HTML & CSS: LEVEL 1

Instructor: Beck Johnson

Week 4

SESSION OVERVIEW

- Review box model of CSS, classes and ids
- Floats and positioning
- Semantic elements and why they matter
- Using web fonts

REVIEW!

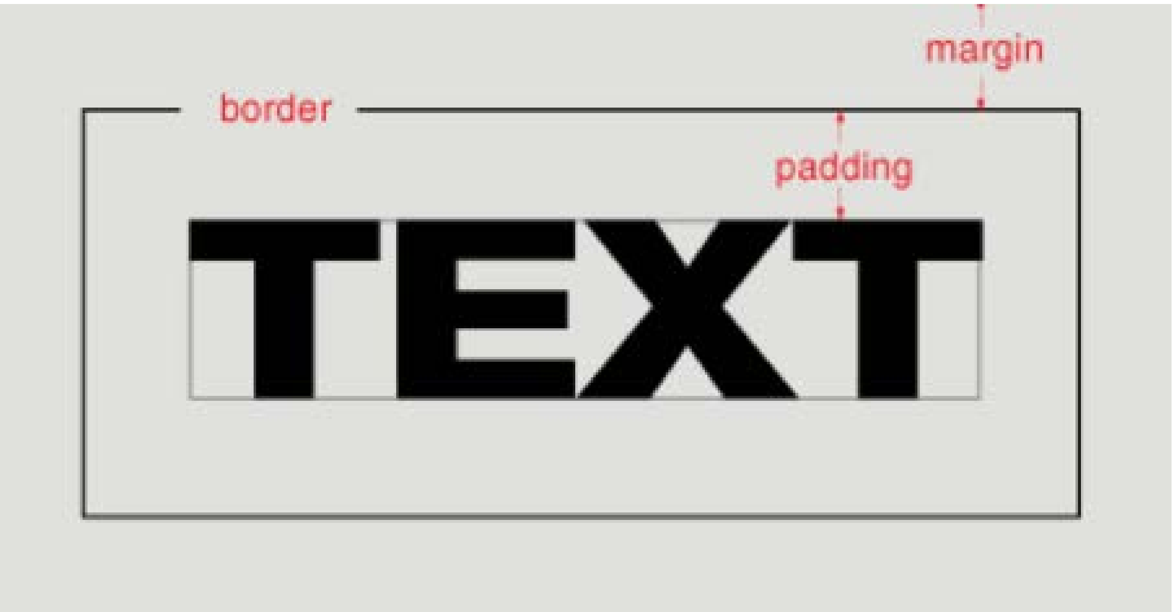
{} CSS BOX MODEL

CONTENT: stuff in the box

PADDING: bubble wrap and packing peanuts

BORDER: sides of the box

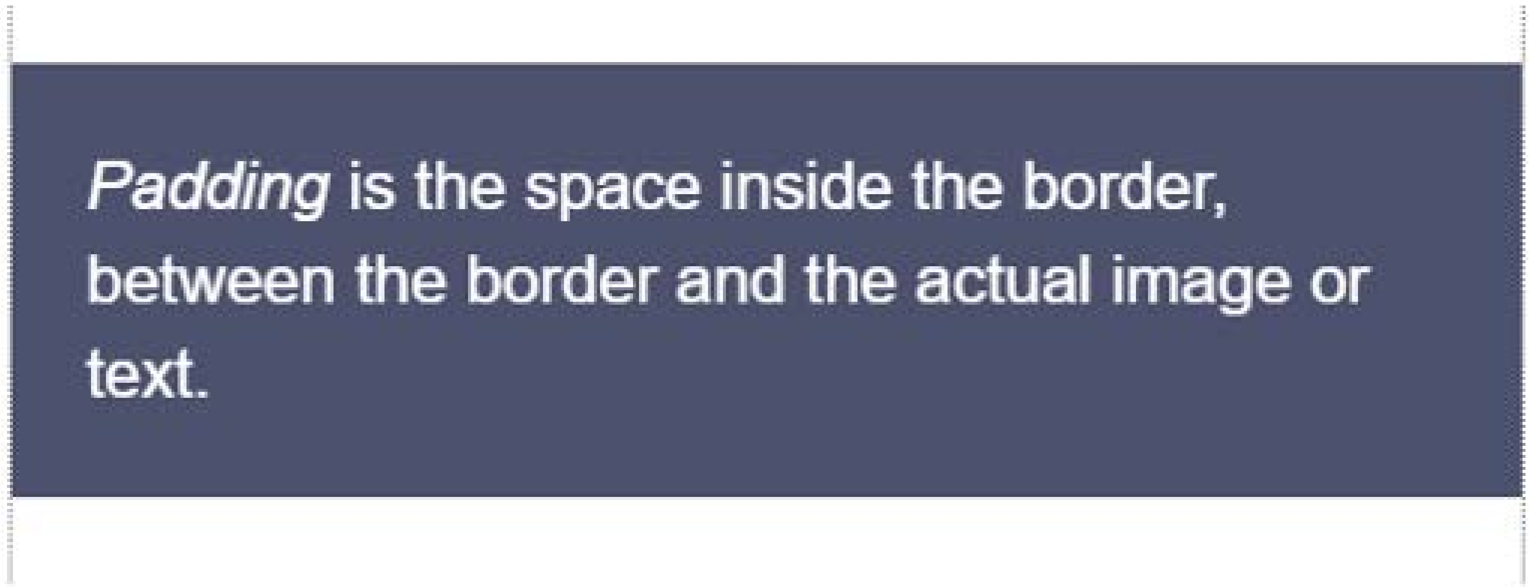
MARGIN: space between multiple boxes



{} PADDING

- Padding creates space **inside** an element

- Padding affects how far content is from the border



Padding is the space inside the border, between the border and the actual image or text.

{} MARGIN

- Margin creates space **outside** an element.
- Margin affects how far elements are from each other

Margin is the space between one object and its surrounding elements.

