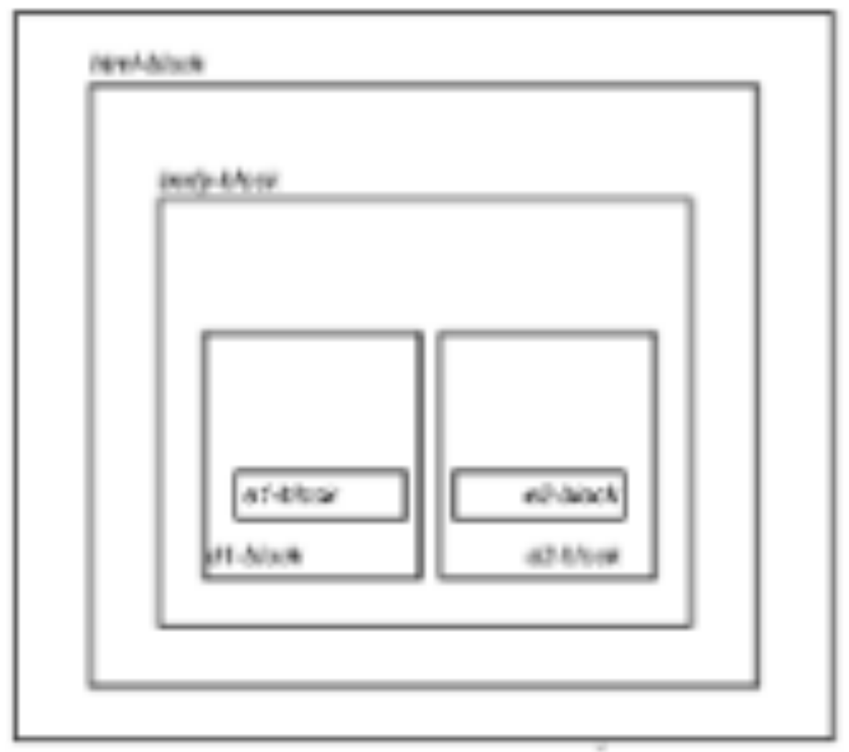
Box Layout

Box Layout



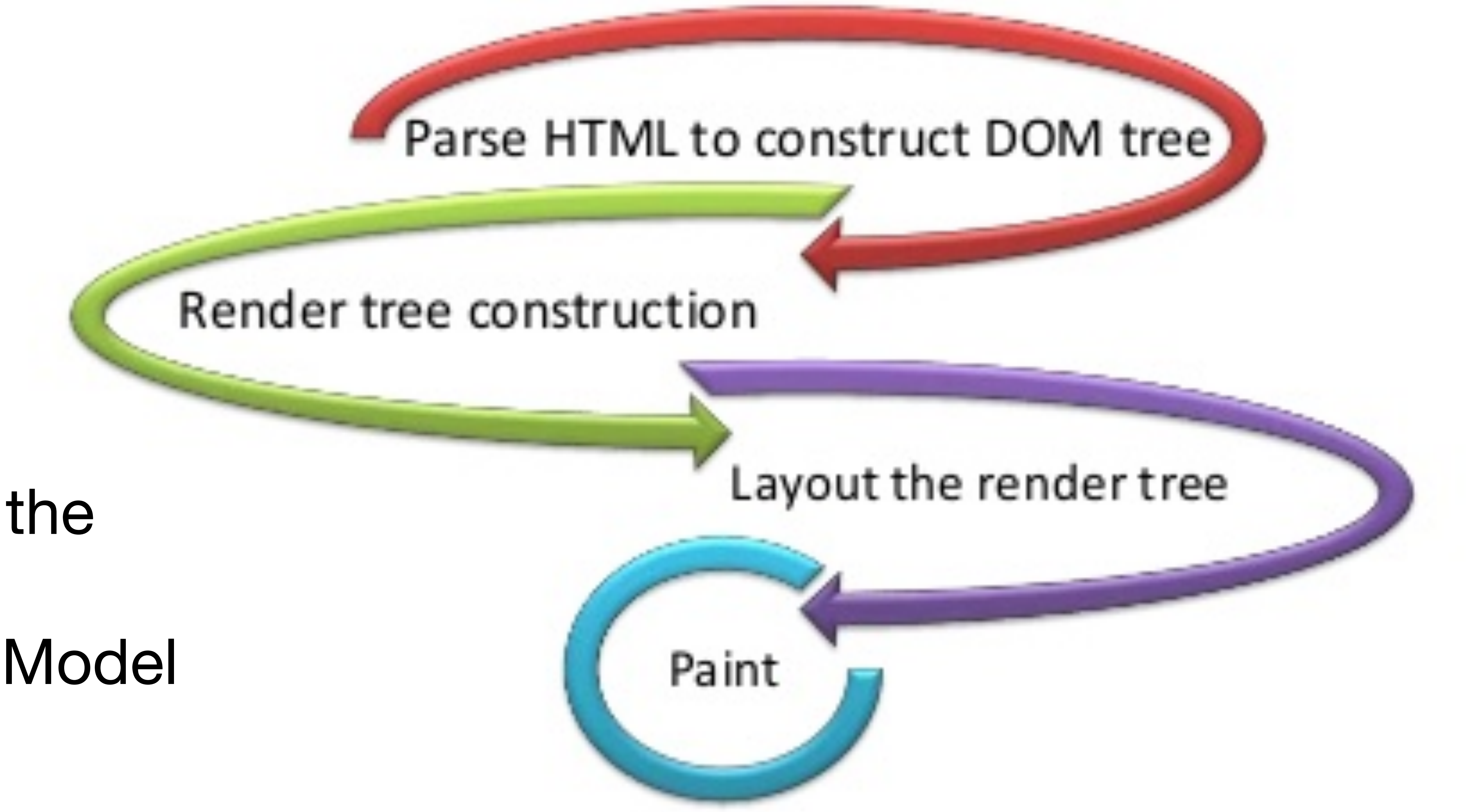
Some important characteristics of the box model layout rules.

