Adobe Dreamweaver 2021

7 – Responsive Design

In the early days of the World Wide Web, websites were viewed primarily on desktop computers or notebook computers. These days though, websites are viewed on a wide range of devices including tv screens, mobile phones and tablets. In more than half of current web users browse websites using smart phones. The problem is devices such as phones have smaller screens. Many websites which are designed using desktop computers don't look good or work well on smaller screens.

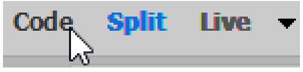
Designing websites so that they work well on a variety of devices (I.e. responding to variations in end devices) is referred to as **Responsive Web Design**.

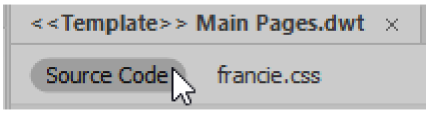
There are many ways to achieve responsive design. In the following examples we will use media queries in our CSS to provide alternate CSS rules when the screen size is smaller than a set width.

Exercise 1. Set the Viewport

We'll start by adding a meta tag in the head section of our document to define the viewport for the page.

1. Open the **Main Pages** template.
2. Switch to **Code** view and make sure you are viewing the **HTML** code and not the **CSS** code.





3. Add the following meta tag to the head section of the HTML code (after the <head> tag).

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

The viewport is the visible area of a page. Different devices with different screen sizes will have different viewports. This line we are adding will ensure that on different devices the width of the page will match the width of the screen being used to view the page. The last part ensures that the page will appear at normal zoom level with it first loads.

4. **Save** the template, updating other pages based on it.
5. Add the same line to the index.html page.

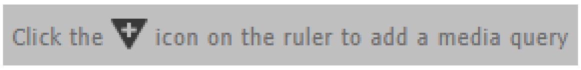
Note The site we been working on has already been designed with certain choices that make it easier to implement a responsive design. For example, setting a max-width property ensures that page elements aren't too big to fit on smaller screens.

Exercise 2. CSS Media Queries

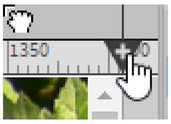
A media query enables us to add a section in our CSS file that will only apply in certain scenarios. We will use a media query to set some CSS rules that will only be used when the web page is being viewed on screens that are smaller than a certain width.

1. Switch to **Live** view.

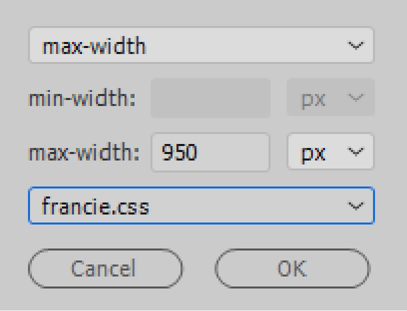
You will see a ruler along the top of the editing area along with a handy tip about media queries.


Click the icon on the ruler to add a media query

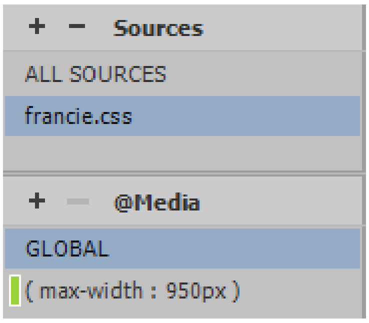
2. Click the **+** button at the end of the ruler.



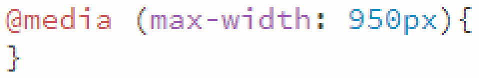
3. When the media query options appear, enter 950px for the mx-width and make sure it is added to the francie.css style sheet. This will create a new section in our CSS file that will only apply when the screen width is less than 950 pixels.


max-width
min-width: px
max-width: 950 px
francie.css

The **@Media** section of the CSS Designer panel will now show 2 options. When the Global option is selected, the **Selectors** section will show selectors that apply to the document normally. This will include all the style rules we already created. When the second option is selected, the **Selectors** section will allow us to add rules that apply to the new media query (I.e. on smaller screens).



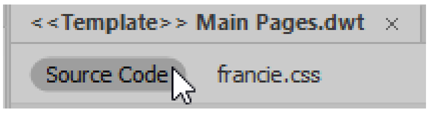
In the source for the francie.css file you will see the following added below the existing style rules.



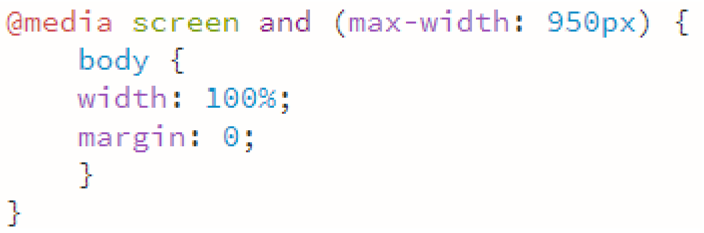
```
@media (max-width: 950px){
}
```

4. Click the (max-width : 950px) media query to select it.

Since we haven't yet created any rules for this media query, the selectors section will be blank. Adding selectors to this section is the same as before.