{} BORDER

Between margin and padding, you can set a `border`

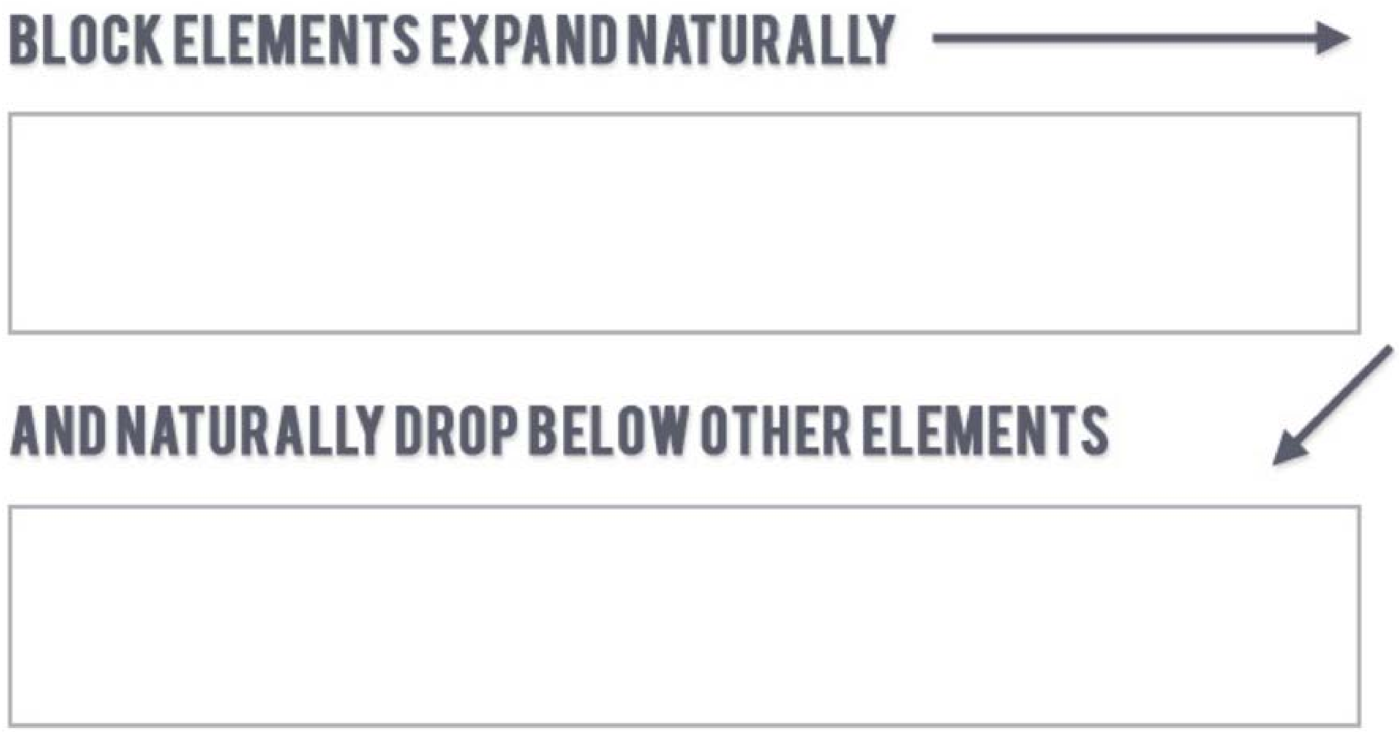
- Width (usually in pixels)
- Border style (solid, dotted, dashed, etc)
- Color

```
p
    border: 2px dotted #ff0000;
}
```

{} BLOCK ELEMENTS

BLOCK ELEMENTS

- Expand naturally to fill their parent container
- Can have margin and/or padding



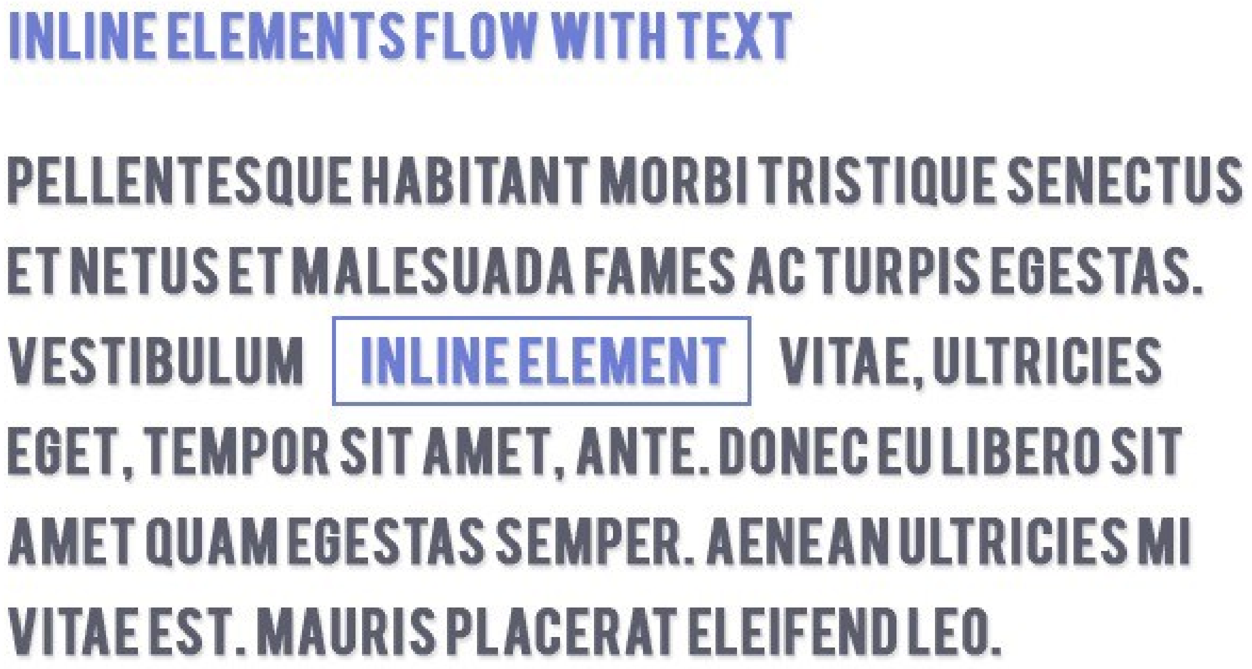
BLOCK ELEMENTS EXPAND NATURALLY

AND NATURALLY DROP BELOW OTHER ELEMENTS

{} INLINE ELEMENTS

INLINE ELEMENTS

- Flow along with text content
- Ignores height, width, top margin, and bottom margin
- Honors left and right margins (and any padding)



