ĐẠI HỌC TÔN ĐỨC THẮNG
TON DUC THANG UNIVERSITY

503073

WEB PROGRAMMING & APPLICATIONS

LESSON 5: BOOTSTRAP

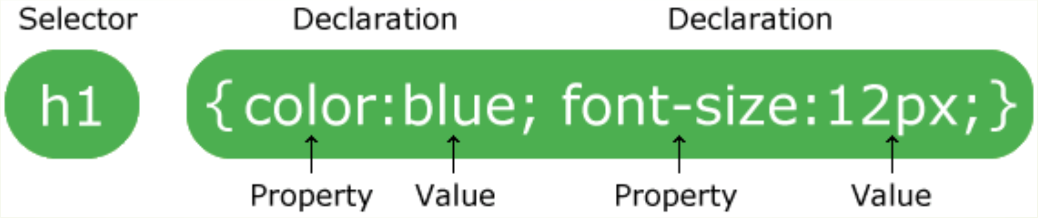
Instructor: Mai Van Manh

Introduction to Bootstrap

- CSS stands for **C**ascading **S**tyle **S**heets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media.
- CSS saves a lot of work. It can control the layout of multiple web pages all at once.
- External stylesheets are stored in ***.css** files.

CSS Syntax

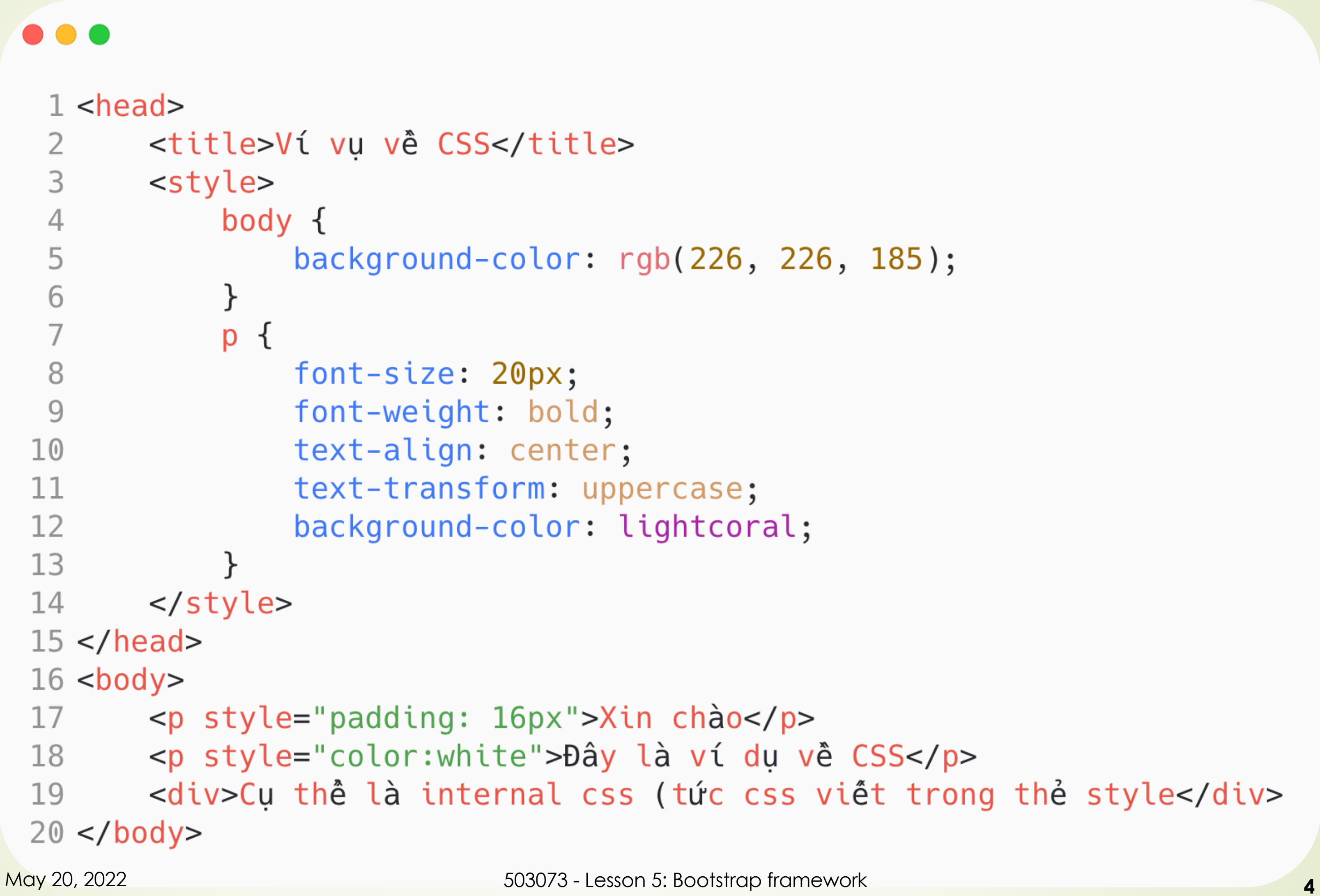
- A CSS rule-set consists of a selector and a declaration block.
  - The selector points to the HTML element you want to style.
  - The declaration block contains one or more declarations separated by semicolons.



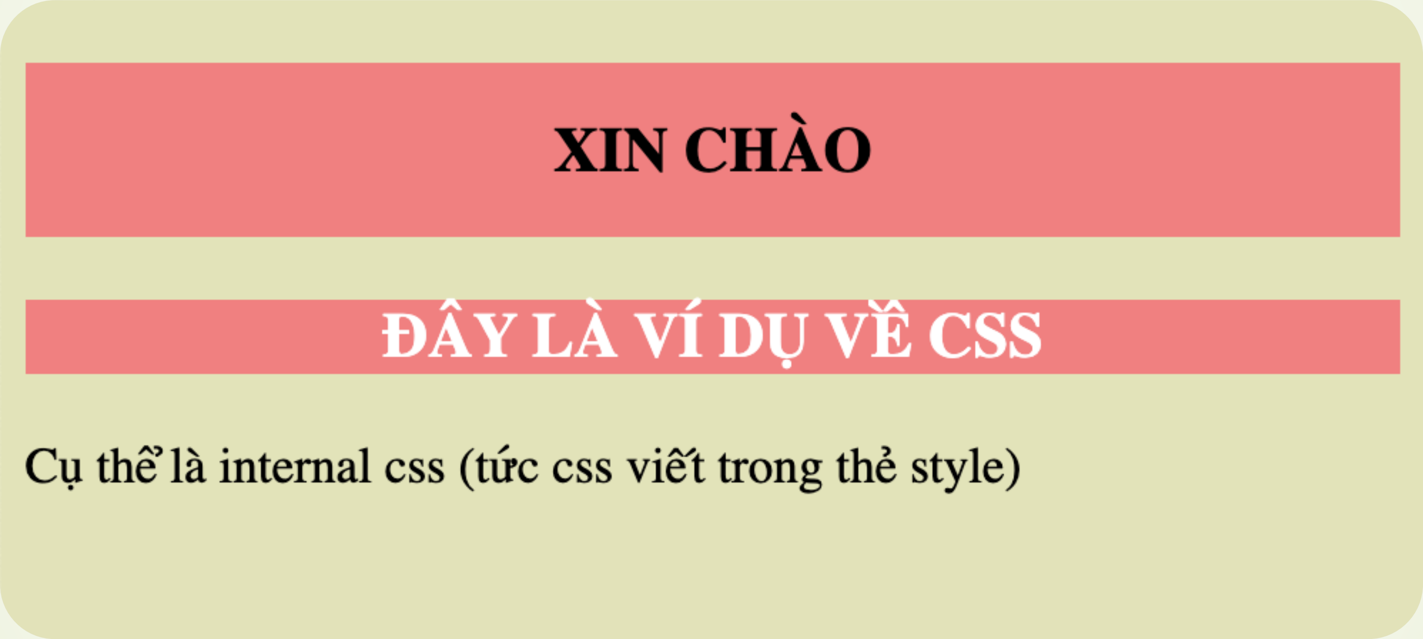
Selector Declaration Declaration

h1 {color:blue; font-size:12px;}

Property Value Property Value



```
<head>
    <title>Ví vụ về CSS</title>
    <style>
        body {
            background-color: rgb(226, 226, 185);
        }
        p {
            font-size: 20px;
            font-weight: bold;
            text-align: center;
            text-transform: uppercase;
            background-color: lightcoral;
        }
    </style>
</head>
<body>
    <p style="padding: 16px">Xin chào</p>
    <p style="color:white">Đây là ví dụ về CSS</p>
    <div>Cụ thể là internal css (tức css viết trong thẻ style</div>
</body>
```



