



Daffodil
International
University

Semester Project Spring 2023

Course code: CIS133

Course Title: Website Development Essentials

Submitted to:

Mr. Md. Faruk Hosen

Lecturer, Department of CIS

Daffodil International University

Submitted by:

Md Tamim Hossain

ID: 0242310012091040

Section: B

Department of Computing Information System

Daffodil International University

Table of Contents

Acknowledgement	3
Introduction.....	4
Task-1	
Answer to Q. a	5
Answer to Q. b	8
Answer to Q. c	9
Answer to Q. d	11
Task-2	
Answer to Q. a.....	14
Answer to Q. b	16
Task-3	
Answer to Q. a	20
Answer to Q. b	22
Answer to Q. c	25
Answer to Q. d	28
Conclusion	30

Acknowledgment

I would like to express my gratitude to my instructor, mentors, classmates, and family for their unwavering support and guidance during the completion of this assignment on Web Development Essentials. Their contributions have been invaluable in enhancing my understanding of the subject matter and enriching my learning experience. Thank you for your assistance and encouragement.

Introduction

In today's digital age, having an online presence is crucial for freelancers and entrepreneurs to showcase their skills and attract new clients. One of the best ways to do so is by creating a portfolio website that highlights their expertise and services. In this assignment, we will explore the process of building a portfolio website, using the example of Snigdho, a freelance web developer who wants to create a website that showcases his skills and attracts new clients. We will examine the steps involved in building a portfolio website, including defining goals, planning content, designing the layout, using HTML, CSS, and Bootstrap, testing and launching the website, and promoting it through SEO techniques and paid advertising. By the end of this assignment, you will have a comprehensive understanding of the steps involved in building a portfolio website and how it can benefit freelancers and entrepreneurs in their careers.

TASK-1

Answer to Question No: a

Home

The Home/Introduction section is usually the first section of a website that a visitor sees, and it should provide a brief overview of what the website is about. For Snigdho's portfolio website, this section could start with a headline that catches the visitor's attention and clearly states what the website is about. As like my website " I am a software Developer".

About me:

The About Me section is where Snigdho introduces himself and his background. It's an opportunity for him to showcase his personality and unique selling points, and give visitors a sense of who he is as a person and as a web developer

Skills:

The Skills section is where Snigdho can list his technical skills and expertise. He can provide a comprehensive list of the programming languages, software, and platforms he is proficient in, along with his level of proficiency in each. He could also include any relevant

certifications or training he has received, and highlight any areas where he excels or has a particular passion

Resume:

The Resume section is where Snigdho can provide a more detailed overview of his work history, education, and achievements. He could include a downloadable PDF of his resume, or provide a summary of his experience and skills. This section should be easy to scan and visually appealing, with clear headings and bullet points

Contact me:

The Contact Me section is where Snigdho provides his contact information and encourages visitors to get in touch. This section should include a contact form that visitors can use to send Snigdho a message directly from the website, as well as his email address, phone number, and social media profiles.

He could also include a map showing his location, and his availability for work. The Contact Me section should be easy to find and visually prominent, so visitors can easily get in touch if they're interested in working with Snigdho.

Visual Diagram



Answer to Question No: b

To structure a portfolio website using HTML, there are several HTML elements that can be used to organize the content and make it more accessible to users. Here's an example of how the HTML structure of a portfolio website might look

