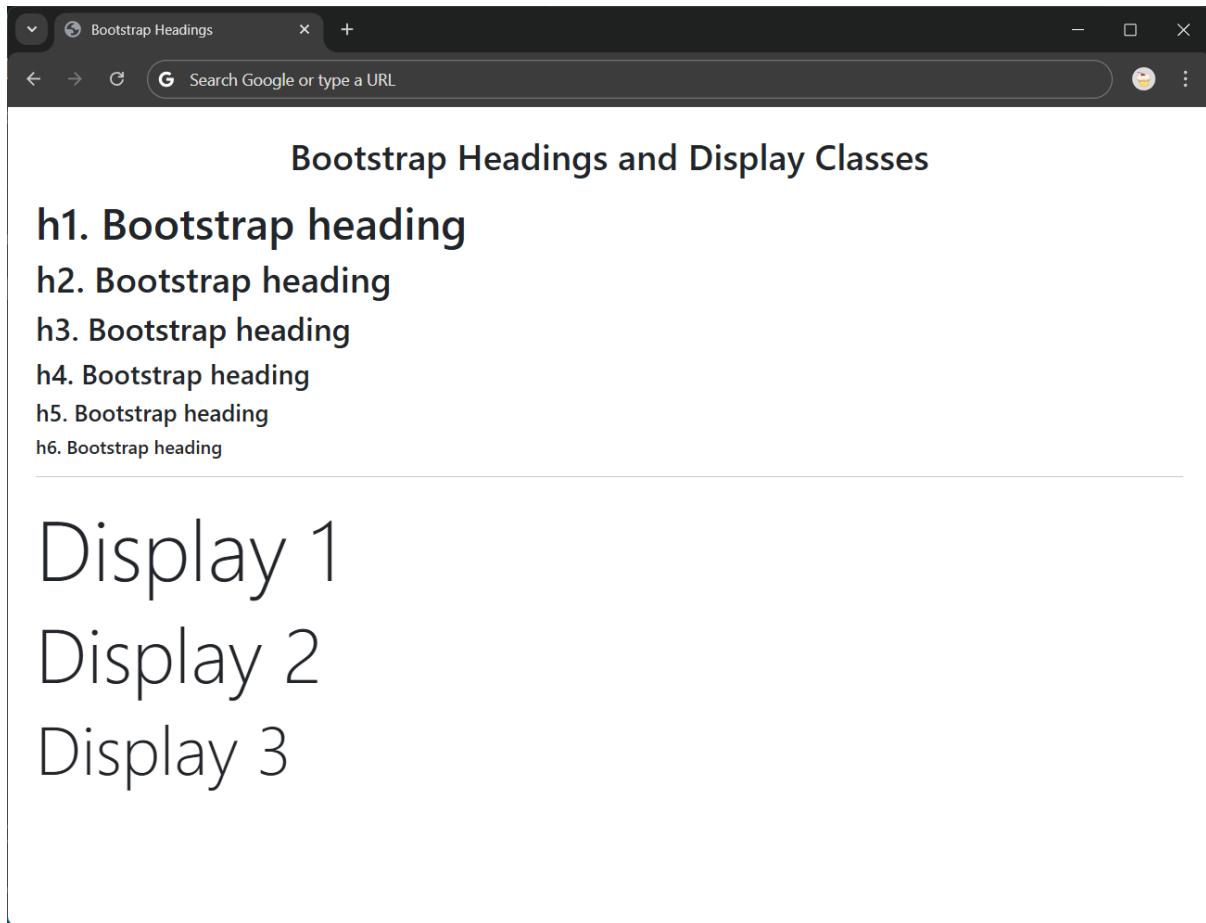
4. Write a program on typography:

```
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Bootstrap Headings</title>
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body class="p-4">
  <h2 class="text-center mb-3">Bootstrap Headings and Display Classes</h2>

  <h1>h1. Bootstrap heading</h1>
  <h2>h2. Bootstrap heading</h2>
  <h3>h3. Bootstrap heading</h3>
  <h4>h4. Bootstrap heading</h4>
  <h5>h5. Bootstrap heading</h5>
  <h6>h6. Bootstrap heading</h6>

  <hr>
  <h1 class="display-1">Display 1</h1>
  <h2 class="display-2">Display 2</h2>
  <h3 class="display-3">Display 3</h3>
</body>
</html>
```

Output:



Explanation:

