CSS LAYOUTS

Box Model

DOM Tree

Linking Files

CSS Properties

QUICK REVIEW

WHAT ARE THE TWO MAJOR TYPES OF HTML TAGS?

WHAT ARE THE THREE PARTS OF EVERY CSS DECLARATION?

WHAT ARE THE THREE BASIC CSS SELECTORS WE LEARNED?

SHOW OF HANDS

Who explored some CSS properties since last class?

IT'S COOL IF YOU DIDNT

I STILL LOVE YA

LETS EXPLORE MORE CSS TOGETHER

TAKE A LOOK AT

ASSIGNMENT 00

LINKING FILES

BASIC LINKAGES

For stylesheets (CSS):

```
<link rel="stylesheet" href="styles/main.css">
```

ABSOLUTE VS RELATIVE

Relative paths link to files on your server, so they are missing the http:// stuff:

```
<link rel="stylesheet" href="styles/main.css">
```

Absolute paths have the http:// stuff. You'll generally see them with anchor tags that link off your server:

```
<a href="https://google.com">Google</a>
```

ABSOLUTE VS RELATIVE

Relative paths can use dot notation to reference folders above them:

```
<link rel="stylesheet" href="../styles/main.css">
```

Relative paths with a slash at the front indicate that the path starts at the root folder:

```
<link rel="stylesheet" href="/styles/main.css">
```

WHEAT DOES EACH OF THESE PATHS MEAN?

```
<link rel="stylesheet" href="styles/main.css">
<link rel="stylesheet" href="../styles/main.css">
<link rel="stylesheet" href="/styles/main.css">
```

ASSIGNMENT 01

1) Your first task is to use the font, background and text classes to improve the look of this page.

2) Use relative links to connect the subfolder to the main page (and vice versa) - along with linking the images.

3) Try to get an image from the main page to show up on the subfolder page.

THE DOM TREE

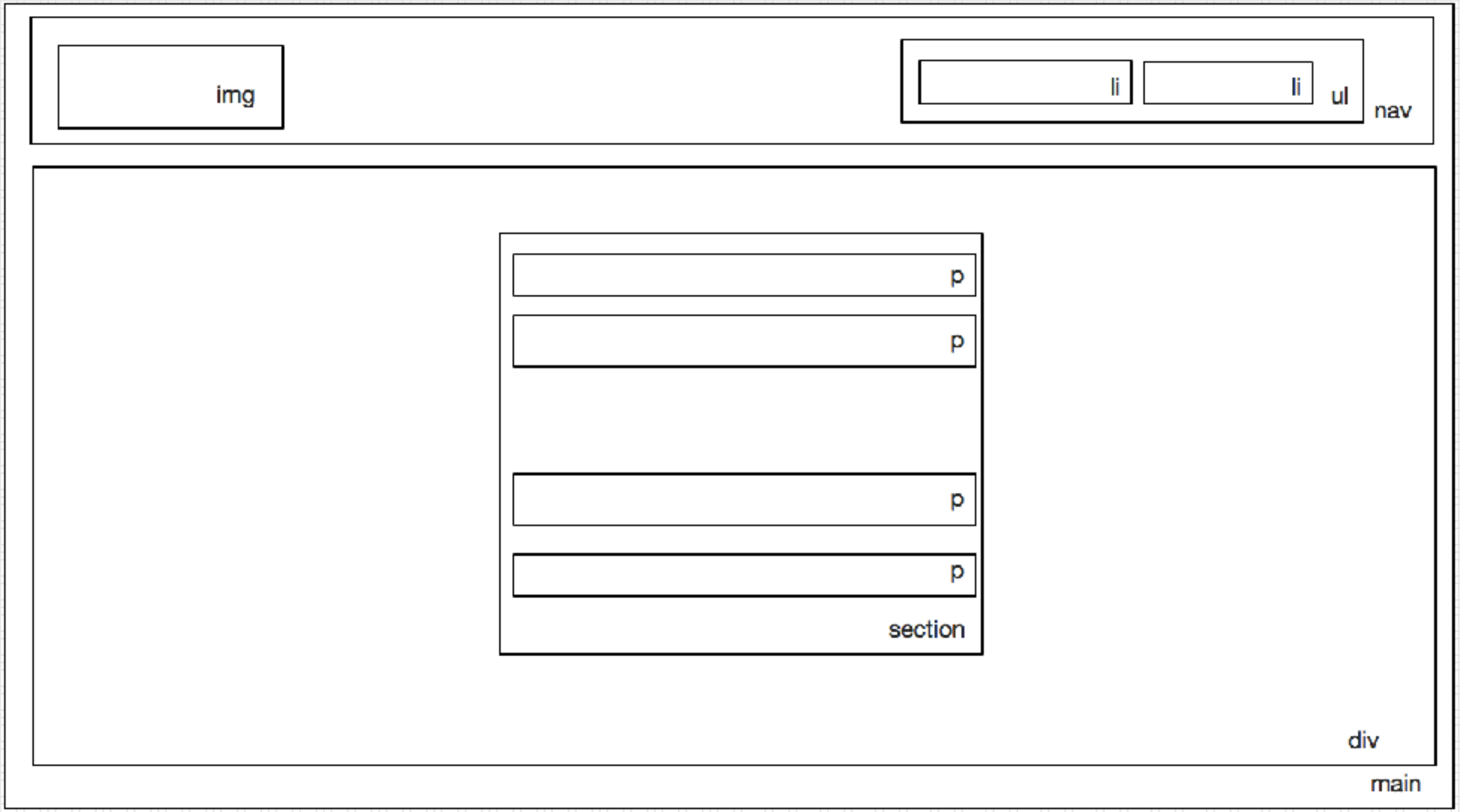
It's not a tree but rather a visual diagram of a webpage's HTML structure