Agenda

- The DOM, HTML Parsing & Rendering
- Formatting concepts in CSS
- A simple, worked example of multicolumn layout
- A worked example of a “tabbed” multicolumn layout

Parsing & Rendering

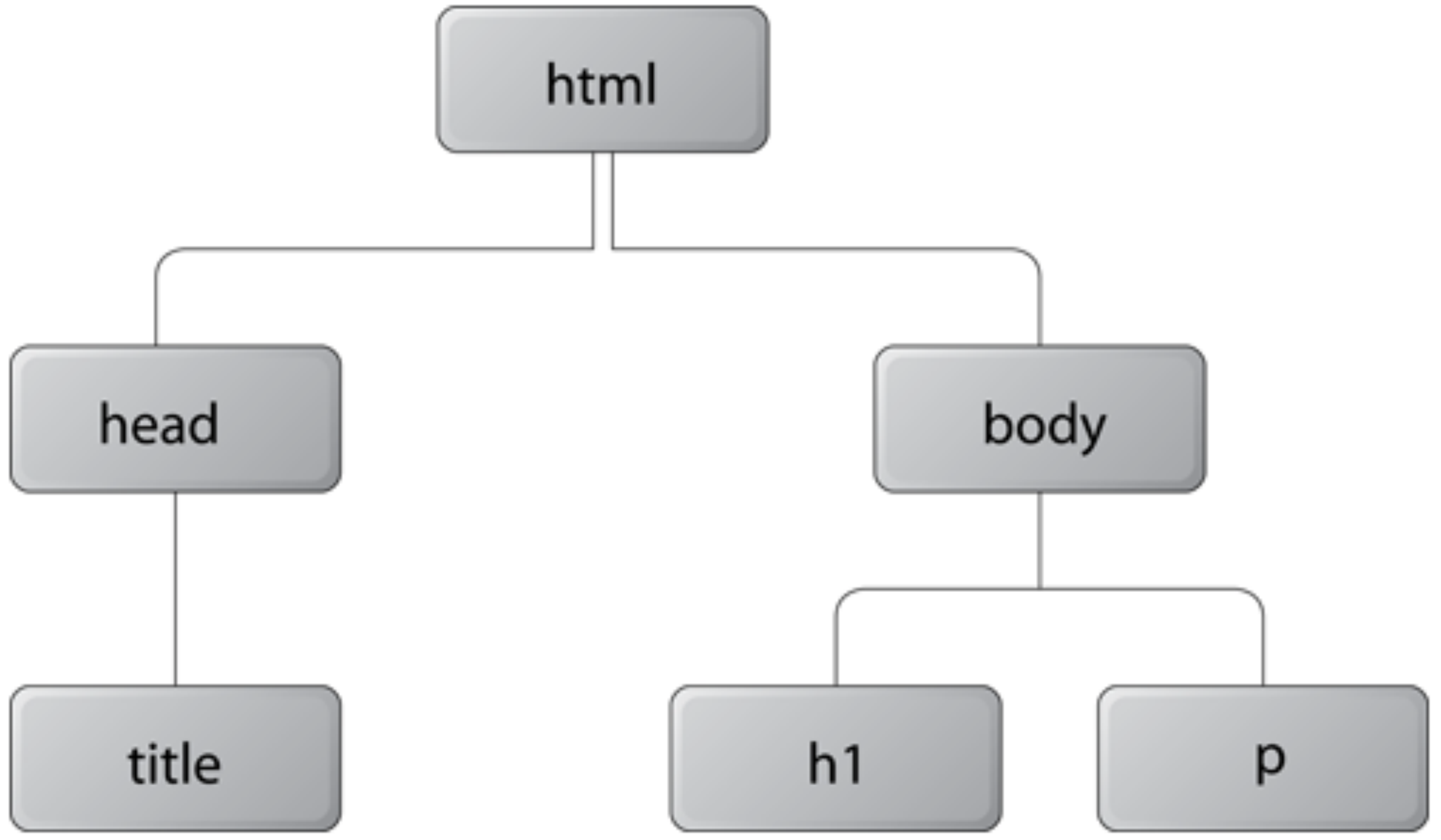
- A web browser typically reads and renders HTML documents in two phases:
  - the parsing phase
  - the rendering phase.
- During the parsing phase, the browser reads the markup in the document, breaks it down into components, and builds a Document Object Model (DOM) tree.
- The DOM is a in-memory data structure, typically traversed by Java Script code



DOM Tree

```
<!DOCTYPE html>
<html>
  <head>
    <title>Widgets</title>
  </head>
  <body>
    <h1>Widgets</h1>
    <p>Welcome to Widgets, the number one company
      in the world for selling widgets!</p>
  </body>
</html>
```

