Exercise 3-5-1: Floating

In the exercises in this chapter, we'll make further improvements to the Jenware home page that we worked on in Chapter 14. If you did not follow along with the exercises in the previous chapter, or if you'd just like a fresh start, there is a copy of the document in its most recent state, *jenware.html*

1. Open the Jenware home page document in a text editor and browser We'll start by removing wasted vertical space next to the product images by floating

the images to the left. We'll use a contextual selector to make sure that we float only those images in the "products" section of the page. While we're at it, let's add a little margin on the right and bottom sides using the `margin` shorthand property.

```
#products img {
float: left;
margin: 0 6px 6px 0;
}
```

Save the document and take a look at it in the browser. You should see the product descriptions wrapping to the right of the images.

2. Next, I'd like the "More about…" links to always appear below the images so they are clearly visible and consistently on the left side of the products section. This change is going to require a little extra markup because we need a way to target just the paragraphs that contain "more about" links. Add the class name "more" to each of the paragraphs that contain links. Here is the first one:

```
<p class="more"><a href="#">More about custom barware...</a></p>
```

Now we can use a class selector to make those paragraphs

clear the floated images.

```
#products .more {
clear: left;
}
```

The following figure shows the new and improved Products section.

Making a navigation bar

Open your copy of *jenware.html* if it isn't already.

1. Start by making the ul element as neutral as possible. The bullets have already been turned off, but we should clear out any padding and margin that might be happening in there.

```
#nav ul {
list-style-type: none;
padding: 0;
margin: 0;
}
```

2. Next float the list items to the left, and clear the following products div.

```
#nav ul li {
...
float: left;
}
#products {
...
clear: both;
}
```

Save the document and take a look at it in a browser.
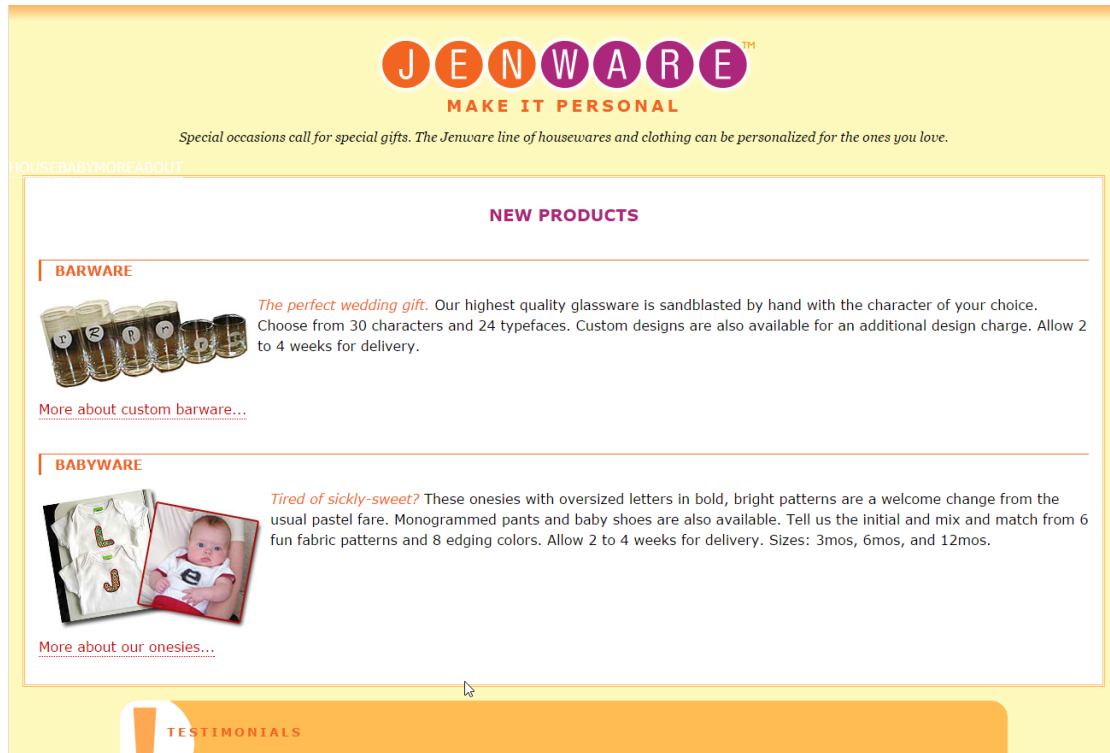
You should see that the links are now lined up pretty tightly, but also that the purple navigation bar has shrunk to nothing— float containment fail! Let's fix it with the overflow technique. And while we're at it, let's do the same for the `#products div` so it is sure to contain the floated images.

```
#nav {
...
overflow: hidden;
width: 100%;
}
#products {
...
overflow: hidden;
}
```