daocloud

FOR PROFESSIONALS

SIGN IN

How is this page structured?



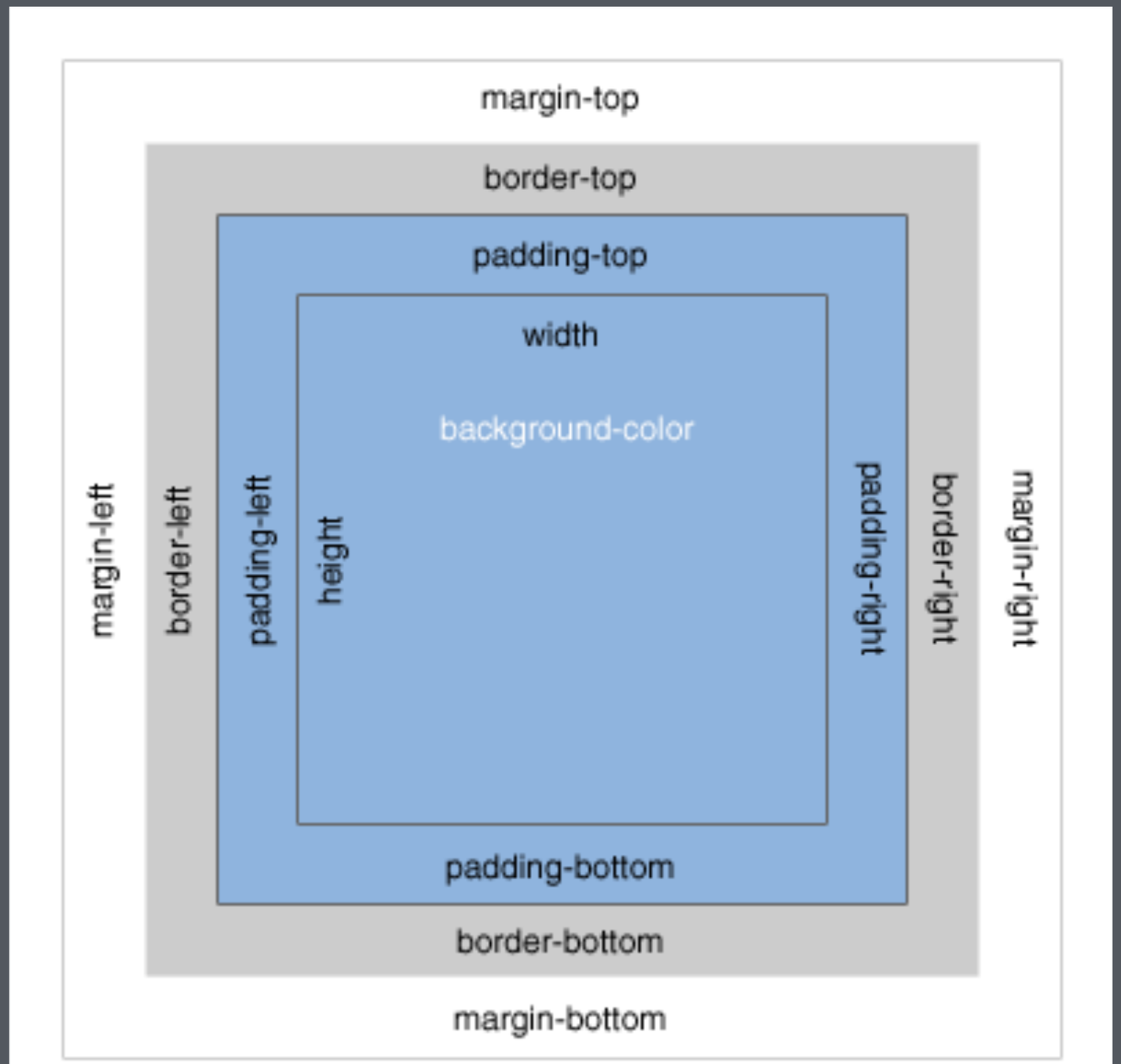
Remember: HTML is subjective (to some degree)