XIN CHÀO

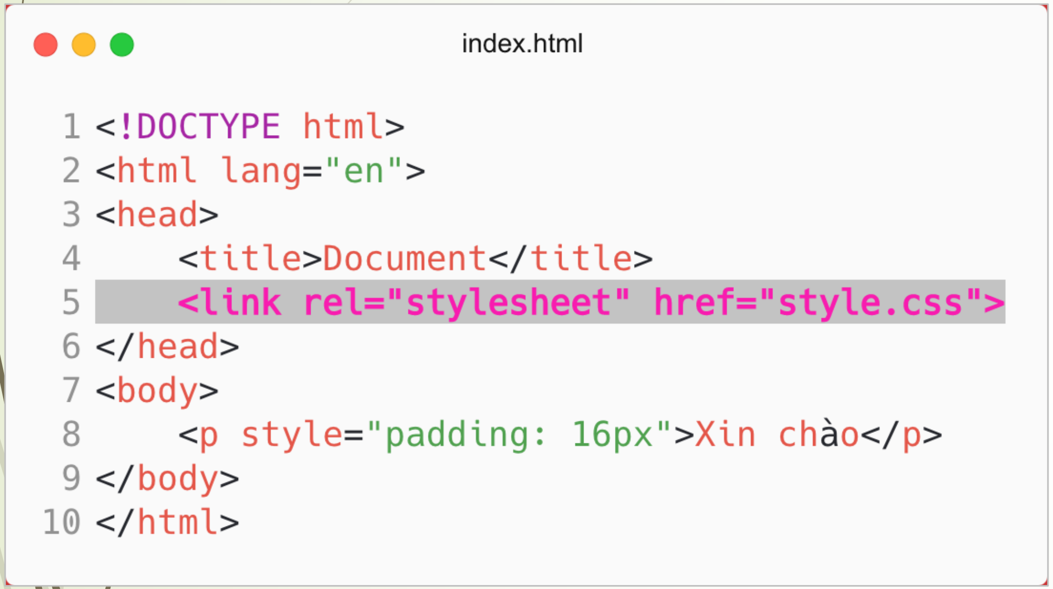
ĐÂY LÀ VÍ DỤ VỀ CSS

Cụ thể là internal css (tức css viết trong thẻ style)

Three Ways to Insert CSS

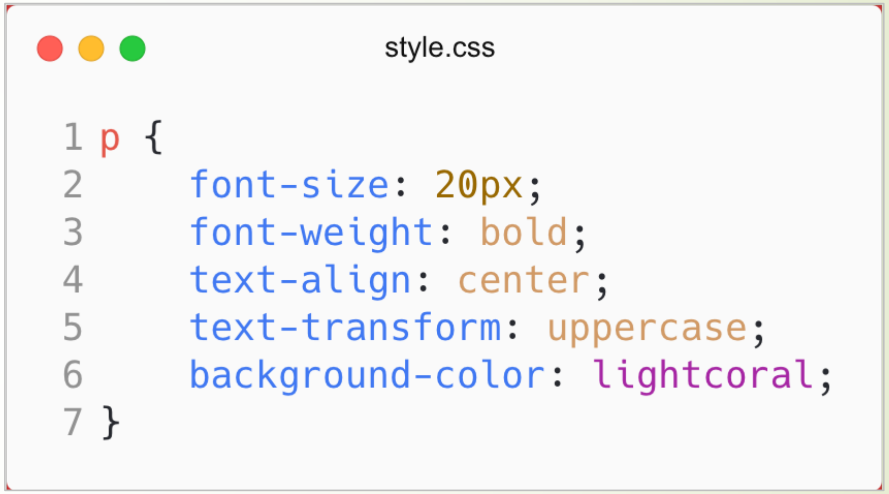
- External CSS
- Internal CSS
- Inline CSS

External CSS



index.html

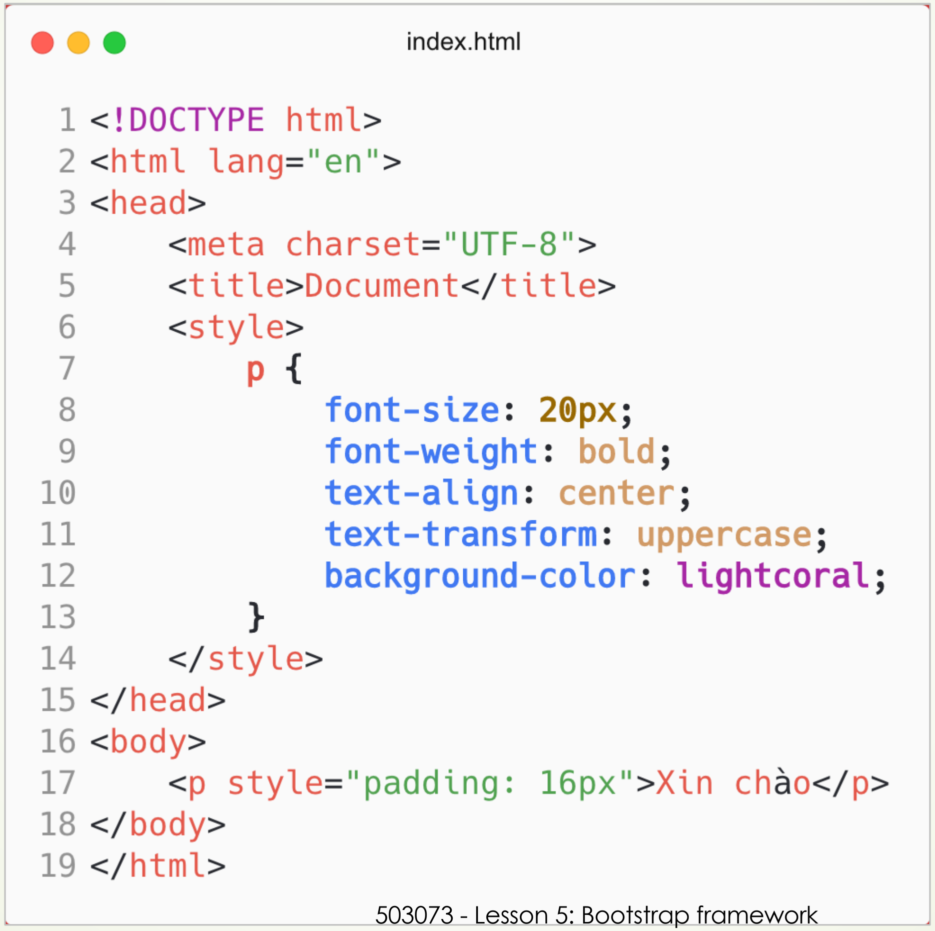
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Document</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <p style="padding: 16px">Xin chào</p>
</body>
</html>
```



style.css

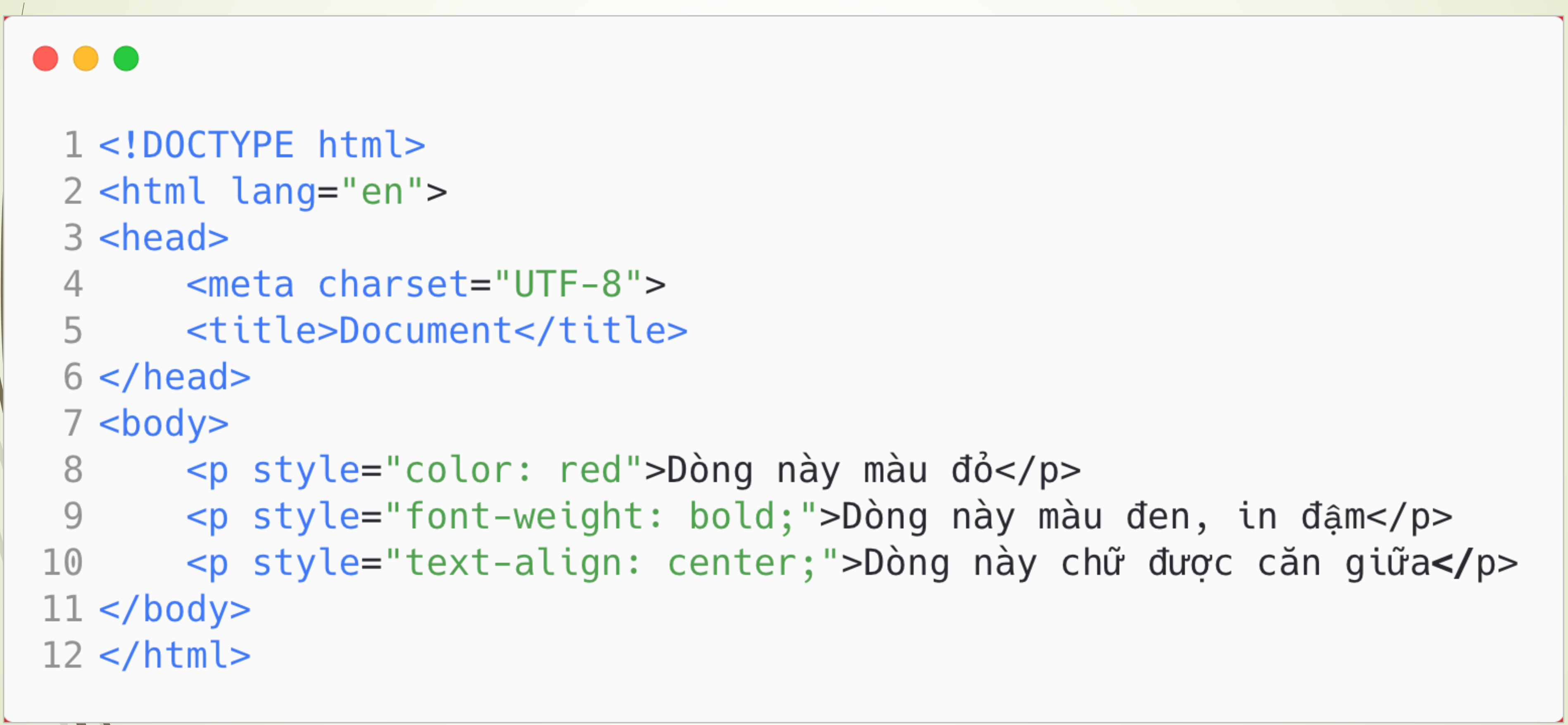
```css
p {
    font-size: 20px;
    font-weight: bold;
    text-align: center;
    text-transform: uppercase;
    background-color: lightcoral;
}
```

Internal CSS