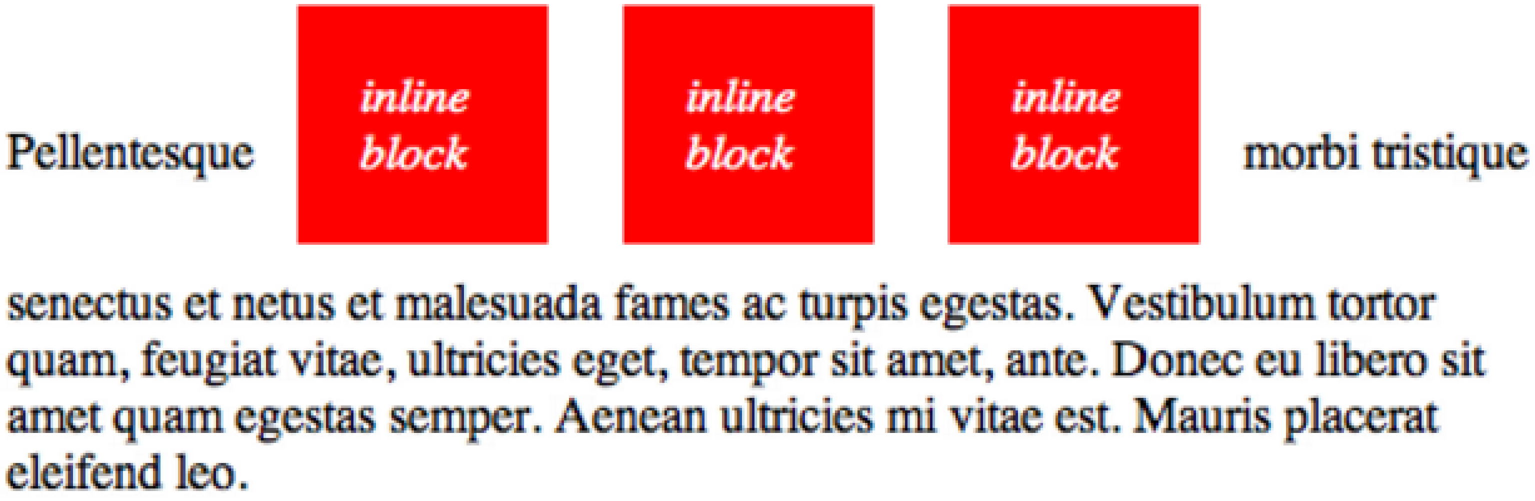
INLINE ELEMENTS FLOW WITH TEXT

PELLENTESQUE HABITANT MORBI TRISTIQUE SENECTUS ET NETUS ET MALESUADA FAMES AC TURPIS EGESTAS. VESTIBULUM INLINE ELEMENT VITAE, ULTRICIES EGET, TEMPOR SIT AMET, ANTE. DONEC EU LIBERO SIT AMET QUAM EGESTAS SEMPER. AENEAN ULTRICIES MI VITAE EST. MAURIS PLACERAT ELEIFEND LEO.

{} INLINE-BLOCK ELEMENTS

INLINE-BLOCK ELEMENTS

- A hybrid of block and inline
- Flow along with text content
- Has height, width, margin, and padding



Pellentesque *inline block* *inline block* *inline block* morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum tortor quam, feugiat vitae, ultricies eget, tempor sit amet, ante. Donec eu libero sit amet quam egestas semper. Aenean ultricies mi vitae est. Mauris placerat eleifend leo.

{} CLASSES AND IDS

You can add `class` and `id` to any HTML element to identify it for styling.

- You decide the `class` and `id` values – be descriptive!

```
<p class="important">Big text</p>

<p class="anyLettersOrNumbersOr_Or-">Still
totally valid</p>
```

{} CLASSES

Multiple elements can have the same `class`

In CSS, target a class with a **period**

```
.kittens { color: gray; }
```

```
<p class="kittens">This will be gray.</p>
```

{} IDS

Only **one** element per page can use the same id

In CSS, target an id with a **hash**:

```
#kittenContainer { color: gray; }
```

```
<div id="kittenContainer">This will be
gray.</div>
```

{} HOW TO CHOOSE BETWEEN THEM

If you think it’s likely or possible that you’ll want to apply the same style to multiple things, definitely use `class`

If your element is guaranteed to be the only one on the page, you can use `id` – or you can still use `class`

If your element needs to be linked to directly, use `id`

QUESTIONS?

WEB LAYOUTS

WEB LAYOUTS

With CSS, we can use a variety of properties to arrange elements on the screen by adjusting the flow of the page.

Basically, you can put elements anywhere...which can be both a good and a bad thing!

3 WEB LAYOUT PROPERTIES

- `display:` dictates how elements behave within the box model
- `float:` moves elements around within the page flow
- `position:` takes elements entirely out of the page flow

DISPLAY PROPERTY

The `display` property tells the browser what type of box model to use:

- `inline`
- `inline-block`
- `block`

This changes how padding, margin, height and width affect an element.

You also can set `display: none` to hide an element entirely.

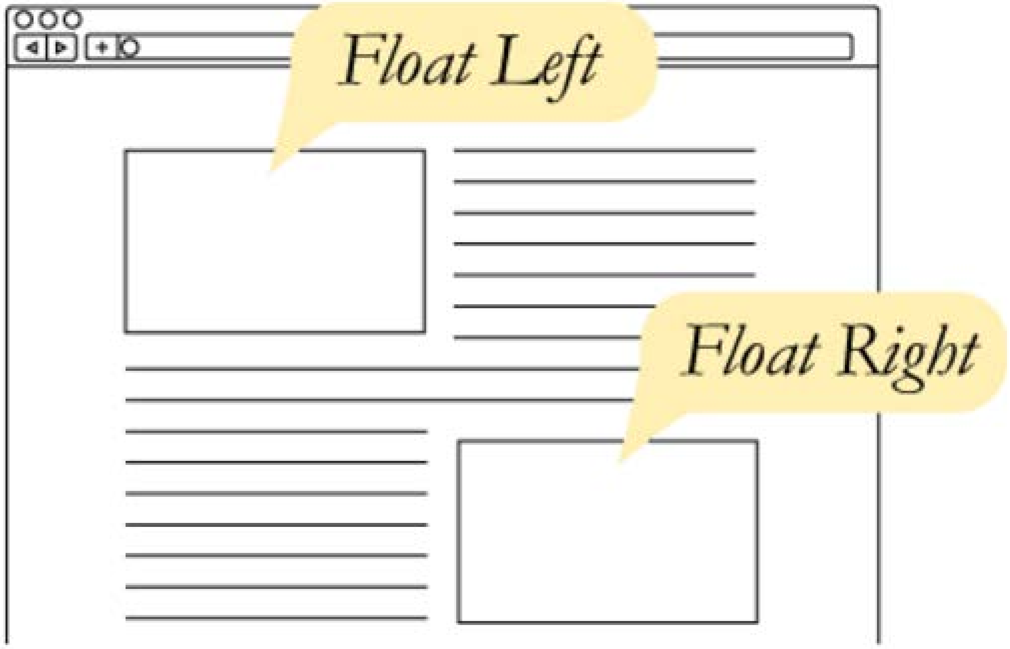
CSS FLOATS

Up to now, elements have displayed sequentially, in the order that you placed them in your HTML.

The `float` property takes an element out of the normal flow and "floats" it to the left or right side of its container.