3. Now we can work on the appearance of the links. Start by making the `a` elements display as block elements instead of inline. Instead of setting specific dimensions for each link, we'll use padding (.5em) to give them a little breathing room inside the border and use margins (.25em) to add space between links. I've added a lavender border as the default, but I brighten it up to white for the `:focus` and `:hover` states.

```
#nav ul li a {
display: block;
padding: .5em;
border: 1px solid #ba89a8;
border-radius: .5em;
margin: .25em;
}
#nav ul a:focus {
```

```
color:#fc6
border-color: #fff;
}
#nav ul a:hover {
color: #fc6;
border-color: #fff;
}
```
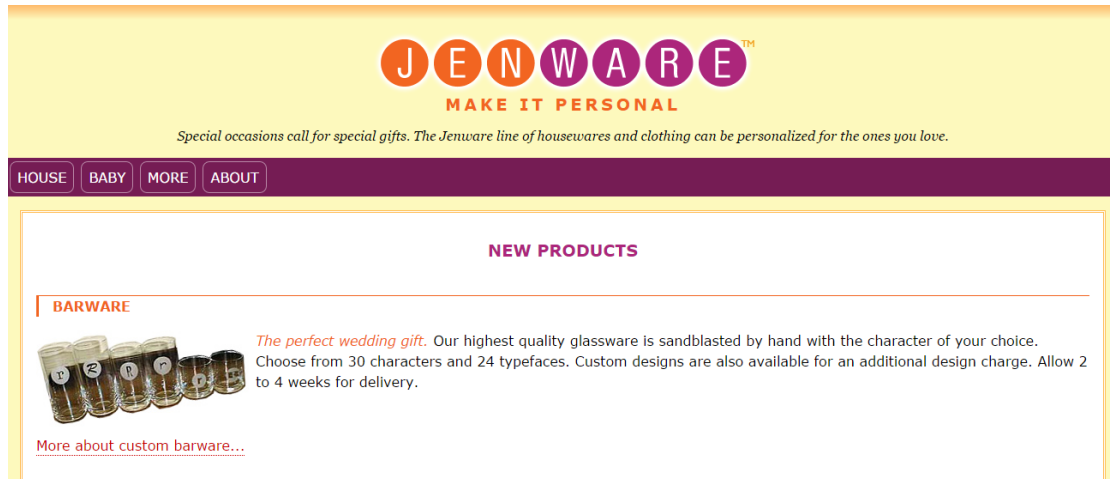

4. Finally, let's center the list in the width of the `nav` section. We can do this by applying a width to the `ul` element and setting its side margins to `auto`. I confess that I had to fiddle around with a few width measurements to arrive at one that fit the entire menu just right (19.5em). If it's too wide, the menu won't be truly centered.

```
#nav ul {
list-style: none;
padding: 0;
margin: 0 auto;
width: 19.5em;
}
```

