



CSS Box Sizing

What is CSS Box sizing Property?

- The CSS box-sizing property is used to specify how to calculate the total height and width of an element.
- It controls the size of an element with a specified height and width.
- It allows you to include the padding and border within the total height and width of the element.

Before starting with CSS box-sizing, let's first understand what issue occurs if we don't use this property.

Without CSS box-sizing

If we don't include box-sizing property, then the box model works like below as default:

width + padding + border = actual visible width of an element's box

height + padding + border = actual visible height of an element's box

It means if we create a box of some specific height and width and then add padding and border to it, it will look wider than the actual width.

To deal with this issue, developers need to adjust the values for height and width to give space for border and padding. Let's understand it with an example:

Example: Let's create two div with same height and width, but with different border and padding.

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <style media="screen">
      .div1{
        height: 150px;
        width: 400px;
        border: 10px solid green;
```

```
}  
.div2{  
  height: 150px;  
  width: 400px;  
  border: 10px solid yellow;  
  padding: 20px; }  
</style>  
</head>  
<body>  
<h1>Without box-sizing property</h1>  
  
<div class="div1">  
  I am the first div with no padding.  
</div>  
<div class="div2">  
  I am the second div with padding added.  
</div>  
</body>  
</html>
```

In the above code, we have created two divs of the same size. In the first div, we have not included padding, whereas, in the second, we added it.

Output:

It will generate the output as:

Without box-sizing property

I am the first div with no padding

I am the second div with padding added.

As we can see from the above output, that we got two div with different sizes. While we have specified the same sizes of both the div, but the second appears bigger than the first.

This is the main issue that occurs in the **CSS box model**. This issue can be resolved with the help of box-sizing property.

With CSS box-sizing property

The above issue can be resolved by using the **CSS** box-sizing property. Now we will use the same code as above, but will also include box-sizing property to it.

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
<head>
<style media="screen">
  .div1{
    height: 150px;
    width: 400px;
```

```
border: 10px solid green;
box-sizing: border-box;

}
.div2{
height: 150px;
width: 400px;
border: 10px solid yellow;
padding: 20px;
box-sizing: border-box;

}
</style>
</head>
<body>
<h1>With box-sizing property</h1>

<div class="div1">
  I am the first div with no padding.
</div>
<div class="div2">
  I am the second div with padding added.
</div>
</body>
</html>
```

In the above code, we have used two divs of the same height and width and also used the box-sizing property for each div.

Output:

With box-sizing property

I am the first div with no padding.

I am the second div with padding added.

As we can see in the above output, we got both div with the same width.

AD

Syntax

Syntax of box-sizing property is given as:

```
box-sizing: border-box; // for border box
box-sizing: content-box; // for content box
```

Value: CSS box-sizing property has two values, which are border-box and content box. These are given as:

- **content-box:** As per the CSS standard, it is the default value of the box-sizing property. By using this value, the height and width properties will only include the content and do not include margin, padding, or border values. It means if we specify the width of the element as 200 px, then the content will be 200px wide, and the border and padding will be added in the final display by making it wider than 200px. For example, if an element has below

size specifications:

{width: 500px;

border: 10px solid black;} then it will display a box of size 510px.

AD

Granular View of Big Data

Distributed Processing Challenges and Spark Performance Tuning Big Da
PySpark

In this, the dimension of value is calculated as below:

