CSS Values and Units

Absolute Length Units

Absolute length units are based on an actual physical unit and are generally considered to be the same size across devices.

Relative Length Units

Relative length units are relative to another element's size or settings. For example, the relative font size of an element may be calculated using the parent element's font size.

Reference :
https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/Values_and_units

Absolute length – Fixed Length value

- ***For example, numeric Values*** – Length has a number followed by a unit like *10px*, *5mm*, *8in* etc.

Relative length –relative to the device

- ***For example, percentages*** – Value will be in percent like 70%, 30% etc. from the parent.

Reference:
https://www.tutorialbrain.com/css_tutorial/css_units/

CSS UNITS

Absolute length

Units	Full Form	Font-size
px	pixel	font-size: 16px
pt	point	font-size: 12pt
pc	pica	font-size: 1pc
in	inches	font-size: 1in
cm	centimeter	font-size: 1cm
mm	millimeter	font-size: 10mm
q	quarter	font-size: 16q

Relative Length TutorialBrain.com

Units	Definition
em	Relative to the font-size of the current element
ex	Relative to the font's x-height
%	Relative to the enclosing parent element in percent
ch	Relative to the width of the digit "0"
rem	Relative to the font-size of the root element
vw	Relative to 1% of the width of the viewport
vh	Relative to 1% of the height of the viewport
vmin	Relative to 1% of the viewport (smaller between vw & vh)
vmax	Relative to 1% of the viewport (bigger between vw & vh)

moz://a

mdn web docs_

https://www.mozilla.org/en-US/

https://developer.mozilla.org/en-US/

https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/Values_and_units

Some absolute equivalences:

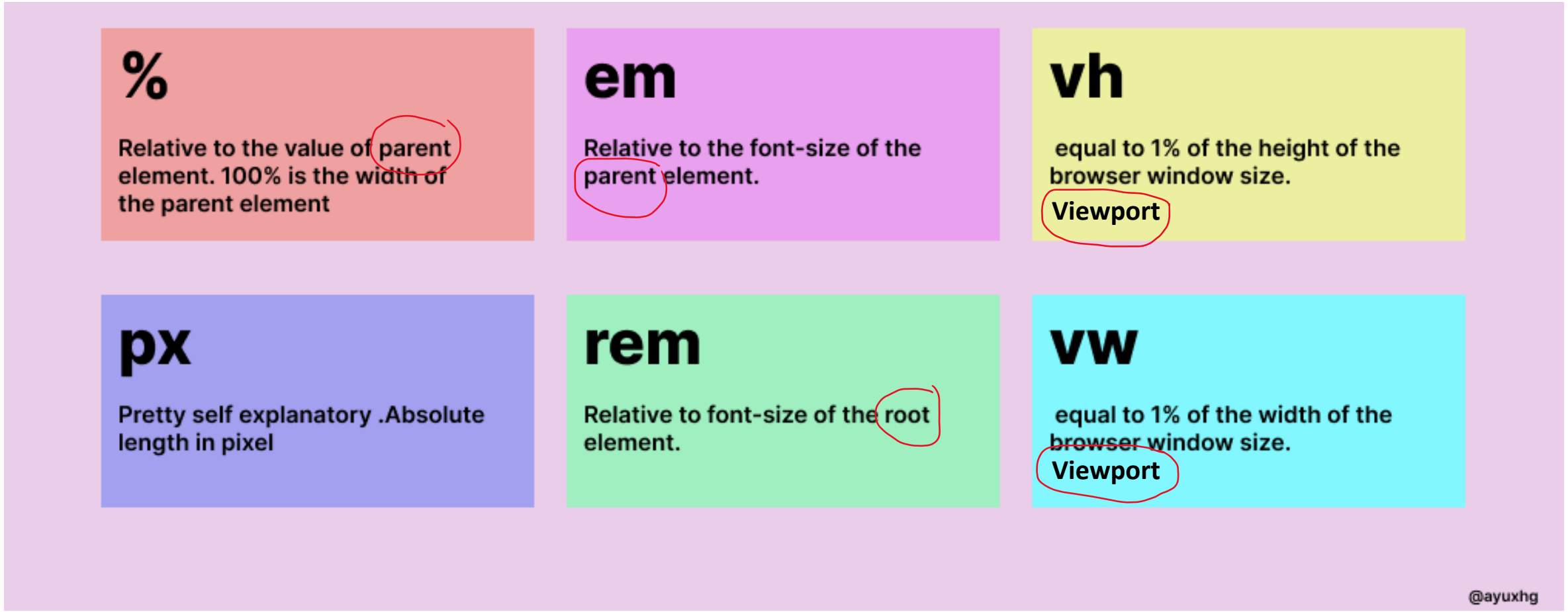
Unit	Name	Equivalent to
cm	Centimeters	1cm = 37.8px = 25.2/64in
mm	Millimeters	1mm = 1/10th of 1cm
Q	Quarter-millimeters	1Q = 1/40th of 1cm
in	Inches	1in = 2.54cm = 96px
pc	Picas	1pc = 1/6th of 1in
pt	Points	1pt = 1/72nd of 1in
px	Pixels	1px = 1/96th of 1in