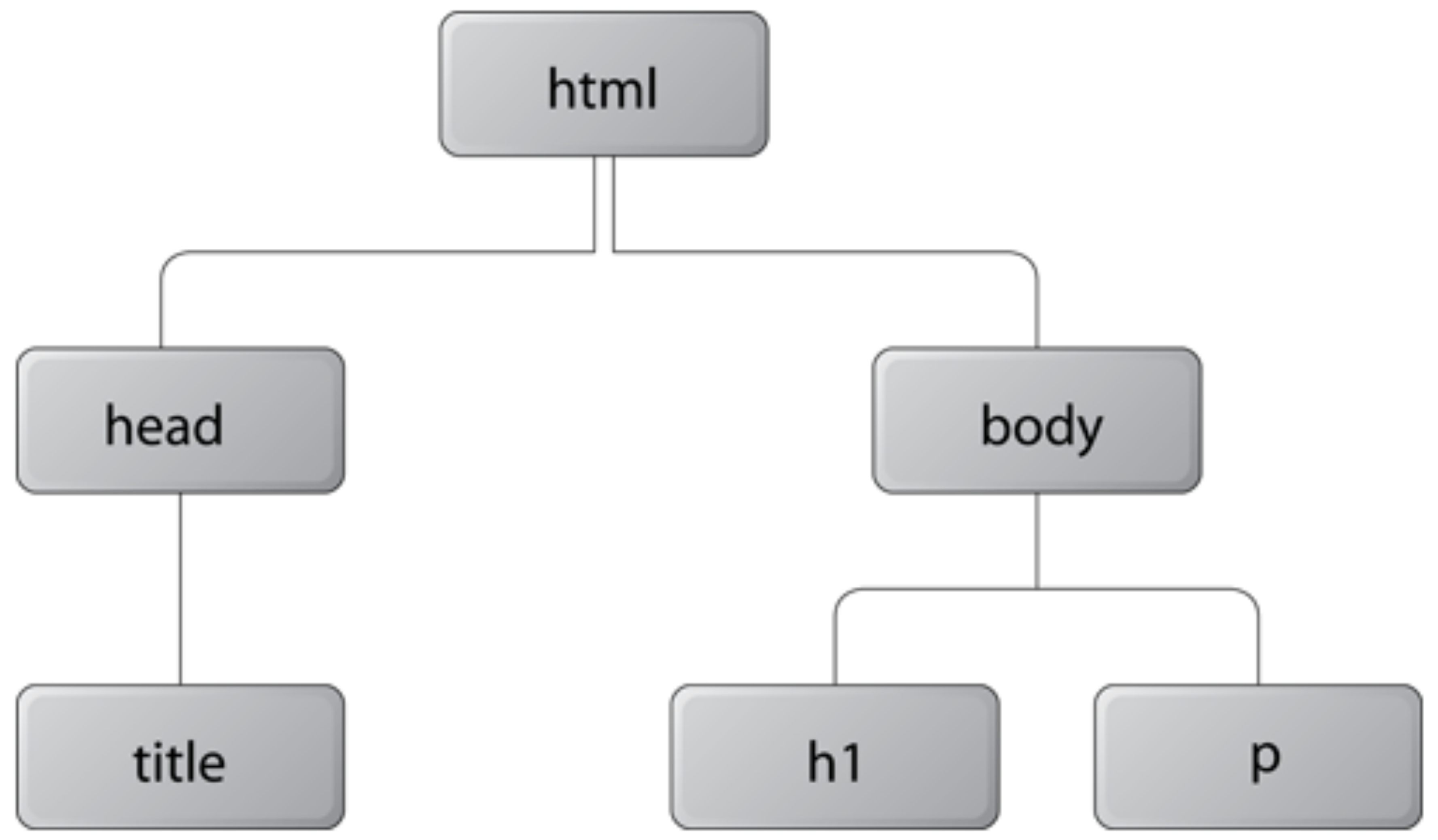
- Each object in the DOM tree is called a node.
- At the top of the tree is a document node, which contains an element node called the root node
- It branches into two child nodes—head and body—which then branch into other children.



