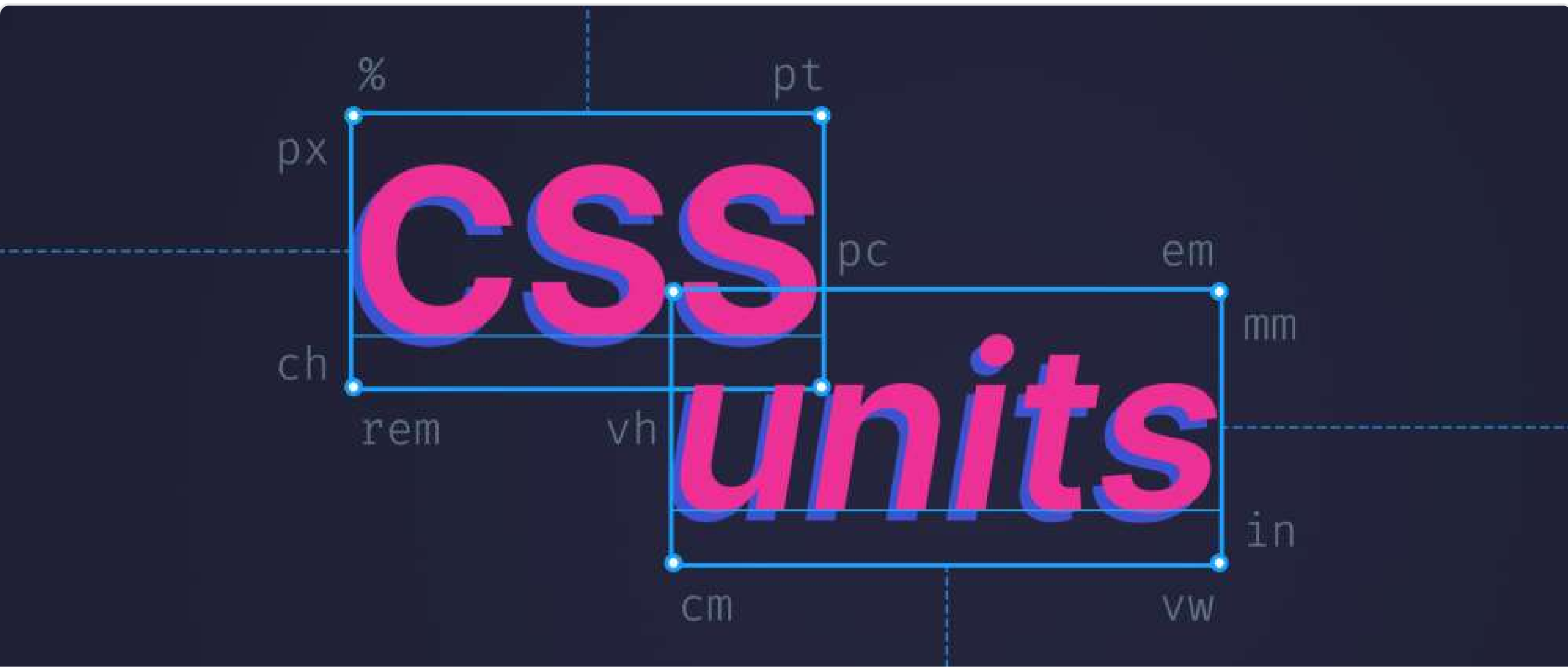


How to decide CSS units for responsive website?