```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Document</title>
    <style>
        p {
            font-size: 20px;
            font-weight: bold;
            text-align: center;
            text-transform: uppercase;
            background-color: lightcoral;
        }
    </style>
</head>
<body>
    <p style="padding: 16px">Xin chào</p>
</body>
</html>
```

Inline CSS



```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Document</title>
</head>
<body>
    <p style="color: red">Dòng này màu đỏ</p>
    <p style="font-weight: bold;">Dòng này màu đen, in đậm</p>
    <p style="text-align: center;">Dòng này chữ được căn giữa</p>
</body>
</html>
```

CSS Priority

1. Important CSS
2. Inline style (inside an HTML element)
3. External and internal style sheets (in the head section)
4. Browser default

CSS Selectors

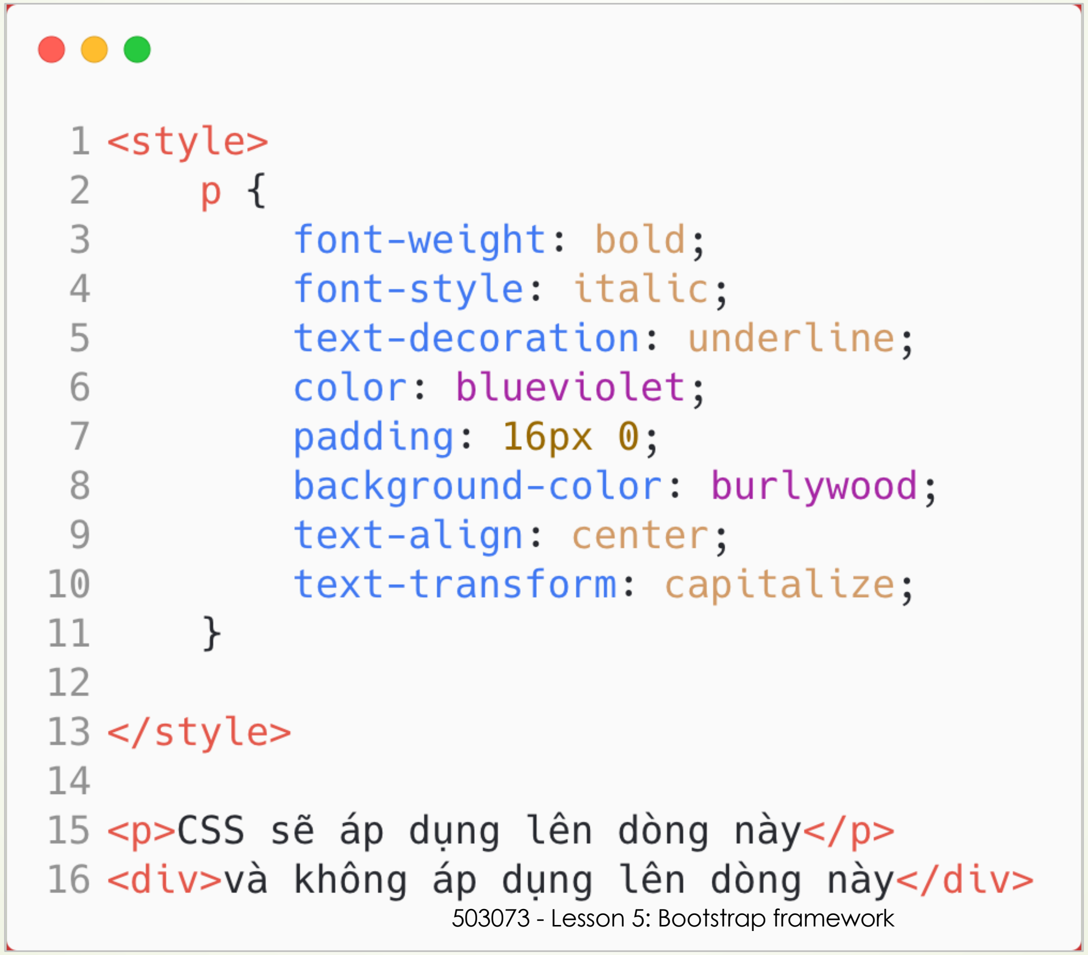
- We can divide CSS selectors into five categories:
  1. Simple selectors: select elements based on name, id, class
  2. Combinator selectors: based on a specific relationship between tags.
  3. Pseudo-class selectors: based on a certain state
  4. Pseudo-elements selectors: select and style a part of an element
  5. Attribute selectors: based on an attribute or attribute value

SIMPLE SELECTORS

Simple Selectors

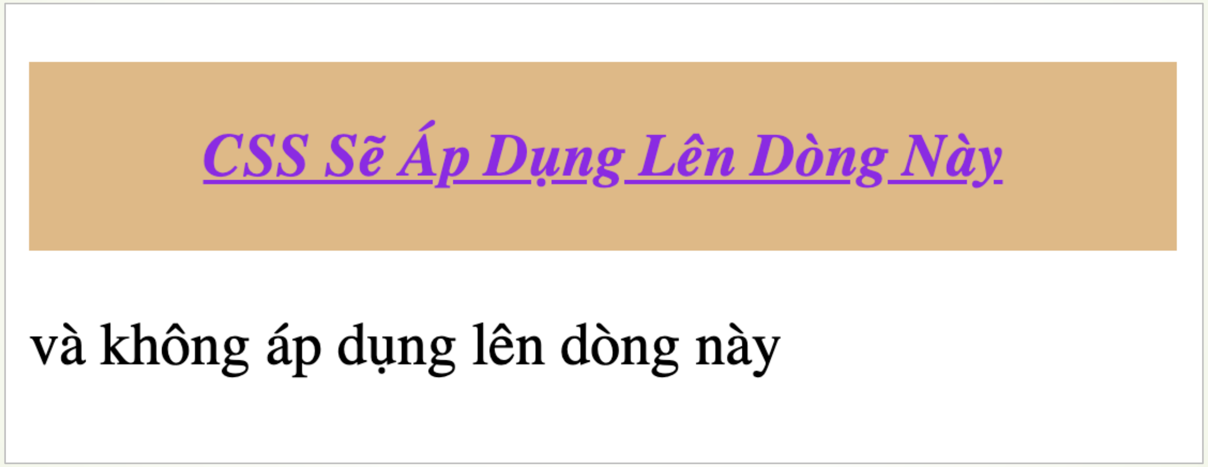
- Name selector
- Id Selector
- Class selector

Name Selectors



```
<style>
    p {
        font-weight: bold;
        font-style: italic;
        text-decoration: underline;
        color: blueviolet;
        padding: 16px 0;
        background-color: burlywood;
        text-align: center;
        text-transform: capitalize;
    }