Default font sizes (in pixels) for h1,2,3,4,5,6 :

Header	Pixels	Ephemeral
h1	32px	2em
h2	24px	1.5em
h3	20.8px	1.3em
h4	16px	1em
h5	12.8px	0.8em
h6	11.2px	0.7em

In most browsers, the default font size is 16px.

CSS Units


%
Relative to the value of parent element. 100% is the width of the parent element
em
Relative to the font-size of the parent element.
vh
equal to 1% of the height of the browser window size.
Viewport
px
Pretty self explanatory .Absolute length in pixel
rem
Relative to font-size of the root element.
vw
equal to 1% of the width of the browser window size.
Viewport
@ayuxhg

CSS Units

px (absolute)

Avoid using `px` for font-sizes. Use mostly for small details like border and shadow.

- fixed in size
- not responsive
- overrides user's browser preferences

% (relative)

Parent

I recommend using percentages for layouts and width/height. For example, laying out links on navbar, placing images inside a div, etc.

- size is defined as percentage of another value (mostly parent element)
- sometimes size is defined as percentage of the element itself

em (relative)

Parent

You can use `em` for font-size and margin/padding. Use `em` when you want to adjust margin/padding based on that element's font-size (if font-size is big, you maybe want bigger spacing).

- changes behavior based on property.
- 1 em = parent font-size
- if parent doesn't have a size, defaults to 16px (body)

rem (relative)

Root

You can also use `rem` for font-size and margin/padding. `rem` is easier to work with than `em` because it's more consistent.

- relative to root HTML, no matter what (default is 16px)
  - you can change the root HTML size. For example, if you change it to 20px, 1 rem will always be 20px.

vw/vh (relative)

Viewport

vw/vh are relative to the width/height of the browser window. 100vw means full width of the screen. Use vw/vh for bigger layouts, like background.

- useful for responsive websites because everything scales

ch

`ch` is relative to the width of the number 0 of the current font. `ch` is used to size the width of a paragraph. In general, you want a 45-70 character wide column for readability.

- You would use it like this:
  max-width: 40ch;
  This sets the width of the column to a maximum of 40 characters per line.

Reference:
https://yurilee.hashnode.dev/css-units-are-confusing-af

Relative Units

In most browsers, the default font size is 16px.

EM: Relative to the parent element

REM: Relative to the root element (HTML tag)

%: Relative to the parent element

VW: Relative to the viewport's width

VH: Relative to the viewport's height

Relative units multiply that number times by the default size:

1em = 16px = 1 * 16
2em = 32px = 2 * 16
.5em = 8px = .5 * 16

Relative vs Absolute Units

If we change the font-size, children will change font-size in relation to their parent :

	`body { font-size: 100%; }`	`body { font-size: 120%; }`	
`font-size: 1em`	The quick brown fox	The quick brown	relative
`font-size: 12pt`	The quick brown fox	The quick brown fox	absolute
`font-size: 16px`	The quick brown fox	The quick brown fox	absolute
`font-size: 100%`	The quick brown fox	The quick brown	relative

<example1/>

<example1/>

Units: px, vw, em

Text Style :

```
.wrapper {
    font-size: 1em;
}
.px {
    width: 200px;
    background-color: powderblue;
}
.vw {
    width: 10vw;
    background-color: green;
}
.em {
    width: 10em;
    background-color: yellow;
}
```

