

CSS

Cascading style sheets

Formatting websites

Position elements & tags

Styling - less boring

R&A HOME - student bo
page

- create a html
website

Save as index.html

<!DOCTYPE html>

<html lang="en-us">

<head>

<meta charset="UTF-8">
<title>PageTitle</title>

</head>

<body> Text
<h1> Your Name </h1>
header one

4
<img src = "imagesource"
alt = "alt description">

any text
<p>Short paragraph1 </p>
<p>Short paragraph2 </p>

 list item 1
 list item 2
 list item 3

<table>
<tr> <th> Title1 </th>
 <th> Title2 </th>
 <th> Title3 </th>
</tr>
<tr> <td> Data1 </td>
 <td> Data2 </td>
 <td> Data3 </td>
</tr>
</table> ...

LINK TO ANOTHER PAGE

LinkText

anchor b reference
another page

</body> close body

</html> close html

If you want it BOLD

 Email :

 blank

somewhere@gmail.com
(text b appear)

How do I make my
index.html public
host the site

① You need a domain name

eg annmcnamara.com

.org

.edu

.gov

need to register a domain
name

now can point domain name
to your web site

need to pay website hosting
server when you register
domain name

godaddy, wix . . .

HOSTING A WEBSITE ON GITHUB

go to github

create new repo

call it your username

annmonamza.github.io

then clone it and
put your index.html
(and other files) into
your rep

clone it

git clone annmonamza.github.io

The copy files into that
cloned directory

Then push as normal

git add -A

git commit -m "index"

git push

If you now type

anmchneramaa.github.io

you can see your web page

-github is hosting our website

Create a new repository

call it myprojects

go to settings

DON'T CREATE ANY BRANCHES

From settings scroll all the way down to GitHub Pages

Under Source

click on use the master
branch for github pages

CSS

Cascading
Style
Sheets

HTML
Hyper
Text
Markup
Language

Breathe life into our webpage

Together they define

Content +
Aesthetics

of web page

HTML alone is like writing
papers in Notepad - unformatted
text content only

HTML + CSS like using Word
Content + Format (color
font alignment, input)

`<style>:`
`</style>`

can change
attributes in
here

css works by hooking
on selectors added into
HTML using classes and
identifiers

Once hooked we apply styles
to those HTML elements using
CSS

`a { background-color: yellow; }`

↑
selector

property
value
separater

↑
declaration
separater

can have as many
property = value ; } as you want inside { }

There are many many
property = value combinations

Tint+Color

- see slide

or

google cheatsheet

04

the style tag

inside <head>

now add <style>

h1 {
 property
 margin-bottom: 15px;
 font-size: 60px;
 color: white;
 text-align: center;
 text-decoration: underline;
 background-color: black;

}

:

</style>

</head>

<head>

<style>

</style>

</head>

YAWE

You could copy and paste
the `<style>` - `</style>` into your other pages

BUT

What if you change
one thing then you
would have to go back
+ change it in everywhere
you pasted it

~~OR PREFERRED WAY~~
you can save it to
a sepeate file

`.CSS`

e.g "mystyle.css"

then you can import that
style and it can be

ste
style) reused in every html page
you reference that file
-external css file.

05- Sh - Dull Gap

```
<link rel = "stylesheet"  
      type = "text/css"  
      href = "style.css">
```

hret could reference a
local file or a webpage
new in same directory

in between <head>/</head>

If the stylesheet you
reference is not clean
then your webpage will
not appear as you wish

- If you forget a ; for
example it won't throw an
error but it won't format

You can link as many .css files as you want

What if you wanted different selectors eg not all h1s to be the same what if you want every h1 to behave a little different

You can create your own scalable selectors

eg

~~p {~~ font-size: 16px;
~~3~~ color: gray;

If you have conflicting styles when linking multiple .css files will style code

```
class  
• p1 {  
= }  
    font-size = 15px;  
    color = gray;
```

```
id  
# p2 {  
= }  
    font-size = 10px;  
    color = black;
```

you can use different prefixes if you want

IN HTML

```
<p class = "p1">
```

```
<p id = "p2">
```

- refer to it as a class

- # refer to it as id

pros & cons of both

Id U class
U .

eg

oo

To

<h

W

U

O

H

T

E

J

You can only reference
a h1 (or any) tag
using an id so if
you have

```
<h1 id="bold-font"  
class="purple">  
text  
</h1>
```

