# Mastering CSS with Mark Lassoff

## Written by Russ Eby and Mark Lassoff

# Section 7: Responsive Design

Section seven is an exciting part of the program and an important one. We're going to be talking about Responsive Design. Here's the idea:

Years ago, it was relatively easy to know what size screen your user had. There were only a couple of popular resolutions. People weren't carrying around mobile devices and tablets. Now there are dozens of common resolutions and multiple resolution displays like Mac's Retina technology.

Windows, Mac, PC, iPhone, and Android devices all have different standard screen resolutions. Therefore when we design a website, it's challenging to target a single resolution. That's where responsive design came in.

In responsive design, we attempt to create sites that are well optimized for large screens, for medium sized tablets and for small devices like your phone.

Let's take a look at how responsive design works.

## Tour of a responsive page

I've created a page called TECHNOnews.

```html
<!DOCTYPE html>
<html>

<head>
    <title>Responsive Design</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link href="reset.css" rel="stylesheet">
    <link href="https://fonts.googleapis.com/css?family=Anton|Domine" rel="stylesheet">
    <style>
        #container,
        body {
            margin: 0;
            padding: 0;
            font-family: 'Domine', serif;
        }
```

```css
header {
    font-family: 'Anton', san-serif;
    font-size: 1.25em;
    text-align: center;
    color: #005;
    background-color: #efefef;
    border-top: 10px solid #aaa;
    padding-bottom: 10px;
    width: 100;
    border-bottom: 5px solid #005;
}

header h1 {
    font-size: 3em;
}

header h4 {
    font-size: .75em;
    margin-top: 3px;
}

.grey {
    color: #aaa;
}

#about,
.story {
    margin: 10px;
    border: 1px solid #005;
    border-radius: 5px;
    padding: 5px;
    background-color: #efefef;
    box-shadow: 2px 2px #ccc;
}

.story {
    background-color: #eef;
}

.story p {
    margin-bottom: 5px;
}

#about h2,
.story h2 {
    font-family: 'Anton', sans-serif;
    font-size: 1.25em;
    margin-bottom: 5px;
}
```

```css
footer {
    text-align: center;
    border-top: 1px solid #aaa;
    bottom: 0px;
    position: fixed;
    width: 100%;
    height: 20px;
    background-color: #eef;
    font-size: .75em;
    margin-top: 10px;
}

footer p {
    margin-top: 3px;
}


img {
    width: 100%;
}

@media screen and (min-width: 600px) {
    header {
        font-size: 2em;
    }

    .story {
        width: 45%;
        float: left;

    }

    .story img {
        width: 50%;
        float: right;
        padding: 5px;
    }
}

@media screen and (min-width: 1024px) {
    header {
        font-size: 3em;

    }

    #about {
        width: 10%;
        float: left;
    }

    .story {
```

```
        width: 45%;
        float: left;
        line-height: 1.2em;
    }


    #stories {
        float: right;
        width: 86%;


    }
    }
    </style>
</head>

<body>
    <div id="container">
        <header>
            <h1><span class="grey">TECHNO</span>news</h1>
            <h2>Technology Headlines 24/7</h2>
            <h4>May 15, 2019</h4>
        </header>
        <section id="about">
            <h2>About Us</h2>
            <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Illum consectetur ius
        </section>
        <section id="stories">
            <div class="story">
                <h2>Keep on Loving You</h2>
                <img src="image.png">
                <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Obcaecati odit num
                    Reprehenderit maxime expedita vitae, obcaecati cumque, autem eum perspicia
                    Ipsam dolorum rerum consectetur quis enim reiciendis modi quam cupiditate
            </div>
            <div class="story">
                <h2>Only the Lonley</h2>
                <img src="image.png">
                <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Obcaecati odit num
                    Reprehenderit maxime expedita vitae, obcaecati cumque, autem eum perspicia
                    Ipsam dolorum rerum consectetur quis enim reiciendis modi quam cupiditate
            </div>
            <div class="story">
                <h2>Pictures of you</h2>
                <img src="image.png">
                <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Obcaecati odit num
                    Reprehenderit maxime expedita vitae, obcaecati cumque, autem eum perspicia
                    Ipsam dolorum rerum consectetur quis enim reiciendis modi quam cupiditate
            </div>
            <div class="story">
                <h2>Long Train Running</h2>
                <img src="image.png">
```

```
            <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Obcaecati odit num
                Reprehenderit maxime expedita vitae, obcaecati cumque, autem eum perspicia
                Ipsam dolorum rerum consectetur quis enim reiciendis modi quam cupiditate
        </div>
    </section>
    <footer>
        <p>Copyright &copy;2019 | Framework Television | Norwalk, CT</p>
    </footer>
</div> <!-- container -->
</body>

</html>
```

Once you have the code keyed into the text editor, open the file in your browser.