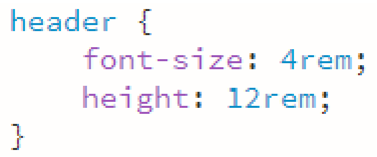


5. Add a new body selector.
6. Add properties to the new body selector that set the body to take up 100% width on smaller screens with no margins on the top or bottom. You can see the CSS code added to the CSS file as below.



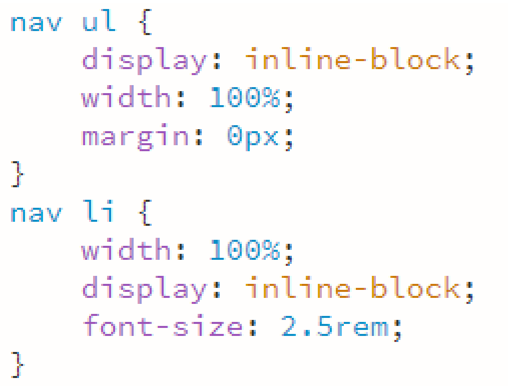
```
@media screen and (max-width: 950px) {
    body {
    width: 100%;
    margin: 0;
    }
}
```

7. Add a rule for the header selector that will make text size and height larger for smaller screens. The additional height will allow for heading text needing to display on two lines instead of one.



```
header {
    font-size: 4rem;
    height: 12rem;
}
```

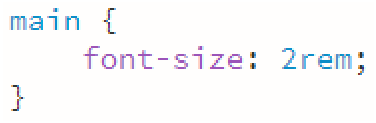
8. The next selectors will adjust the navigation so that it fits better on smaller screens.



```
nav ul {
    display: inline-block;
    width: 100%;
    margin: 0px;
}
nav li {
    width: 100%;
    display: inline-block;
    font-size: 2.5rem;
}
```

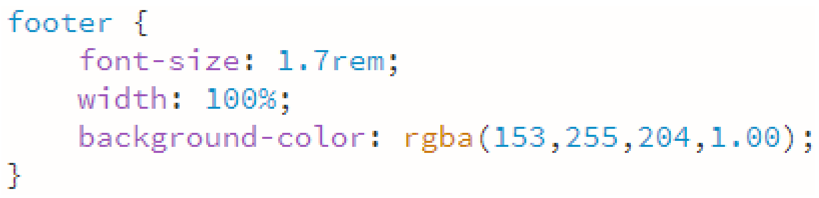
Instead of the navigation buttons displaying next to each other, they will display as 100% width blocks under each other. The text size has also been adjusted to better suit smaller screen sizes.

9. Add a rule for the main section so that the text size is larger on smaller screens.



```
main {
    font-size: 2rem;
}
```

10. Lastly, add a new selector that will make the footer selection to display better on smaller screens.



```
footer {
    font-size: 1.7rem;
    width: 100%;
    background-color: rgba(153,255,204,1.00);
}
```

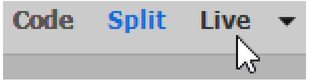
11. From the **File** menu select **Save All** to make sure all pages and the CSS file are saved.

Note You can create more than one media query. E.g. you could have one that's intended for phone size screens and another one that's intended more for tablet size screens.

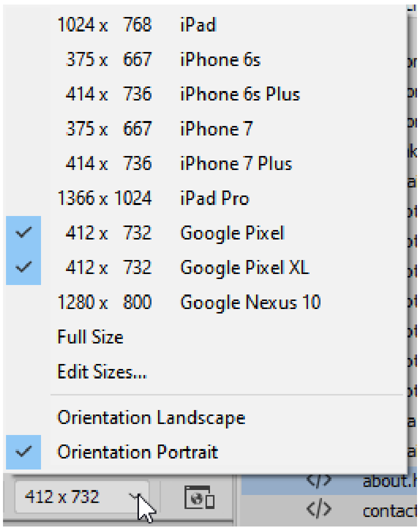
Exercise 3. Testing on Smaller Screens

Now you will need to see how it looks on smaller screens. One way is to save the whole website and copy it on to your phone to view it that way but that wouldn't be the best solution. Here's a couple of other things to try.

