

[≡ MENU](#)

The HTML Article vs Section Elements: Everything On Their Difference

by Position is Everything



The HTML article vs section is a debate that has been around ever since these elements were first introduced. Moreover, the HTML article element is used **to wrap independent content** and there does not have to be any context. On the other hand, the section in HTML documents is used **to divide specific content** on your web page into sections.

Sometimes it is difficult to see the difference between the article and section elements, so keep reading this article as we explore and explain their main differences.

Contents [\[show\]](#)

The HTML Article Element

Before we explain and exemplify the main **difference between article and section HTML elements**, we must discuss their primary functions. The `<article>` HTML element is used when it has **independent and self-contained content**. Any specific article on your web page should convey meaning without being dependent on other articles on the same web page.

To do so, you are going to use the article HTML tag and wrap the content between the opening and closing tags. You should use the **HTML article element** in the following situations:

- Writing a forum post
- Creating a blog post
- Publishing a news story

This element does not have any styling values and is simply going **to divide your content**. However, you can use specific CSS styling properties to change the visual output. We are going to provide you with the two syntaxes so you can see how simple it is to implement them.

– Creating a Basic HTML Syntax

Writing the **correct syntax** when using the **HTML article element** does not require a lot of effort. You are usually supposed to add a heading and a paragraph inside the opening and closing tags. Moreover, you can create **multiple HTML article blocks** on your web page and then style them using specific CSS properties. Our example is going to contain three HTML articles and each is going to be assigned with a specific class.

Let us observe the following HTML syntax example:

```
<article class="all-browsers">
<h1> Most Popular Web Browsers </h1>
<article class="browser">
<h2> Google Chrome </h2>
<p> Google Chrome is one of the most widely used browsers today. It was
released in 2008 by Google. </p>
</article>
<article class="browser">
<h2> Mozilla Firefox </h2>
<p> Mozilla Firefox recently became the most most popular browser. It was
```

```
designed by Mozilla. </p>
</article>
<article class="browser">
<h2> Microsoft Edge </h2>
<p> Microsoft Edge replaces Internet Explorer. Microsoft designed and
released it in 2015. </p>
</article>
</article>
```

The **three articles are wrapped inside a single HTML article**. Web developers recommend doing this to create a more professional and [correct HTML syntax](#). We have included the classes in these elements so you can style them using specific CSS styling properties.

– Creating a CSS Syntax

The easiest way to change the visual output of any HTML element is by using specific **CSS styling properties**. Experts recommend writing this syntax on a separate document for easier and smoother navigation. The CSS properties we have included are going to change the margins and background color. However, you can include other properties if you want to change other default values.

Let us observe the following **CSS syntax**:

```
<style>
.all-browsers {
margin: 0;
padding: 7px;
background-color: blue;
}
.all-browsers > h1, .browser {
margin: 9px;
padding: 4px;
}
.browser {
background: grey;
}
.browser > h2, p {
margin: 5px;
```

```
font-size: 85%;  
}  
</style>
```

If you include the **CSS syntax** in the same HTML document, you are supposed to wrap them in the HTML style element. There are several CSS classes in this example and each class contains different styling properties. Moreover, this is all it takes to learn about the HTML article element, so we can now move on to the HTML section element.

The HTML Section Element

The HTML section element represents a **generic sectioning element** because it divides the document into thematically defined pieces. As with the HTML article, this element is also followed by a heading and a paragraph. Web developers usually use this element **to divide long documents into chapters**, create different tabs, and divide various themes on your web page. Furthermore, you are supposed to use the following tags inside the HTML section element:

- **<main>**
- **<header>**
- **<footer>**
- **<nav>**
- **<aside>**

These are some of **the most common tags** used when creating a separate section on your web page. This element will not divide the content into specific blocks for styling purposes because it only created thematical groups. If you want to make separate blocks, you should use the HTML div element. Now, let us take a look at some examples.

– Creating a Basic HTML Syntax

It is quite easy to create a fully-functional **HTML section syntax** because the method is similar to the article element. You just have to wrap the content inside the opening and closing tags.

We are going to provide you with an example that is going to have other elements too, so you can see the **function of the HTML section element** more

easily. This element usually does not require any CSS styling properties:

```
<section>
<h1> Advertisement </h1>
<p><a href="http://www.AnExampleWebPage.com"> The new frogger shaper
5000. Everything you need to keep fit this winter! </a> </p>
</section>
<main>
<h1> How to keep yourself fit during the pandemic? </h1>
<p> During the global pandemic it is very important to have a healthy
lifestyle... </p>
</main>
<section>
<h1> Other articles </h1>
<ul>
<li> <a href="eat-vegetables.html"> Include healthy vegetables in your diet
</a> </li>
<li> <a href="eat-fruits.html"> Include various fruits in your diet </a> </li>
</ul>
</section>
```

You can **wrap various elements inside your HTML section** element, as you can see. In the first section element, we included the common heading and paragraph HTML elements. However, in the second section element, we created an unordered list including two list items.

This is going to create two different sections on your web page, including the main section between.

– Creating a CSS Syntax

As we have previously stated, this element **does not require a lot of styling attributes**. However, we are going to provide you with a simple CSS syntax that is going to hide specific content for improved assistive technology. This is going to help people who use screen readers.

Please observe the following **CSS example**, that contains three styling properties:

```
.hidden {  
  position: absolute;  
  top: -9999px;  
  left: -9999px;  
}
```

Though you can include other styling properties for specific elements in the previous example, it is not required. This **simple CSS syntax** is enough, not only to aid people who use screen readers but also to improve your web page's SEO (Search Engine Optimization).

The HTML Section vs Article Elements: Noticing Their Difference

Now that you are familiar with the **HTML section and article elements**, we are ready to start discussing their differences.

