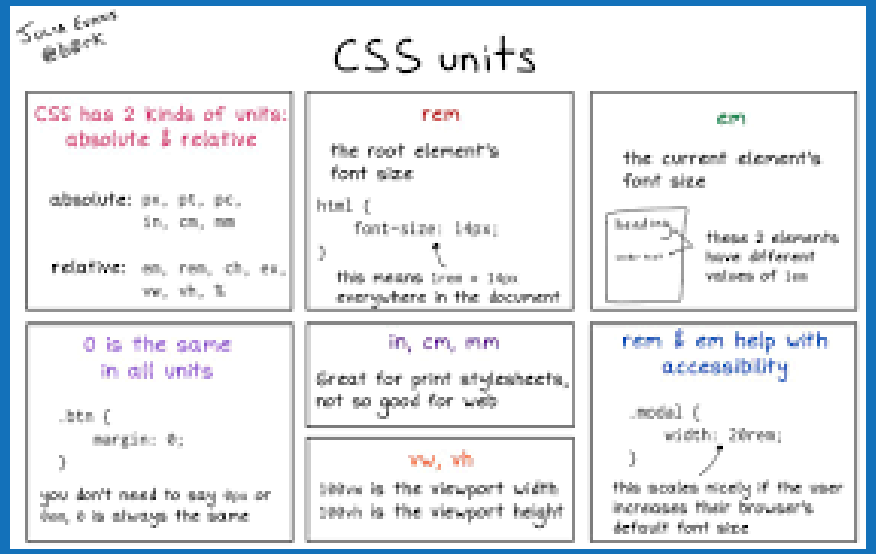
netlink
digital solutions

www.netlink.com

CSS UNITS (INTRODUCTION)



Introduction

- CSS has several options for which units to use when determining the size of various CSS properties.
- Learning all your options for CSS units can be key for styling in a way that’s easy to manage and looks great on any screen.

What is a CSS Unit?

- A CSS unit determines the size of a property you're setting for an element or its content.

- For example, if you wanted to set the property margin of a paragraph, you would give it a specific value. This value includes the CSS unit.

Let's look at a small example:

```
p {
  margin: 20px;
}
```

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.

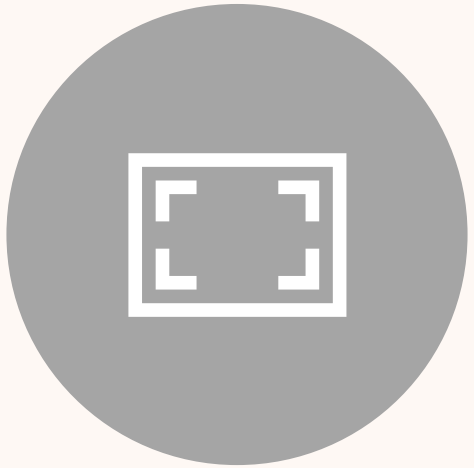
@ayuxhg

“What’s the best unit to use here?”

“When to use which unit?”