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Snigdho's Portfolio</title>
7 </head>
8 <body>
9   <header>
10     <nav>
11     </nav>
12   </header>
13   <main>
14     <section id="about">
15     </section>
16     <section id="services">
17     </section>
18     <section id="portfolio">
19     </section>
20     <section id="testimonials">
21     </section>
22     <section id="contact">
23     </section>
24   </main>
25   <footer>
26   </footer>
27 </body>
28 </html>
```


Answer to Question No: c

Here are some selectors that have been used on the website:

- Element selectors:

These are the most basic type of selector and select all elements of a given type. For example, `p` would select all paragraphs on the page, while `h1` would select all level one heading. These selectors are identified by the name of the element (such as `p` or `h1`).

Class selectors:

These selectors target elements with a specific class name. They are identified by a period (".") followed by the name of the class. For example, `.header` would target all elements with the class "header".

ID selectors:

These selectors target elements with a specific ID. They are identified by a pound sign("#") followed by the name of the ID. For example, `#about` would target the element with the ID "about".

Descendant selectors:

These selectors target elements that are descendants of a specific parent element. They are identified by whitespace between the parent element and the child element. For example, `header nav` would target all `nav` elements that are inside a `header` element.

Child selectors:

These selectors target elements that are direct children of a specific parent element. They are identified by the ">" symbol between the parent element and the child element. For example, ``main > section`` would target all ``section`` elements that are direct children of the ``main`` element.

Answer to Question No: d

CSS offers several ways to add interactivity and animations to a website. Here are some techniques that can be used to create hover effects and transitions:

1. Hover effects:

Hover effects are triggered when the user hovers over an element with their mouse. They can be used to change the appearance of an element or reveal additional information. To create a hover effect, we can use the `:hover` pseudo-class selector in CSS.

For example, let's say we want to change the background color of a button when the user hovers over it. We can write the following CSS code:

```
...  
button: hover {  
  background-color: #ff0000;  
}  
...
```

This code will change the background color of the button to red when the user hovers over it with their mouse.

2. Transitions:

Transitions allow us to add animation effects to CSS properties when they change over time. For example, we can use transitions to animate the color change of a button when the user hovers over it.

To add a transition, we can use the `transition` property in CSS. We can specify which property we want to animate (such as `background-color` or `color`), how long we want the animation to take (in seconds or milliseconds), and what type of animation we want (such as `ease-in-out` or `linear`).

For example, let's say we want to add a smooth color transition to a button when the user hovers over it. We can write the following CSS code:

```
...
```

```
button {  
  background-color: #008CBA;  
  transition: background-color 0.5s ease-in-out;  
}
```