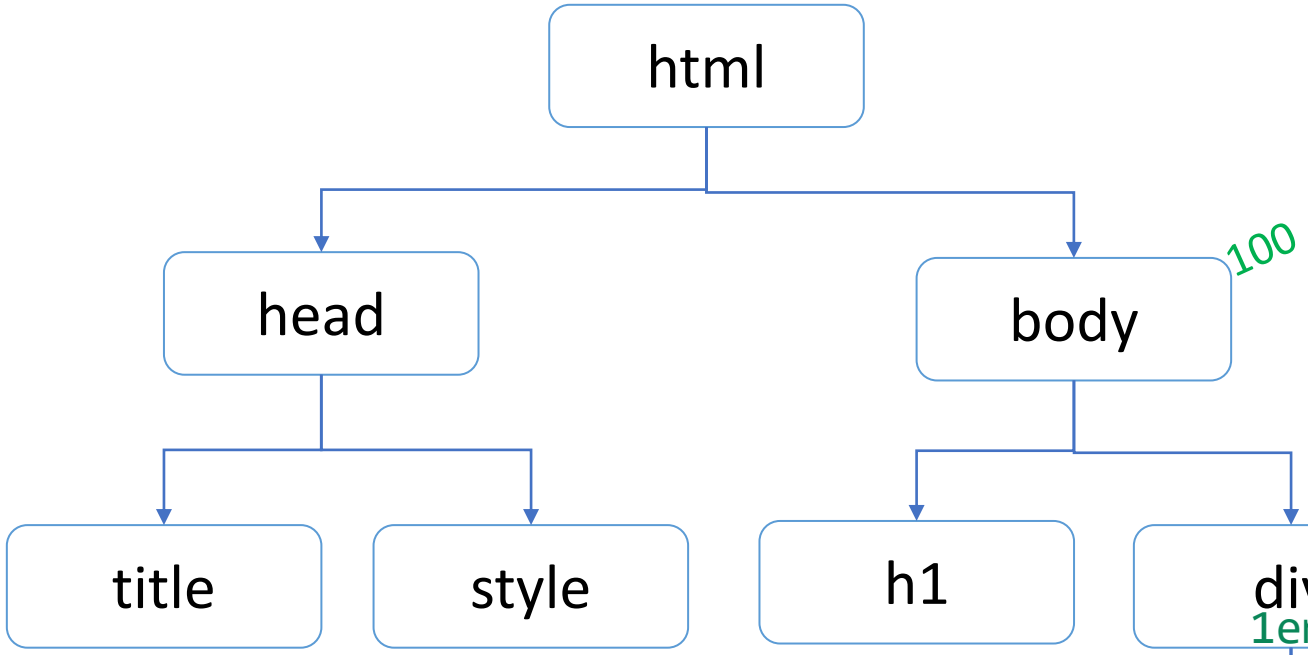
HTML :

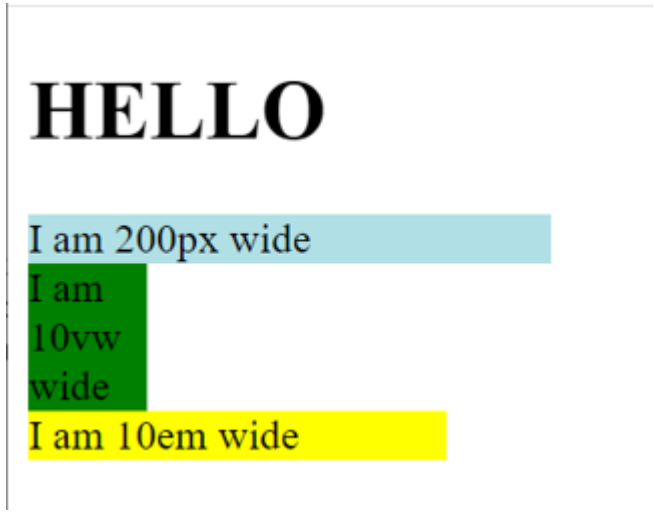
```
<div class="wrapper">

    <div class="px">I am 200px wide</div>

    <div class="vw">I am 10vw wide</div>

    <div class="em">I am 10em wide</div>

</div>
```

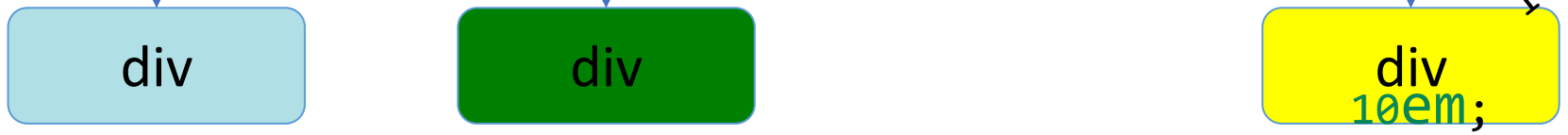
px - absolute
vw - relative to viewport (window size)
em – relative to the parent size