Parents, Grandparents, Children, Grandchildren, Siblings

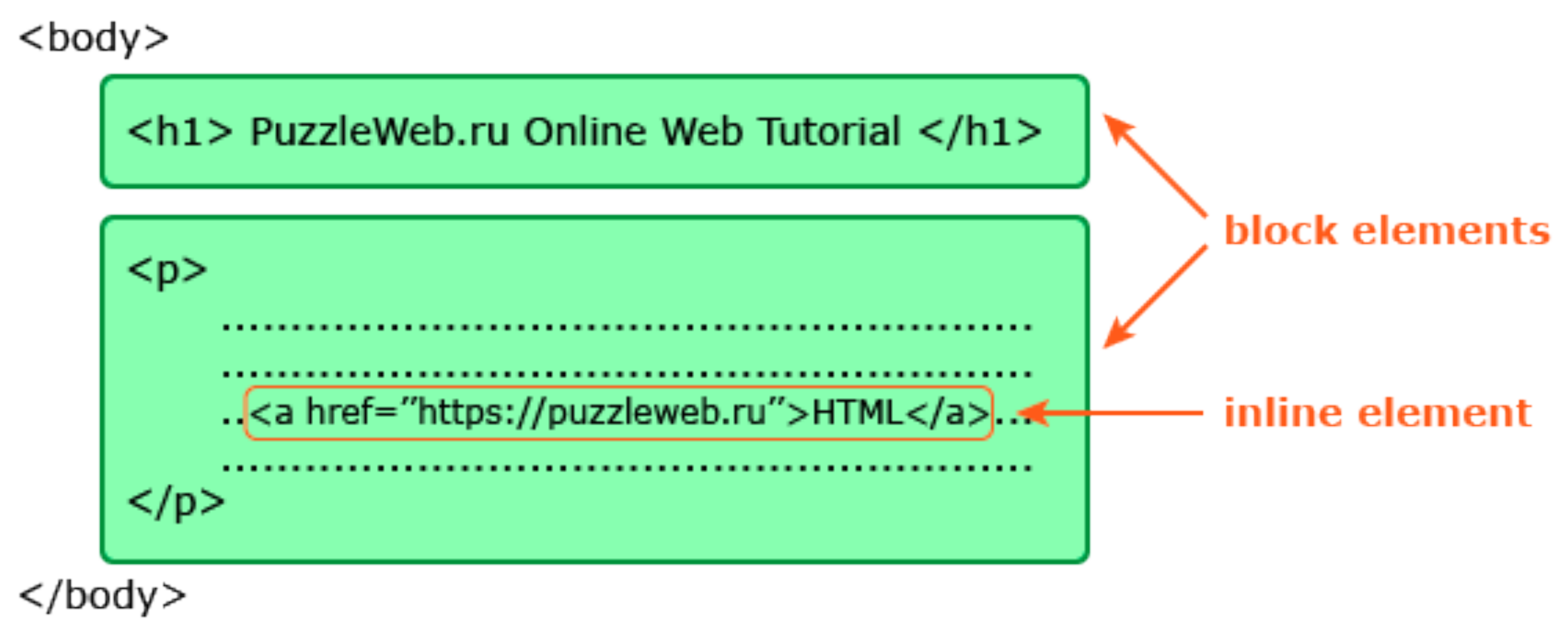
Inheritance

- A child node is structurally subordinate to its parent node - the child's tags are nested inside the tags of the parent.
- A node can be called a descendant node if it's a child, grandchild, and so on, of another node.
- A node can be called an ancestor node if it's a parent, grandparent, and so on, of another node.
- Nodes that have the same parent are called siblings. .