</style>

<p>CSS sẽ áp dụng lên dòng này</p>
<div>và không áp dụng lên dòng này</div>
```

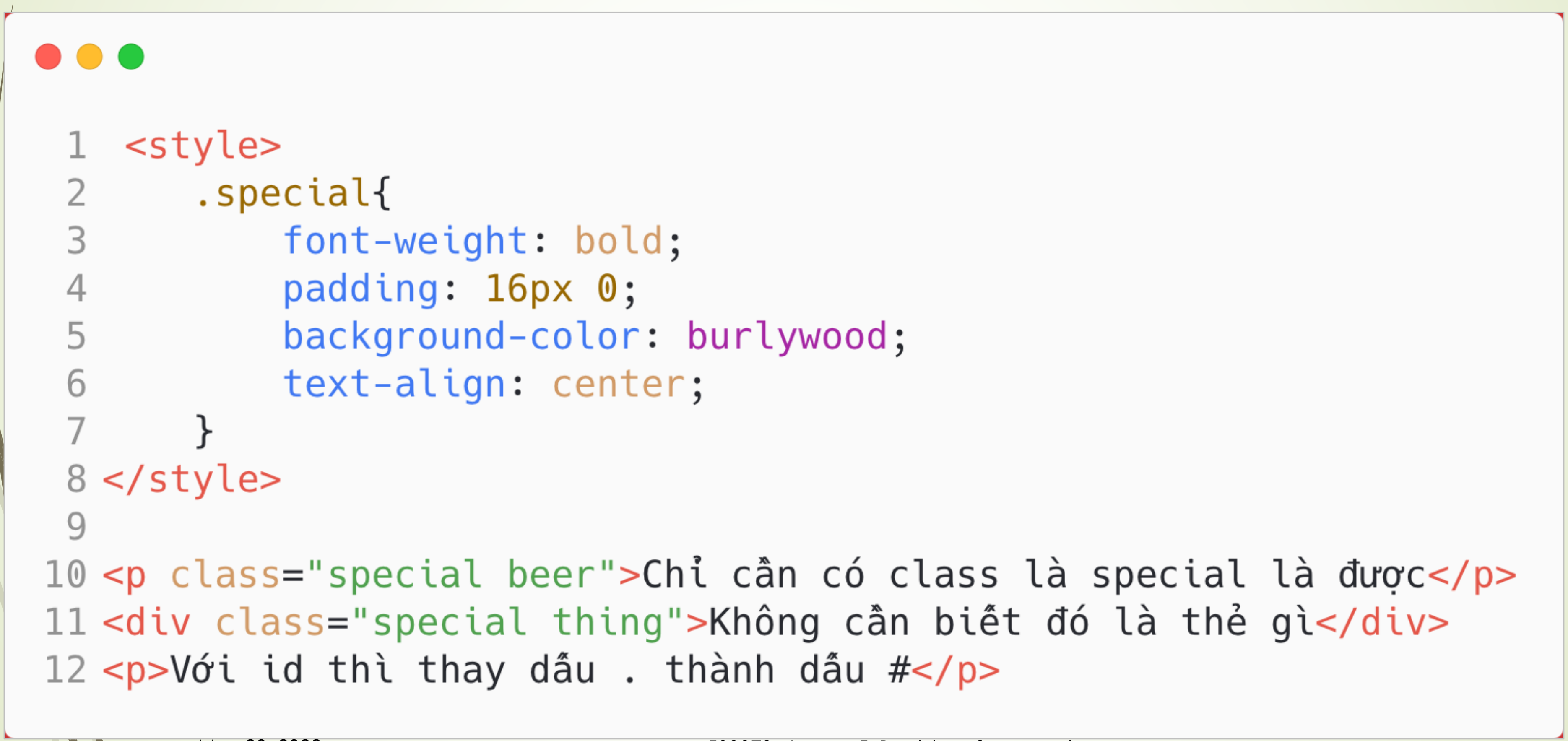
Name Selectors



CSS Sẽ Áp Dụng Lên Dòng Này

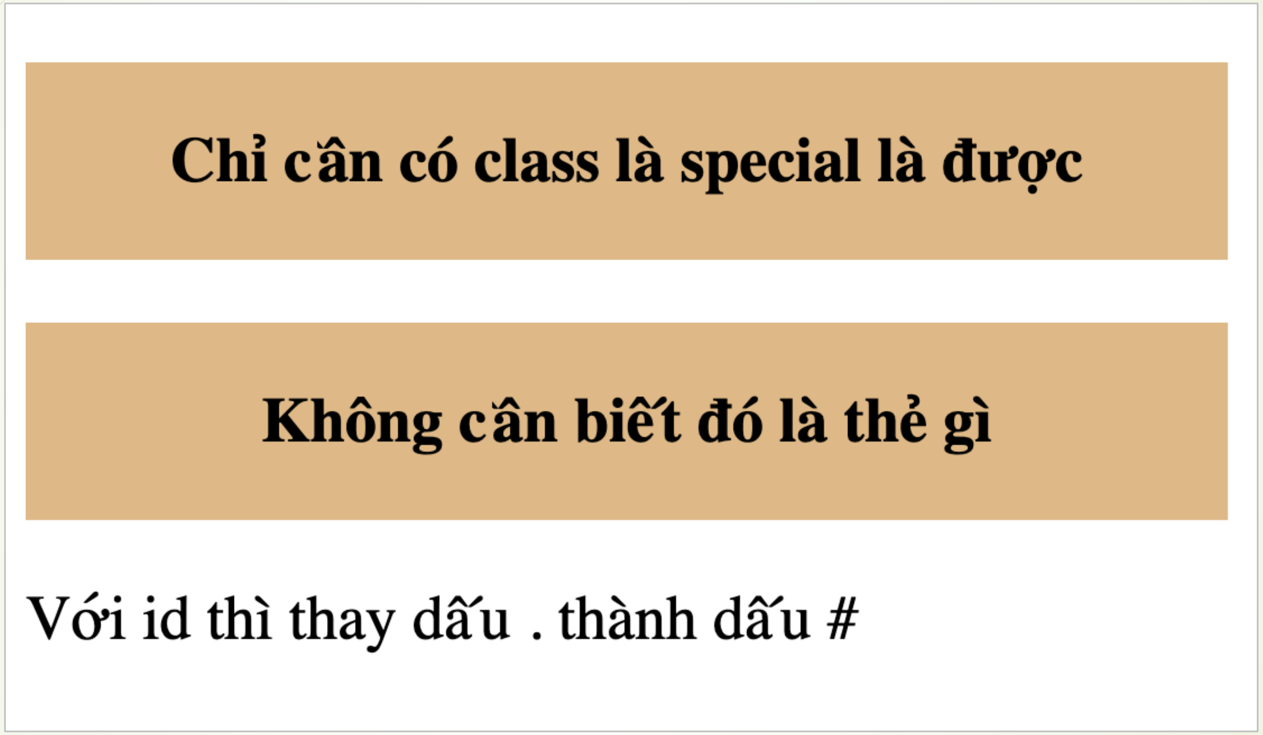
và không áp dụng lên dòng này

Id & Class Selectors



```
<style>
    .special{
        font-weight: bold;
        padding: 16px 0;
        background-color: burlywood;
        text-align: center;
    }
</style>

