

Shri Vaishnav Vidhyapeeth Vishwavidhyalaya  
Shri Vaishnav Institute of Information Technology



Year (2022-23)

**Program : B.Tech, Department: IT**

**Subject:- INTERNSHIP/PROJECT , Subject Code :- BTCS801**

***Submitted by:*** Prashant Vishvkarma [19100BTIT06591]

***Under the guidance of:*** Prof. Priyanka Gupta (Mentor) & Prof. Rahul Choudhary

## Assignment-I

### Q 1) What is HTML? discuss about it

Ans:

HTML stands for Hyper Text Markup Language. It is a language of World Wide Web. It is a standard text formatting language which is used to create and display pages on the Web. It makes the text more interactive and dynamic. It can turn text into images, tables, links.

**Hyper Text:** HyperText simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. HyperText is a way to link two or more web pages (HTML documents) with each other.

**Markup language:** A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

**Web Page:** A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. **With the help of HTML only, we can create static web pages.**

Hence, HTML is a markup language which is used for creating attractive web pages with the help of styling, and which looks in a nice format on a web browser. An HTML document is made of many HTML tags and each HTML tag contains different content

**Let's see a simple example of HTML.**

1. `<!DOCTYPE>`
2. `<html>`
- 3.
4. `<head>`
5. `<title>Web page title</title>`
6. `</head>`
7. `<body>`
8. `<h1>Write Your First Heading</h1>`
9. `<p>Write Your First Paragraph.</p>`
10. `</body>`
11. `</html>`

## Description of HTML Example

**<!DOCTYPE>:** It defines the document type or it instruct the browser about the version of HTML.

**<html >:** This tag informs the browser that it is an HTML document. Text between html tag describes the web document. It is a container for all other elements of HTML except <!DOCTYPE>

**<head>:** It should be the first element inside the <html> element, which contains the metadata (information about the document). It must be closed before the body tag opens.

**<title>:** As its name suggested, it is used to add title of that HTML page which appears at the top of the browser window. It must be placed inside the head tag and should close immediately. (Optional)

**<body> :** Text between body tag describes the body content of the page that is visible to the end user. This tag contains the main content of the HTML document.

**<h1> :** Text between <h1> tag describes the first level heading of the webpage.

**<p> :** Text between <p> tag describes the paragraph of the webpage.

## Brief History of HTML

In the late 1980's , a physicist, Tim Berners-Lee who was a contractor at CERN, proposed a system for CERN researchers. In 1989, he wrote a memo proposing an internet based hypertext system.

**Tim Berners-Lee** is known as the father of HTML. The first available description of HTML was a document called "HTML Tags" proposed by Tim in late 1991. The latest version of HTML is HTML5, which we will learn later in this tutorial.

## HTML Versions

Since the time HTML was invented there are lots of HTML versions in market, the brief introduction about the HTML version is given below:

**HTML 1.0:** The first version of HTML was 1.0, which was the barebones version of HTML language, and it was released in 1991.

**HTML 2.0:** This was the next version which was released in 1995, and it was standard language version for website design. HTML 2.0 was able to support extra features such as form-based file upload, form elements such as text box, option button, etc.

**HTML 3.2:** HTML 3.2 version was published by W3C in early 1997. This version was capable of creating tables and providing support for extra options for form elements. It can also support a web page with complex mathematical equations. It became an official standard for any browser till January 1997. Today it is practically supported by most of the browsers.

**HTML 4.01:** HTML 4.01 version was released on December 1999, and it is a very stable version of HTML language. This version is the current official standard, and it provides added support for stylesheets (CSS) and scripting ability for various multimedia elements.

**HTML5 :** HTML5 is the newest version of HyperText Markup language. The first draft of this version was announced in January 2008. There are two major organizations one is W3C (World Wide Web Consortium), and another one is WHATWG( Web Hypertext Application Technology Working Group) which are involved in the development of HTML 5 version, and still, it is under development.