#beginners #css #100daysofcode #webdev

Akram Narejo Aug 6 · 3 min read

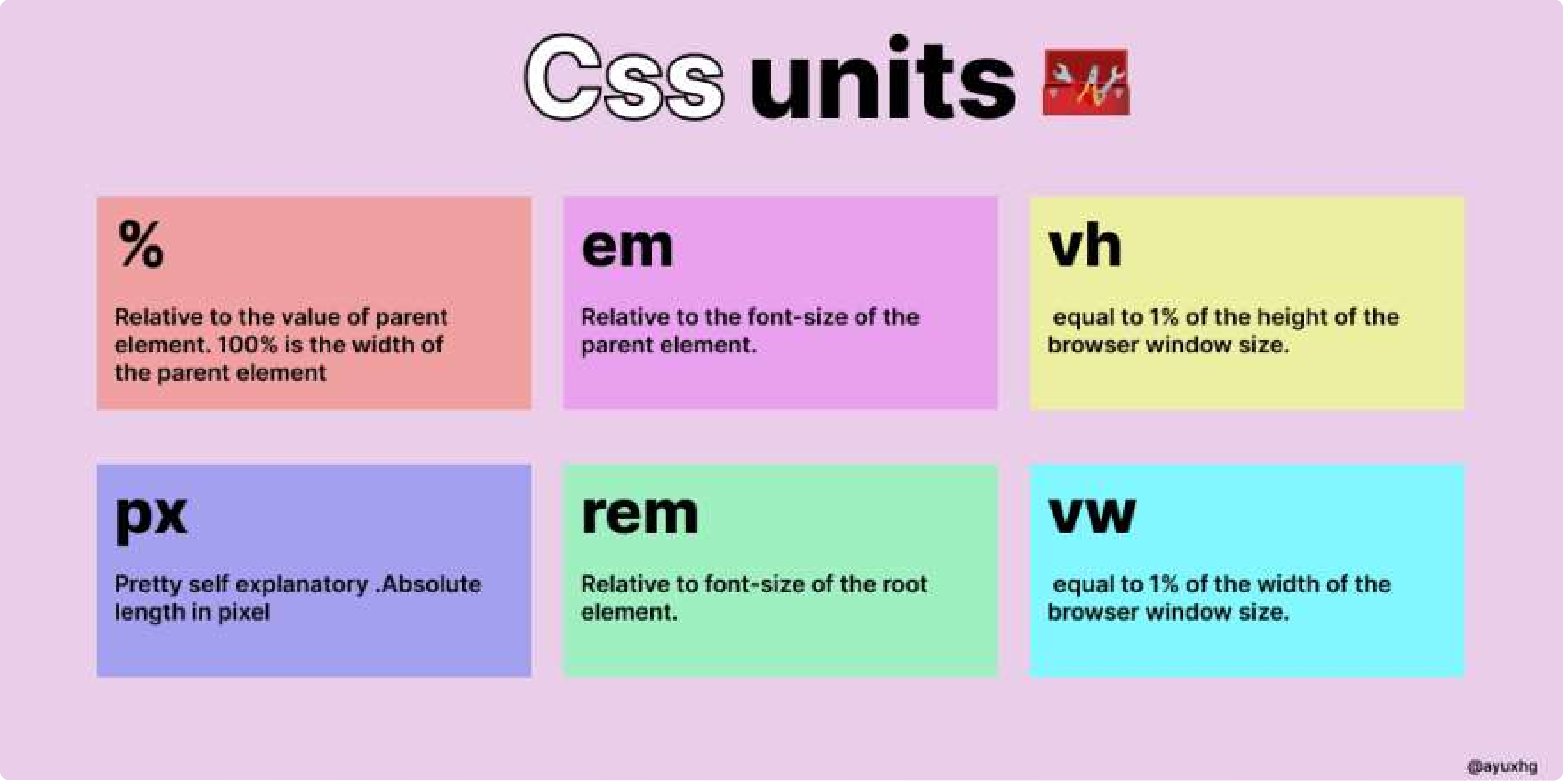
When it comes to create a responsive website to fit the every device, It is important to know the right **CSS units**. But before diving into decision making, let's classify them to know their class.

1. Absolute units
   - px - Pixel
   - pt - Point
   - pc - Picas
   - in - Inch
   - cm - Centimeter
   - mm - Millimeter

2. Percentage units
   - percentage % unit

3. Relative units
   - Relative to font size
     - em
     - rem - root em
   - Relative to view port/document
     - vw
     - vh
     - vmax
     - vmin

We are learning here the most common units:



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.

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.

@ayuxhg

px unit

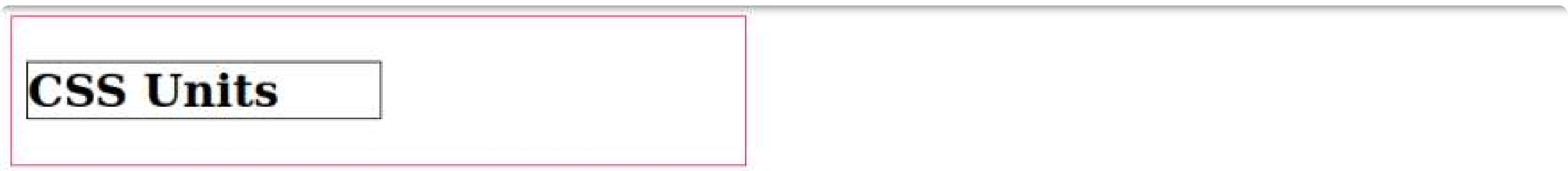
In absolute px unit is only used for screen (interface) and rest of units are for print. px unit is not a good choice, actually this is not for scaling. px unit is fixed in size no matter what screen size you choose. That's why *px unit is always preferred for borders because borders also kept fixed across all screen sizes*.