<p class="special beer">Chỉ cần có class là special là được</p>
<div class="special thing">Không cần biết đó là thẻ gì</div>
<p>Với id thì thay dấu . thành dấu #</p>
```

Id & Class Selectors



Chỉ cần có class là special là được

Không cần biết đó là thẻ gì

Với id thì thay dấu . thành dấu #

COMBINATOR SELECTORS

Combinator Selectors

- Descendant selector (space)
- Child selector (>)
- Adjacent sibling selector (+)
- General sibling selector (~)

PSEUDO-CLASS SELECTORS

Pseudo-class Selectors

- A pseudo-class is used to define a special state of an element.
- Examples:
  - Anchor Pseudo-classes: a:link, a:visited, a:active
  - :hover
  - :first-child, :last-child
  - :checked, :empty, :disabled, :focus, :invalid
  - :nth-child(n), :last-of-type, :not(selector)
- https://www.w3schools.com/css/css_pseudo_classes.asp

CSS PSEUDO-ELEMENTS

CSS Pseudo-elements

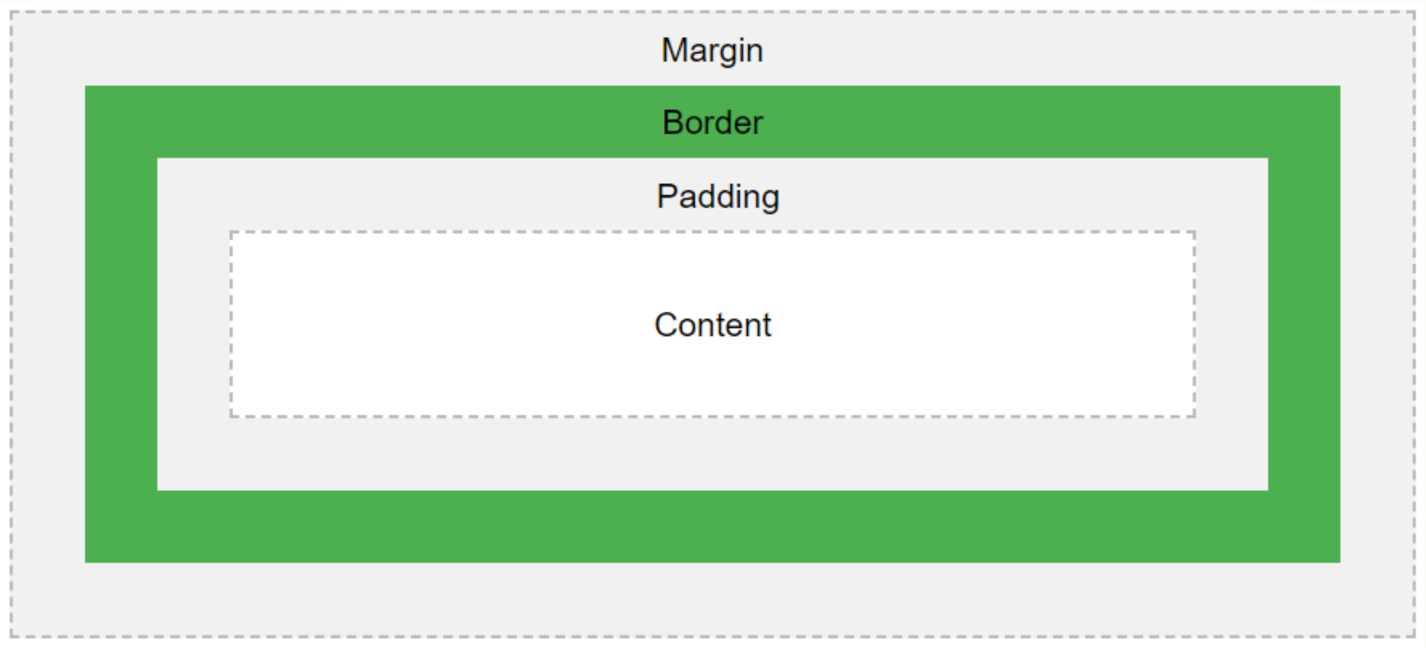
- A CSS pseudo-element is used to style specified parts of an element.
- For example, it can be used to:
  - Style the first letter, or line, of an element
  - Insert content before, or after, the content of an element.

CSS Pseudo-elements

- ::first-line
- ::first-letter
- ::before
- ::after
- ::selection

Box Model

- All HTML elements can be considered as boxes.
- It consists of: margins, borders, padding, and the actual content.



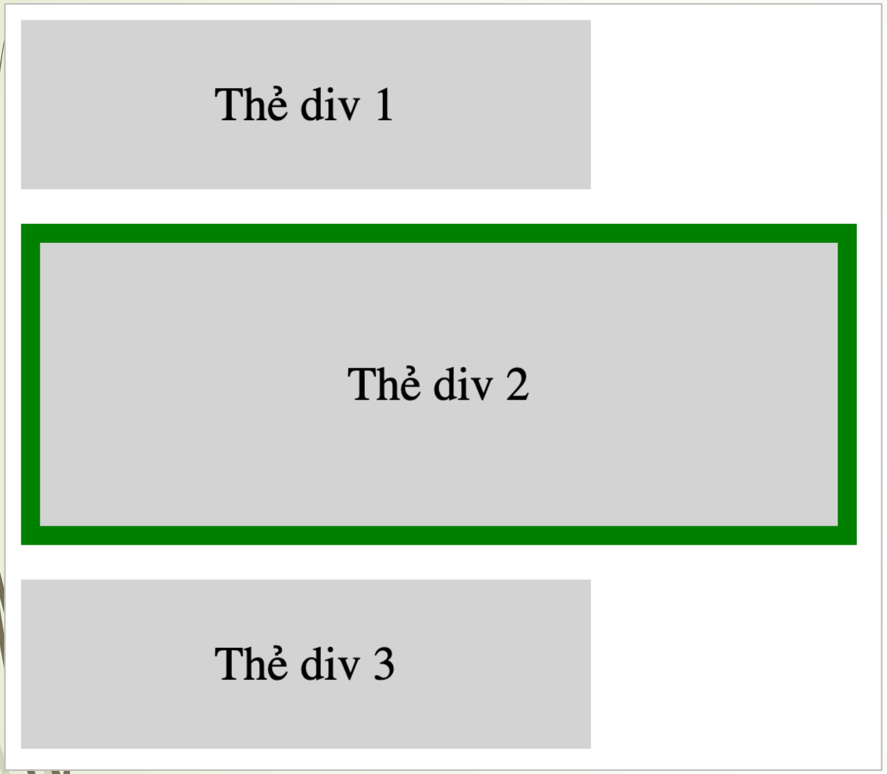
Box Model: Width and Height

- By default:
  - When you set the width and height properties of an element with CSS, you just set the width and height of the content area.
  - To calculate the full size of an element, you must also add **padding**, and **borders.**

(by default)

Width = Content width

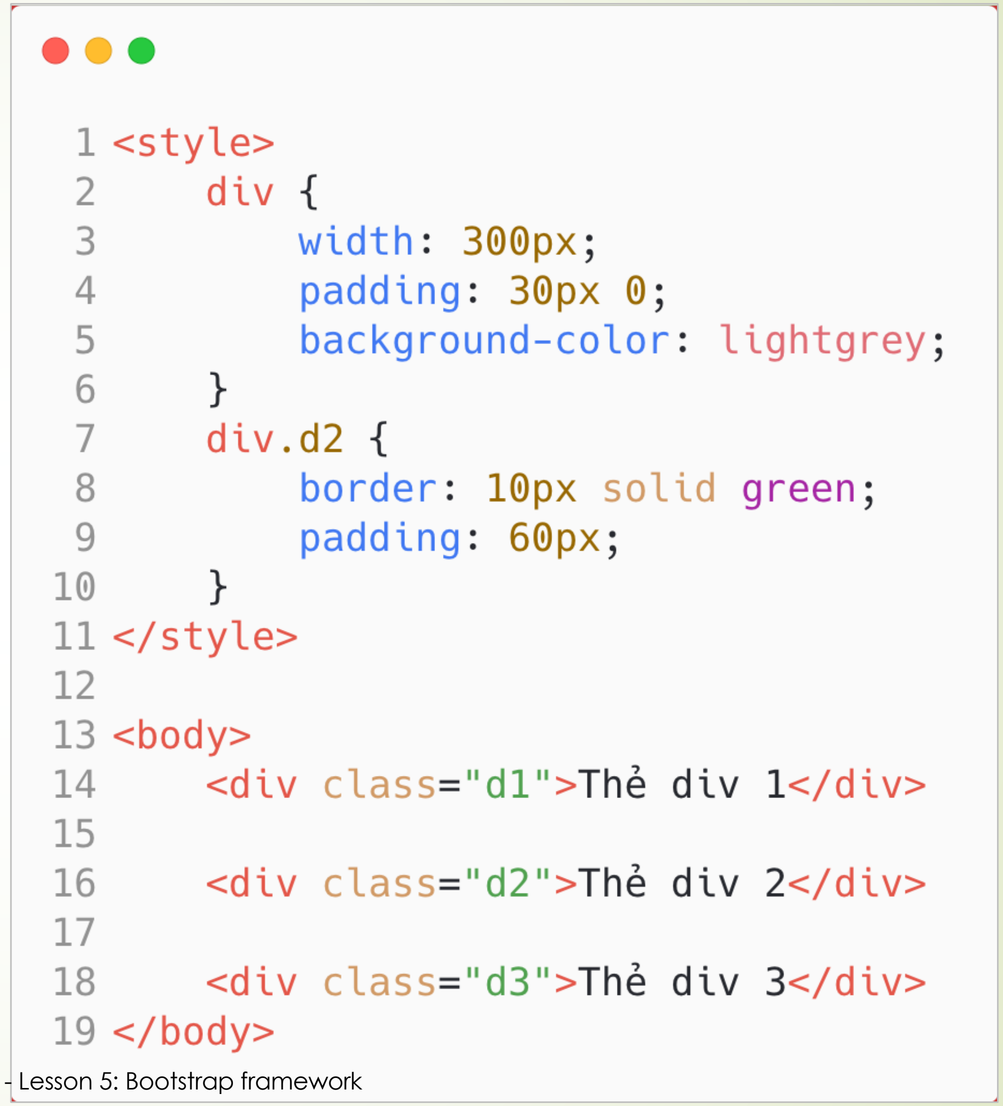
~~Width = Content width + padding + border~~



Thẻ div 1

Thẻ div 2

Thẻ div 3



```
<style>
    div {
        width: 300px;
        padding: 30px 0;
        background-color: lightgrey;
    }
    div.d2 {
        border: 10px solid green;
        padding: 60px;
    }
</style>