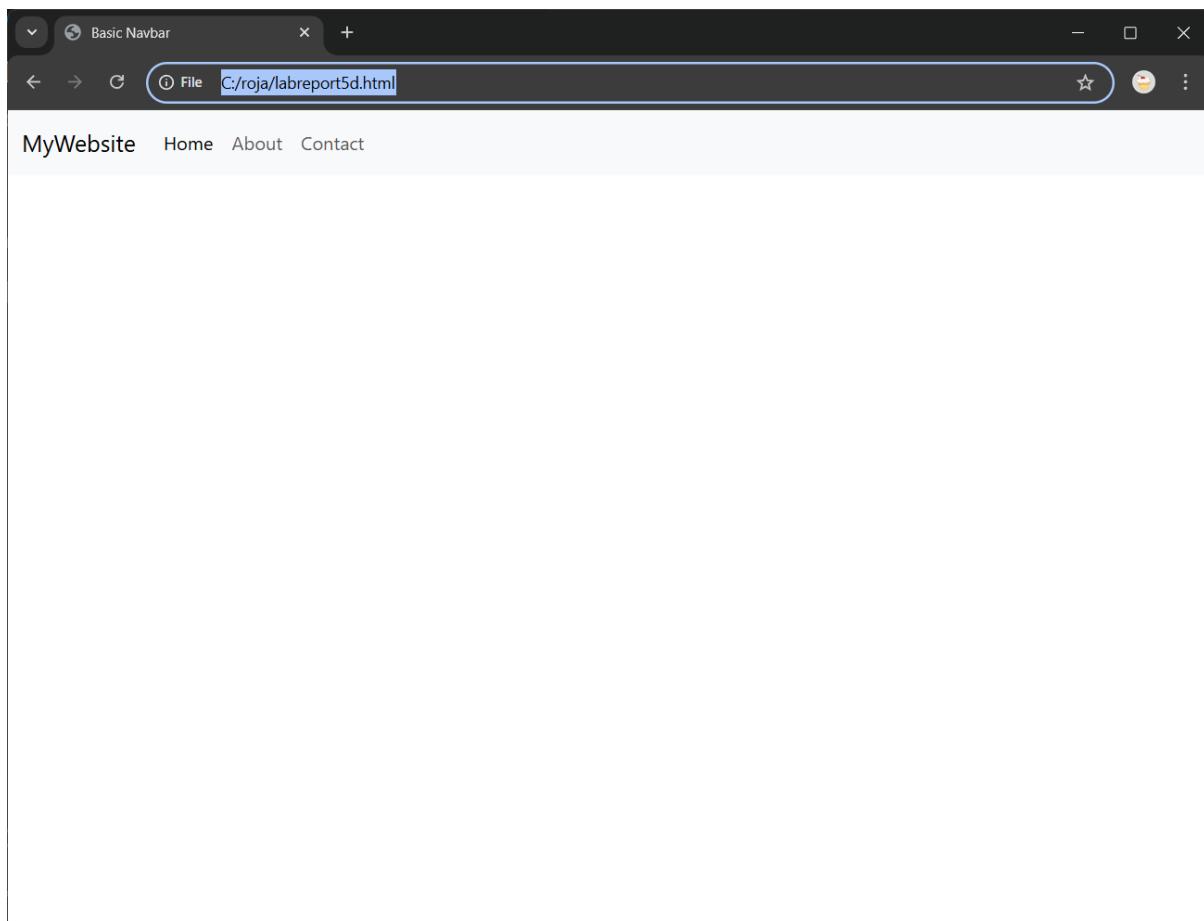
- Bootstrap provides default styles for HTML headings (h1–h6).
- The .display-* classes (from .display-1 to .display-6) create larger, more prominent text.
- These are commonly used for titles or hero text in web pages.

5.write a program on bootstrap-navbar:

```
html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Basic Navbar</title>
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
```

```
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">MyWebsite</a>
    <div class="collapse navbar-collapse">
      <ul class="navbar-nav">
        <li class="nav-item"><a class="nav-link active" href="#">Home</a></li>
        <li class="nav-item"><a class="nav-link" href="#">About</a></li>
        <li class="nav-item"><a class="nav-link" href="#">Contact</a></li>
      </ul>
    </div>
  </div>
</nav>
</body>
</html>
```

Output:



Explanation:

The .navbar class creates a navigation bar.

.navbar-light bg-light gives a light background with dark text.

navbar-brand shows the site name or logo.

.navbar-nav and .nav-item define the navigation links

The <!DOCTYPE html> tag defines the document as an HTML5 web page.

The <meta name="viewport" content="width=device-width, initial-scale=1"> ensures the page is responsive and adapts to all screen sizes.

The Bootstrap CSS file is linked through a CDN, allowing use of built-in Bootstrap styles.

The <nav> element with class navbar creates a responsive navigation bar section.

The class navbar-expand-lg allows the navbar to expand horizontally on large screens and stack vertically on smaller devices.

The class navbar-light bg-light sets a light background color with dark text for good contrast.

The <div class="container-fluid"> ensures the navbar spans the full width of the screen.

The defines the website name or logo displayed on the left side.

The unordered list <ul class="navbar-nav"> contains the navigation links, each inside a <li class="nav-item">.