The length unit in CSS is of two types:-

ABSOLUTE LENGTH



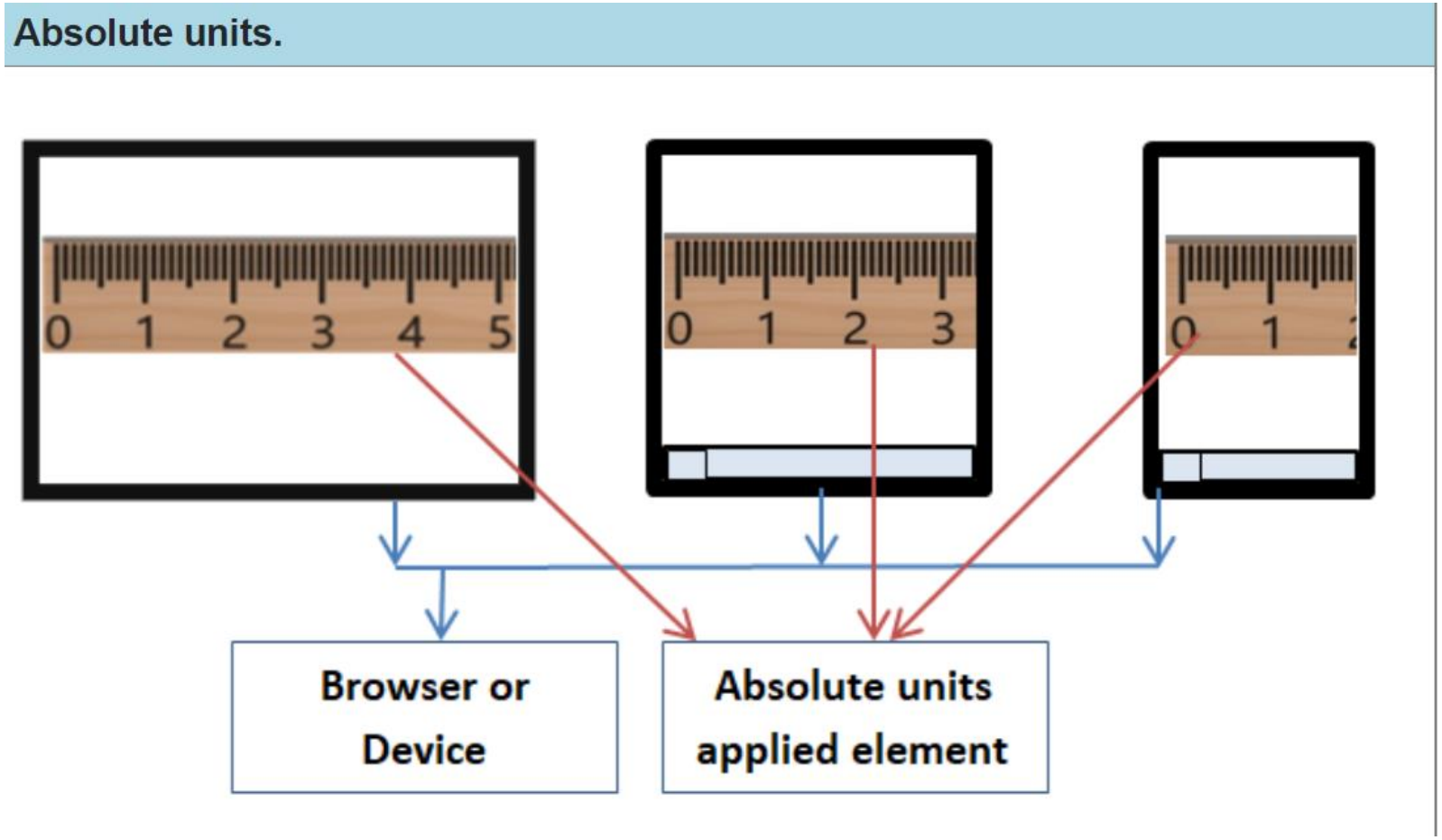
RELATIVE LENGTH

Absolute Units

- Absolute units will be the same regardless of screen size or other settings.

- Absolute units are fixed across everywhere else. Also, it remains absolute if the output device has a higher resolution.

