Topic/Title: Media Query Breakpoints

Keywords/Questions:

Mobile-friendly test

Making separate mobile site

Responsive Design

CSS breakpoints (Media query breakpoints)

@media print{ }

@media speech{ }

@media (max-width:900px){ }

@media(max-width:90px) and (min-width:50px) { }

Mobile-friendly first design

Laptop-friendly first design

Notes:

Google rankings are affected by whether if your web site is mobile friendly or not.

Mobile-Friendly test:- tells you whether your website is mobile friendly or not.

-just type mobile friendly test on google or use this link:

-https://search.google.com/test/mobile-friendly

Ways to make your website mobile-friendly:-

-Making a separate mobile site

-it will redirect users if they're trying to browse from mobile

-many big websites do this for example when you try to access Facebook from a mobile phone browser it will take us to m.facebook.com instead of facebook.com. You can access the mobile site from desktop by going to m.facebook.com

-this is a lot of work since you are designing 2 separate web sites.

-Making your website responsive

-this is what Angela recommends as the best solution.

Making websites responsive with CSS native functionality:

-Media query breakpoints(aka CSS breakpoints)

-when using CSS media queries you need to turn off or delete code that runs bootstrap because it might affect the media queries.

Summary:

Mobile-Friendly test:-tests whether your website is mobile friendly. -important for google search website ranking

-https://search.google.com/test/mobile-friendly

Making websites mobile-friendly:-making a separate mobile site -making your website responsive

Media query breakpoints(CSS breakpoints):-make your website more responsive -turnoff bootstrap when using this

@media print{ } :- property displayed while printing.

@media speech{ } :- applied when using a screen reader.

@media (min-width:900px) and (max-width:1000px){ }:-when both properties are satisfied for our browser screen

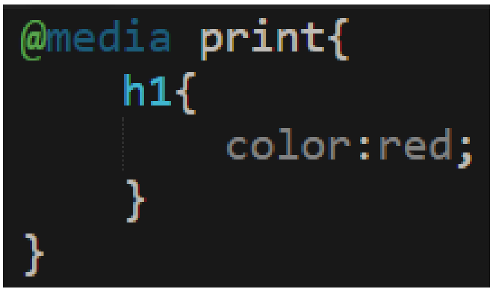
Methods for designing mobile friendly websites: -make it mobile-friendly first then tablet then laptop or - vice versa

More Notes

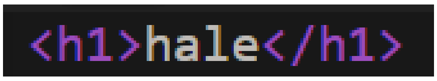
@media print

-the property will be displayed only when we print the file

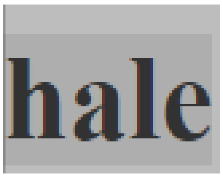
- don't forget to turnoff or delete bootstrap

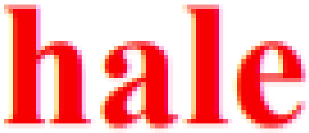
- 

```
@media print{
    h1{
        color:red;
    }
}
```



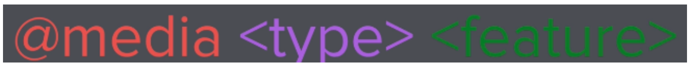
```
<h1>hale</h1>
```





@media speech

-activated if the website is being read to a visually impaired person by a screen reader



```
@media <type> <feature>
```

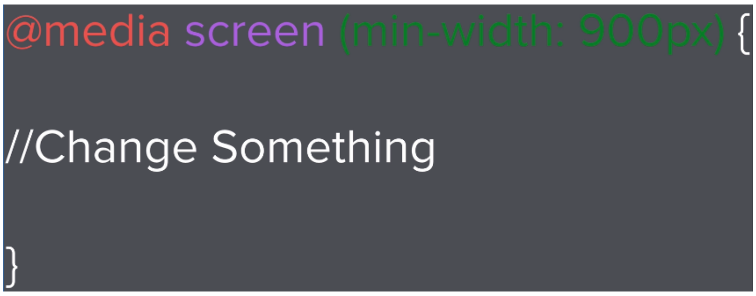
:-Usually either a <type> or a <feature> is applied at once. We don't use them both in the same line.(Not sure this is my opinion)

-@media:-this keyword says that everything that comes afterwards is a media query.

-<type>:-is the type of media or medium we are selecting on

- the code should only be activated if the web site is being displayed on a screen, or if it's being printed, or if it's on a screen reader, all of those kind of thing

-<feature>:- a condition for the property to be displayed. For example:-

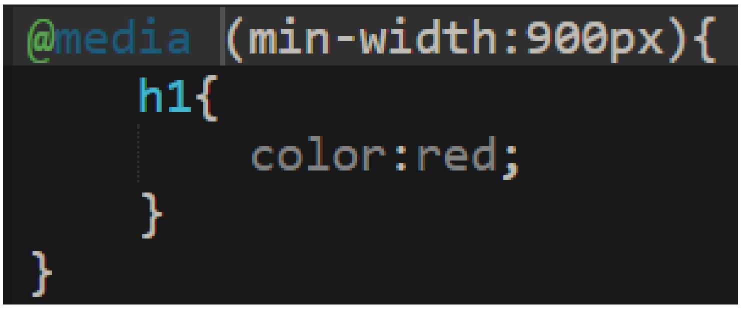


```
@media screen (min-width: 900px){

//Change Something

}
```

-the property within this { }is applied when we have a large screen, a screen > 900px(a laptop screen).



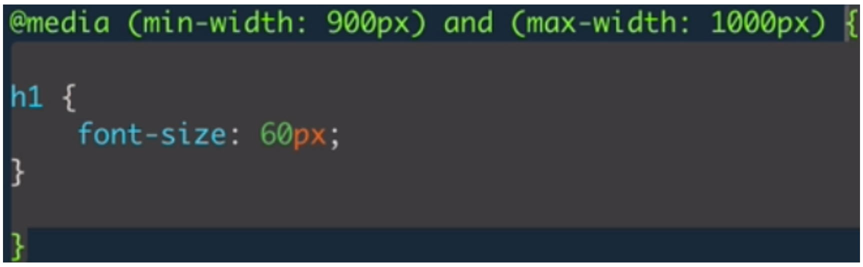
```
@media (min-width:900px){
    h1{
        color:red;
    }
}
```

-we shouldn't include screen when we write our code

@media(max−width:900px):

−displayed when the screen size is less than 900px(less than a laptop screen)

We can use more than one media queries in the same line



```
@media (min-width: 900px) and (max-width: 1000px) {

h1 {
    font-size: 60px;
}

}
```

−applied when we have browser screen size that is between 900 and 1000px.

−We can check the screen size of our browser on chrome by touching inspect(or going to developer tools) and trying to increase the screen size of my browser. On the right hand top edge you will be able to see the size of your browser.

−When when you're designing a responsive web site there's really two ways or two directions where you can go about doing this. You can either go **mobile first,** so you start designing a web site at sort of this size, and then you start looking at how you can make it look good on laptop, or the other way, which is what we've done basically, is designed our web site **for desktop and then made it responsive and look good on the smaller sizes.** Now there's a lot of debate about which direction is better, but what Angela recommends is that you actually try doing it both ways, so starting by designing and creating your HTML and CSS for a mobile size web site, and then trying to make it responsive on tablet and laptop, and also going the other direction for maybe another web site project, and you'll find out which you prefer.