./html

HTML tree representation of Example 1





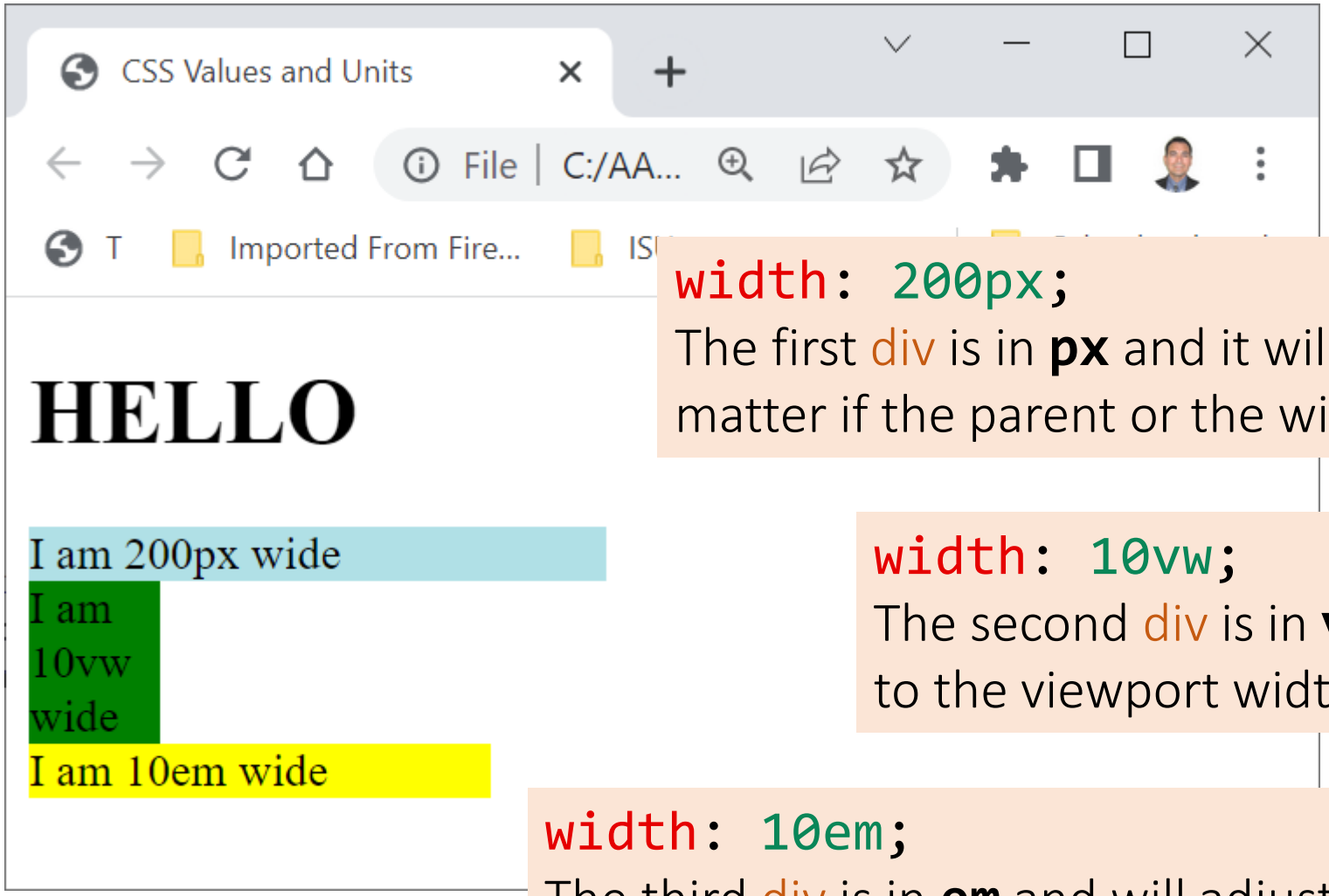



```
class="px"
200px
Absolute
```

```
div class="vw"
10vw
Relative to viewport
Change size according viewport
```

```
div class="em"
10em
Relative to parent
Parent div depend on body
```

`px` - absolute
`vw` - relative to viewport (window size)
`em` – relative to the parent size



`width: 200px;`
The first div is in **px** and it will keep its size no matter if the parent or the window change.