TRY OUT

ASSIGNMENT 02

1) Work in groups of 2, take 20 minutes and draw a DOM tree for the sample included in this assignment

THE BOX MODEL



BOX MODEL BASICS

-Every block level element on a webpage is a box (`div, section, ul, nav, header, footer, etc`).

-All block elements have `padding, border, margin`

-`padding + border` are inside the box (count towards `height + width`)

-`margin` is outside the box (does NOT count towards `height + width`)

HTML BLOCK VS INLINE

-Block level elements (`div, section, ul, nav, header, footer, etc`) inherit the box model.

-Inline level elements do not inherit the box model (`span, img, sub, sup, textblock`).

-Very hard to know which are which when you're starting out. You can apply (**`display: block;`**) to an inline element and it will become a block level element.

-Refer here for list of inline elements: https://developer.mozilla.org/en-US/docs/Web/HTML/Inline_elements

NOW BACK TO ASSIGNMENT 01

1) Apply the box model CSS elements to the HTML in the sample. Adjust the box model in unique ways to test what happens.

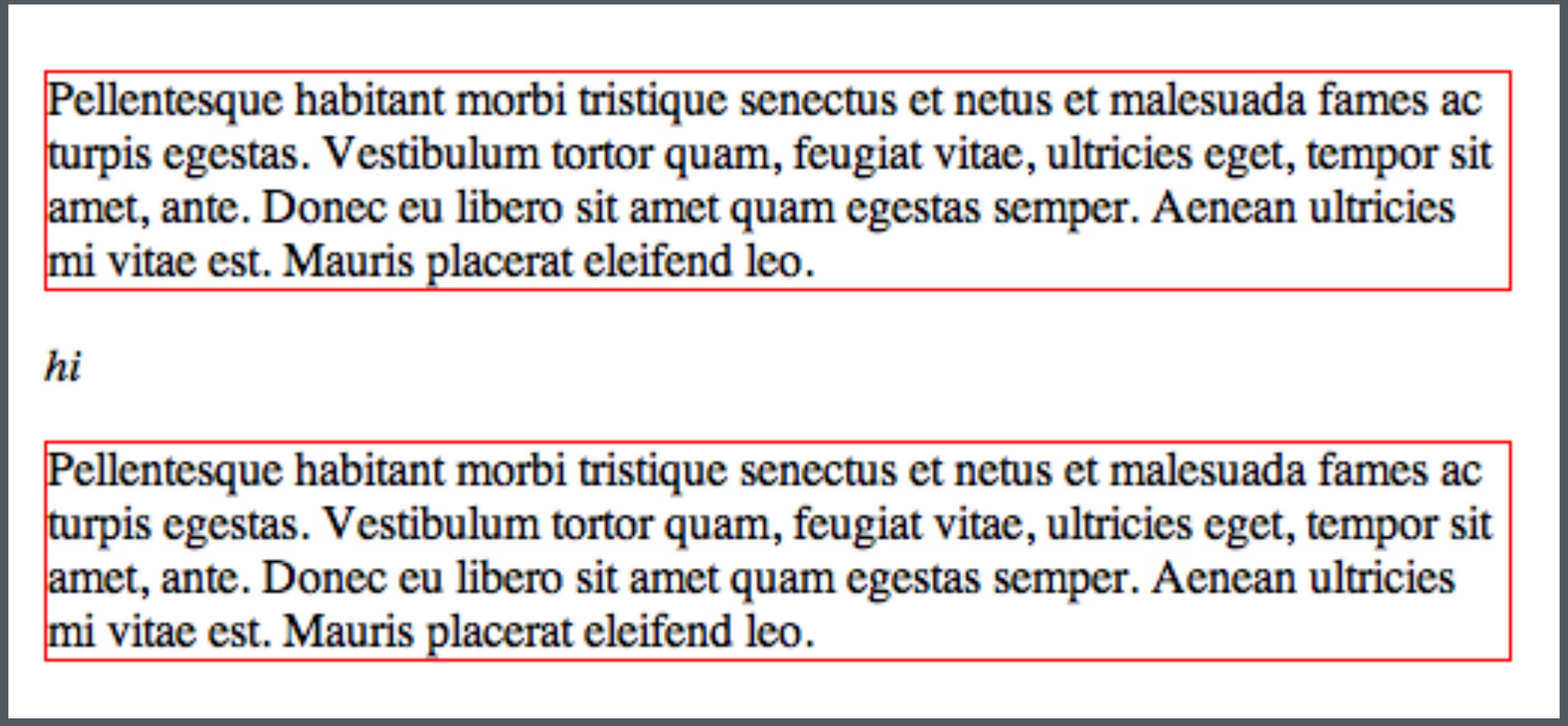
2) If you're feeling really ambitious, try to align the three elements horizontally.

CSS POSITION

POSITIONING ELEMENTS

`-display` controls behavior of the box content sits in:

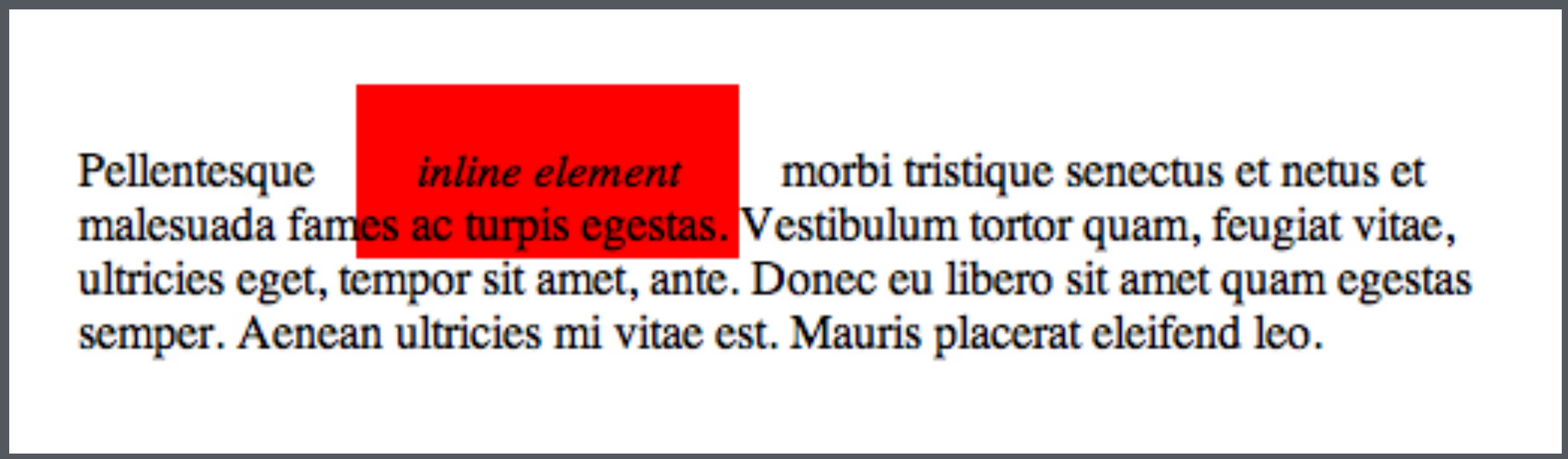
`display: block;` - element takes up as much width as possible, following element drops to a new line