The following Figure shows the way your navigation should look when you view it in the browser.
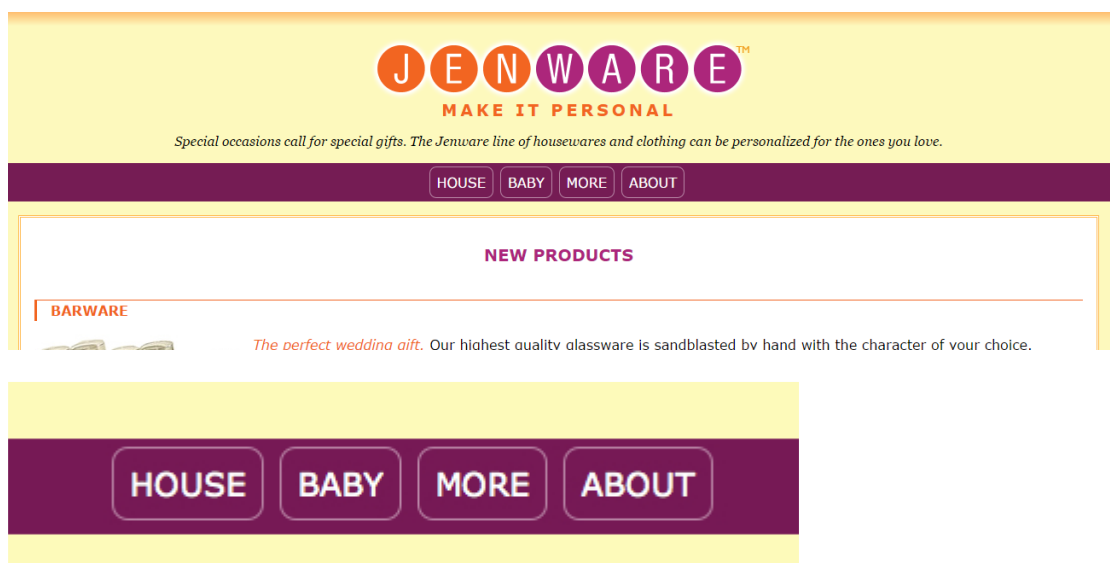
Creating columns with floats

The layout we've been using for the Jenware site might be a good starting point for a small-screen device, but it gets awkward in larger browser windows. In this exercise, we'll write styles to give the page a fluid two-column layout using floats. I recommend making a copy of your current Jenware file and renaming it *jenware-float.html*. That will keep a copy fresh for the next exercise, and you won't need to undo what you've done here.

What we're going to do is give the `#products div` a width, float it to the left, and allow the Testimonials box to flow around it on the right side, creating a second column. I want this layout to resize proportionally to always fill the width of the screen, so I'm going to use percentages for all the horizontal measurements (that means making a few changes to our prior code).

1. Start by setting the width of the `#products div` to 55% and floating it to the left. Currently the padding and margins are set at 1em all around, but change the left and right padding and margins to 2% for this fluid layout. That means the Products box is now taking up roughly 63% of the width of the screen (2% + 2% + 55% + 2% + 2%), plus a few pixels more for the borders. The following Figure shows the results of these changes.

```
#products {
    background-color: #FFF;
    line-height: 1.5em;
    padding: 0 2%;
    border: double #FFBC53;
    margin: 0 2%;
    clear: both;
    overflow:hidden;
    width:55%;
    float:left;
    }
```
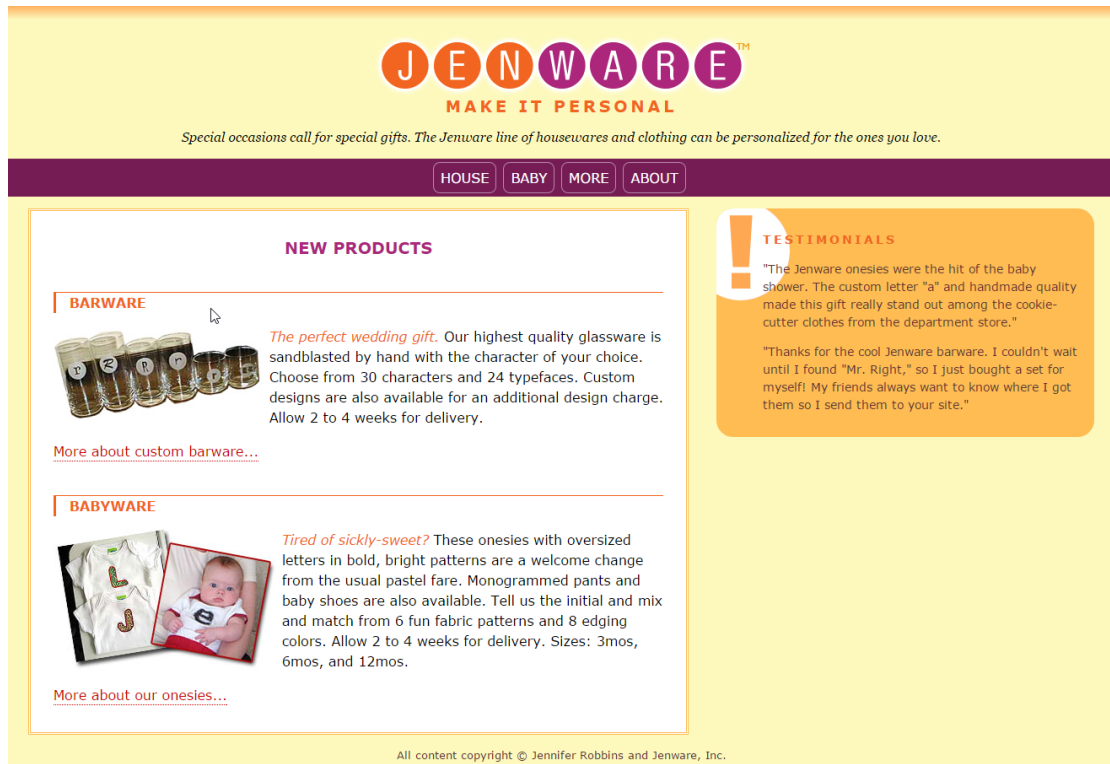
In addition, set the top margin of `#products` to zero.

```
#products {
background-color: #FFF;
line-height: 1.5em;
padding: 1em 2%;
border: double #FFBC53;
margin: 0 2% 1em;
clear: both;
float: left;
width: 55%;
}
```