<body>
    <div class="d1">Thẻ div 1</div>

    <div class="d2">Thẻ div 2</div>

    <div class="d3">Thẻ div 3</div>
</body>
```

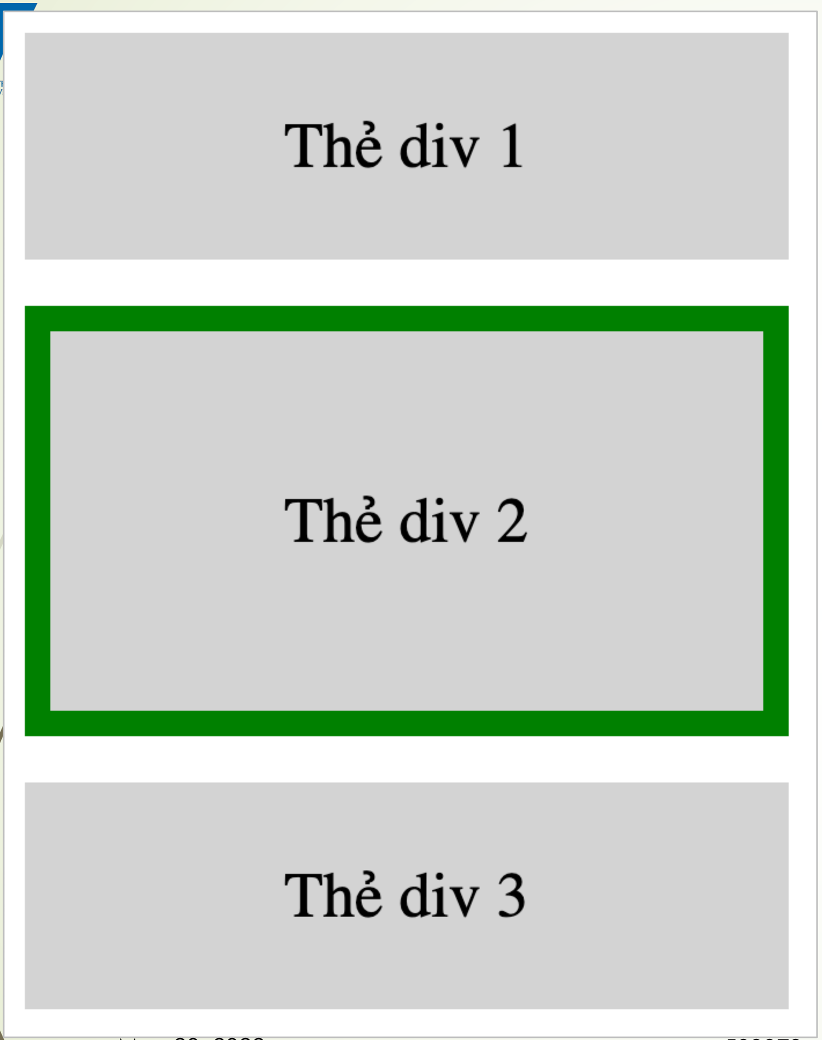
Box Model: Box sizing

- Setting the css box-sizing property to border-box makes sure that the padding and border are included in the total width and height of the elements.

(box-sizing: border-box)

~~Width = Content width~~

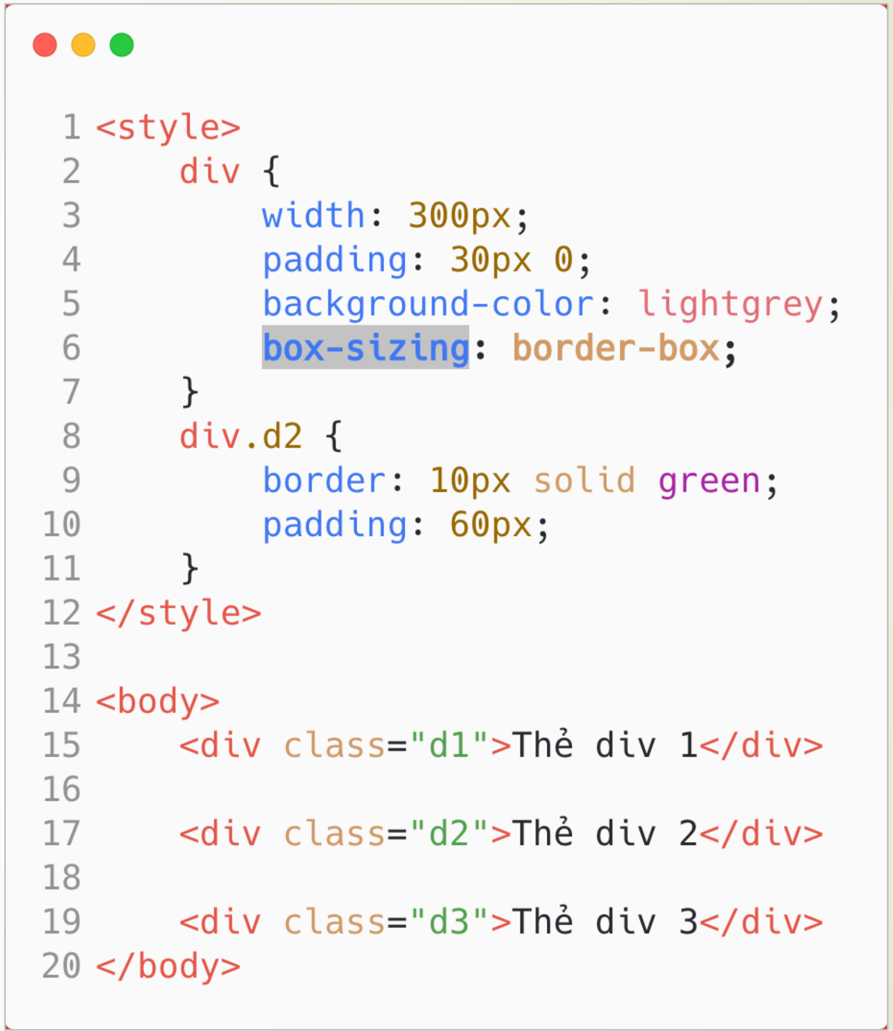
Width = Content width + padding + border



Thẻ div 1

Thẻ div 2

Thẻ div 3



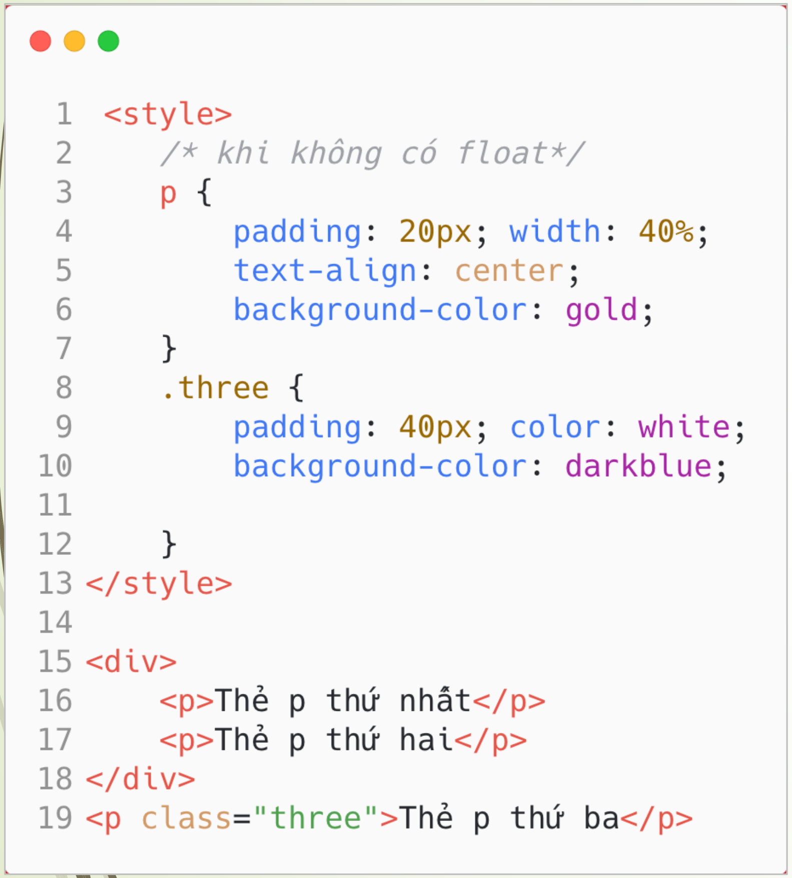
```
<style>
    div {
        width: 300px;
        padding: 30px 0;
        background-color: lightgrey;
        box-sizing: border-box;
    }
    div.d2 {
        border: 10px solid green;
        padding: 60px;
    }
</style>

<body>
    <div class="d1">Thẻ div 1</div>

    <div class="d2">Thẻ div 2</div>

    <div class="d3">Thẻ div 3</div>
</body>
```

FLOATING

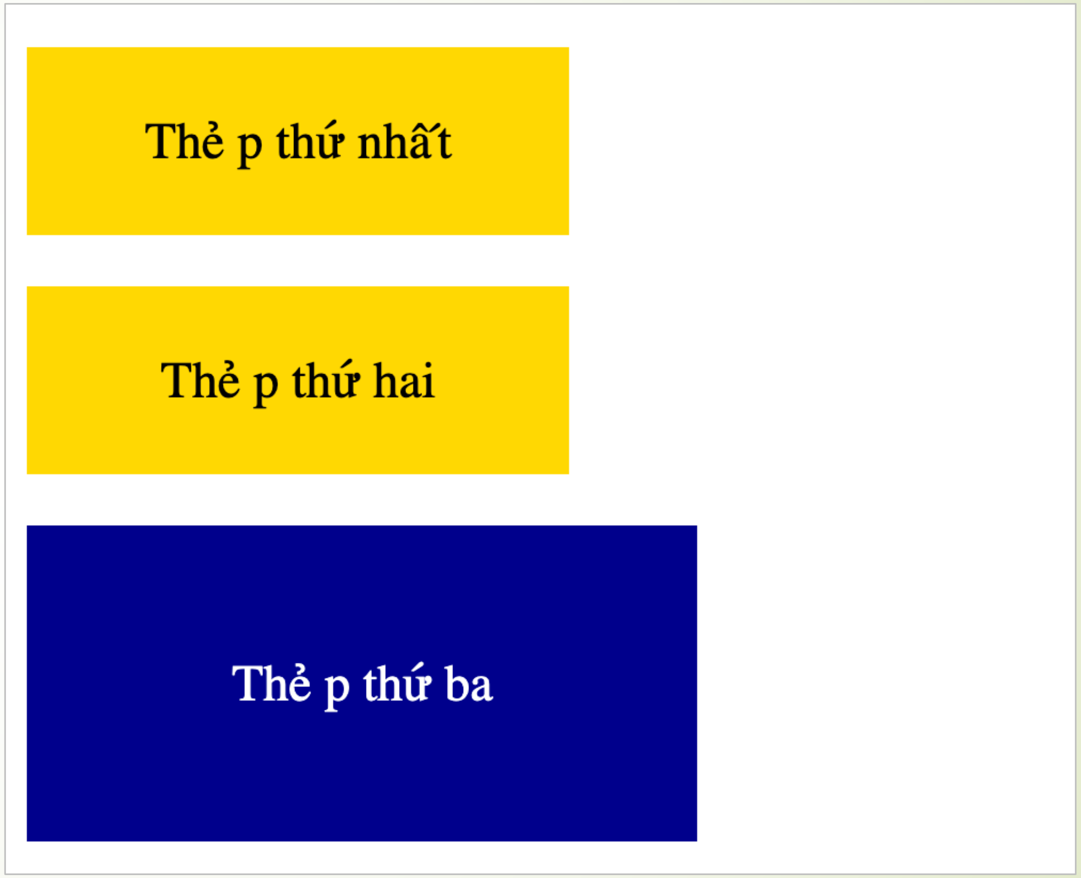
CSS Float

- The CSS float property specifies how an element should float.
- The CSS clear property specifies what elements can float beside the cleared element and on which side.

Without floating



```
<style>
    /* khi không có float*/
    p {
        padding: 20px; width: 40%;
        text-align: center;
        background-color: gold;
    }
    .three {
        padding: 40px; color: white;
        background-color: darkblue;