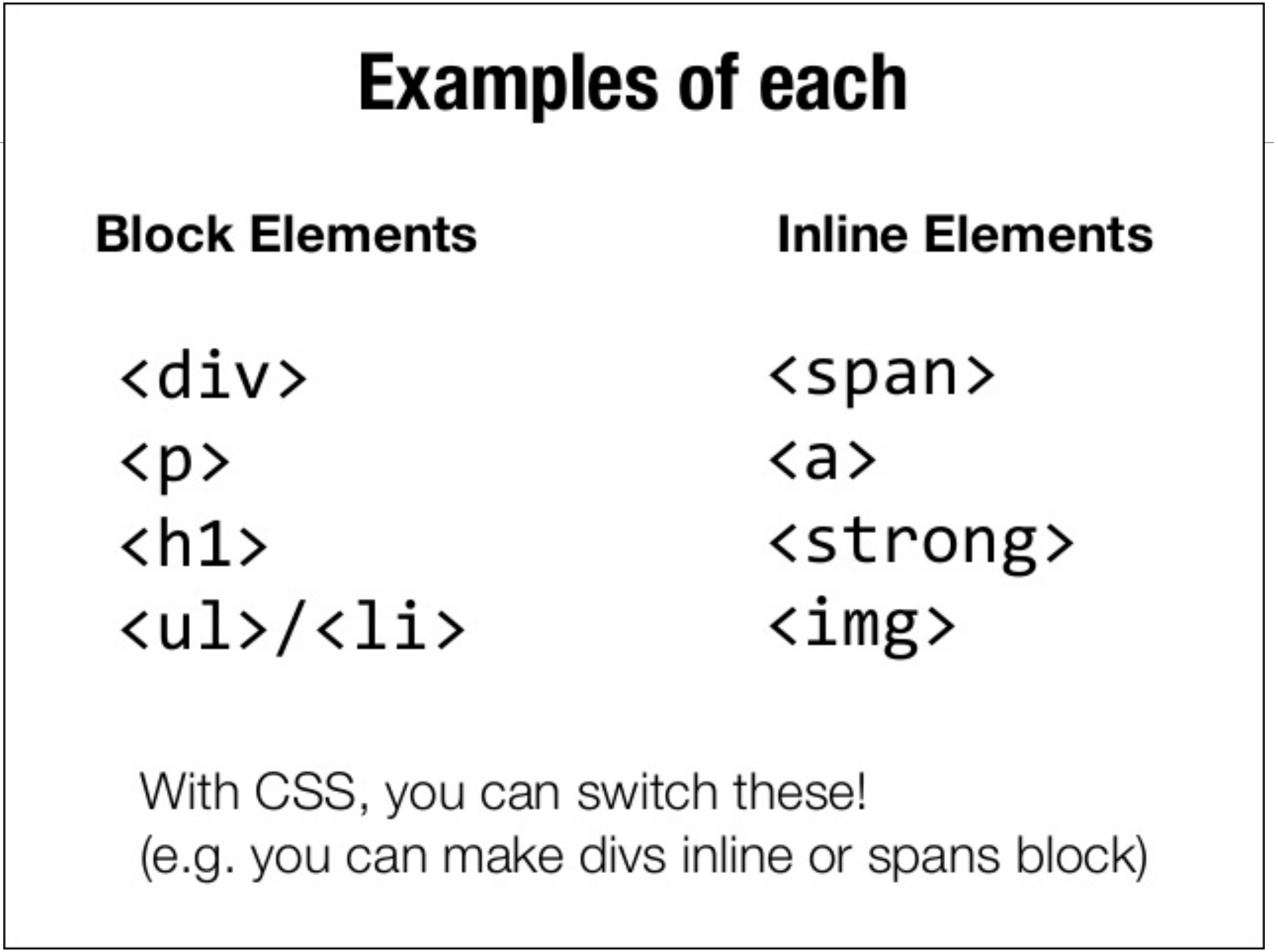


Block & Inline

- When the DOM tree has been constructed, and any CSS style sheets have been loaded and parsed, the browser starts the rendering phase.
- Each node in the DOM tree will be rendered as zero or more boxes.
- A box is always rectangular, it has four sides with a 90° angle between each side.
- HTML Elements are either block-level or inline-level.
  - block-level HTML elements generate block boxes - with line breaks between them
  - inline-level HTML elements generate inline boxes - without line breaks



```
<body>
    <h1> PuzzleWeb.ru Online Web Tutorial </h1>
    <p>
        ..<a href="https://puzzleweb.ru">HTML</a>..
    </p>
</body>
```



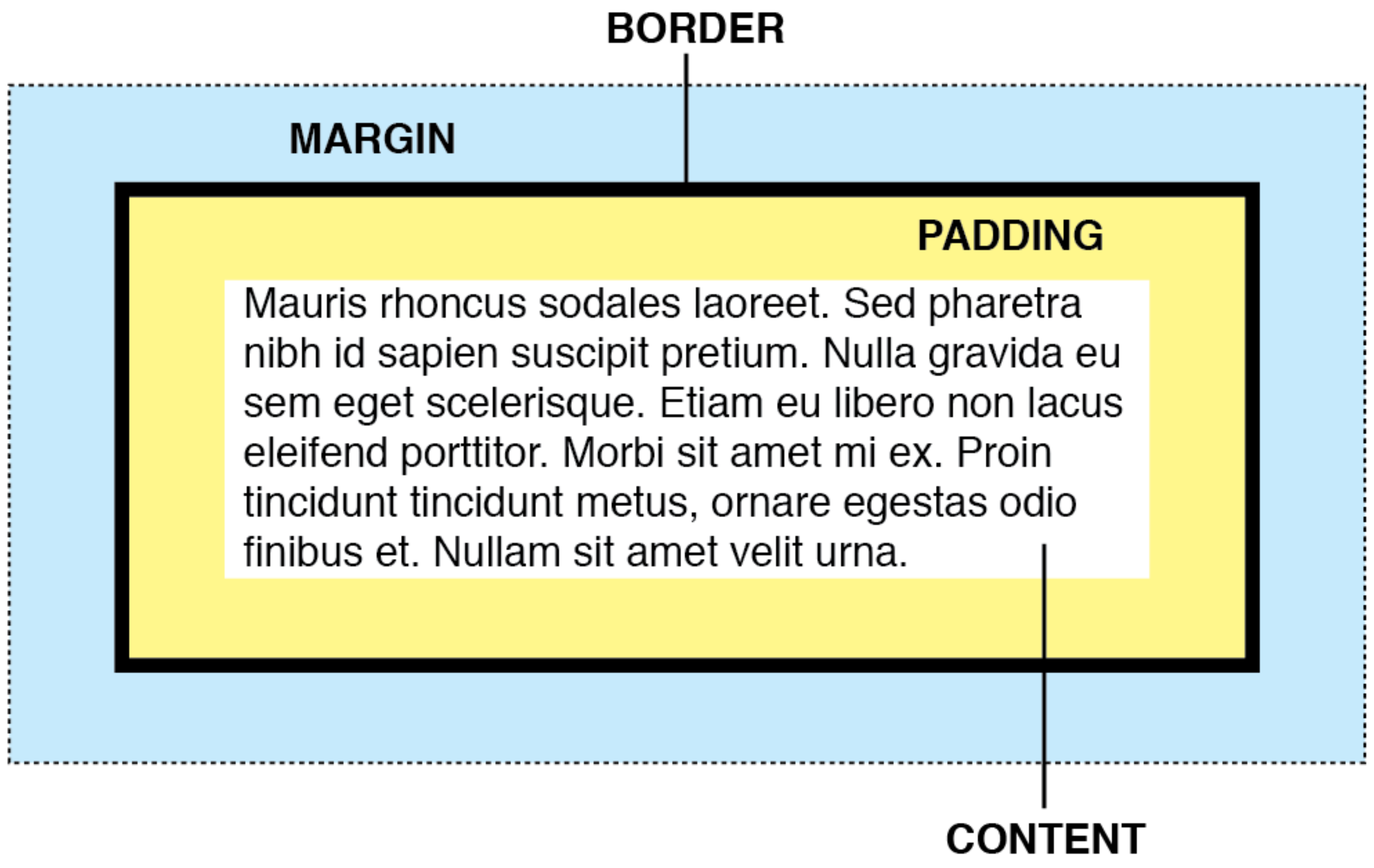
Examples of each

Block Elements	Inline Elements
`<div>`	`<span>`
`<p>`	`<a>`
`<h1>`	`<strong>`
`<ul>/<li>`	`<img>`

With CSS, you can switch these!
(e.g. you can make divs inline or spans block)

CSS Box Model

- Some key concepts:
  - Box Terminology
  - Calculating Box Dimensions
  - The Containing Block
  - Collapsing Margins


BORDER
MARGIN
PADDING
Mauris rhoncus sodales laoreet. Sed pharetra nibh id sapien suscipit pretium. Nulla gravida eu sem eget scelerisque. Etiam eu libero non lacus eleifend porttitor. Morbi sit amet mi ex. Proin tincidunt tincidunt metus, ornare egestas odio finibus et. Nullam sit amet velit urna.
CONTENT

Box Terminology

- The content area in the centre, the padding around the content area, the border area, and the margin area.