There are some interesting behaviors to observe here. The Testimonials text has moved up to the right of the Products box, which is expected, but the Testimonials box (with the exclamation point graphic) is hidden behind the Products box. Only the content wraps; the element box just moves up and does not resize.

2. Time to get that Testimonials box into shape. What we need to do is adjust the margins, specifically to make the left margin on the Testimonials box wide enough that it clears the Products box. The Products box is taking up a hair more than 63% of the width of the page, so let's give the Testimonials box a left margin of 64% to accommodate it and add a little space between. I've also set a narrow right margin of 2% (remember the order of the declaration values is Top, Right, Bottom, Left). Reload the page, and the Testimonialsbox should be centered in the right column.

```
#testimonials {
...
margin: 1em 10%;/* delete */
margin: 1em 2% 1em 64%;
```

```
        }
```

3. Just a few more tweaks here. Clear the copyright paragraph so it appears at the bottom of the page. Finally, I think the "New Products" h2 would look better left-aligned in this layout, so let's adjust that too.

```
        p#copyright {
        ...
        clear: left;
        }
        #products h2 {
        ...
        text-align: center left;
        }
```

The results are shown in the following Figure.
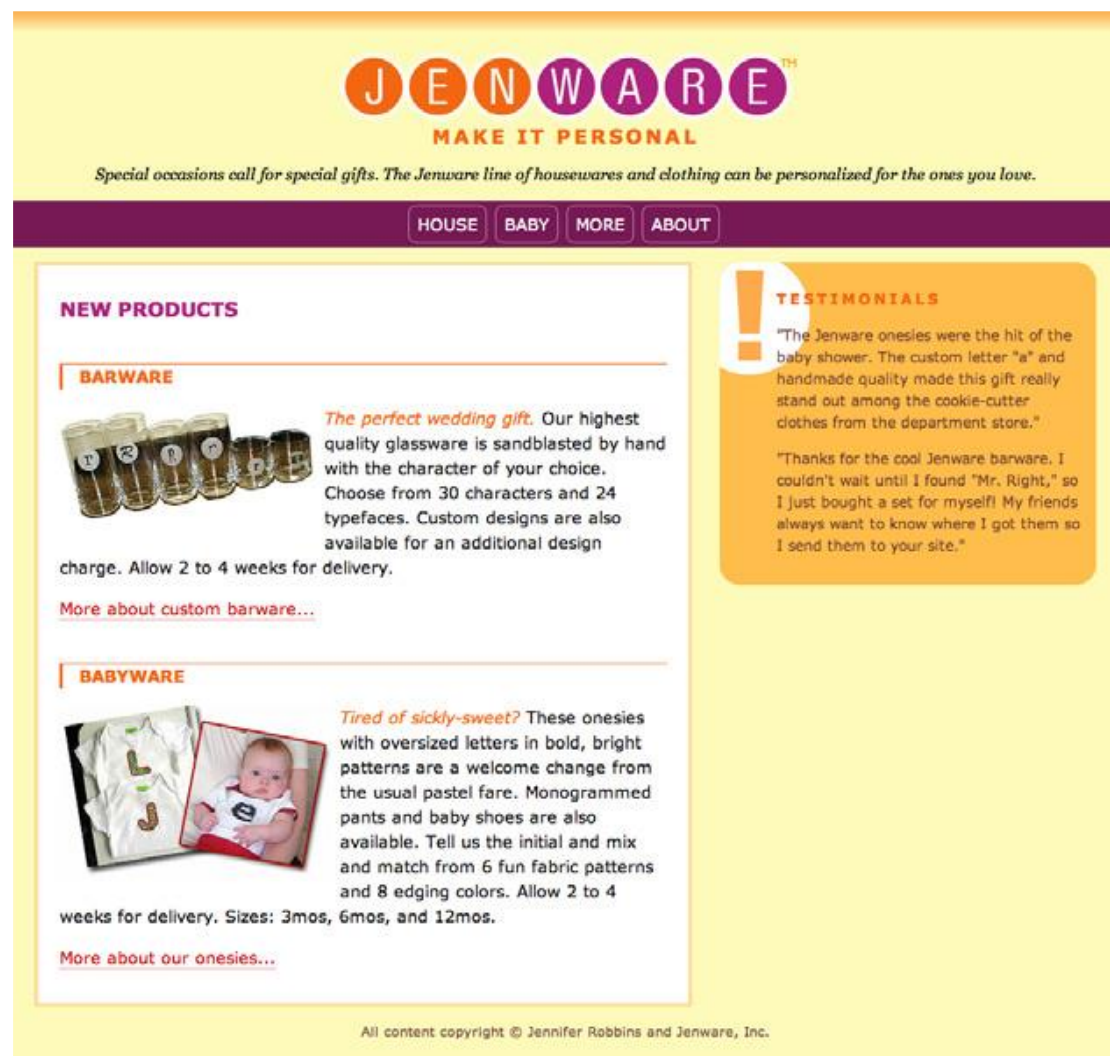
3-5-2 Positioning

Absolute positioning

In this exercise, we'll use absolute positioning to add an award graphic to the site and to create a