> Note: Obtain the Latin placeholder text at http://www.lipsum.com.

Let's bring up the browser's developer tools. For Chrome, click the three dots in the upper right corner to get to the browser's menu, click `more tools` and then `developer tools`.

One of the tools offered in developers tools is a screen emulator which lets us emulate different mobile devices. For example, we can emulate the screen of the iPhone 6/7/8. Or the Pixel 2. There is also has a responsive mode. In responsive mode, we can test different sized screens. In responsive mode, expand the browser out as wide as you can.

Along the top of the web page, you will see a bar with different sizes. Clicking the center will emulate a small mobile device that is 320px wide. You can use the scroll controls to emulate scrolling through the page on a small device's web browser.

Notice that when you click on the screen to scroll the mouse pointer changes into a circle, which shows you how large a finger is compared to the buttons. If the circle overlaps more than one button, you should consider spreading out the elements in your design.

By clicking on the top bar in responsive mode, the screen will change to a medium-sized mobile screen, followed by a large mobile screen.

The idea here is that the design will work across different screens. The text may reflow. The images may get a little bigger. But in general, it's the same design. But would this design work just as well on a larger device?

Clicking further out along the top bar, you'll reach a tablet view. Notice on the table size the design is different. The stories now are now placed in two columns.

The images take up a small portion of each column while displaying a mobile screen the images stretch across the screen.

In full widescreen mode, you'll notice that the design spreads across three different columns.

In summary, we have a design that is optimized and works for each screen size. Some details, like changing the size of the font for TECHNOnews in the header, are also adjusted, so the design works well for each

device. The text gets bigger on tablets and then on large screens even bigger.

## Code of a responsive page

Let's take a look at the code and the mechanics of how you create a responsive design.

When coding a responsive design, the first thing you should do is place an appropriate `meta` tag in the document `head`.

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

This is the viewport meta tag. The viewport meta tag is designed, essentially, to instruct the browser to size itself appropriately to the size of the screen.

- `width=device-width` says to the browser, "Ok, the width of the browser should be the width of the screen".
- `initial-scale=1` says we want everything scaled at 100%. We don't want people to adjust the scale with a pinch gesture.

## Reset CSS

In the head section of the document, we've also linked to a reset CSS.

```
<link href="reset.css" rel="stylesheet">
```

If you haven't used a reset CSS before, the purpose is to reset all of starting of the starting points of all your CSS elements at zero, making the random values typically found in the reset style sheet meaningless.

The one I like to use is at [Meyer Web](). It's in the Public Domain, so there is no cost to use it.

The beauty of using a reset.css is that you'll never have to guess how many pixels the margin of an `h1` is or how much padding there is for the `h2` underneath it. You don't need to override anything, but instead, style all elements from scratch.

I like doing it this way, and I think you'll like it too.

Looking inside the Reset CSS code, you'll notice that all of the tags are set to:

- margin: 0;
- padding: 0;
- border: 0;
- font-size: 100%;
- font: inherit;
- vertical-align: baseline;

All the text markup tags are set to

- display: block;

There are a few other styling rules that are also zeroed out.

You're starting from no style rules what-so-ever.

To use the reset.css, download the style sheet and copy it into a document, then saved it as reset.css. Finish up by linking it into the HTML with a `link` tag.

### Google Fonts API

Next, in the document head, I linked to the Google Fonts API. I'm using a couple of fonts: Anton and Domine. I chose these fonts from the over 900 available at Google Fonts.

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

# HTML Tour

Let's examine the HTML first and then look at the CSS that is styling it.