AD

Width= width of the content

Hight= height of the content

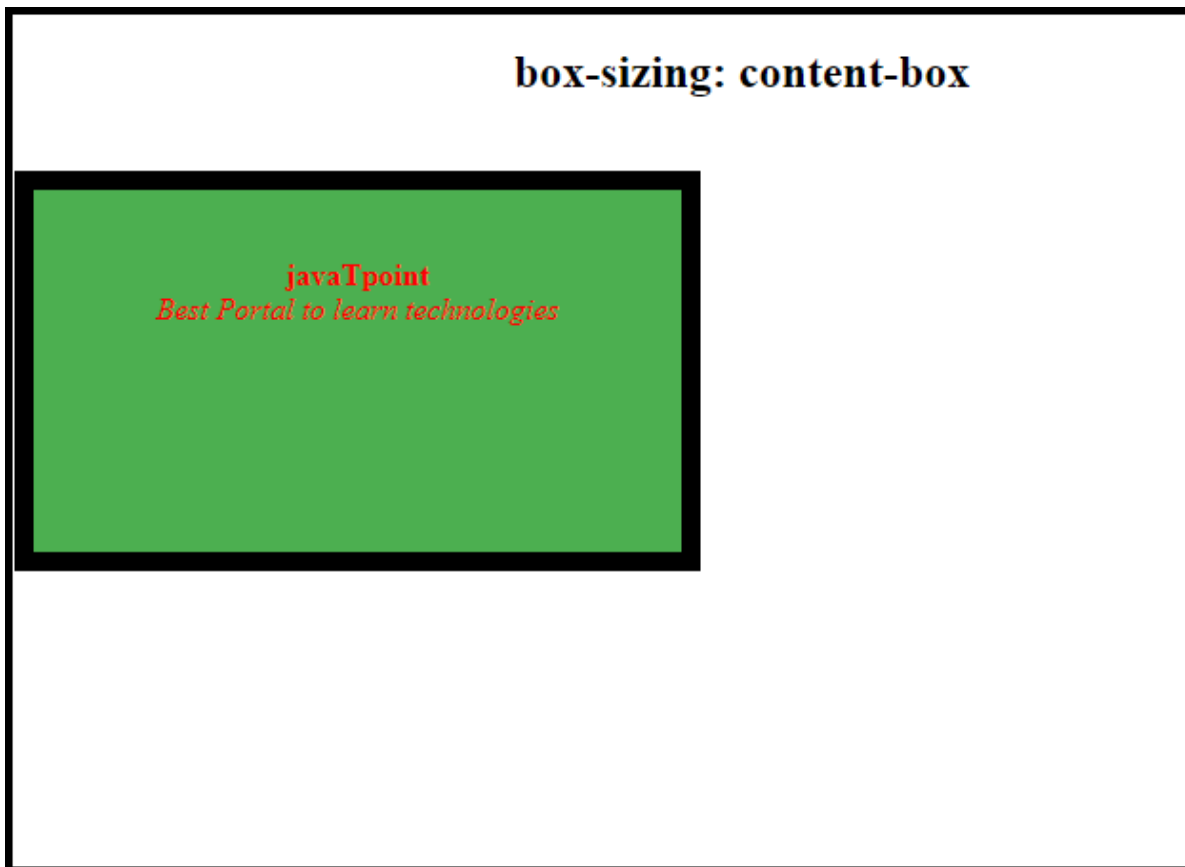
Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>box-sizing Property</title>
    <style>
      div {
        width: 300px;
        height: 150px;
        padding: 20px;
        border: 10px solid black;
        background: #4caf50;
        color: red;
        box-sizing: content-box;}
    </style>
  </head>
  <body style = "text-align: center;">
    <h2>box-sizing: content-box</h2>
    <br>
    <div>
      <p><b>javaTpoint</b><br>
```

```
<i>Best Portal to learn technologies</i></p>  
</div>  
</body>  
</html>
```

In the above code, we have used the content-box value of the box-sizing property.

Output:



- **border-box:** The border-box is a widely used value of the box-sizing property. It informs the browser to adjust the border and padding within the specified height and width of an element. It means if we specify the width of an element as 200 px, then the specified border and padding will be included within 200 px, and the content box will adjust the extra width. For example, if an element has the following size specifications:

```
{width: 500px;  
border: 10px solid black;}
```

The browser will display a box of size 500, in which 480px wide area for width will be included.

In this, the width and height are calculated by including content, border, and padding. **But it does not include the margin.**

The dimensions of the element are calculated as:

Width = border + padding + width of the content

Height = border + padding + height of the content.

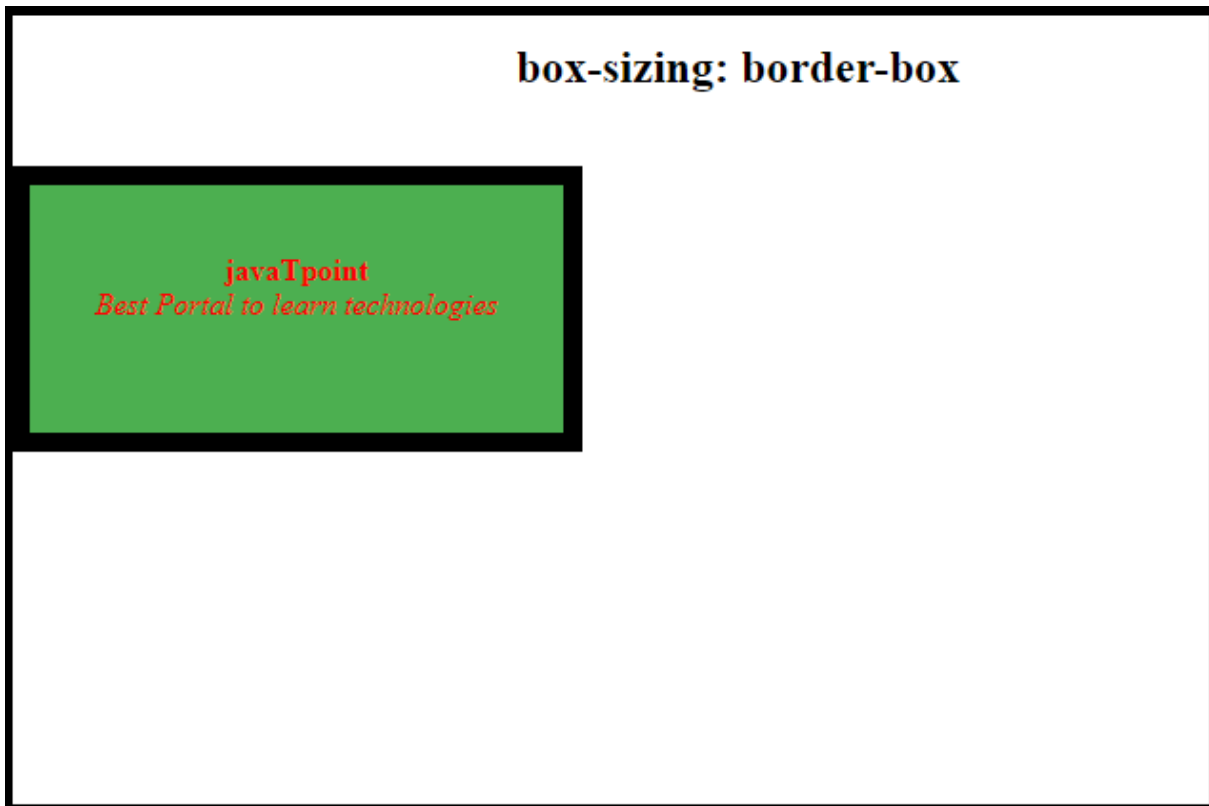
Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>box-sizing Property</title>
    <style>
      div {
        width: 300px;
        height: 150px;
        padding: 20px;
        border: 10px solid black;
        background: #4caf50;
        color: red;
        box-sizing: border-box;
      }
    </style>
  </head>
  <body style = "text-align: center;">
    <h2>box-sizing: border-box</h2>
    <br>
    <div>
      <p><b>javaTpoint</b><br>
      <i>Best Portal to learn technologies</i></p>
    </div>
  </body>
```

```
</html>
```

In the above code, we have used the border-box value of the box-sizing property.

Output:

[< Prev](#)[Next >](#)

AD



For Videos Join Our Youtube Channel: [Join Now](#)


Feedback


- Send your Feedback to feedback@javatpoint.com

Help Others, Please Share





Learn Latest Tutorials


 **Splunk tutorial**
Splunk


 **SPSS tutorial**
SPSS

 **Swagger tutorial**
Swagger


 **T-SQL tutorial**
Transact-SQL


 **Tumblr tutorial**
Tumblr


 **React tutorial**
ReactJS

 **Regex tutorial**
Regex

 **Reinforcement learning tutorial**
Reinforcement Learning


 **R Programming tutorial**
R Programming

 **RxJS tutorial**
RxJS

 **React Native tutorial**
React Native

 **Python Design Patterns**
Python Design Patterns


 **Python Pillow tutorial**
Python Pillow


 **Python Turtle tutorial**
Python Turtle

 **Keras tutorial**
Keras

Preparation

 **Aptitude**
Aptitude

 **Logical Reasoning**
Reasoning

 **Verbal Ability**
Verbal Ability


 **Interview Questions**
Interview Questions



Company
Interview
Questions


Company Questions

Trending Technologies




Artificial
Intelligence

Artificial
Intelligence




AWS Tutorial

AWS




Selenium
tutorial

Selenium




Cloud
Computing

Cloud Computing




Hadoop tutorial

Hadoop




ReactJS
Tutorial

ReactJS




Data Science
Tutorial

Data Science




Angular 7
Tutorial

Angular 7




Blockchain
Tutorial

Blockchain




Git Tutorial

Git



Machine
Learning Tutorial

Machine Learning



DevOps
Tutorial


DevOps

B.Tech / MCA




DBMS tutorial

DBMS




Data Structures
tutorial

Data Structures




DAA tutorial

DAA




Operating
System

Operating System



Computer
Network tutorial

Computer Network



Compiler
Design tutorial

Compiler Design



Computer
Organization and
Architecture

Computer
Organization




Discrete
Mathematics
Tutorial

Discrete
Mathematics



Ethical Hacking

Ethical Hacking



Computer
Graphics Tutorial

Computer Graphics














Software
Engineering



html tutorial

Web Technology

		Software Engineering	
 Cyber Security tutorial Cyber Security	 Automata Tutorial Automata	 C Language tutorial C Programming	 C++ tutorial C++
 Java tutorial Java	 .Net Framework tutorial .Net	 Python tutorial Python	 List of Programs Programs
 Control Systems tutorial Control System	 Data Mining Tutorial Data Mining	 Data Warehouse Tutorial Data Warehouse	

AD