You have more flexibility
if you separate out
attributes into classes

eg

- blue
- small-font
- big-font

then you have a
similar class you
are referring to

TO CONCATENATE THEM

<hi class="blue smallfont"
MAKE ME PURPLE </hi>

When you create an id,
you can only consume it
once - you can only
have it once per tag

better to use classes

and keep classes simpler
i.e. they only do one
thing

POSITIONING

Box Model

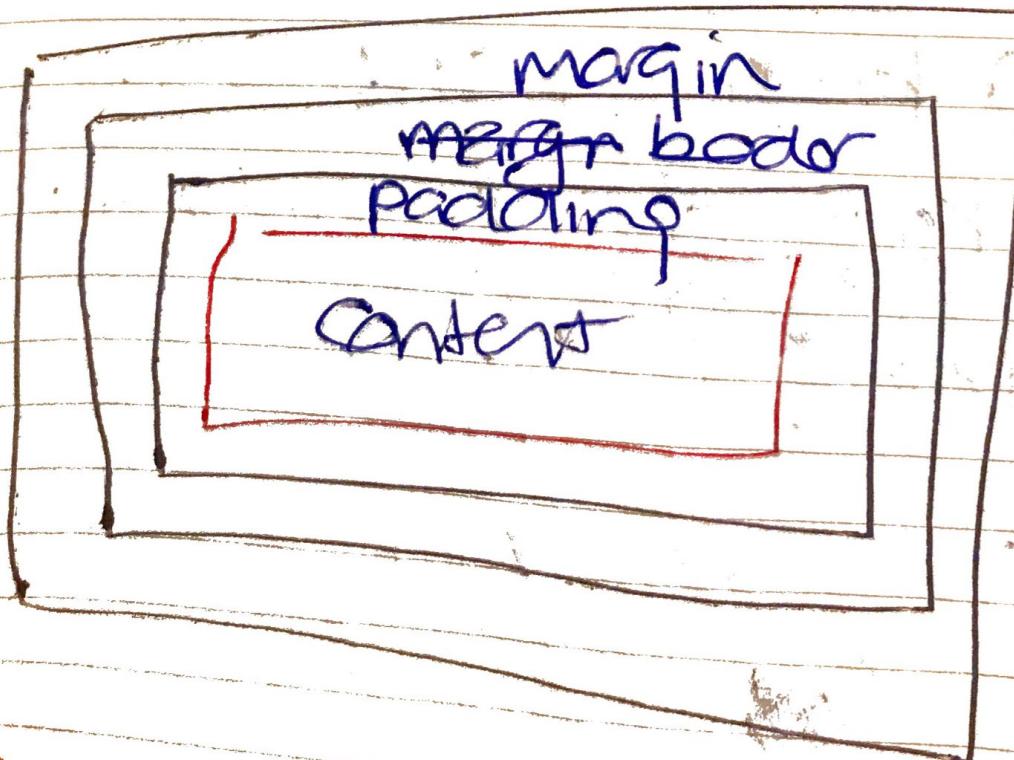
In CSS every element rests within a series of boxes

Each box has customizable space properties:

margin
border
padding

Typical spacing values:

20px 10px 10px 20px
TOP RIGHT BOTTOM LEFT



task 1 CSS summary

<style>

• style 2

position: static;

border: 3px solid #7BADD1;

absolute

positioned relative to the nearest positioned ancestor

fixed

positioned relative to viewport

- always stays in same pos

relative

relative to its normal position

static (default)

Top Bottom
left right
width height
static

Position = static (default)

Four Boxes placed STATICALLY

<style>

• static {

position: static;

border: 3px solid #7B0000;

absolute

positioned relative to the nearest positioned ancestor

fixed

positioned relative to viewport

- always stays in same pos

relative

relative to its normal position

static (default)

Top Bottom
Left Right
Next Previous
Static

sticky

zIndex

<div class="relative">

This div has pos relative

To parent box

<div class="absolute">

This div elt has position absolute

</div> </div>

<div>

</div> is a division or section in HTML

gives you flexibility

to place items where

you want them on screen

Just showing under the hood here there are more sophisticated tools to do this

"
for fixed even if you scroll up and down it stays fixed to view port

- good for important messages

- sticky

-webkit-sticky /*Safari*/

can say both for different browsers

Stays at top of browser while you are scrolling

z-index

Because the image has a z-index of -1 it will be placed behind the text

next bootstrap