two-column layout. Open the *jenware.html* (or *jenware_ch15.html*) in a text editor. You should be starting with the single-column layout with floated images and a horizontal menu.

1. Let's pretend that Jenware.com won the "Awesome Site of the Week" award, and now we have the privilege of displaying a little award banner on the home page. Because it is new content, we'll need to add it to the markup. Because it is non-essential information, we'll add the image in a new `div` at the very end of the document, after the copyright paragraph.

```
<div id="award">
<img src="images/awesomesite.gif" alt="awesome site of the week">
</div>
```

Just because it is at the end of the document source doesn't mean it needs to display at the bottom of the page. We can use absolute positioning to place the `#award` `div` in the top-left corner of the browser window for all to see by adding a new rule to the style sheet that positions the `div`, like so:

```
#award {
position: absolute;
top: 35px;
left: 25px;
}
```

Save the document and take a look. Resize the browser window very narrow, and you will see that the positioned award image overlaps the header content. Notice also that when you scroll the document, the image scrolls with the rest of the page. Try playing around with other offset properties and values to get a feel for positioning in the browser window (or the "initial containing block" to use the correct term).

2. In previous Exercise, we created two columns with a float. Now let's do the same thing with absolute positioning. This time we'll make the Testimonials box a fixed width and allow the Products box to flex to fill the remaining space. This is just another common layout approach that I want you to get a feel for.