`width: 10vw;`
The second div is in **vw** and will adjust according to the viewport width (windows width size)

`width: 10em;`
The third div is in **em** and will adjust its size according to the parent size. The parent `wrapper` is also relative to HTML which is finally set fixed.

<example2/>

<example2/> unit em vs unit rem

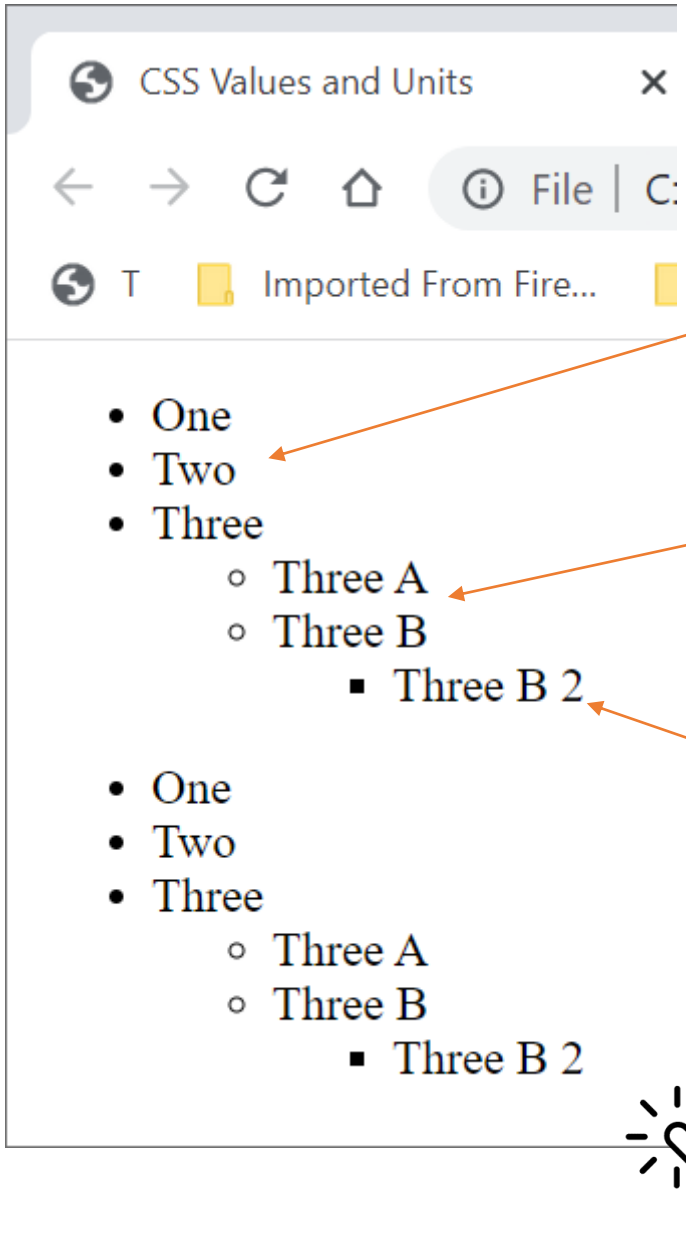
Difference between the unit em and the unit rem:

em unit means “My parent element's font-size“

rem unit means "The root element's font-size"

HTML :

```
<ul class="ems">
    <li>One</li>
    <li>Two</li>
    <li>Three
      <ul>
        <li>Three A</li>
        <li>Three B
          <ul>
            <li>Three B 2</li>
          </ul>
        </li>
      </ul>
    </li>
  </ul>

  <ul class="rems">
    <li>One</li>
    <li>Two</li>
    <li>Three
      <ul>
        <li>Three A</li>
        <li>Three B
          <ul>
            <li>Three B 2</li>
          </ul>
        </li>
      </ul>
    </li>
  </ul>
```