### **Features of HTML**

- 1) It is a very **easy and simple language**. It can be easily understood and modified.
- 2) It is very easy to make an **effective presentation** with HTML because it has a lot of formatting tags.
- 3) It is a **markup language**, so it provides a flexible way to design web pages along with the text.
- 4) It facilitates programmers to add a **link** on the web pages (by html anchor tag), so it enhances the interest of browsing of the user.
- 5) It is **platform-independent** because it can be displayed on any platform like Windows, Linux, and Macintosh, etc.
- 6) It facilitates the programmer to add **Graphics, Videos, and Sound** to the web pages which makes it more attractive and interactive.
- 7) HTML is a case-insensitive language, which means we can use tags either in lower-case or upper-case.

**NOTE: It is recommended to write all tags in lower-case for consistency, readability, etc.**

## Q.2) What are Tags?

Ans:

### HTML Tags

HTML tags are like keywords which defines that how web browser will format and display the content. With the help of tags, a web browser can distinguish between an HTML content and a simple content. HTML tags contain three main parts: opening tag, content and closing tag. But some HTML tags are unclosed tags.

When a web browser reads an HTML document, browser reads it from top to bottom and left to right. HTML tags are used to create HTML documents and render their properties. Each HTML tags have different properties.

An HTML file must have some essential tags so that web browser can differentiate between a simple text and HTML text. You can use as many tags you want as per your code requirement.

- All HTML tags must enclosed within < > these brackets.
- Every tag in HTML perform different tasks.
- If you have used an open tag <tag>, then you must use a close tag </tag> (except some tags)

### Syntax

<tag> content </tag>

### HTML Tag Examples

**Note: HTML Tags are always written in lowercase letters. The basic HTML tags are given below:**

<p> Paragraph Tag </p>

<h2> Heading Tag </h2>

<b> Bold Tag </b>

<i> *Italic Tag* </i>

<u> Underline Tag </u>

### Unclosed HTML Tags

Some HTML tags are not closed, for example br and hr.

**<br> Tag:** br stands for break line, it breaks the line of the code.

**<hr> Tag:** hr stands for Horizontal Rule. This tag is used to put a line across the webpage.

## **HTML Meta Tags**

DOCTYPE, title, link, meta and style

## **HTML Text Tags**

<p>, <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <strong>, <em>, <abbr>, <acronym>, <address>, <bdo>, <blockquote>, <cite>, <q>, <code>, <ins>, <del>, <dfn>, <kbd>, <pre>, <samp>, <var> and <br>

## **HTML Link Tags**

<a> and <base>

## **HTML Image and Object Tags**

<img>, <area>, <map>, <param> and <object>

## **HTML List Tags**

<ul>, <ol>, <li>, <dl>, <dt> and <dd>

## **HTML Table Tags**

table, tr, td, th, tbody, thead, tfoot, col, colgroup and caption

## **HTML Form Tags**

form, input, textarea, select, option, optgroup, button, label, fieldset and legend

## **HTML Scripting Tags**

script and noscript

**Note:** We will see examples using these tags in later chapters.

**Q. 3) What is CSS? And discuss about it.**

Ans:

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

**Brief History of CSS**

CSS was developed by **W3C (World Wide Web Consortium)** in 1996 for a rather simple reason. HTML element was not designed to have tags that would help format the page. You were only supposed to write the markup for the web page.

Tags like **<font>** were introduced in HTML version 3.2, and it caused quite a lot of trouble for web developers. Due to the fact that web pages have different fonts, colored backgrounds, and multiple styles, it was a long, painful, and expensive process to rewrite the code. Thus, CSS was created by W3C to solve this problem.

CSS is not technically a necessity, but you probably wouldn't want to look at a web page that features only HTML elements as it would look completely bare-boned.

**CSS Advantages on Web Pages**

The difference between a web page that implements CSS and one that doesn't is massive and surely noticeable.

You might have seen a website that fails to load completely and has a white background color with most of the text being blue and black. This means that the CSS part of the web page didn't load correctly or it doesn't exist altogether.

That's what web pages with only HTML look like, and I think you'd agree that that's not very appealing. Before using CSS, all of the stylizing had to be included into the HTML markup. This means web developers had to separately describe the background color, font size, alignments, etc.

CSS lets you stylize everything on a different file, thus creating the design there and later on integrating the CSS files on top of the HTML markup. This makes the actual HTML markup much cleaner and easier to maintain.