As the document stands now, if we position the Testimonials `div`, it will be relative to the browser window, which is not what we want. We want it to always appear under the `#nav` `div`, so we'll start by creating a new containing block after `#nav` that holds the products and testimonial `div`s and will serve as the new positioning context.

This is going to require some changes to the markup. Wrap `#products` and `#testimonials` in a new `div` with

an `id` of "content." The structure of the document should look like this:

```
<div id="content">
<div id="products"> ... </div>
<div id="testimonials">... </div>
</div>
<p class="copyright">...</p>
```

3. Now we can turn the "content" `div` into a containing block simply by positioning it with the "unmoved-relative-position" trick:

```
#content {
position: relative;
}
```

4. With that in place, we can position the `#testimonials` box in the top-right corner of the `#content div`. Add the position as well as top and right properties to the `#testimonials` rule as shown next. In addition, make the content 14 ems wide. Adjust the top margin to 0, and change the left and right margins from 10% to just 1em.

```
#testimonials {
…
margin: 0 1em;
position: absolute;
top: 0;
right: 0;
width: 14em;
 }
```

5. If you save the file and take a look in the browser, you should see the Testimonials box in the right corner, plopped right on top of the Products box. The next step is to put a right margin on the Products box to make a space for the Testimonials. But how much space? Let's calculate like web geeks do.

6. The Testimonials box has approximately 3.5 ems of left padding (55px), 14-em-wide content, 1 em of right padding, and a 1-em right margin, for a total of 19.5em. If we make the right margin on `#products` 20.5em, that will make space for the Testimonials box plus a little space in between the columns. We'll do it using the TRBL shorthand, as shown here.

```
#products {
…
margin: 1em 20.5em 1em 1em;
…
}
```

Save the document and look at it in the browser. Resize the window and compare how the boxes behave compared to the previous floated column example.
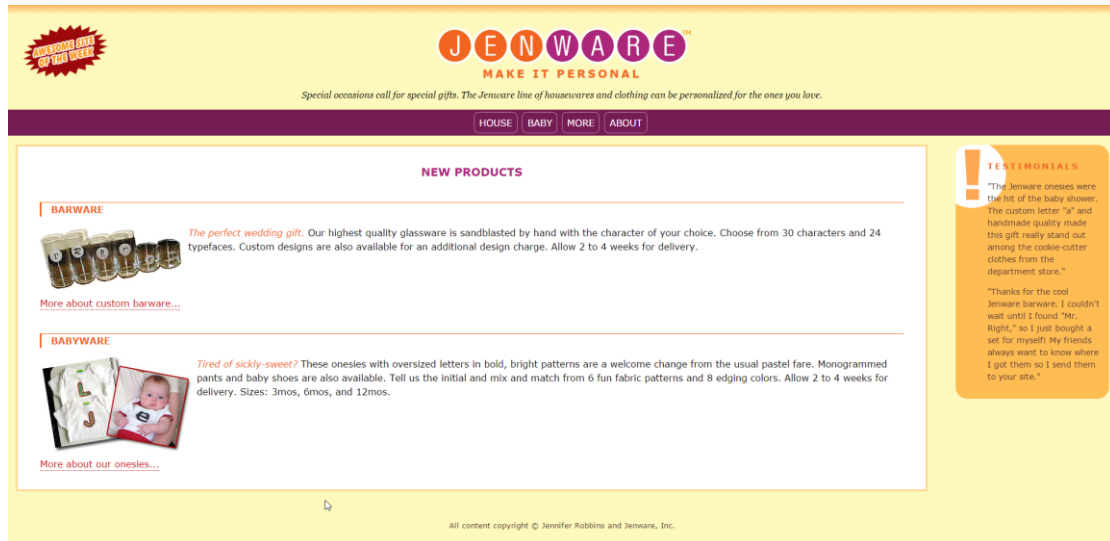
Fixed positioning

This should be simple. Open the Jenware page and edit the style rule for the `#award div` to make it `fixed` rather than `absolute`.

```
#award {
position: fixed;
top: 35px;
left: 25px;
}
```

Save the document and open it in a browser. However, when you scroll the page, you will see that the award now stays put where we positioned it in the browser window