% unit

This is used for setting the width of element and it's always relative to its immediate parent element size. If there's no defined parent then by default body is considered the parent.

Considering a box with 500px of width and a h1 element inside

```
.box{
  width: 500px;
  border: 1px solid crimson;
  padding: 10px;
}
h1{
  width: 50%;
  border: 1px solid;
}
```



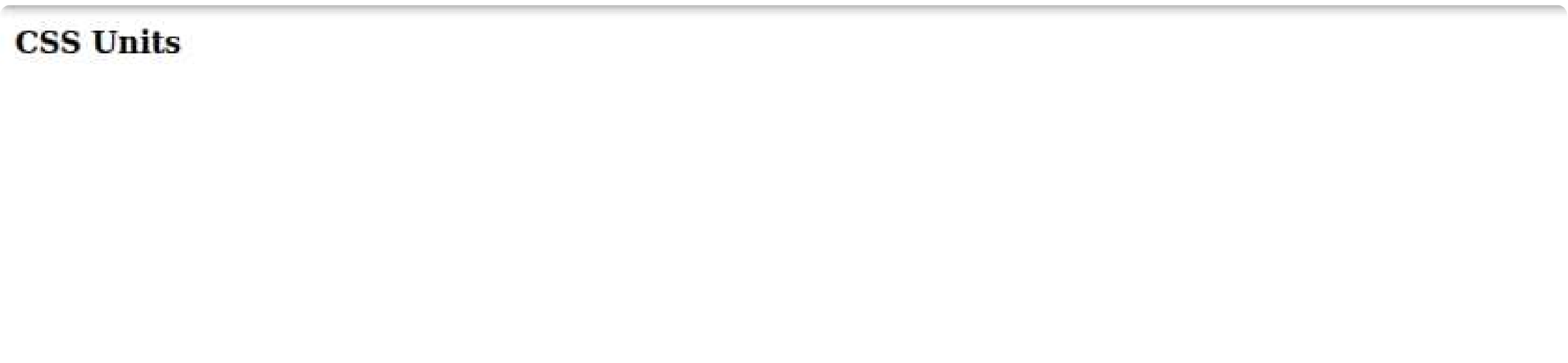
CSS Units

☞ *if there's no parent defined then root will be considered as the default parent.*

em unit

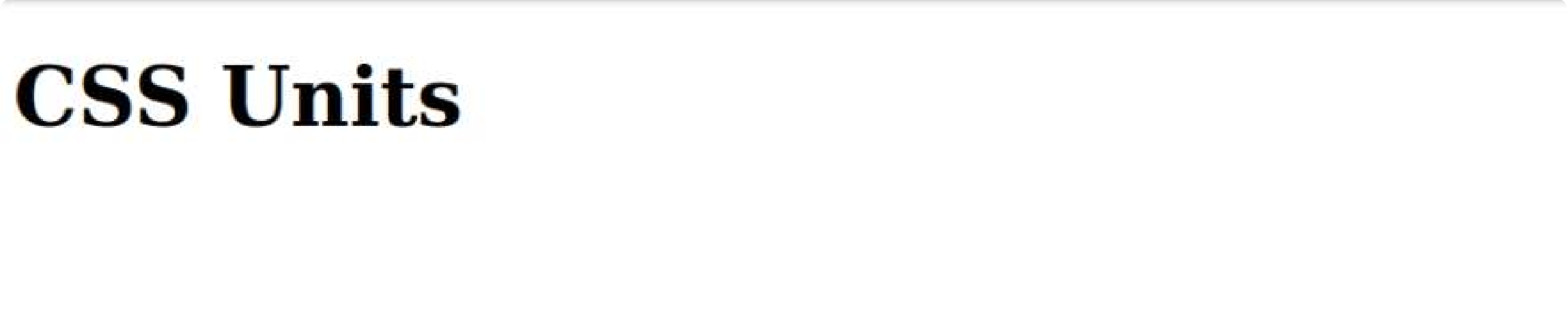
em unit is always relative to the font size of it's immediate parent. 1em == to the size of parent's font size. Default font size is 16px if not overridden, let's say in parent element the font size is 48px then in the child element 1em == 48px.

```
h1{
  font-size: 1em; /* now 1em == 16px */
}
```