- This allows other content to flow around it

```
img { float: left; }
```

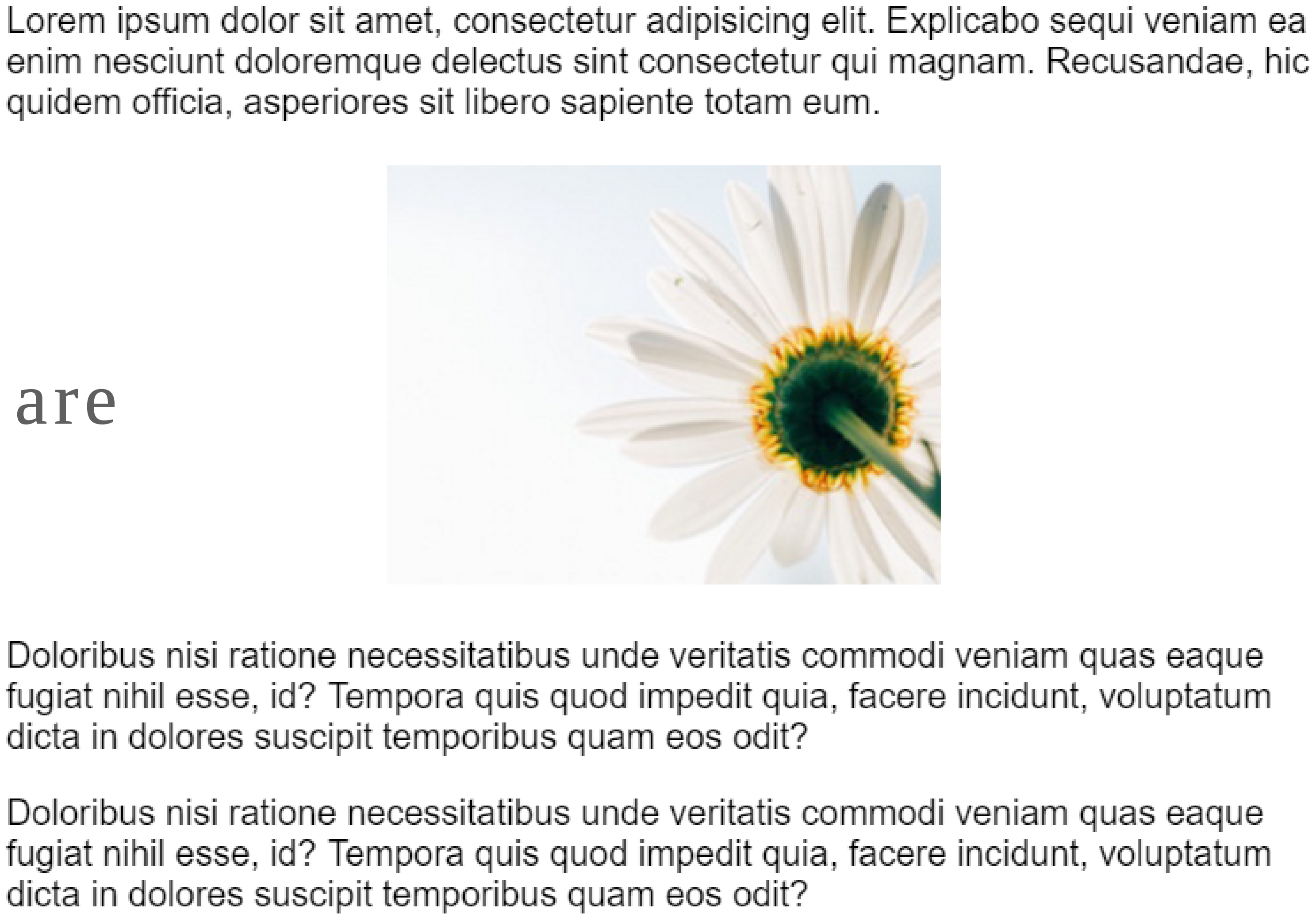


CSS FLOATS

The three values for `float` are:

- `left`
- `right`
- `none`

By default, elements are `float: none`



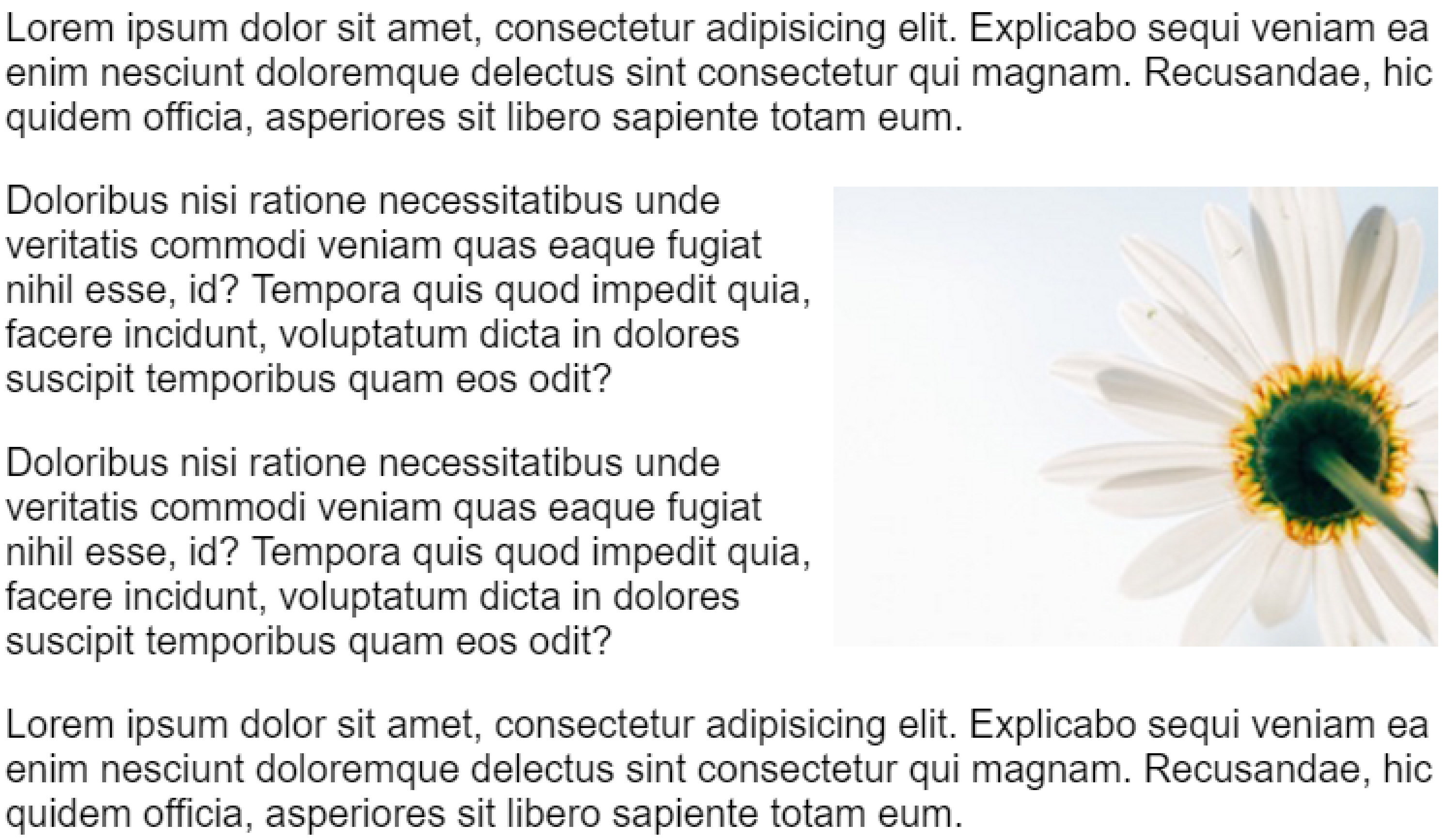
Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo sequi veniam ea enim nesciunt doloremque delectus sint consectetur qui magnam. Recusandae, hic quidem officia, asperiores sit libero sapiente totam eum.

Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

CSS FLOATS

```
img { float: right; }
```



Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo sequi veniam ea enim nesciunt doloremque delectus sint consectetur qui magnam. Recusandae, hic quidem officia, asperiores sit libero sapiente totam eum.

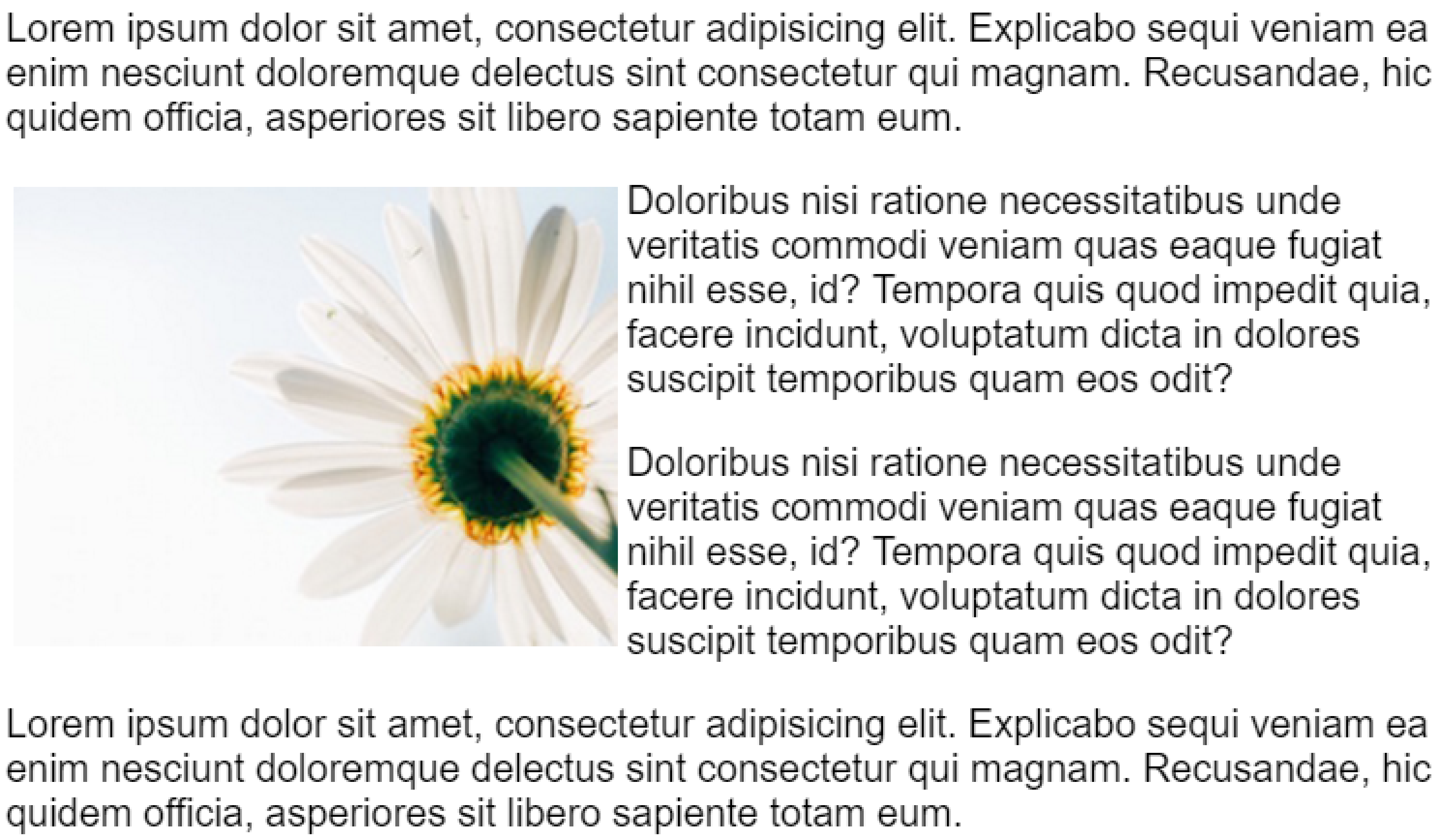
Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo sequi veniam ea enim nesciunt doloremque delectus sint consectetur qui magnam. Recusandae, hic quidem officia, asperiores sit libero sapiente totam eum.

CSS FLOATS

```
img { float: left; }
```



Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo sequi veniam ea enim nesciunt doloremque delectus sint consectetur qui magnam. Recusandae, hic quidem officia, asperiores sit libero sapiente totam eum.

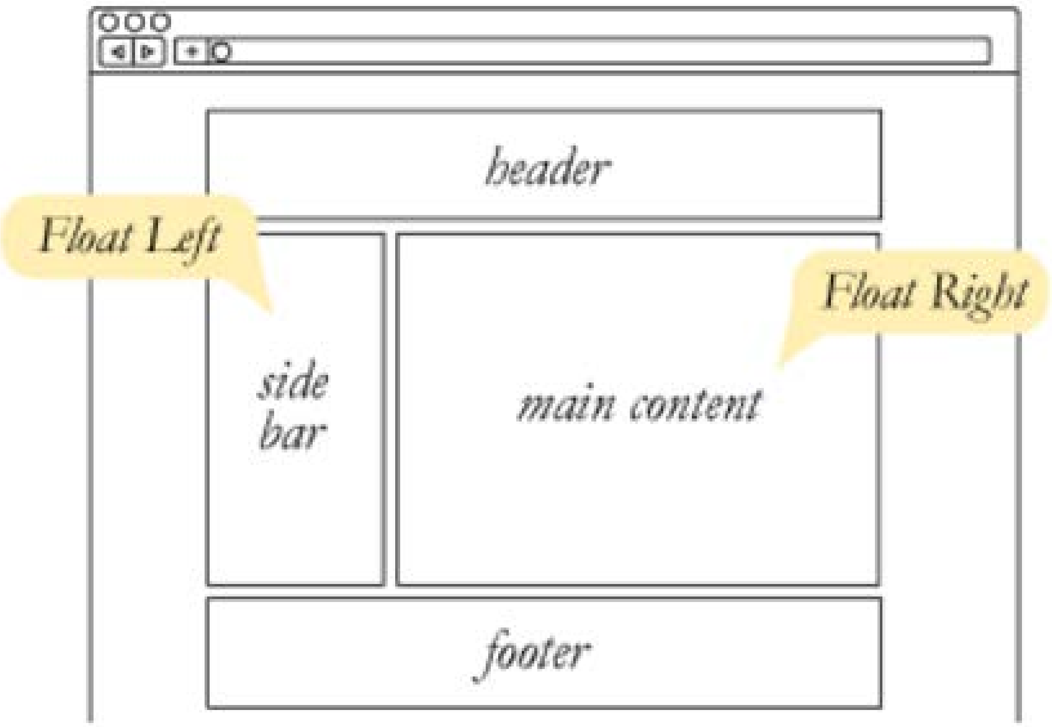
Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