Absolute Unit



Absolute Units

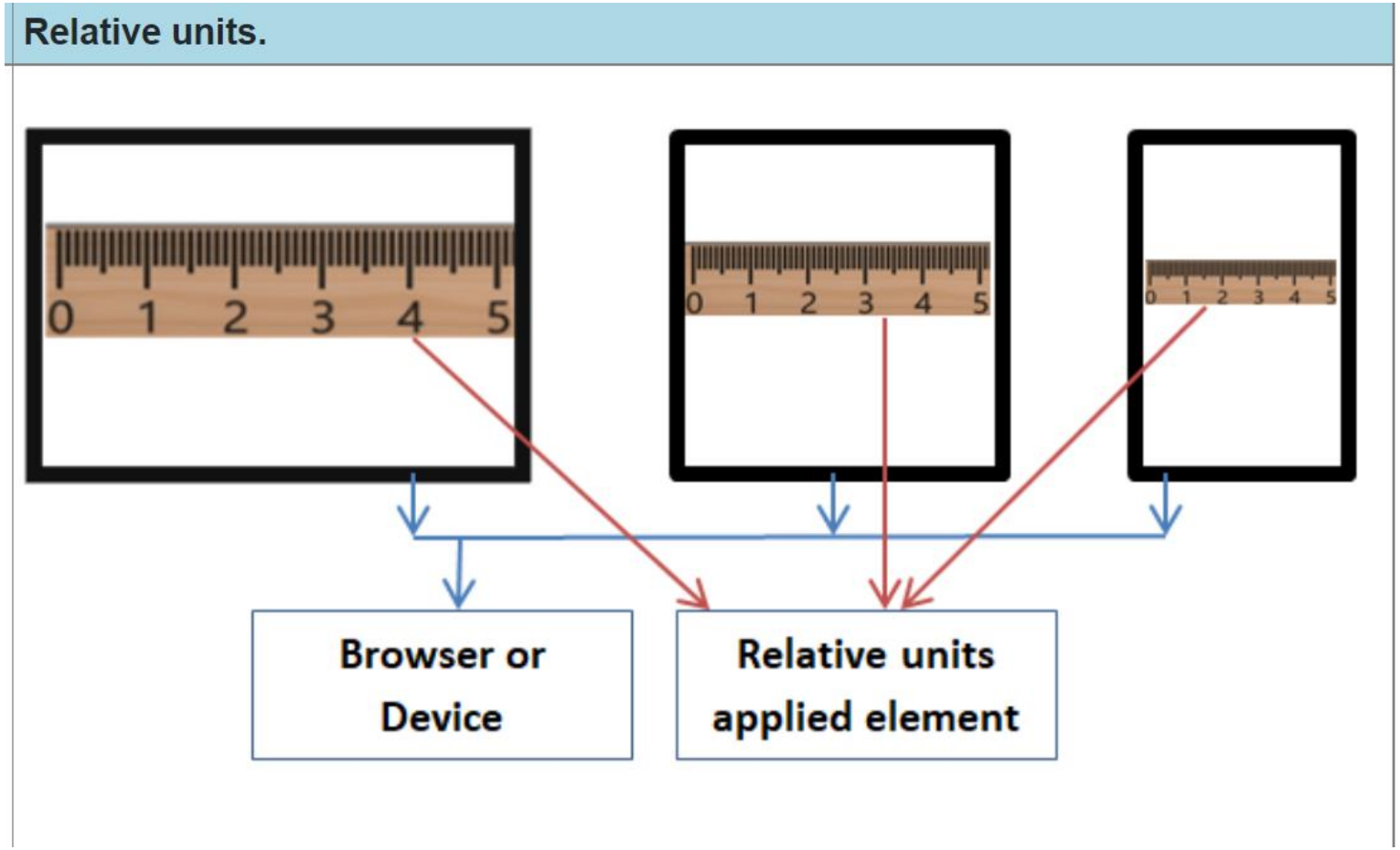
in	inches - 1in is equal to 2.54cm.
cm	centimeters - 1cm is equal to 96px/2.54.
mm	millimeters - 1mm is equal to 1/10 of 1cm.
pt	points — the points used by CSS are equal to 1/72nd of 1in.
pc	picas — 1pc is equal to 12pt.
px	pixel units — 1px is equal to 0.75pt.

Relative Unit

Relative units change relative to the element’s current font-size or other settings.

It is good for use on-screen, if screen size varies so much depending on the device, then these relative length units are perfect because it changes with the different rendering mediums.

Relative Unit



⭐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)

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)

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)

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)

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

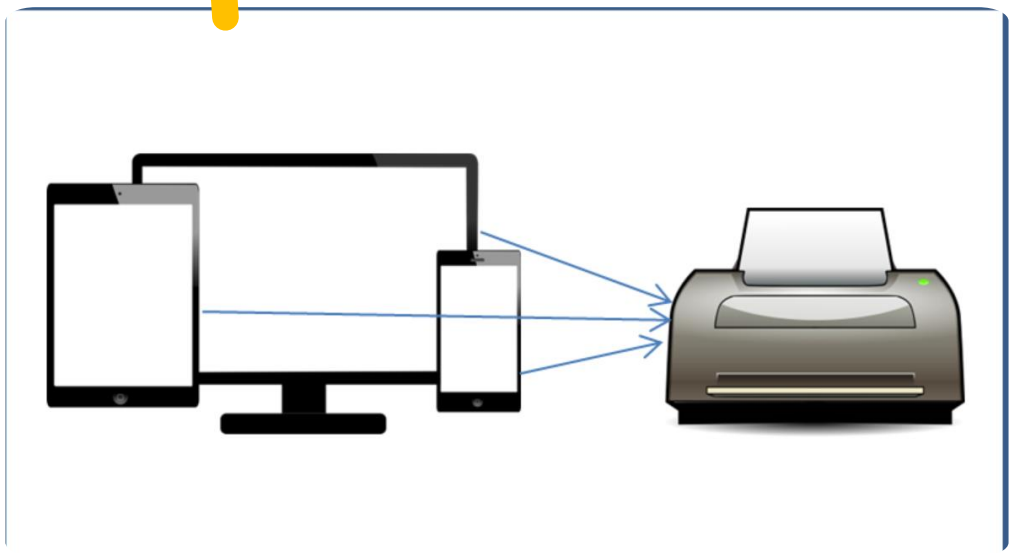
ch

`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.

Different devices

- Web pages can be viewed using different types of devices (cellphones, tablets, laptops, PCs).
- Each device height and width can be different.
- **It is better to use relative units in web pages (for screen use).**
- **For printing purpose, absolute size can be used.**



Thank you

netlink
digital solutions