CSS Units

```
.container{
   font-size: 48px;
/* or 3em, because default font-size is 16px
& it's parent is body so, 3*16px will be 48px */
}
h1{
   font-size: 1em; /* now 1em == 48px */
}
```



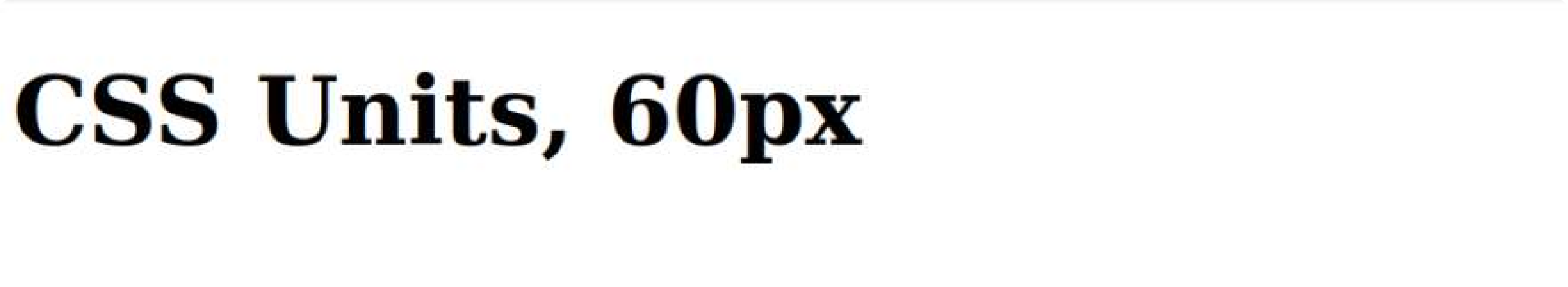
CSS Units

we can use this unit for margin and padding because it can allow us to use flexible spacing around the boxes according to our font-size of the element. So, element font-size will change according to device size therefore, spacing around the element will also change respectively.

rem unit

r stands for *root em* and unlike em it's always relative to the root font size no matter what font size it's very next parent element has. if the root has redefined font-size like 60px then 1rem == 60px in child.

```
html{
   font-size: 60px;
}
.container{
   font-size: 16px;
}
h1{
   font-size: 1rem;
}
```



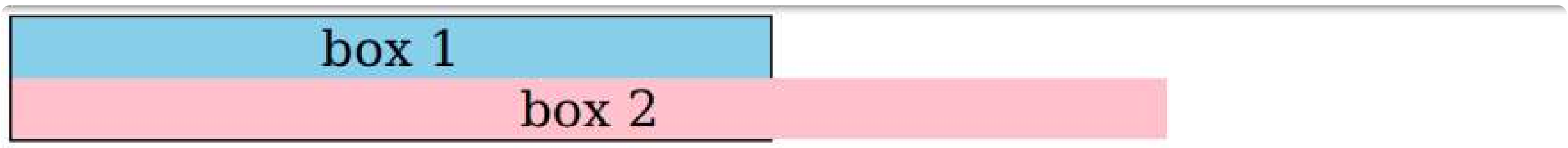
CSS Units, 60px

vw unit

vw stands for the viewport width which means vw is always relative to the 1% of the width of root, irrespective of parent element's width. So, if 1vw == 1% then 100vw == 100% of the viewport width.

let's consider a following example where the width of one child is relative to parent size and other's is relative to root.