We have a container that contains the elements. I think it's helpful to have a logical division to be the container for everything. Otherwise, the body is the container, and the body is not a suitable container.

```
<div id="container">
    ... All of the elements
<div>
```

Next, we have a header section at the top. I'm just using `h1`, `h2` and `h4` that I later styled.

```
<div id="container">
    <header>
        <h1><span class="grey">TECHNO</span>news</h1>
        <h2>Technology Headlines 24/7</h2>
        <h4>May 15, 2019</h4>
    </header>
    ... All of the elements
<div>
```

The next section is the About us section. It contains an `h2` tag, and some Lorem Ipsum copied in.

```
<div id="container">
    ... Header Section

    <section id="about">
        <h2>About Us</h2>
        <p>...Lorem Ipsum</p>
    </section>

    ... All of the elements
<div>
```

The next section of the document contains all the news stories. There are four news stories on the web site assigned the `story` class.

Each story has an `h2` for the title and an image. Which for demonstration purposes I'm using the same one over again. Also, Lorem Ipsum text lives in a `p` tag because the text doesn't matter while designing this responsive page.

```
<div id="container">
    ... Header Section
    ... About Section

    <section id="stories">
        <div class="story">
            <h2>Keep on Loving You</h2>
            <img src="image.png">
            <p>... Lorem Ipsum</p>
        </div>
        ... Second Story
        ... Third Story
        ... Forth Story
    </section>

    ... All of the elements
<div>
```

And at the very bottom, we have the page `footer` which just contains copyright information which we'll see at the bottom of the page.

```
<div id="container">
    ... Header Section
    ... About Section
    ... Stories Section

    <footer>
        <p>Copyright &copy;2019 | Framework Television | Norwalk, CT</p>
    </footer>
<div>
```

# CSS

Now that we've reviewed the HTML for this page let's take a look at the CSS.

When working with responsive design, I highly recommend that you start with the mobile layout and expand to the larger screens, adding and enhancing elements as you go. There are different opinions on what CSS styles are going to provide an optimal experience for mobile users and full-screen users. I generally have to experiment to get a design that is well optimized on all screens.

The initial CSS described below is targeted for mobile device screens.

First, on the `#container` and `body` we set margin and padding to zero. (This probably duplicates the functionality of the reset.css, but I like to see these items to eliminate any awkward margin space around the site.)

In this CSS segment, the `font-family` was set to Domine, one of the Google fonts imported in the document `head`.

```
#container, body {
    margin: 0;
    padding: 0;
    font-family: 'Domine', serif;
}
```

The next part of the CSS is the header section.

The header is pretty straight forward. There's no responsive design used here-- Just CSS that you should already be familiar with.

```
header {
    font-family: 'Anton', san-serif;
    font-size: 1.25em;
    text-align: center;
    color: #005;
    background-color: #efefef;
    border-top: 10px solid #aaa;
    padding-bottom: 10px;
    width: 100%;
    border-bottom: 5px solid #005;
}
```

Next, we're going to set the font-size for the `h1` and `h4` elements. We'll also add some margin to the `h4` to spread provide some breathing room in the design.

```
header h1 {
    font-size: 3em;
}

header h4 {
    font-size: .75em;
    margin-top: 3px;
}
```

The class grey is coloring the word TECHNO. Looking back at the HTML you'll notice that the phrase TECHNO exists in a `span` with a class of `grey`.

```
.grey {
    color: #aaa;
}
```

The About section and the story class have these styles in common:

```
#about,
.story {
    margin: 10px;
    border: 1px solid #005;
    border-radius: 5px;
    padding: 8px;
    background-color: #efefef;
    box-shadow: 2px 2px #ccc;
}
```

For the story class, which is applied to all the news stories, we change the background color.

```
.story {
    background-color: #eef;
}
```

This next rule will give us a small space at the bottom of each story.

```
.story p {
    margin-bottom: 5px;
}
```

For the h2 in the `#about` and `story` elements, again some common styling:

```
#about h2,
.story h2 {
    font-family: 'Anton', sans-serif;
    font-size: 1.25em;
    margin-bottom: 5px;
}
```

For the footer at the very bottom of the page:

```css
footer {
    text-align: center;
    border-top: 1px solid #aaa;
    bottom: 0px;
```

The `position: fixed` will allow the footer stick to the bottom of the page even if the user scrolls through the page.

```css
    position: fixed;
    width: 100%;
    height: 20px;
    background-color: #efefef;
    font-size: .75em;
    margin-top: 10px;
}

footer p {
    margin-top: 3px;
}
```

We're giving the images a width of 100%:

```css
img {
    width: 100%;
}
```

## Media Queries

Next, in the CSS, we have the media queries which are, essentially, where the responsive design part of the CSS begins. Media queries are the heart of responsive design.

