UNIT 9: RESPONSIVE WEB

Review Guide

RESPONSIVE WEB DESIGN

As more and more users are accessing the internet from smart phones and tablets, this means that your site needs to be able to perform well on much smaller screens. However, simply making everything smaller might make parts of your page difficult to read or use. So what's a possible solution? Responsive Design, is designing sites for the optimal viewing experience for every device on which a site could be viewed.

To master responsive design, there are three areas to focus: Fluid Layouts, Media Queries, and Ems.

FIXED-WIDTH VS FLUID LAYOUTS

"Fixed-Width" layout means that regardless of our browser or device screen size, our blocks will be the same, fixed pixel width.

Fixed width layouts are fine for being viewed on large screens, but you will quickly experience problems when you launch your page on a smaller screen.

Fixed Width Example:

```
div {
    min-width: 50px;
    max-width: 100px;
    }
```

Fluid layouts (also known as "Liquid" Layouts) offer a solution to this issue. While fixed width layouts are defined with hard-coded pixel values, Fluid layouts define space with percentages. By using percentages, we are able to create layouts that shrink and expand relative to the browser and device size.

Fluid Width Example:

```
div {
    min-width: 50%;
    max-width: 100%;
    }
```

MEDIA QUERIES

Media Queries allow us to retrieve information about the type of device that's viewing our site and deliver CSS styles accordingly. This allows us to make changes that go beyond just shrinking the space of elements on the screen. We can unfloat and rearrange objects, alter our navigation items, and hide and show elements, all based on the width of the device that is rendering our page.

In the following example, we are telling the browser that is loading our page, "If the maximum width of the browser is larger than 500 pixels, change the font-size of the body to 150% of the default font-size."

```
@media screen and (min-width: 500px) {
    body {
      font-size: 150%;
      }
    }
```

EMS

Ems are unit of measurement, like pixels, but equal to the current font size. Unlike pixels, which are absolute, ems are relative to their parent's font size.

The em allows us to define and measure elements in relative terms. In combination with our media queries, this allows you to set the typography sizing just by setting the font size value on the body.

Fixed Width Example:

```
#fixed-unit {
    background-color: #FF6A5E;
    width: 40px;
    height: 40px;
    }
```

Em Example

```
#em-unit {
    background-color: #FF6A5E;
    width: 1em;
    height: 1em;
    }
```

QUESTIONS TO ASK YOUR MENTOR

1. Do I need to use fluid layouts, media queries, and ems in order to make my page responsive?

UNIT 9: RESPONSIVE WEB

Review Guide

2. When starting from scratch, what is the best process for building a responsive website?
3. How can I test my page to ensure that it will be responsive across different browsers (without having to go to each different type of browser on every device)?
4. Are there other approaches to building pages for various browsers and devices?
5. How do I determine which elements of my page to highlight as the screen size “shrinks”?