In short, with CSS features you don't need to repeatedly describe how individual elements look. This saves time, shortens the code, and makes it not as prone to errors.

CSS lets you have multiple styles on one HTML page, therefore making the customization possibilities almost endless. Nowadays, this is becoming more a necessity than a commodity.

### How Does CSS Work?

CSS uses a simple English based syntax with a set of rules that govern it. Like we've mentioned before, HTML was never intended to use style elements, only the markup of the page. It was created to merely describe the content. For example: **<p>This is a paragraph.</p>**.

But how do you style the paragraph? The CSS syntax structure is pretty simple. It has a selector and a declaration block. You select an element and then declare what you want to do with it. Pretty straightforward, right?

However, there are rules you have to remember. The structure rules are pretty simple, so don't worry.

The selector points to the HTML elements you want to style. The declaration block contains one or more declarations separated by semicolons.

Each declaration includes a CSS property name and a value, separated by a colon. A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces.

Let's look at an example:

All of the **<p>** elements will be colored blue and bolded.

**<style>**

**p {**

**color: blue;**

**text-weight: bold;**

**}**

**<style>**

In another example, all **<p>** elements will be center-aligned, be 16x wide and pink.

**<style>**

**p {**



```
text-align: center;
```

```
font-size: 16px;
```

```
color: pink;
```

```
}
```

```
</style>
```

Now let's talk about the different styles of CSS. They are Inline, External and Internal.

### **Internal, External and Inline CSS Styles**

We'll go over each style briefly, for an in-depth explanation of each method, there will be a link below the overview.

Let's start by talking about the **Internal** style. CSS styles done this way are loaded each time an entire website is refreshed, which may increase loading time. Additionally, you won't be able to use the same CSS style on multiple pages as it's contained within a single page. However, this also comes with benefits. Having everything on one page makes it easier to share the template for a preview.

The **External** method might be the most convenient one. Everything is done externally on a **.css** file. This means you can do all the styling on a separate file and apply the CSS to any page you want. The External style might also improve loading times.

Lastly, we will talk about the **Inline** style of CSS. Inline works with specific elements that have the `<style>` tag. Each component has to be stylized, so it might not be the best or fastest way to handle CSS. But it can come in handy. For example, if you want to change a single element, quickly preview changes, or maybe you don't have access to the CSS files.

#### **Q.4 What does CSS do? Why use CSS?**

**Ans:**

- You can add new looks to your old HTML documents.
- You can completely change the look of your website with only a few changes in CSS code.

These are the three major benefits of CSS:

##### **1) Solves a big problem**

Before CSS, tags like font, color, background style, element alignments, border and size had to be repeated on every web page. This was a very long process. For example: If you are developing a large website where fonts and color information are added on every single page, it will become a long and expensive process. CSS was created to solve this problem. It was a W3C recommendation.

##### **2) Saves a lot of time**

CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file.

##### **3) Provide more attributes**

CSS provides more detailed attributes than plain HTML to define the look and feel of the website.

## Q5. How to Link CSS to Html?

Ans:

In HTML, we can easily link the style sheet to the Html document in the following different three methods:

1. Using an Inline Style
2. Using an Embedded Style or Internal Style
3. Using an External style

### Using an Inline Style

It is an easiest method for adding the CSS style to our Html document or code. But we cannot reuse this method, so reusability is the disadvantage in this method. If we want to add the CSS using inline style to our Html document then we have to follow the steps which are given below:

**Step 1:** Firstly, we have to type the Html code in any text editor or open the existing Html file in the text editor in which we want to link the CSS using Inline Style.