Doloribus nisi ratione necessitatibus unde veritatis commodi veniam quas eaque fugiat nihil esse, id? Tempora quis quod impedit quia, facere incidunt, voluptatum dicta in dolores suscipit temporibus quam eos odit?

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Explicabo sequi veniam ea enim nesciunt doloremque delectus sint consectetur qui magnam. Recusandae, hic quidem officia, asperiores sit libero sapiente totam eum.

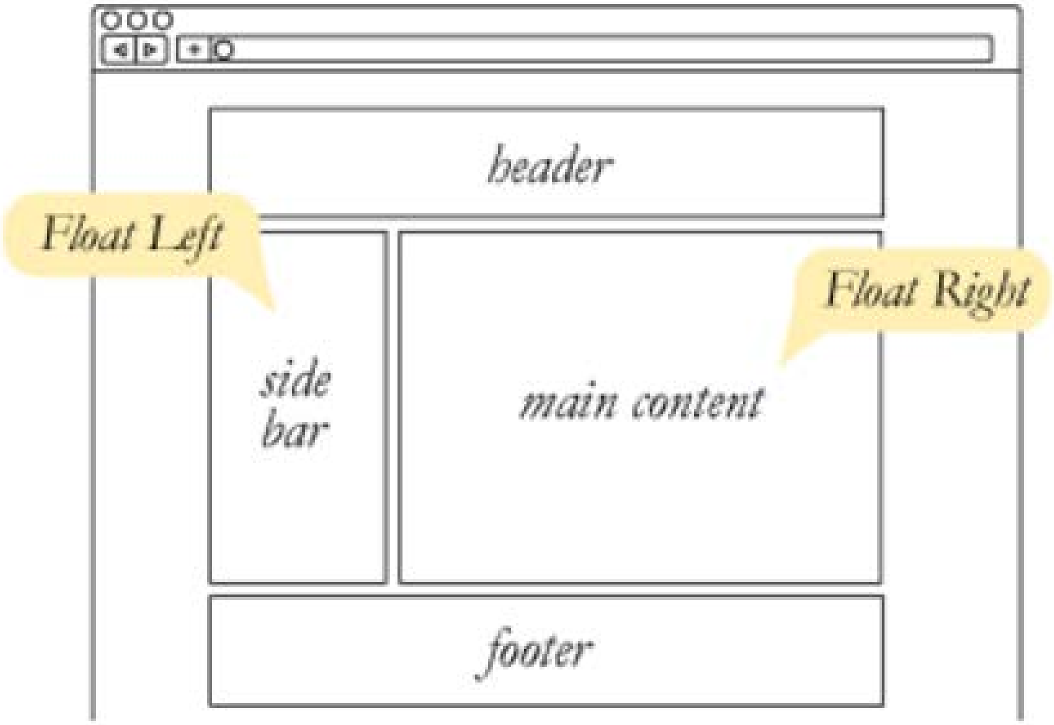
CSS FLOATS

`float` can be used to float text around images, but it also can be used to create entire page layouts.



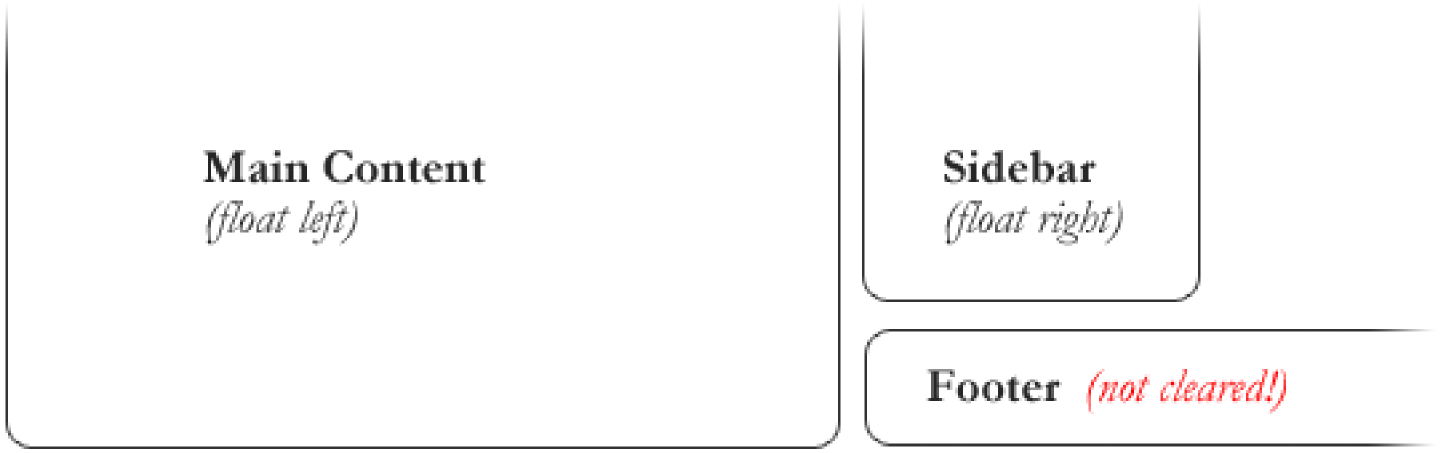
CSS FLOATS

`float` can be used to float text around images, but it also can be used to create page layouts.



THE CLEAR PROPERTY

- CSS float’s sister is the **clear** property.
- An element that has the **clear** property set on it will not move up adjacent to the float like the float desires, but **will move itself down past the float**.



(uncleared float)

THE CLEAR PROPERTY

- One of the trickiest things about floats is when to stop “floating”

- You can give any element the **clear:both;** style to prevent it from floating.

```
#sidebar {
   clear: both;
}
```

THE MAGIC FLOAT FIX

The most common fix today though is the self-clearing float.