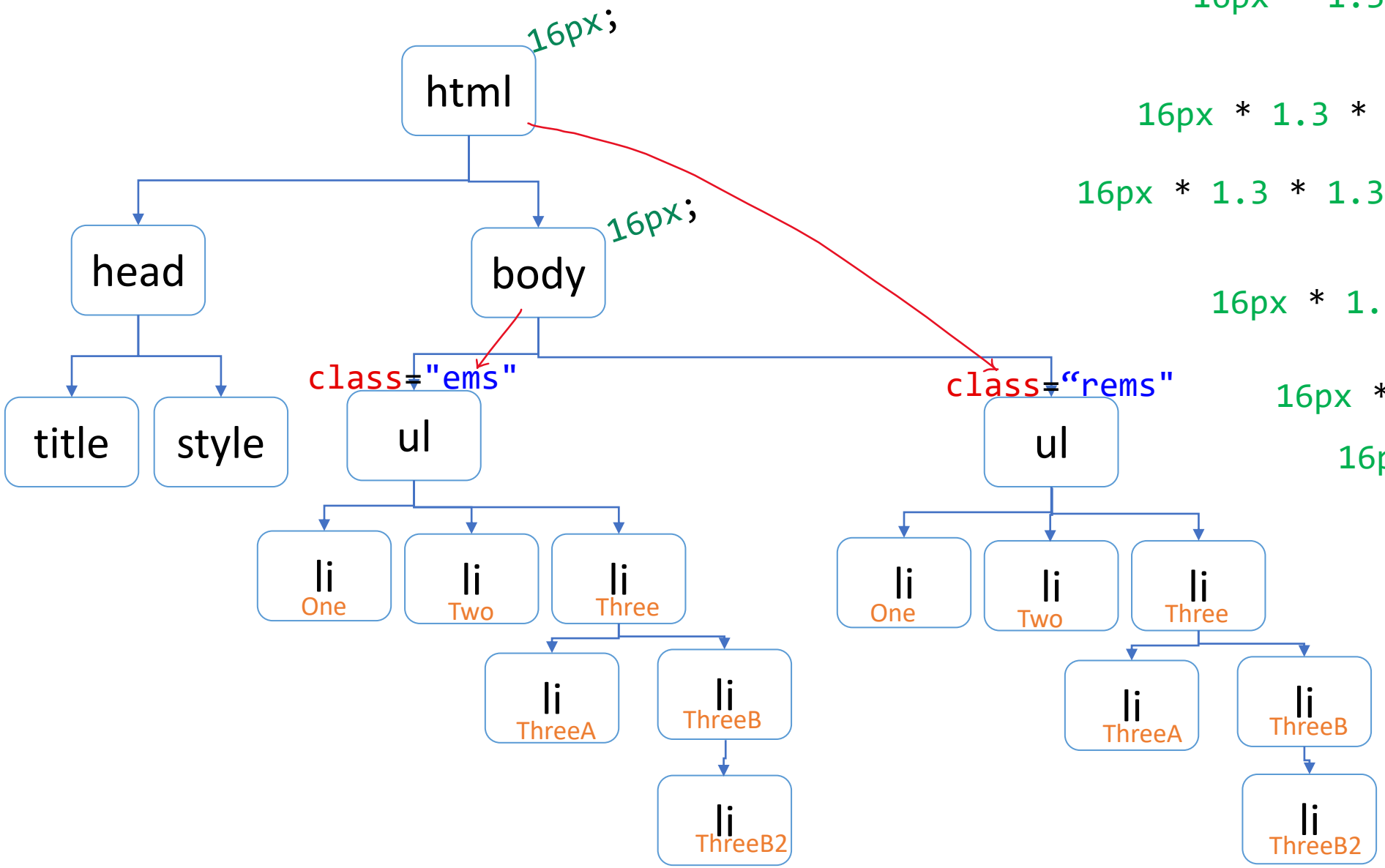


Both lists have three sub-elements.

And the third sub-element has two sub-sub elements.

Finally, the sub-sub-element B hast one sub-sub-sub-element B 2.

HTML tree representation of Example 2



16px * 1.3
- One
- Two
- Three
  - Three A

16px * 1.3 * 1.3
  - Three B

16px * 1.3 * 1.3 * 1.3 ▪ Three B 2

Always relative to parent

16px * 1.3
- One
- Two
- Three

16px * 1.3
  - Three A
  - Three B

16px * 1.3 ▪ Three B 2

Always relative to root

```css
html {
    font-size: 16px;
}
```

```css
.ems li {
    font-size: 1.3em;
}
```

```html
<ul class="ems">
    <li>One</li>
    <li>Two</li>
    <li>Three
      <ul>
        <li>Three A</li>
        <li>Three B
          <ul>
            <li>Three B 2</li>
          </ul>
        </li>
      </ul>
    </li>
</ul>
```