1. `<!Doctype Html>`
2. `<Html>`
3. `<Head>`
4. `<Title>`
5. Link the CSS using Inline style to Html
6. `</Title>`
7. `</Head>`
8. `<Body>`
9. This page helps you to understand how to link the CSS to the Html page. `<br>`
10. `<br>`
11. And, this section helps you to understand how to link the CSS using Inline Style.
12. `</Body>`
13. `</Html>`

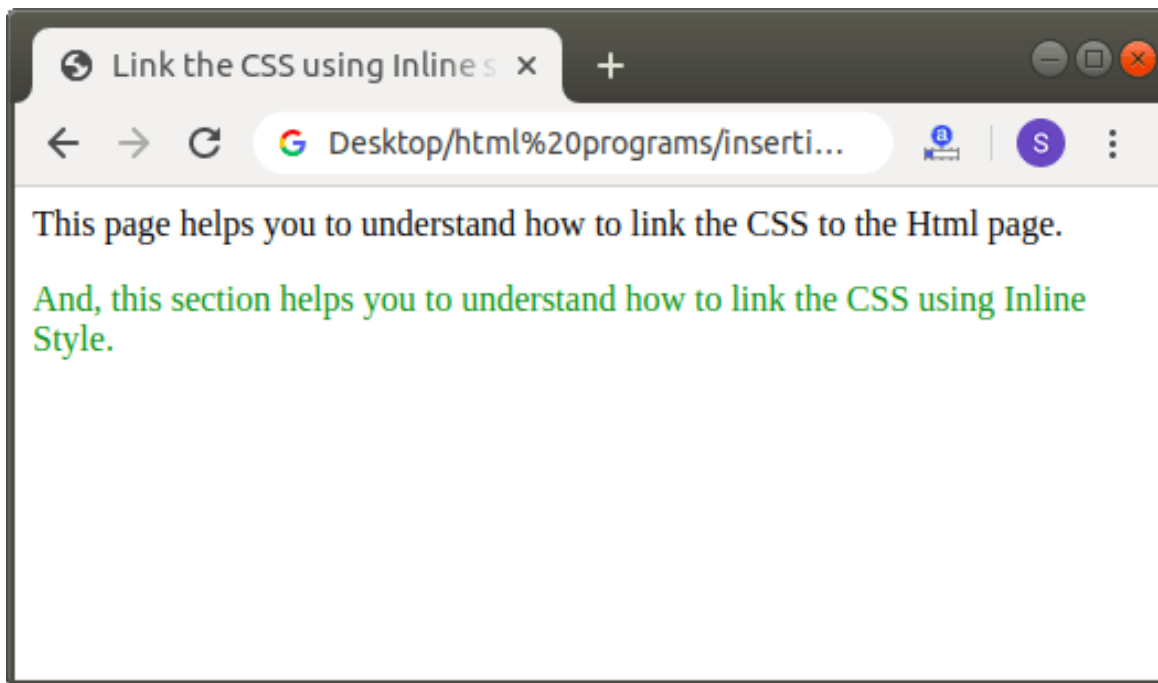
**Step 2:** Now, we have to use the style attribute at the starting of that text on which we want to use CSS. So, we have to type the style attribute within a particular tag for linking the CSS to Html using inline style as shown in the following block:

`<p style=" "> Any text </p>`

**Step 3:** Now, we have to give the property in the style attribute as shown in the following block:

1. `<!Doctype Html>`
2. `<Html>`
3. `<Head>`
4. `<Title>`
5. Link the CSS using Inline style to Html
6. `</Title>`
7. `</Head>`
8. `<Body>`
9. This page helps you to understand how to link the CSS to the Html page. `<br>`
10. `<p style="color:green;">`
11. And, this section helps you to understand how to link the CSS using Inline Style.
12. `</p>`
13. `</Body>`
14. `</Html>`

**Step 4:** Now, save the Html Code and then run it. When the code is successfully executed in the browser then it will show the output. The below screenshot shows the output of the above Html code:



## Using an Internal StyleSheet

Those sheets which only affects the Html document in which they are embedded in are known as **Internal Style Sheets**. These style sheets are defined between the starting and closing of **<head>** tag.

If we want to add the CSS using Internal style sheet to our Html document then we have to follow the steps which are given below:

**Step 1:** Firstly, we have to type the Html code in any text editor or open the existing Html file in the text editor in which we want to link the CSS using Internal style sheet.