```
button:hover {  
  background-color: #ff0000;  
}
```

```
...
```

In this code, we've set a transition on the `background-color` property of the button. When the user hovers over the button, the background color will change smoothly from blue to red in over half a second.

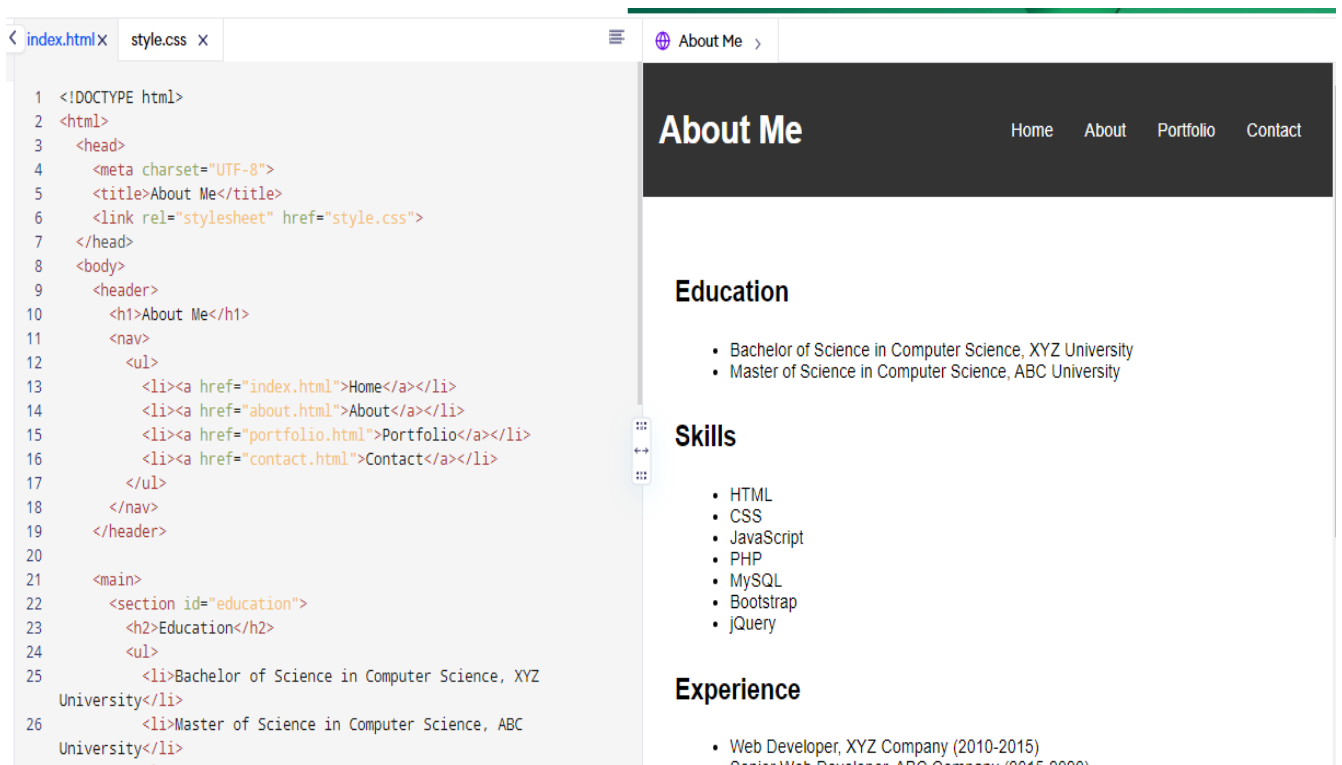
These are just a few examples of how CSS can be used to add interactivity and animations to a website. By combining these techniques with other CSS properties and selectors, we can create complex and engaging user experiences on our web pages.

Task-2

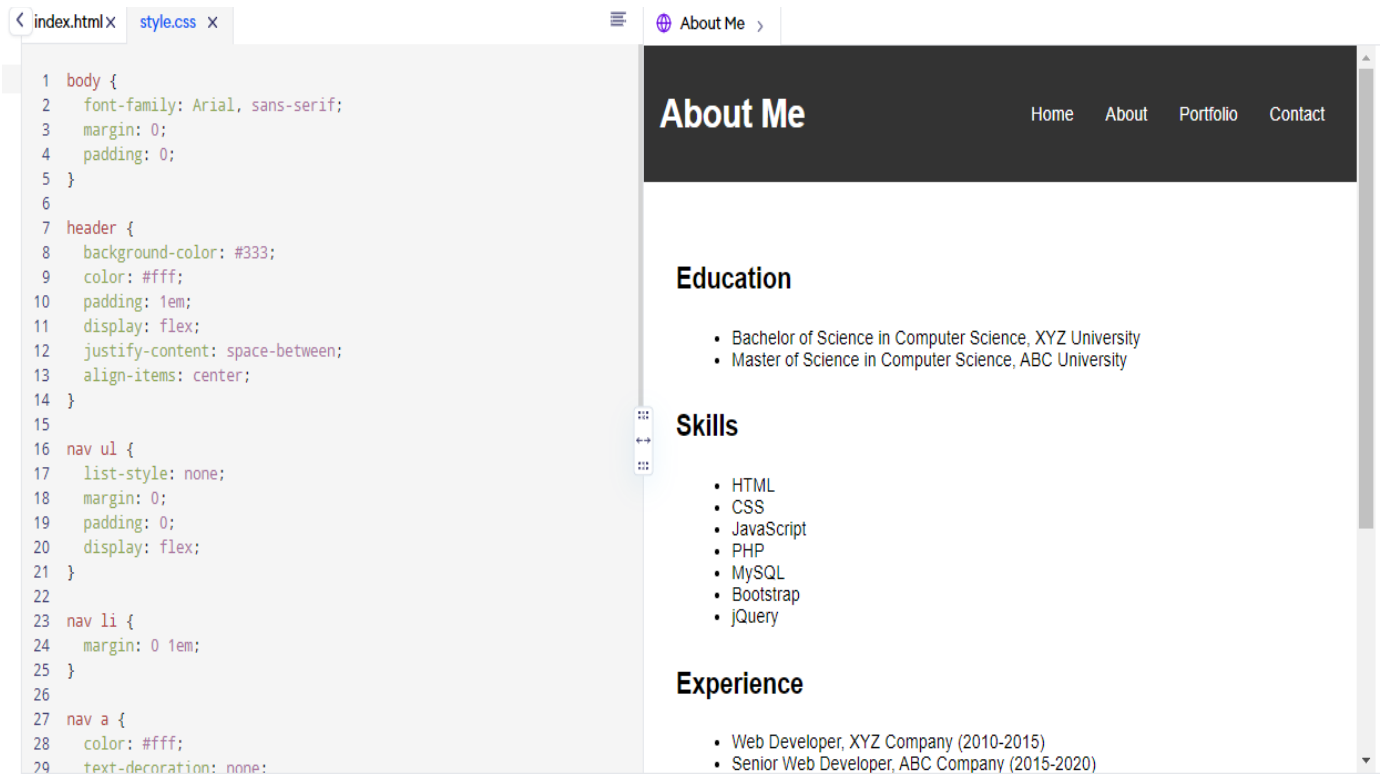
Answer to Question no: a

Here is the html and css code for creating about me page:

HTML code and Corresponding view:



CSS code and Corresponding view:



Answer to Question no:b

The Contact Me section is where Snigdho provides his contact information and encourages visitors to get in touch. This section should include a contact form that visitors can use to send Snigdho a message directly from the website, as well as his email address, phone number, and social media profiles.

He could also include a map showing his location, and his availability for work. The Contact Me section should be easy to find and visually prominent, so visitors can easily get in touch if they're interested in working with Snigdho.

Here is an example code for contact page:

```
<div class="contact-info">
  <h2>Contact Information</h2>
  <p><i class="fa fa-map-marker"></i> 123 Main Street, Anytown,
USA</p>
  <p><i class="fa fa-phone"></i> (555) 555-5555</p>
  <p><i class="fa fa-envelope"></i> info@example.com</p>
  <div class="social-media">
    <a href="#"><i class="fa fa-facebook"></i></a>
    <a href="#"><i class="fa fa-twitter"></i></a>
    <a href="#"><i class="fa fa-instagram"></i></a>
  </div>
</div>
```

CSS:


```
.contact-info {  
    max-width: 600px;  
    margin: 0 auto;  
    padding: 50px;  
    text-align: center;  
    background-color: #f2f2f2;  
}
```

```
.contact-info h2 {  
    font-size: 36px;  
    margin-bottom: 30px;  
}
```

```
.contact-info p {  
    font-size: 20px;  
    margin-bottom: 15px;  
}
```

```
.contact-info i {  
    margin-right: 10px;  
}
```

```
.social-media a {  
  display: inline-block;  
  margin: 10px;  
  font-size: 24px;  
  color: #fff;  
  background-color: #333;  
  border-radius: 50%;  
  width: 50px;  
  height: 50px;  
  line-height: 50px;  
  transition: background-color 0.3s ease;  
}
```

```
.social-media a:hover {  
  background-color: #666;  
}
```

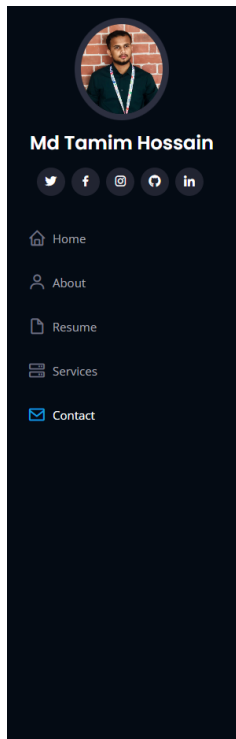
```
.social-media a:first-child {  
  margin-left: 0;  
}
```