Always relative to parent

If we set the font-size of the element `<li>` to `1.3em`,
every child will increase its size by 1.3 times *or grand-child*

`em` **unit means "My parent element's font-size"**

```html
<ul class="rems">
  <li>One</li>
  <li>Two</li>
  <li>Three
    <ul>
      <li>Three A</li>
      <li>Three B
        <ul>
          <li>Three B 2</li>
        </ul>
      </li>
    </ul>
  </li>
</ul>
```

Always relative to root

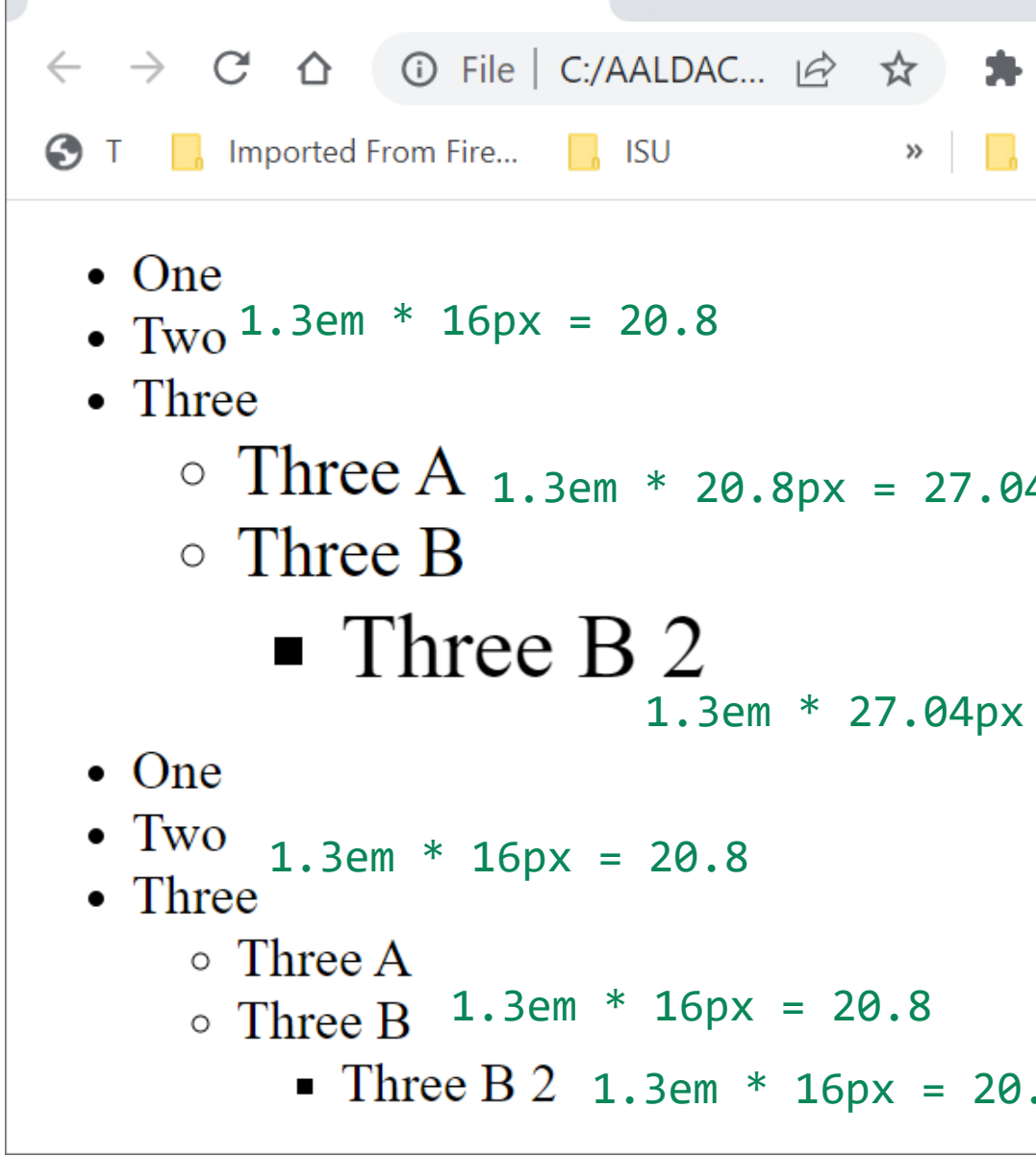
```css
.rems li {
    font-size: 1.3rem;
}
```