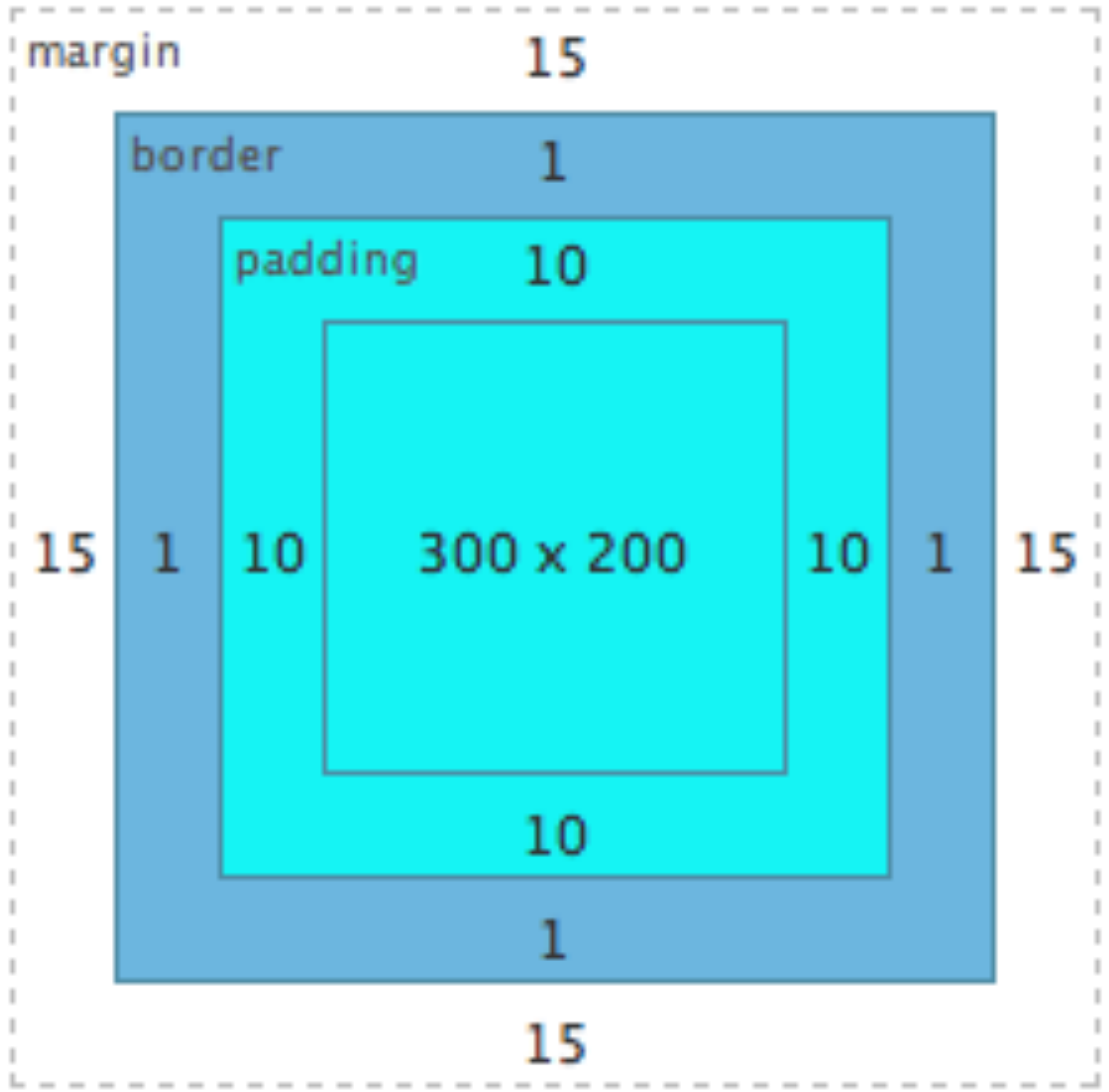
```
.box  {
  width: 300px;
  height: 200px;
  padding: 10px;
  border: 1px solid #000;
  margin: 15px;
}
```

The outer edge of the content area is called the content edge or inner edge;

the outer edge of the padding area is called the padding edge;