You can use a **pseudo-element** on the parent of the floated elements to create a **“self-clearing”** float.

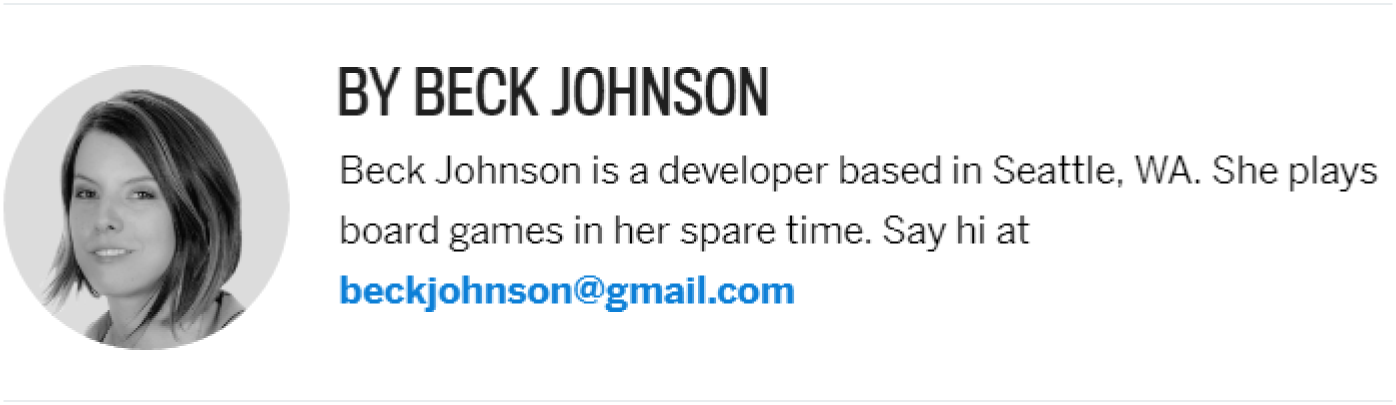
```
.clearfix:after {
   content: “”;
   display: block;
   height: 0;
   clear: both;
}
```

PRACTICE TIME!

ASSIGNMENT

Create a sidebar on your page that is either floated left or right

Make an "about me" section at the bottom of the page



BY BECK JOHNSON

Beck Johnson is a developer based in Seattle, WA. She plays board games in her spare time. Say hi at beckjohnson@gmail.com

CSS POSITIONING

CSS POSITIONING

The `position` property specifies how an element is positioned on the page. Possible values are:

- `relative`
- `absolute`
- `fixed`
- `static`

The default `position` is `static`, which just means that the element obeys whatever its box model rules tell it to do.

CSS POSITIONING

`position: absolute` is a powerful tool that allows you to place any page element exactly where you want it, down to the pixel.

When an element has `position: absolute`, it is entirely removed from the normal flow of the page

This is a very powerful type of positioning that allows you to literally place any page element exactly where you want it. You use the positioning attributes top, left bottom and right to set the location. Remember that these values will be relative to the next parent element with relative (or absolute) positioning. If there is no such parent, it will default all the way back up to the <html> element itself meaning it will be placed relatively to the page itself.The trade-off, and most important thing to remember, about absolute positioning is that these elements are removed from the flow of elements on the page. An element with this type of positioning is not affected by other elements and it doesn't affect other elements. This is a serious thing to consider every time you use absolute positioning. It's overuse or improper use can limit the flexibility of your site.

CSS POSITIONING

We can dictate where elements go on the page down to the pixel!

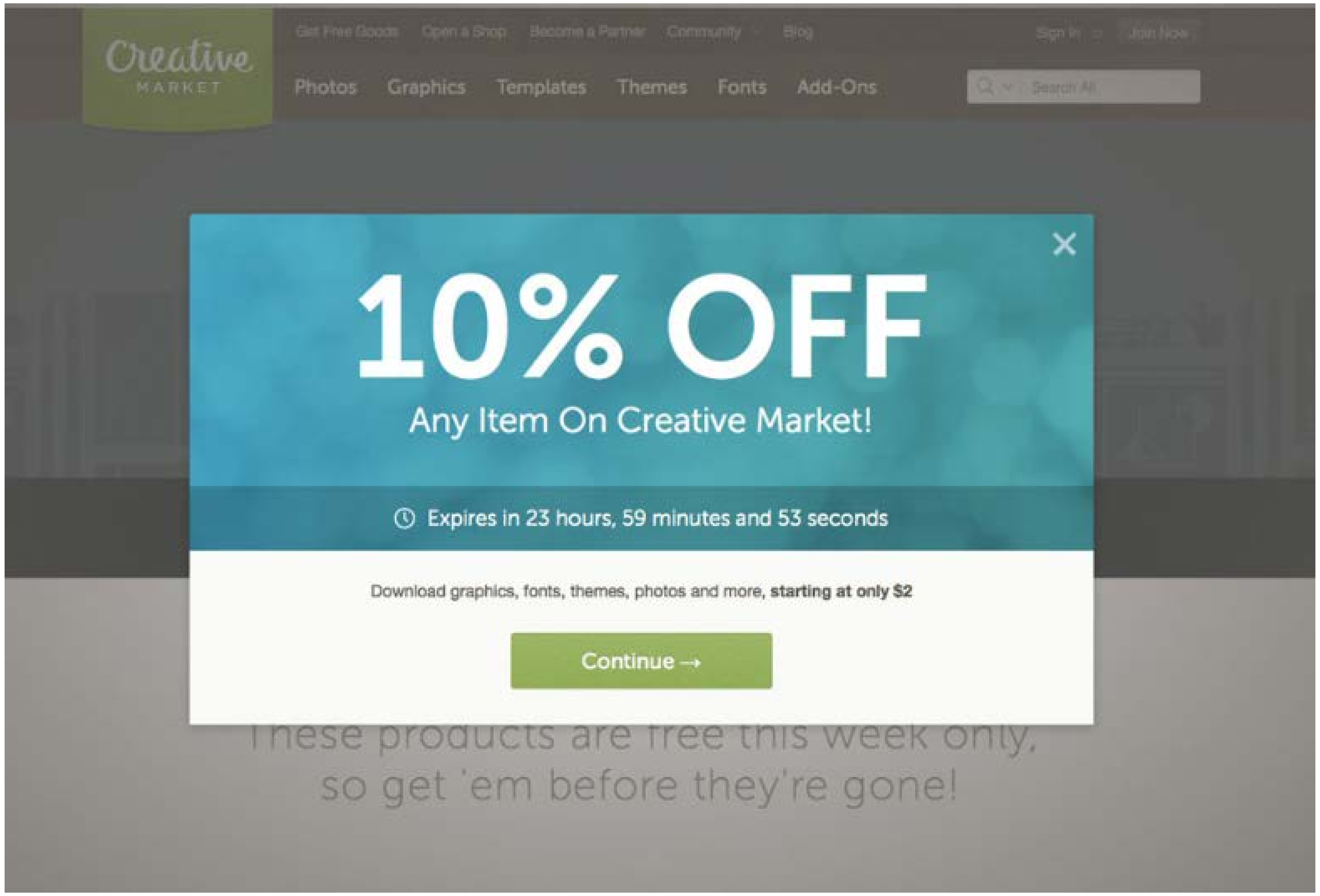
left, right, top, bottom

Can tweak positively or negatively

```
nav {
   position: absolute;
   right: -10px;
   top: 30px;
}
```

ABSOLUTE POWER

`position: absolute` is commonly used when creating page modals that pop up over other content



10% OFF

Any Item On Creative Market!