The first media query takes effect and provides CSS rules for a screen that is minimally 600 pixels wide. The changes inside this media query are applied to any screen that is 600 pixels wide but not more than 1024 pixels wide. At a width of 1024 pixels, the second media query takes control.

First we have the query itself:

```css
@media screen and (min-width: 600px) {
```

When the browser is 600 pixels wide or more, these rules take effect. This is a common breaking point for Tablet devices.

```
    header {
        font-size: 2em;   // Increases to 2em;
    }

    .story {
        width: 45%;       // instead of 100% width now will be 45%
        float: left;      // Floats to the left giving us columns.
    }

    .story img {
        width: 50%;      // instead of 100% width now 50%
        float: right;    // floats to the right
        padding: 5px;    // adds a slight padding
    }
}
```

Our next breaking point and media query are for screens 1024 pixels wide or more.

```
@media screen and (min-width: 1024px) {
```

This is a common breaking point for laptop and desktop computers. We'll be changing the overall design to a three column display.

```
    header {
        font-size: 3em;     // Again increasing the font size
    }

    #about {
        width: 10%;         // instead of 100% width now 10%
        float: left;        // floating left to make a column
    }

    .story {
        width: 45%;         // instead of 100% width now will be 45%
        float: left;        // Floats to the left giving us columns.
        line-height: 1.2em;
    }

    #stories {
        float: right;       // floating to the right for the about section
        width: 86%;         // instead of 100% width, now 86% for the about to fit in
    }
}
```

That's how responsive design works. We're able to modify the design for different size screens. And hopefully, optimize the users' experience whether they are using a phone, tablet, or full-size computer.

## Submit this

With the provided HTML, add styling to make an attractive, responsive design within the following guidelines.

1. The articles should appear stacked with the image above the `h2` above the `p`.
2. The logoImage always appears at the top but is adjusted in size depending on the width of the screen.3. When the page is in tablet size, the articles are paired up creating two columns.4. When the page is in laptop size, the articles now will be four columns, and the footer is spread out across the bottom.

Please use the same Dropbox link used in the other Mastering CSS Assignments. Please ensure that your filename includes your last name and the assignment number.

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
</head>

<body>
    <div>
        <header>
            <img id="logoImage" src="http://placehold.it/300x200">
        </header>
        <section>
            <article>
                <img src="http://placebear.com/150/100">
                <h2>Article #1</h2>
                <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Accusamus, at! Ra
                    Accusantium cumque aspernatur recusandae eum in hic fugiat, sunt maxime ab
                    Ipsum dolores, optio nesciunt, nisi quo corporis neque numquam, doloribus
                </p>
            </article>
            <article>
                <img src="http://placebear.com/150/100">
                <h2>Article #2</h2>
                <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Accusamus, at! Ra
                    Accusantium cumque aspernatur recusandae eum in hic fugiat, sunt maxime ab
                    Ipsum dolores, optio nesciunt, nisi quo corporis neque numquam, doloribus
                </p>
            </article>
            <article>
                <img src="http://placebear.com/150/100">
                <h2>Article #3</h2>
                <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Accusamus, at! Ra
                    Accusantium cumque aspernatur recusandae eum in hic fugiat, sunt maxime ab
                    Ipsum dolores, optio nesciunt, nisi quo corporis neque numquam, doloribus
```

```html
                    </p>
            </article>
            <article>
                <h2>Article #4</h2>
                <img src="http://placebear.com/150/100">
                <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Accusamus, at! Ra
                    Accusantium cumque aspernatur recusandae eum in hic fugiat, sunt maxime ab
                    Ipsum dolores, optio nesciunt, nisi quo corporis neque numquam, doloribus
                </p>
            </article>
        </section>
        <footer>
            <ol>
                <li>Facebook</li>
                <li>Twitter</li>
                <li>Snapchat</li>
                <li>instagram</li>
                <li>LinkedIn</li>
            </ol>
        </footer>
    </div>
</body>

</html>
```