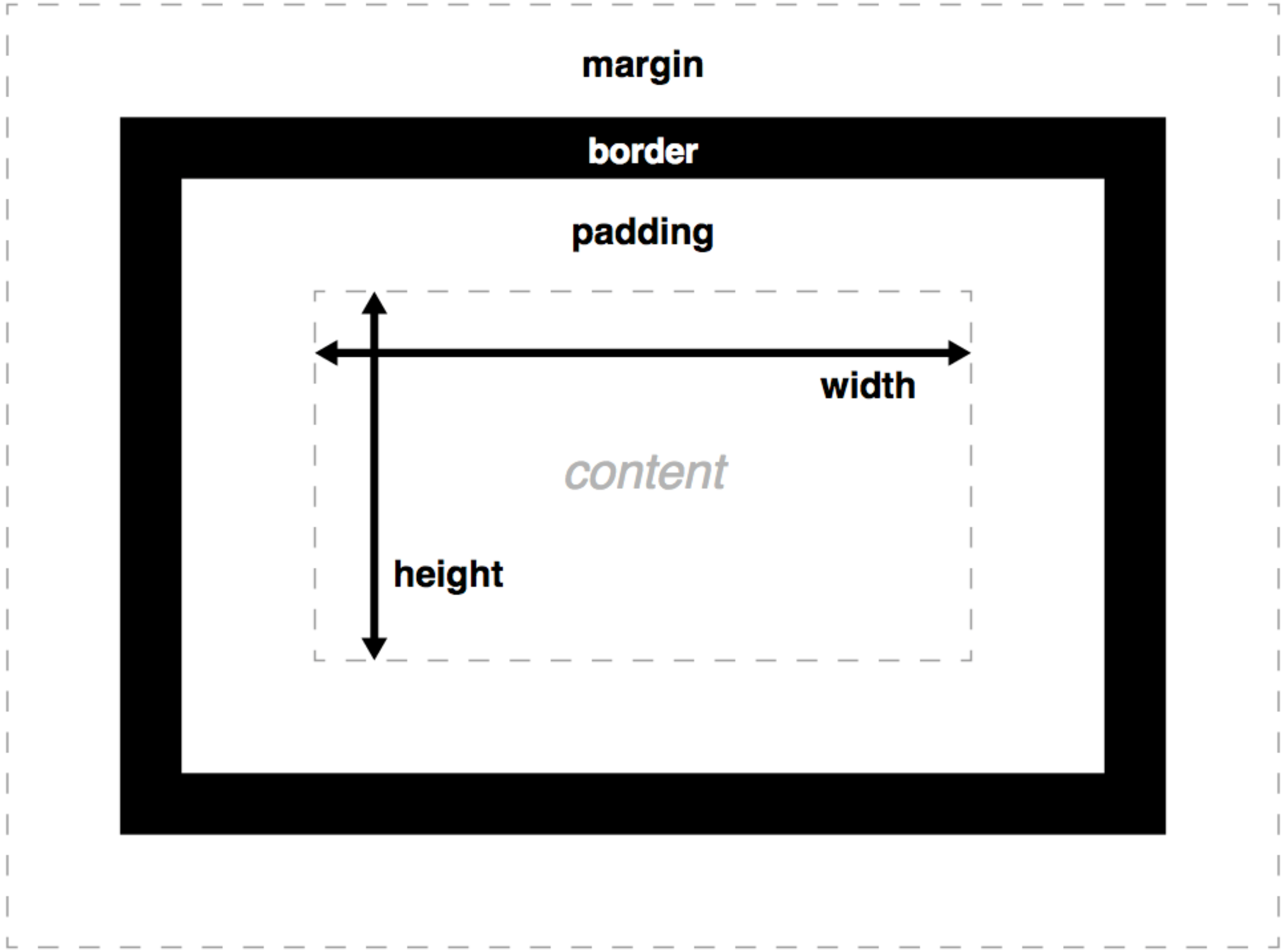
the outer edge of the border area is called the border edge;