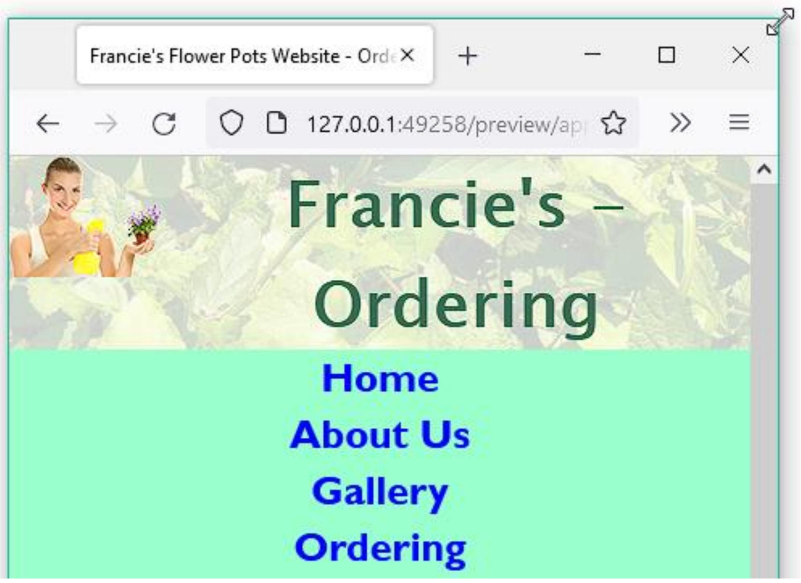
1. Open one of the pages in Dreamweaver and switch to **Live** view.



2. Use the Size options at the bottom of the editing area to see how your design looks at different sizes.



3. Preview it in an external browser **[F12]** and then simply re-size the browser window to see how the page looks at different sizes.


Francie's Flower Pots Website - Ord
127.0.0.1:49258/preview/ap
Francie's –
Ordering
Home
About Us
Gallery
Ordering

Note As mentioned in an earlier section, image maps don't work at different sizes so using one as the navigation for the main page like we have isn't so good for responsive design. Normally, we could set CSS rules so that images like the ones on the main page will scale to fit better on smaller screens but doing so would break the image map as it is currently.

If you've made this far you should know enough to experiment with different HTML elements and CSS properties to tailor the site even futher. Once you're done with that, continue to the last exercise to bring it all together for the **Nuklear Family Website**.

Code Reference

Once complete, the CSS code for the site should look like the example below.

```
@charset "utf-8";

/* general layout */
* {
   margin: 0;
}
html {
   background-color: rgba(153,255,204,1.00);
   background-image: url(images/background.jpg);
   background-repeat: no-repeat;
   background-size: cover;
   background-attachment: fixed;
   font-family: "Gill Sans", "Gill Sans MT", "Myriad Pro", "DejaVu Sans Condensed",
Helvetica, Arial, sans-serif;
   font-size: 10px;
          height:100%;
}
body {
   width: 80%;
   max-width: 1000px;
   height: 100%;
   margin: 0px auto;
}


/* type */
a {
   text-decoration: none;
}
a:link {
   color: blue;
}
a:visited {
   color: purple;
}
a:active {
   text-decoration: underline;
   color: red;
}
a:hover {
   color: red;
   border-top: thin solid blue;
   border-bottom: thin solid blue;
}
.francie {
   color: #660000;
   background-color: #99ffcc;
}


/* header layout */
header {
   height: 75px;
   background-color: rgba(255,255,255,0.80);
   font-family: "Lucida Grande", "Lucida Sans Unicode", "Lucida Sans", "DejaVu Sans", 
Verdana, sans-serif;
   font-size: 4.5rem;
   text-align: center;
   font-weight: bold;
   color: rgba(61,106,83,1.00);
}
#page_logo {
   float: left;
}


/* nav layout */
nav {
   font-size: 1.7rem;
   line-height: 3.5rem;
   font-weight: bold;
}
nav ul {
   padding: 0;
   list-style: none;
   display: flex;
}
nav ul li {
   display: inline;
   text-align: center;
   flex: 1;
}
nav a {
   display: block;
   color: blue;
   background-color: rgba(153,255,204,1.00);
   text-decoration: none;
}
nav a:hover {
   background-color: rgba(255,255,255,0.90);
   color: rgba(61,106,83,1.00);
   border: 0;
   text-decoration: none;
}
nav a:visited {
    color: blue;
}
nav a:active {
    background-color: white;
}



/* main layout */
main {
    padding: 5px;
    min-height: 100%;
    background-color: rgba(255,255,255,0.90);
    font-size: 1.7rem;
}
tr {
    vertical-align: top;
}
tr:nth-child(even) {
    background-color: rgba(153,255,204,0.30);
}


/* footer layout */
footer {
    background-color: rgba(153,255,204,1.00);
    font-size: 1.4rem;
    text-align: center;
    clear: both;
    position: fixed;
             bottom:0;
    width: 80%;
    max-width: 1000px;
}
footer li {
    list-style: none;
    text-align: center;
    padding: 0 5px;
    display: inline;
    font-size: 2rem;
}
#about_francie_image {
    float: right;
    width: 400px;
    max-width: 50%;
}
```

```
    /* Mobile Device Layout */
@media screen and (max-width: 950px) {
    body {
    width: 100%;
    margin: 0;
    }
    header {
        font-size: 4rem;
        height: 12rem;
    }
    nav ul {
        display: inline-block;
        width: 100%;
        margin: 0px;
    }
    nav li {
        width: 100%;
        display: inline-block;
        font-size: 2.5rem;
    }
    main {
        font-size: 2rem;
    }
    footer {
        font-size: 1.7rem;
        width: 100%;
        background-color: rgba(153,255,204,1.00);
    }
}
```