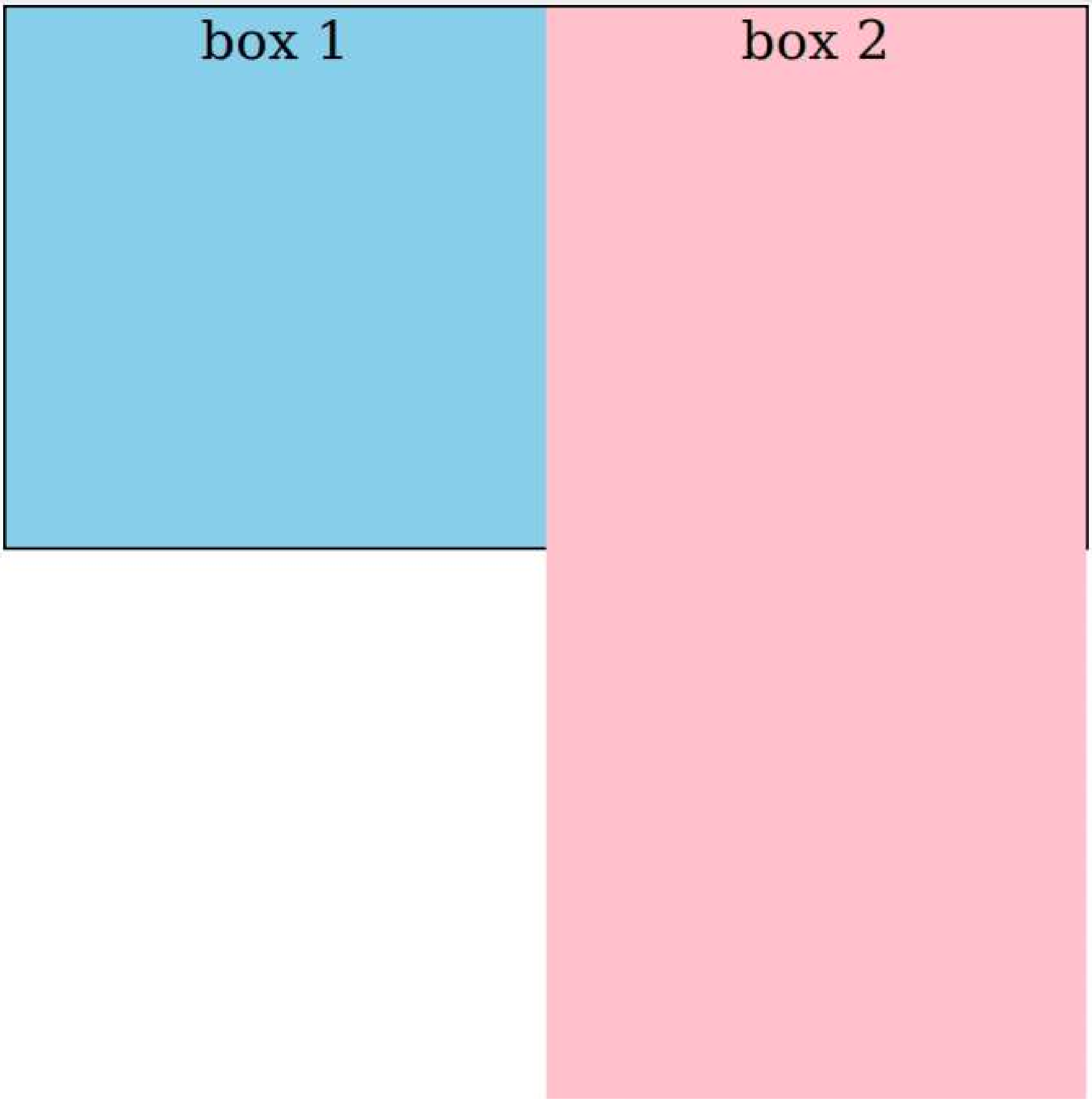
```
.container{
  width: 600px;
  border: 2px solid black;
  text-align: center;
  font-size: 20px;
}
.box1{
  width: 100%;
  background: skyblue;
}
.box2{
  width: 70vw;
  background: pink;
}
```



vh unit

vh stands for viewprot height like vw it's also relative to the 1% height of root/document.
let's consider a following example where the height of one child is relative to parent size and other's is relative to root.

```
.container{
  border: 2px solid;
  font-size: 40px;
  width: 800px;
  height: 400px;
  display: flex;
  text-align: center;
  margin: 0 auto;
}
.box1{
  height: 100%;
  width: 50%;
  background: skyblue;
}
.box2{
  height: 100vh;
  background: pink;
  width: 50%;
}
```



Summary

- px unit for borders.
- % unit for width relative to parent.
- em unit for margin and padding relative to fonts size of element.
- rem unit for fonts size relative to root.
- vw and vh for width and height relative to root.

These are the 6 css unit, which are the most commonly used to make the website responsive.

Hope, I have helped you to understand these concepts somehow. Your feedback will be appreciated 👏

I would love to connect with forntend devs on linkedin

Discussion (3)

G Paterne • Aug 8

Très bonne explication. Car, certains utilisent les unités dans le désordre.

Thomas-Max99 • Aug 8

Surtout les développeurs-youtubeurs

Akram Narejo • Aug 8

merci à vous deux.

Code of Conduct • Report abuse

Akram Narejo

I'm a frontend developer, code the better UI for the better UX. I'm open to work.

LOCATION
Pakistan

EDUCATION
Bachelors in Computer Science

WORK
Frotnend Dev at Fiverr

JOINED
Jul 29, 2020

More from Akram Narejo

How to push remote changes to github with another account
#github #productivity #webdev #beginners

diff and patch explained
#github #git #python #beginners