If we set the font-size of the element `<li>` to `1.3rem`,
children won't increase its size because they depend on **root** `html`

`rem` **unit means "The root element's font-size"**

```css
html {
    font-size: 16px;
}

.ems li {
    font-size: 1.3em;
}

.rems li {
    font-size: 1.3rem;
}
```

```html
<ul class="ems">
  <li>One</li>
  <li>Two</li>
  <li>Three
    <ul>
      <li>Three A</li>
      <li>Three B
        <ul>
          <li>Three B 2</li>
        </ul>
      </li>
    </ul>
  </li>
</ul>

<ul class="rems">
  <li>One</li>
  <li>Two</li>
  <li>Three
    <ul>
      <li>Three A</li>
      <li>Three B
        <ul>
          <li>Three B 2</li>
        </ul>
      </li>
    </ul>
  </li>
</ul>
```



<example3/>

<example3/> Unit % (percentages)

Percentage % is always relative to some other value of the parent.

For example:

- If you set an element's `font-size` as a percentage, it will be a percentage of the **font-size** of the element's **parent**.
- If you use a percentage for a `width` value, it will be a percentage of the **width** of the **parent**.

Reference:
https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/Values_and_units

CSS Style :

```css
.px {
  width: 200px;
}

.percent {
  width: 40%;
}

.wrapper {
  width: 400px;
  border: 5px solid purple;
}
```

HTML :

```html
<div class="px">I am 200px wide</div>
<div class="percent">I am 40% wide</div>

<div class="wrapper">
    <div class="px">I am 200px wide</div>
    <div class="percent">I am 40% wide</div>
</div>
```

I am 200px wide
I am 40% wide

I am 200px wide
I am 40% wide

The parent of this ** ** is **<body>**

```
<div class="px">I am 200px wide</div>
<div class="percent">I am 40% wide</div>
```

Not setting a width on the HTML and body elements will default to the full size of the screen.

Try shrinking the page width

px is absolute.

200px will be the same no matter if the div is wrapped in a parent div.

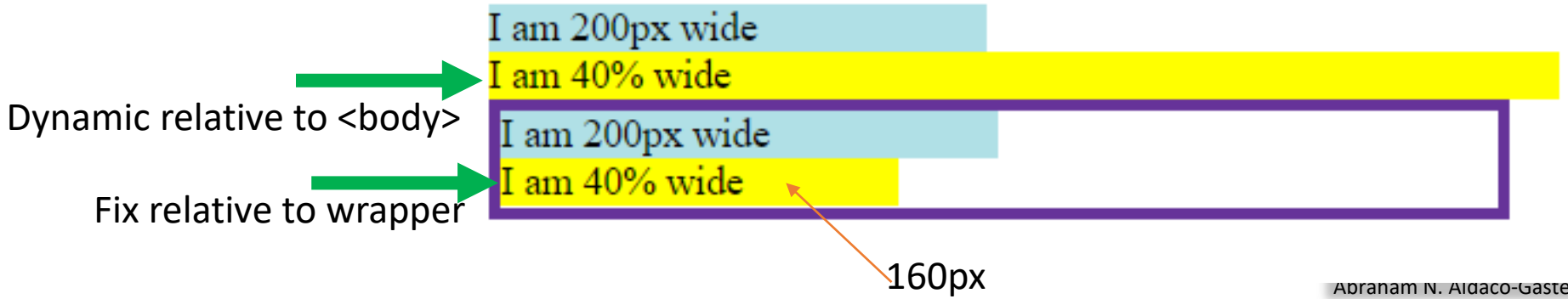
```
<div class="px">I am 200px wide</div>
<div class="percent">I am 40% wide</div>
<div class="wrapper">
    <div class="px">I am 200px wide</div>
    <div class="percent">I am 40% wide</div>
</div>
```

% is relative.

The **second** 40% is less wide compared to the first 40%, because the **second** is wrapped in a parent of 400px.

```
.wrapper {
  width: 400px;
  border: 5px solid rebeccapurple;
}

.px {
  width: 200px;
}

.percent {
  width: 40%;
}
```

```
<div class="px">I am 200px wide</div>

<div class="percent">I am 40% wide</div>

<div class="wrapper">

    <div class="px">I am 200px wide</div>

    <div class="percent">I am 40% wide</div>

</div>
```



< Go to the activity 04 />