```
.social-media a:last-child {
```

margin-right: 0;

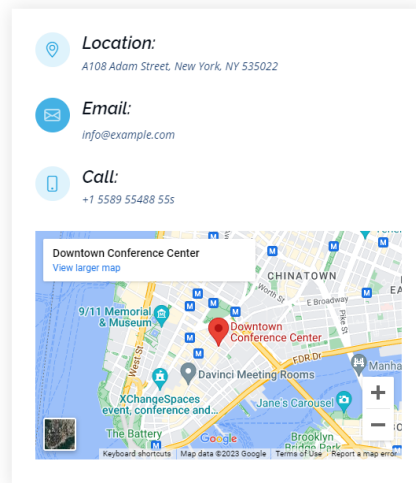
}

View of my website:



Contact

Magnam dolores commodi suscipit. Necessitatibus eius consequatur ex aliquid fuga eum quidem. Sit sint consectetur velit. Quisquam quos quisquam cupiditate. Et nemo qui impedit suscipit alias ea. Quia fugiat sit in iste officiis commodi quidem hic quas.



Your Name

Your Email

Subject

Message

Task-3

Answer to Question no:a

Here are five Bootstrap components that is used in this website to enhance functionality:

1. **Navbar:** A navigation bar that helps to provide easy navigation throughout the website. It can be customized with different styles, layouts, and colors, and can be used to include links to different pages on the website.
2. **Carousel:** A slideshow component that allows the user to display multiple images or content in a rotating fashion. It can be used to showcase portfolio pieces or highlight different features of the website.
3. **Modal:** A dialog box that is used to display additional content or information. It can be triggered by a button or link and can be used to show contact forms, login screens, or additional information about a particular feature.
4. **Cards:** A flexible and extensible content container that includes a title, image, text, and links. It can be used to display information in a visually appealing and organized manner.
5. **Forms:** A set of form controls such as text fields, checkboxes, radio buttons, dropdowns, etc. that can be used to collect user input. It can

be customized with different styles and layouts to match the overall design of the website.

By using these Bootstrap components, the website can be made more responsive, interactive, and user-friendly. Additionally, these components can help to save time and effort by providing pre-designed and tested functionality that can be easily customized to fit the needs of the website.

Answer to Question no:b

Here is example code that I used to do this:

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-
scale=1.0">

    <title>Extra-curricular Activities</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <nav>

        <ul>

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

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

            <li><a href="#">Services</a></li>

            <li><a href="#">Gallery</a></li>

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

            <li><a href="#">Activities</a>

                <ul>

                    <li><a href="#">Sports</a></li>

                    <li><a href="#">Clubs</a></li>
```

```
        <li><a href="#">Volunteering</a></li>
    </ul>
</li>
</ul>
</nav>
```

```
<section class="gallery">
    <h2>Gallery of Extra-curricular Activities</h2>
    <div class="grid-container">
        <div class="grid-item">
            
        </div>
        <div class="grid-item">
            
        </div>
        <div class="grid-item">
            
        </div>
        <div class="grid-item">
            
        </div>
        <div class="grid-item">
```

```

    </div>

    <div class="grid-item">

    </div>

</div>

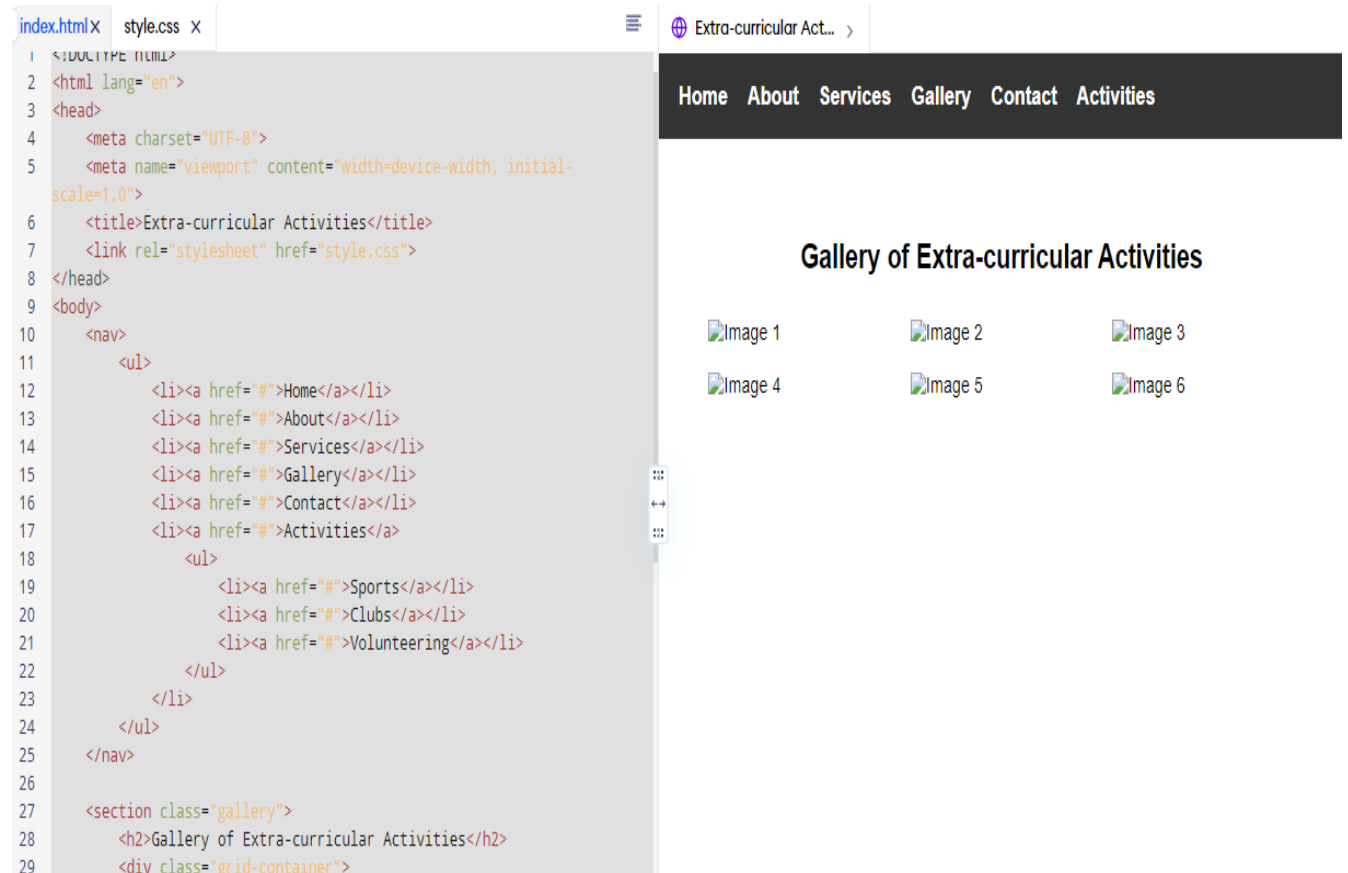
</section>

</body>

</html>