images courtesy of css-tricks.com

POSITIONING ELEMENTS

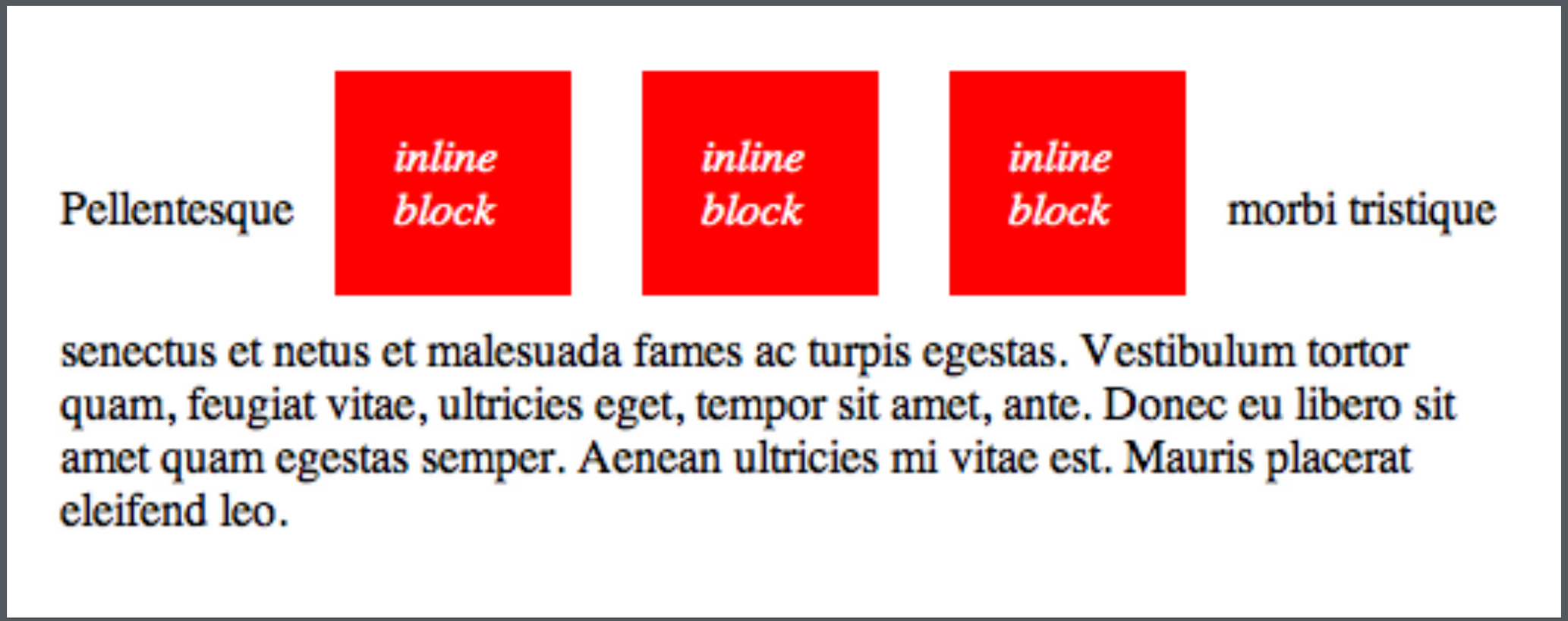
`display: inline;` - element takes up only as much width as it needs, `padding / margins` only work left + right, not top and bottom. Top and bottom spacing is controlled by `line-height` property because the content is in line.



images courtesy of css-tricks.com

POSITIONING ELEMENTS

`display: inline-block;` - combination of the two concepts. The item(s) will display inline with other items but you can use all margin, padding, height and width properties on an inline-block.



images courtesy of css-tricks.com

POSITIONING ELEMENTS

`display: none;` - nothing shows up at all. It's in the DOM the browser sees but the user won't see it. Seems useless now but just wait till we hit Javascript. You're going to love display none.

FOR NEXT TIME

Layout with CSS Flexbox
Come to Office Hours!