1. **<!Doctype Html>**
2. **<Html>**
3. **<Head>**
4. **<Title>**
5. Link the CSS using Inline style to Html
6. **</Title>**
7. **</Head>**
8. **<Body>**
9. This page helps you to understand how to link the CSS to the Html page. **<br>**
10. **<br>**
11. And, this section helps you to understand how to link the CSS using Internal Style Style.
12. **</Body>**
13. **</Html>**

**Step 2:** Now, we have to place the style tag within the starting and closing of **<head>** tag, just after the **<title>** tag. We described this step in the following block:

1. **<Head>**
2. **<Title>**
3. Link the CSS using Internal Style Sheet to Html
4. **</Title>**
5. **<style>**
6. ....
7. ....
8. **</style>**
9. **</Head>**

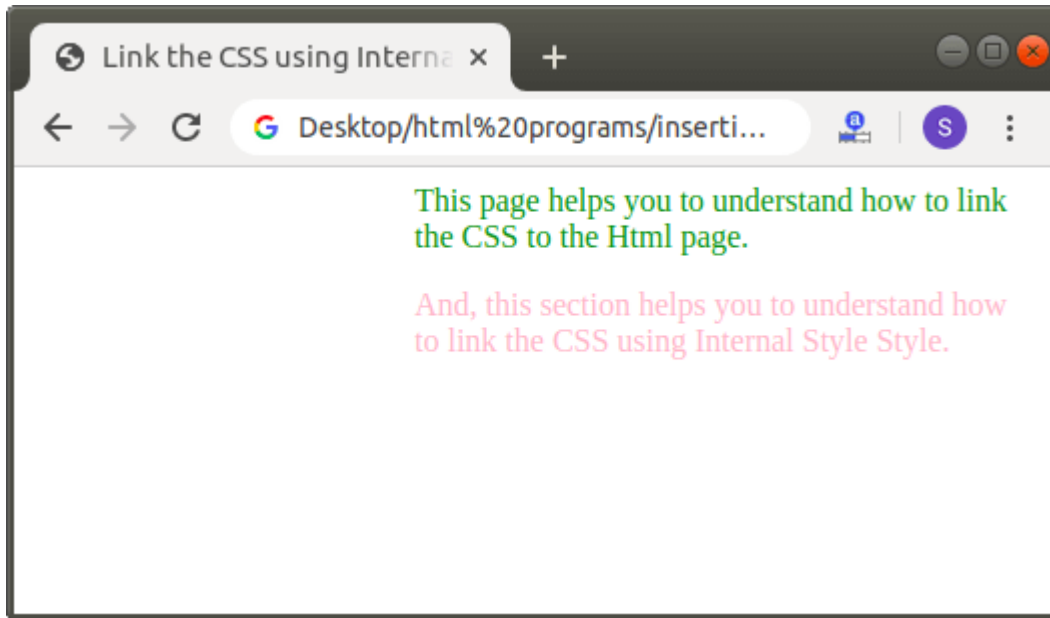
**Step 3:** Now, we have to use the attribute of style tag whose name is "type". So, we have to always start the <style> tag like this:

1. **<style type="text/css">**

**Step 4:** Now, we have to add those elements which we want to use on the text in the same Html page. We can add these elements within the style tag which is defined in the <head> tag.

1. <!Doctype Html>
2. <Html>
3. <Head>
4. <Title>
5. Link the CSS using Internal Style Sheet to Html
6. </Title>
7. <style>
8. Body
9. {
10. color:green;
11. margin-left:200px;
12. }
13. p
14. {
15. color:pink;
16. }
17. </style>
18. </Head>
19. <Body> This page helps you to understand how to link the CSS to the Html page.
20. <p>
21. And, this section helps you to understand how to link the CSS using Internal Style Style.
22. </p>
23. </Body>
24. </Html>

**Step 5:** And, at last save the Html file and run it. When the code is successfully executed by the browser then it will show the output. The below screenshot provides the output of the above Html code:



### Using an external Style

Those files which contains only the CSS format or code only are known as external style sheet files or CSS files. The extension of these files must end with the .css extension. These files are different from Html files and can be easily included in the Html files using the **<link>** tag.