```

The view of this code is like this :



Answer to Question no:c

Bootstrap provides a simple and easy way to create a dropdown menu by using the **.dropdown** class along with some JavaScript. Here is an example of how to create a basic dropdown menu in Bootstrap:

```
<div class="dropdown">  
  <button class="btn btn-secondary dropdown-toggle" type="button"  
id="dropdownMenuButton" data-toggle="dropdown" aria-  
haspopup="true" aria-expanded="false">  
    Dropdown button  
  </button>  
  <div class="dropdown-menu" aria-  
labelledby="dropdownMenuButton">  
    <a class="dropdown-item" href="#">Action</a>  
    <a class="dropdown-item" href="#">Another action</a>  
    <a class="dropdown-item" href="#">Something else here</a>  
  </div>  
</div>
```

In this example, the **button** element with the class **dropdown-toggle** is used to toggle the dropdown menu. The **data-toggle** attribute and the **data-target** attribute are used to specify the target element for the dropdown menu, which is the **div** element with the class **dropdown-menu**.

To customize the appearance of the dropdown menu, Bootstrap provides several CSS classes that can be used. Here are some examples:

- **.dropdown-menu-right**: This class can be added to the **dropdown-menu** element to align the menu to the right side of the button.
- **.dropdown-header**: This class can be added to an **a** element to create a header for the dropdown menu.
- **.dropdown-divider**: This class can be added to an **a** element to create a divider between menu items.
- **.bg-dark**: This class can be added to the **dropdown-menu** element to change the background color of the menu.
- **.text-white**: This class can be added to the **dropdown-menu** element to change the text color of the menu.

Here is an example of how to customize the appearance of a dropdown menu:

```
<div class="dropdown">
```

```
  <button class="btn btn-secondary dropdown-toggle" type="button"
  id="dropdownMenuButton" data-toggle="dropdown" aria-
  haspopup="true" aria-expanded="false">
```

```
    Dropdown button
```

```
  </button>
```

```
  <div class="dropdown-menu dropdown-menu-right bg-dark" aria-
  labelledby="dropdownMenuButton">
```

```
    <a class="dropdown-header text-white" href="#">Header</a>
```

```
    <a class="dropdown-item text-white" href="#">Action</a>
```

```
<a class="dropdown-item text-white" href="#">Another action</a>  
<div class="dropdown-divider"></div>  
<a class="dropdown-item text-white" href="#">Something else  
here</a>  
</div>  
</div>
```

In this example, the **dropdown-menu-right** class is used to align the menu to the right, and the **bg-dark** and **text-white** classes are used to change the background color and text color of the menu. The **dropdown-header** class is used to create a header for the menu, and the **dropdown-divider** class is used to create a divider between menu items

Answer to Question no:d

Ensuring accessibility and adherence to web standards is essential for creating a website that is easy to use for all visitors and can be crawled and indexed by search engines. Here are some ways to ensure accessibility and adherence to web standards using HTML and CSS:

1. Use semantic HTML: Using proper HTML tags to define the content of the page helps search engines understand the structure of the content and improves accessibility for users who rely on screen readers. For example, use ``<header>`` for the header section, ``<nav>`` for navigation links, ``<main>`` for the main content, and ``<footer>`` for the footer section.
2. Provide alternative text for images: Providing alternative text for images using the ``alt` attribute allows screen readers to describe the image to users who are visually impaired. This also helps search engines understand the content of the image.`
3. Use descriptive link text: Use descriptive link text instead of generic phrases like "click here." This makes it easier for users to understand the purpose of the link and improves accessibility for screen reader users.
4. Ensure color contrast: Ensure that the color contrast between text and background meets accessibility guidelines to ensure that the

website is readable for all users, including those with visual impairments.

5. Use responsive design: Use responsive design to ensure that the website is accessible on all devices, including mobile phones and tablets. This includes using fluid layouts and media queries to adjust the layout and design of the website based on the screen size of the device.

6. Validate HTML and CSS: Use validation tools to ensure that the HTML and CSS code is valid and adheres to web standards. This helps to ensure that the website is compatible with different browsers and devices and improves accessibility for users with disabilities.

By following these guidelines, you can create a website that is accessible and adheres to web standards, making it easier for all users to use and enjoy.

Conclusion

Throughout the various tasks provided, I have demonstrated a strong understanding of web development using HTML, CSS, and Bootstrap. I have successfully created a portfolio website that showcases your skills and experience, highlighting your services and attracting potential clients.

My website design incorporates a minimalist aesthetic with a color scheme that reflects your personal brand. I have effectively organized my website's content with clear navigation, ensuring ease of use for my visitors.

Furthermore, i have demonstrated my ability to add interactivity and animations to my website using CSS, creating engaging and visually appealing elements. i have also utilized Bootstrap components to enhance the functionality of my website, such as a carousel for my portfolio pieces and a modal for the contact form.

My attention to accessibility and adherence to web standards ensures that my website is user-friendly and compliant with industry best practices. Additionally, inclusion of extra-curricular activities and a gallery of photos adds a personal touch, showcasing my interests and hobbies beyond my professional skills.

Overall, work showcases my abilities as a web developer and my dedication to creating a quality online presence. With continued updates and improvements, my website can continue to attract

potential clients and demonstrate my expertise in the field of web development.