    }
</style>

<div>
    <p>Thẻ p thứ nhất</p>
    <p>Thẻ p thứ hai</p>
</div>
<p class="three">Thẻ p thứ ba</p>
```

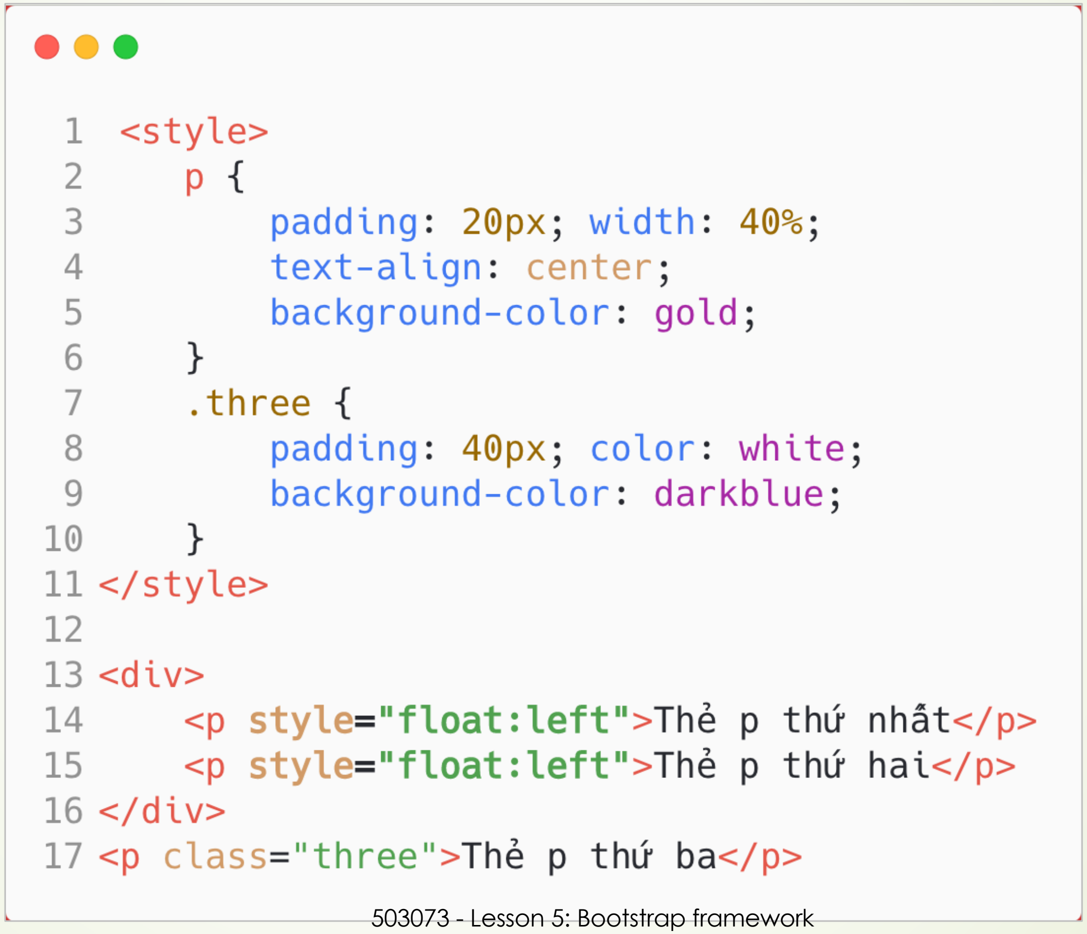


Thẻ p thứ nhất

Thẻ p thứ hai

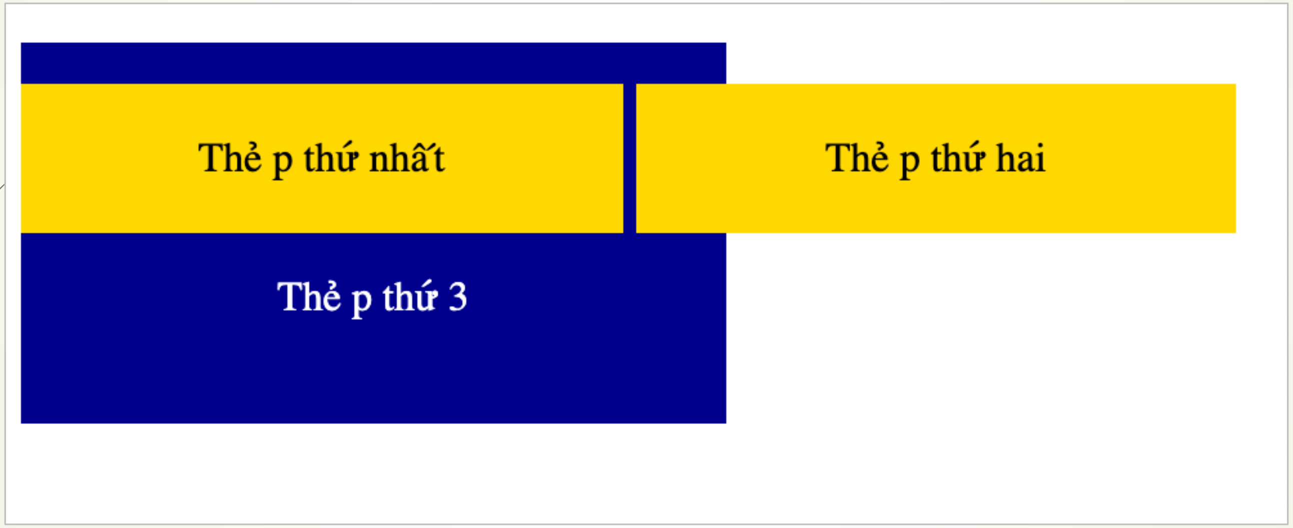
Thẻ p thứ ba

With float



```
<style>
    p {
        padding: 20px; width: 40%;
        text-align: center;
        background-color: gold;
    }
    .three {
        padding: 40px; color: white;
        background-color: darkblue;
    }
</style>

<div>
    <p style="float:left">Thẻ p thứ nhất</p>
    <p style="float:left">Thẻ p thứ hai</p>
</div>
<p class="three">Thẻ p thứ ba</p>
```

With float

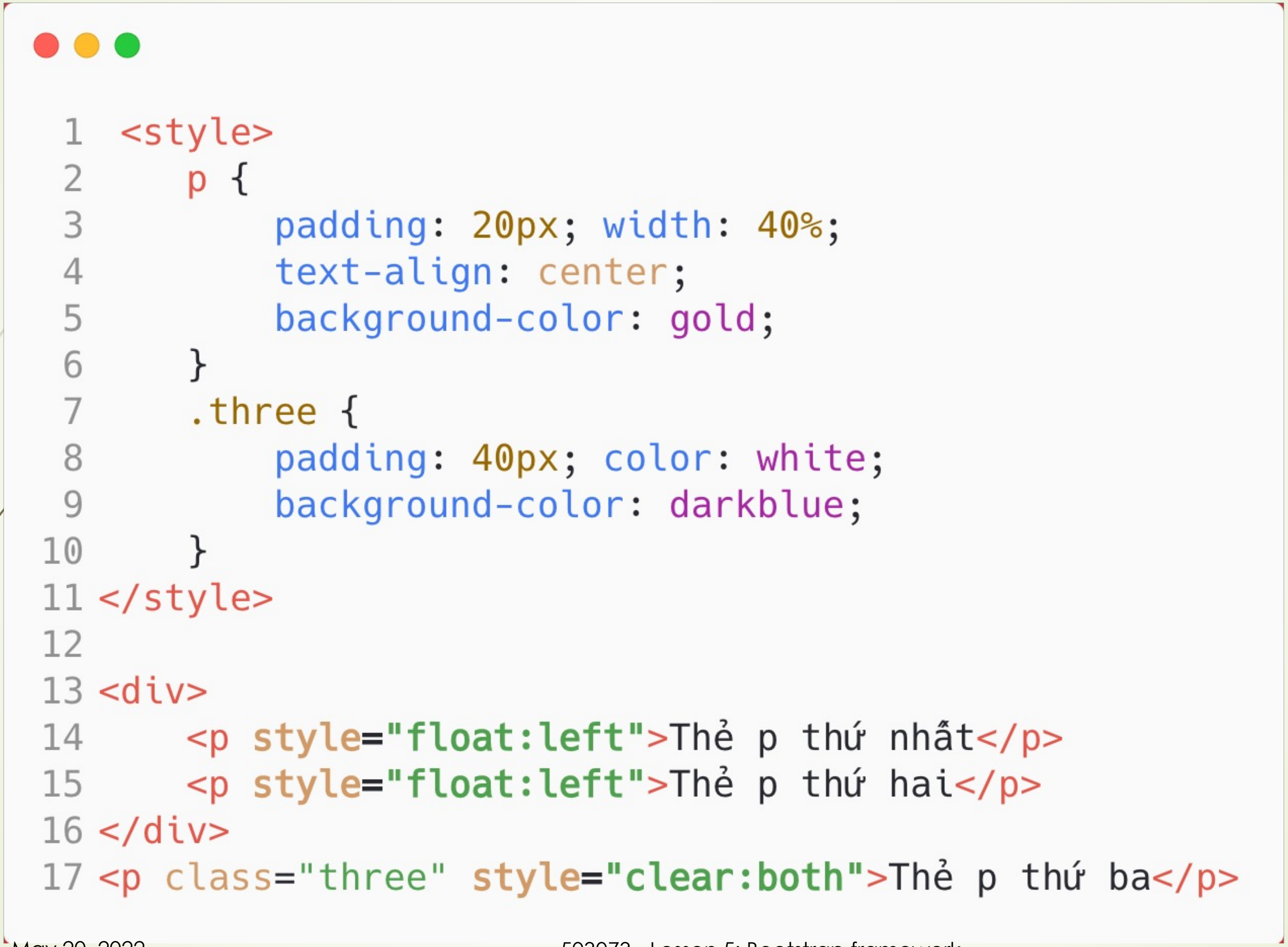


Thẻ p thứ nhất

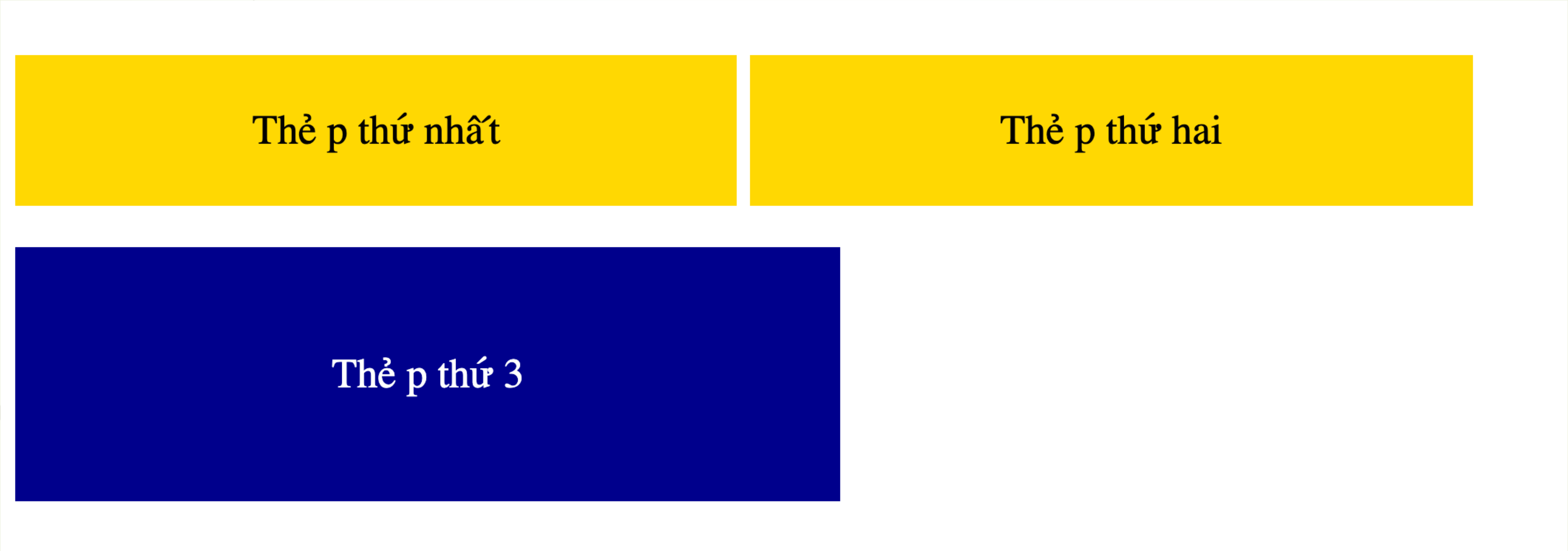
Thẻ p thứ hai

Thẻ p thứ 3

THE CLEAR PROPERTY