and the outer edge of the margin area is called the margin edge or outer edge



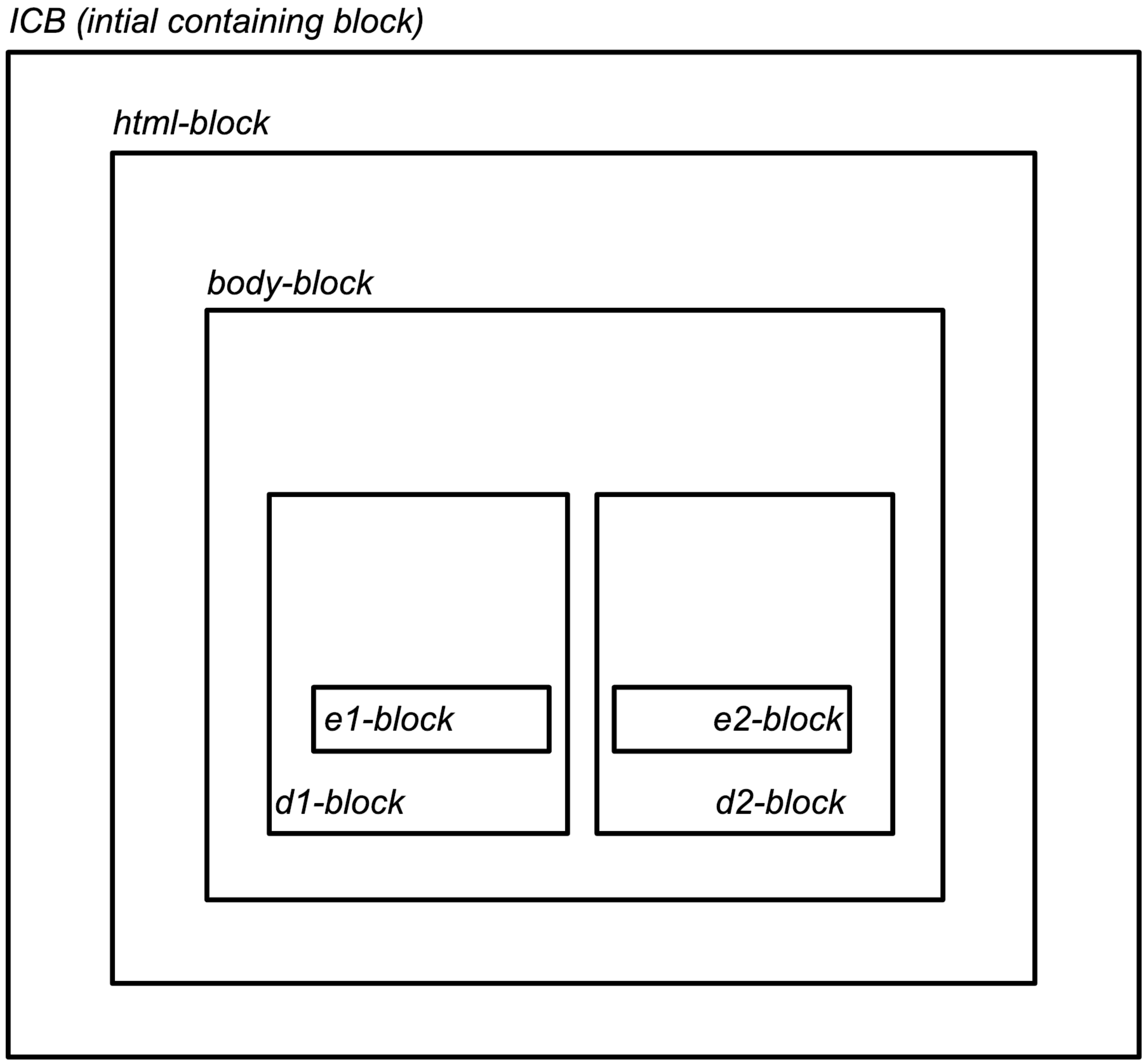
Calculating Box Dimensions

- Block-level elements can only be rectangular.
- Calculate the overall dimensions of a block-level element by taking into account the height and width of the content area, as well as any margins, padding, and borders that are applied to the element.
- There are some short circuits to this!



The Containing Block

- CSS rendering comprises the tasks of laying out and rendering numerous boxes. Element boxes are positioned within a formatting context, which, by default, is provided by the box generated by a parent element.
- When we specify the positions or dimensions of element boxes, we're doing so relative to what's known as the **containing block,** which is a very important concept in CSS layout.
- The containing block for the root element is called the **initial containing block**, and has the same dimensions as the **viewport** - generally the browser window
- The containing block for other elements is generally the immediate parent



Collapsing Margins

- When the vertical margins of two elements are touching, only the margin of the element with the largest margin value will be honoured, while the margin of the element with the smaller margin value will be collapsed to zero

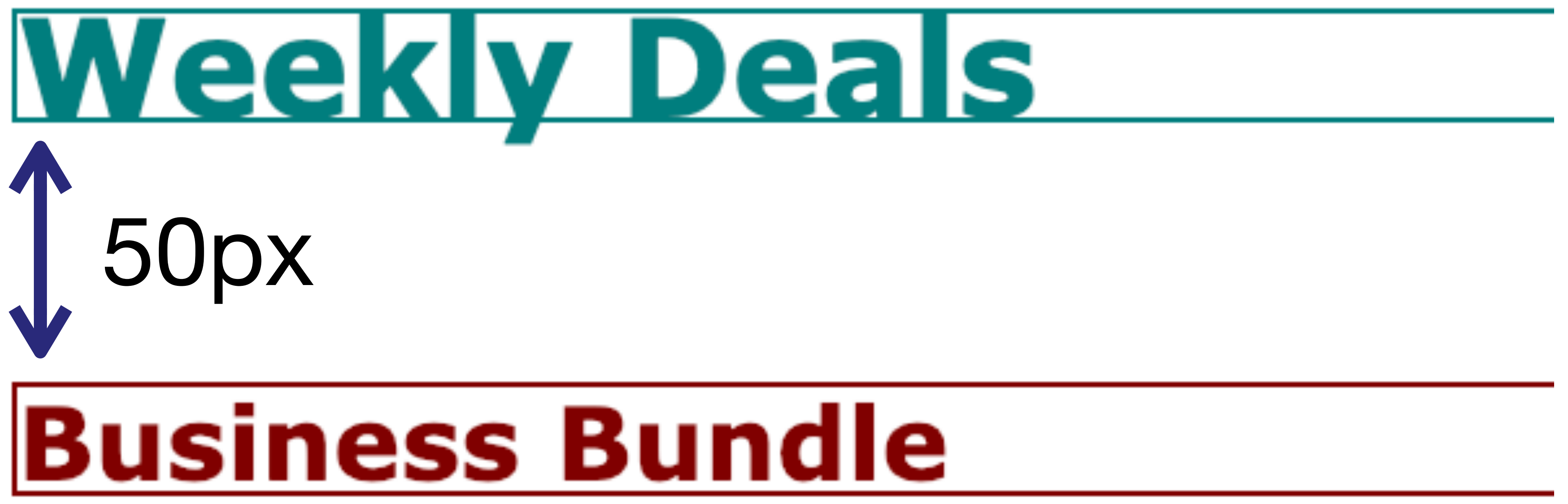
- In this example, the gap between the elements is only 50px, and the other (smaller if equal) margin has collapsed to zero

- There are exceptions to this rule…

```
<h1>Weekly Deals</h1>
<h2>Business Bundle</h2>
```

```
h1 {
  border: solid 1px;
  margin-bottom: 50px;
}

h2 {
  border: solid 1px;
  margin-top: 50px;
}
```


Weekly Deals
50px
Business Bundle