Expires in 23 hours, 59 minutes and 53 seconds

Download graphics, fonts, themes, photos and more, **starting at only $2**

These products are free this week only, so get 'em before they're gone!

POSITION: FIXED

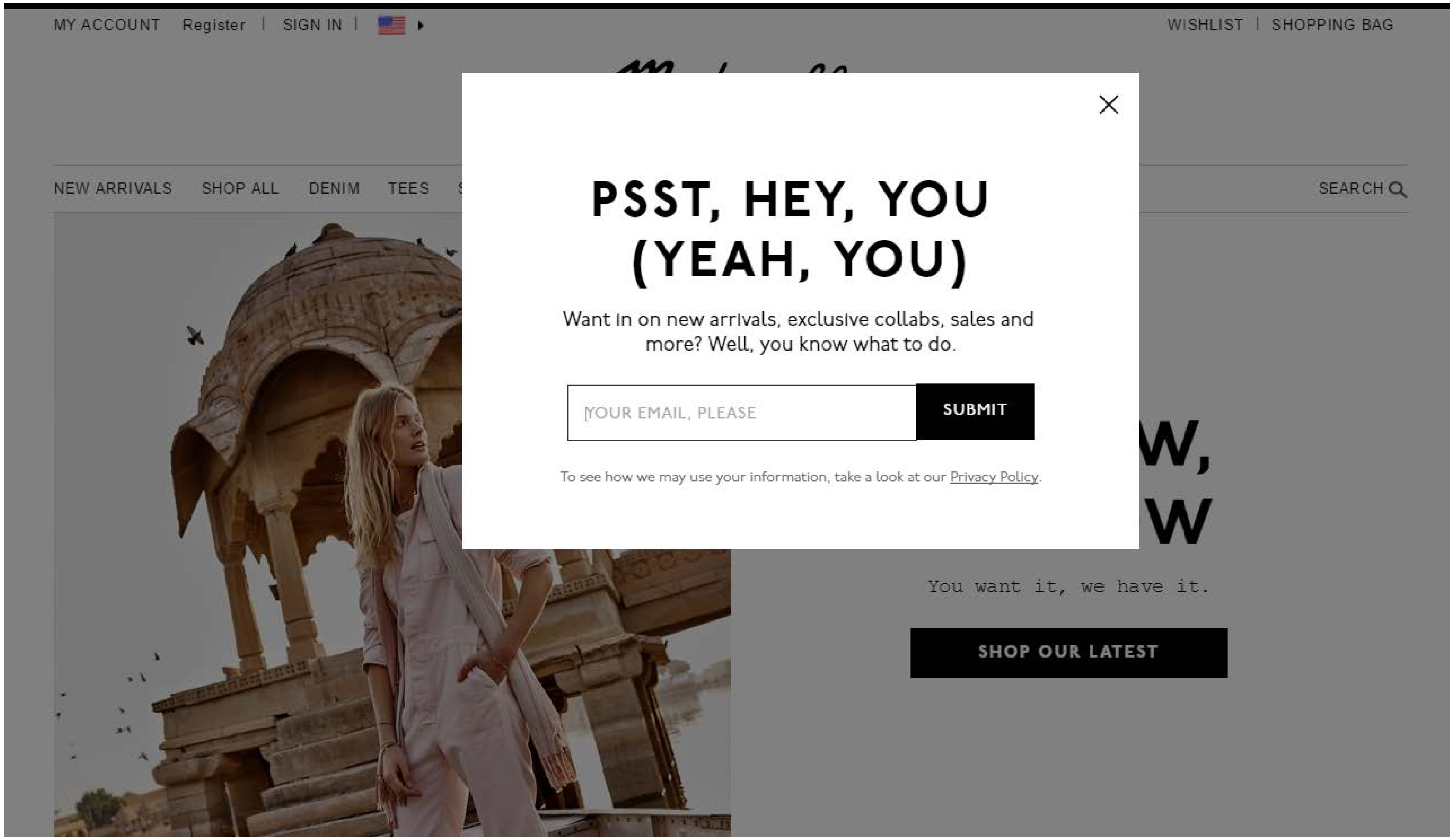
`position: fixed` makes content “stick” to the browser window, regardless of where the user scrolls.

Commonly used to make headers, nav, or footers that follow the page as it scrolls.

```
nav {
    position: fixed;
    left: 0;
    top: 0;
}
```

ABSOLUTE POWER

The greyed-out background uses `position: fixed` to cover the entire page



PSST, HEY, YOU (YEAH, YOU)

Want in on new arrivals, exclusive collabs, sales and more? Well, you know what to do.

To see how we may use your information, take a look at our Privacy Policy.

You want it, we have it.

PRACTICE TIME!

ASSIGNMENT

Create a new page that has a modal popup covering the main content

WEB FONTS

WEB FONTS

Remember that `font-family` looks for a font installed on the user's local machine.

```
body { font-family: Tahoma, sans-serif; }
```

If the Tahoma font isn't found, the browser will default to a generic sans-serif font instead.

What if you want to use an interesting font that most people aren't likely to have installed?

WEB FONTS

The absolutely easiest way to get custom fonts is to link to a Google font stylesheet in the head of your page:

```
<link href="https://fonts.googleapis.com/css?family=Roboto"
rel="stylesheet">
```

Then just use the font by name in `font-family`, just like you would a web-safe font:

```
p { font-family: Roboto; }
```

WEB FONTS

The other way to ensure people see the correct font is to download the font onto their computer when they load your page.

- Have to consider licensing fees – most fonts cost money

ICON FONTS

Font Awesome is a free font that is used for icons

- Heavily used in real-world projects

To use Font Awesome, add this `script` to the head of your page:

```
<script
src="https://use.fontawesome.com/2e71adfc30.js">
</script>
```

PRACTICE TIME!

ASSIGNMENT

Find a free font from Google fonts and use it on your site

“HOMEWORK”

- Practice!

- Optional: read chapters 15 and 17 of *HTML and CSS: Design and Build Websites*