```
 <style>
    p {
        padding: 20px; width: 40%;
        text-align: center;
        background-color: gold;
    }
    .three {
        padding: 40px; color: white;
        background-color: darkblue;
    }
</style>

<div>
    <p style="float:left">Thẻ p thứ nhất</p>
    <p style="float:left">Thẻ p thứ hai</p>
</div>
<p class="three" style="clear:both">Thẻ p thứ ba</p>
```



Thẻ p thứ nhất

Thẻ p thứ hai

Thẻ p thứ 3

WATCH A DEMO VIDEO

Responsive Web Design

Responsive Web Design

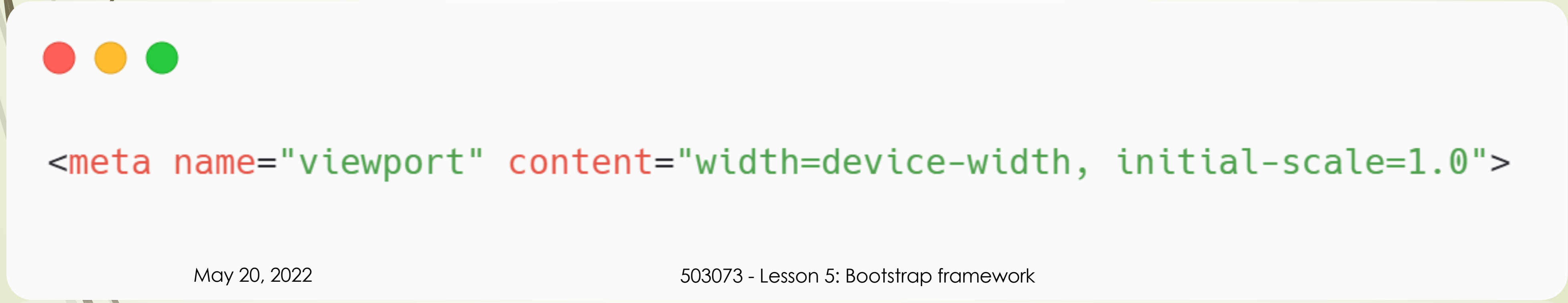
- Responsive web design makes your web page look good on all devices.
- Responsive web design uses only HTML and CSS.
- Responsive web design is not a program or a JavaScript.

Viewport

- The viewport is the user's visible area of a web page.
- The viewport varies with the device, and will be smaller on a mobile phone than on a computer screen.
- Before tablets and mobile phones, web pages were designed only for computer screens, and it was common for web pages to have a static design and a fixed size.
- Then, when we started surfing the internet using tablets and mobile phones, fixed size web pages were too large to fit the viewport. To fix this, browsers on those devices scaled down the entire web page to fit the screen.

Setting the Viewport

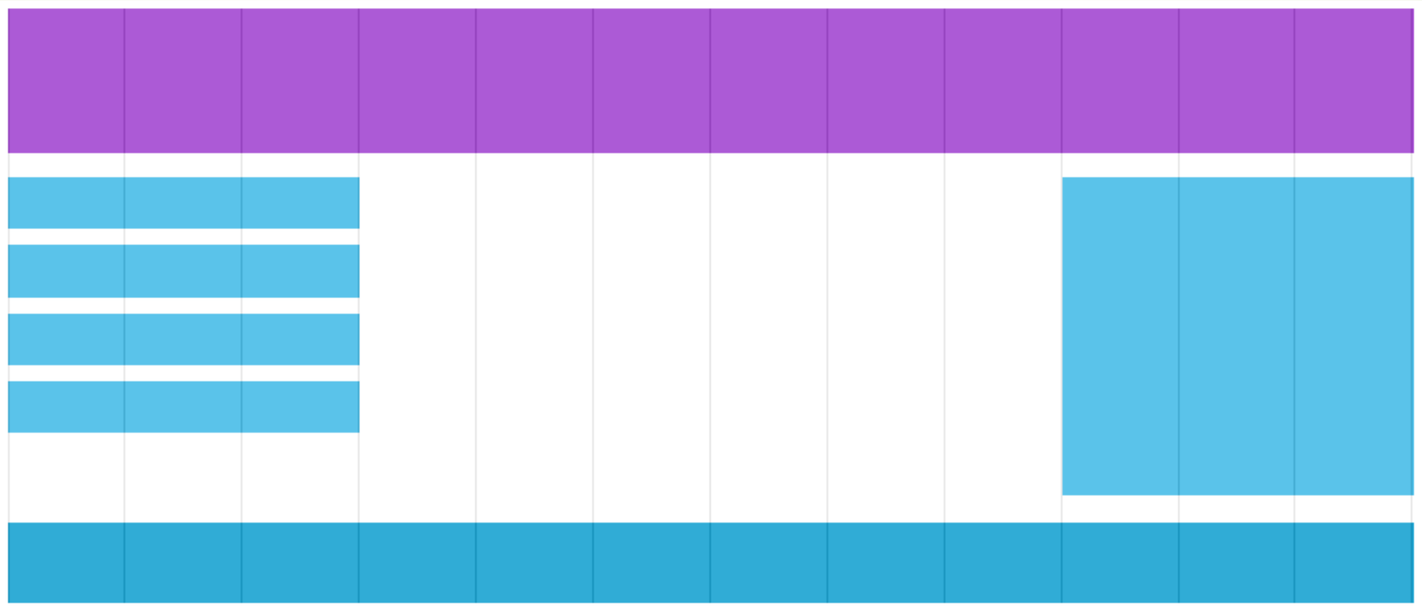
- The width=device-width part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).
- The initial-scale=1.0 part sets the initial zoom level when the page is first loaded by the browser.



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

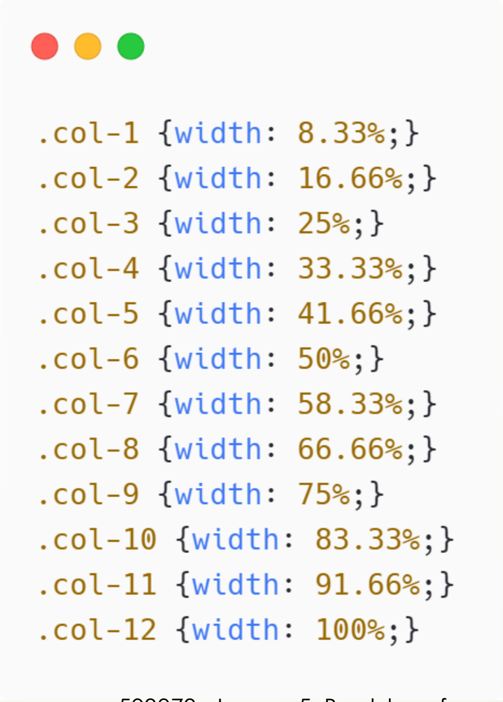
Gridview

- Many web pages are based on a grid-view, which means that the page is divided into columns.



Gridview

- A responsive grid-view often has 12 columns, and has a total width of 100%, and will shrink and expand as you resize the browser window.



```
.col-1 {width: 8.33%;}
.col-2 {width: 16.66%;}
.col-3 {width: 25%;}
.col-4 {width: 33.33%;}
.col-5 {width: 41.66%;}
.col-6 {width: 50%;}
.col-7 {width: 58.33%;}
.col-8 {width: 66.66%;}
.col-9 {width: 75%;}
.col-10 {width: 83.33%;}
.col-11 {width: 91.66%;}
.col-12 {width: 100%;}
```