On a general note, web developers recommend using the HTML **article element over the section element**. A simple reason that supports this opinion is that the article element constructs an optimal display for users who have a smaller screen. On the other hand, if you want to do the same with the HTML section element, you are going to use specific CSS styling properties.

The easiest way to take notice of how different they are is to take a look at an example that contains both elements. We are going to create **specific classes** in several HTML elements, so later on we can apply specific CSS styling properties. The article element is going to contain independent content, whereas the section element is going to divide the specific section on your web page.

– Looking at a Complete Example Containing Both Elements

Since these two elements are used to **wrap specific content** on your web page, they must be included in the body of your HTML document. Moreover, we are going to include three HTML div elements. later on in this article, we are going to explain the reason why you should also do this in your HTML document.

Let us observe the following HTML example:

```
<!DOCTYPE html>
<html>
<head>
<title>
Using both tags in a single HTML document
</title>
</head>
<body>
<center>
<h1> Everything About Your Computer </h1>
<p> The Best Computer Science Portal </p>
<div class="container">
<div class="article">
<article>
<h2> This is where we include the article tag </h2>
<p>
The article tag is placed above the <h2> element
</p>
</article>
<h2> This part does not contain the HTML article tag </h2>
<p>
The header in this part is going to appear larger than before
</p>
</div>
<div class="section">
<section>
<h2> This is where we include the section tag</h2>
<p>
The section tag is placed above the <h2> element
</p>
</section>
<h2> This part does not contain the HTML section tag </h2>
<p>
The header in this part is going to appear larger than before
</p>
</div>
</div>
</center>
</body>
</html>
```

This is a **complete HTML syntax** and it contains various elements you can use alongside the article and section elements. There is a slight difference in the heading size if you wrap them in either of these tags.

On the other hand, if you do not wrap the heading, it is going to appear larger. Next, let us see what is **the best way to style this HTML syntax** using specific styling properties.

– Styling the Elements Using Specific CSS Properties

Including **CSS properties** in your document is the best way to style your HTML elements. As mentioned before, the reason why we have included classes in several HTML elements is to style them using the CSS properties. You should **write this syntax on a separate document** and then link both files. This is the best way to avoid the creation of unnecessarily complicated syntaxes.

Let us observe the following CSS syntax:

```
<style>
.container {
width: 700px;
height: auto;
}
.section {
float: left;
}
.article {
float: right;
}
h1 {
color: orange;
}
</style>
```

As you can see, here we have only included several CSS properties. However, they are enough to change the default values and **visually alter the output**. You can add other CSS properties if you want to change the [background color](#), margins, etc.

Is There an Alternative to These Two HTML Elements?

Yes, there is another HTML element you can use instead of the article and section elements. In a previous section of this article, we said that the **HTML div element** is **one of the best** elements you can use to wrap, style, and divide specific content on your web page. If you think that distinguishing when is the right time to create an article or a section on your web page is difficult, the easiest solution is to opt for the HTML div element.

Web developers recommend using this element because you do not have to think so much about **the type of content** you include on your web page. You can also use the HTML div element when wrapping specific content that is not semantically related to each other. It is regarded as one of the best HTML elements you can use to create specific blocks on your web page for styling purposes. Our advice is to give each element a try and see which one suits you the most.

Wrapping Things Up

The difference between the **HTML article and section elements** can sometimes be difficult to notice. That is why this article taught you everything you need to know about their difference but it also explained the following important notes:

- Both elements can be used to divide the content on your web page
- The HTML article element is used to wrap independent and self-contained content
- The HTML section element is used to divide thematically defined pieces
- You can apply specific CSS styling properties to change the visual output
- The HTML article element creates an optimal screen size
- You can use both elements in a single HTML document
- The HTML div element is the alternative option to these two elements



There are plenty of HTML elements that have a similar function and that is why it is important to know **when to properly use each**. After carefully reading this guide you have learned everything about the slight differences between the HTML article and section elements.

5/5 - (20 votes)

Author

Recent Posts



Position Is Everything

Your Go-To Resource for Learn & Build: CSS, JavaScript, HTML, PHP, C++ and MYSQL. [Meet The Team](#)

HTML

- < The HTML Date Input: Entering a Date Through Input Fields
- > The HTML Button Link: Redirecting Users To Different Pages

Leave a Comment

☐ Save my name, email, and website in this browser for the next time I comment.

ABOUT US



Position Is Everything: *The latest Coding and Computing News & Tips.*

Get all of your questions and queries expertly answered in a clear, step-by-step guide format that makes understanding a breeze.

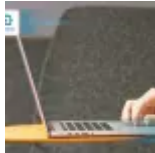
[LEARN MORE](#)

RECENT POSTS



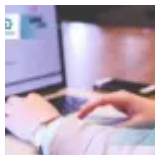
[Systemerror Parent Module Not Loaded Cannot Perform Relative Import](#)

By Position is Everything



[The Mbstring Extension Is missing.please Check Your PHP Configuration](#)

By Position is Everything



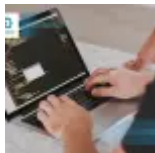
[This App Cannot Be Installed Because Its Integrity: Fixes](#)

By Position is Everything



[This Application Has No Explicit Mapping for /Error, so You Are Seeing This as a Fallback.](#)

By Position is Everything



[Tuple Object Does Not Support Item Assignment: How To Solve?](#)

By Position is Everything

[About Us](#)

[The Team](#)

[Privacy Policy](#)

[Cookie Policy](#)

[Contact Us](#)

[Site Map](#)

Sister Website: [Timerminutes.ninja](#)

© 2022 Position Is Everything • All right reserved