If we want to add the CSS using External style sheet to our Html document then we have to follow the steps which are given below:

**Step 1:** Firstly, we have to type the Html code in any text editor or open the existing Html file in the text editor in which we want to link the CSS file:

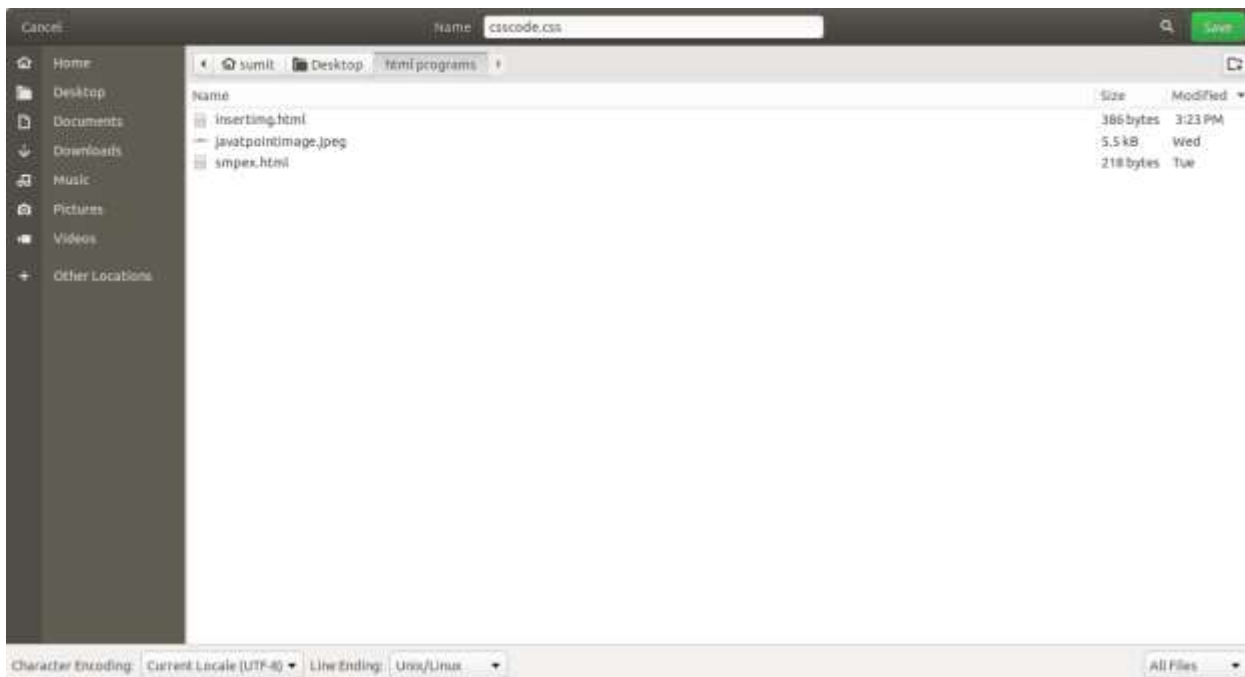
1. **<!Doctype Html>**
2. **<Html>**
3. **<Head>**
4. **<Title>**
5. Link the CSS code using External style sheet to Html
6. **</Title>**
7. **</Head>**
8. **<Body>**

9. This page helps you to understand how to link the CSS to the Html page. <br>
10. <br>
11. And, this section helps you to understand how to link the CSS using External Style Style.
12. </Body>
13. </Html>

**Step 2:** Now, we have to create the CSS file. So, open the text editor and type the CSS code in the file.

1. Body
2. {
3. color:green;
4. margin-left:200px;
5. }
6. p
7. {
8. color:pink;
9. }

**Step 3:** And, then save the file with the .css extension.





**Step 4:** Again, come to the Html file. And then, we have to place the cursor within the starting and closing of <head> tag, just after the <title> tag. And, then type the <tag> with its attributes and their values. We described it in the following block:

1. <!Doctype Html>
2. <Html>
3. <Head>
4. <Title>
5. Link the CSS code using External style sheet to Html
6. </Title>
7. <link rel="stylesheet" href="csscode.css">
8. </Head>
9. <Body>
10. This page helps you to understand how to link the CSS to the Html page.
11. <p>
12. And, this section helps you to understand how to link the CSS using External Style Style.
13. </p>
14. </Body>
15. </Html>

**Step 5:** Now, we have to save the Html file at the same location or directory in which CSS file is saved. And, then run the Html file in a browser. The output of the above html code is shown in the following screenshot:

