



Create well-formatted HTML Tables
Understand when and when not to use tables
Explain the role of the form element
Utilize a variety of form controls
Define complex HTML forms
Add HTML5 Form Validations
Ensure accessibility of our forms

Unit Goals

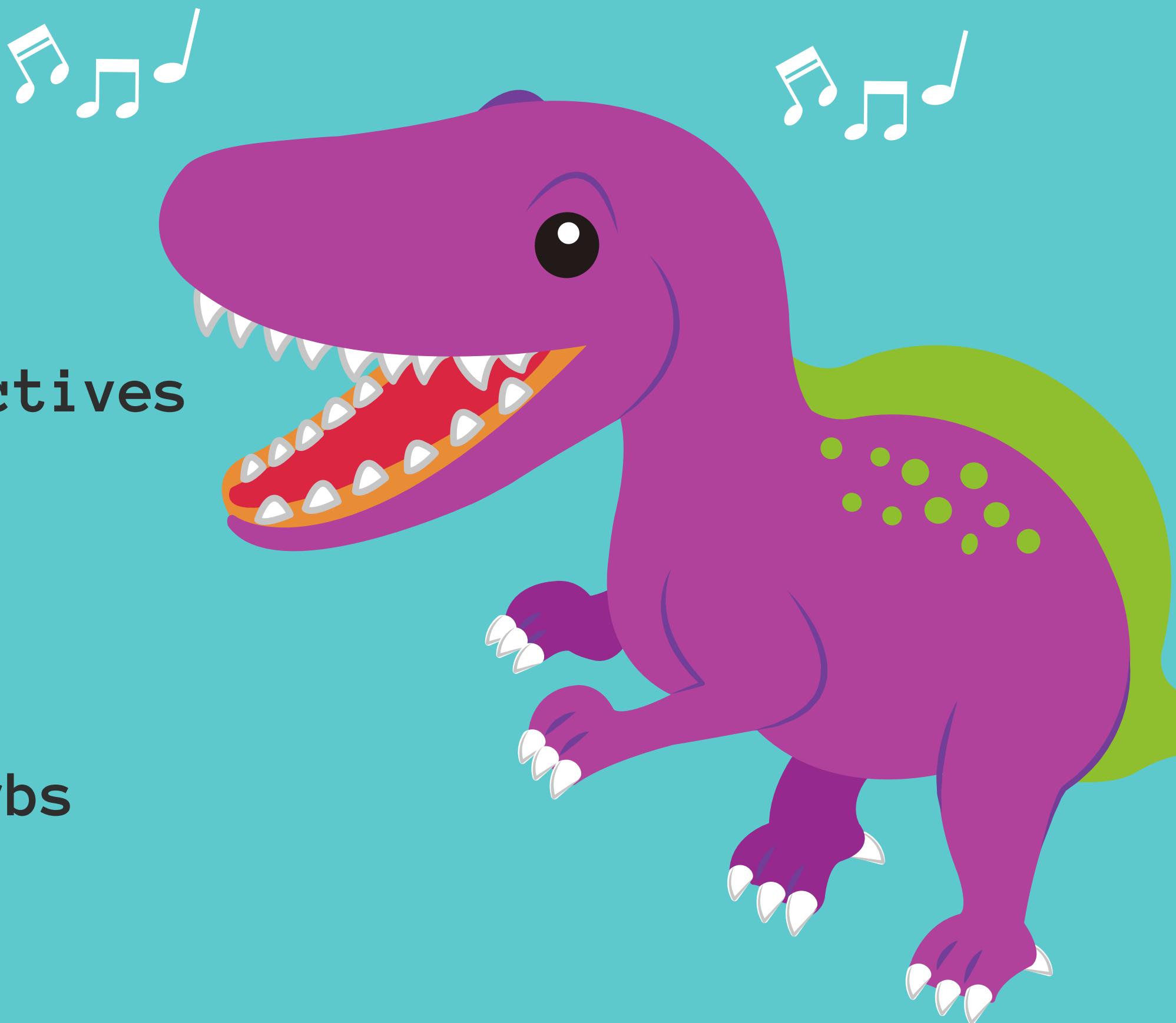
Things We'll Cover

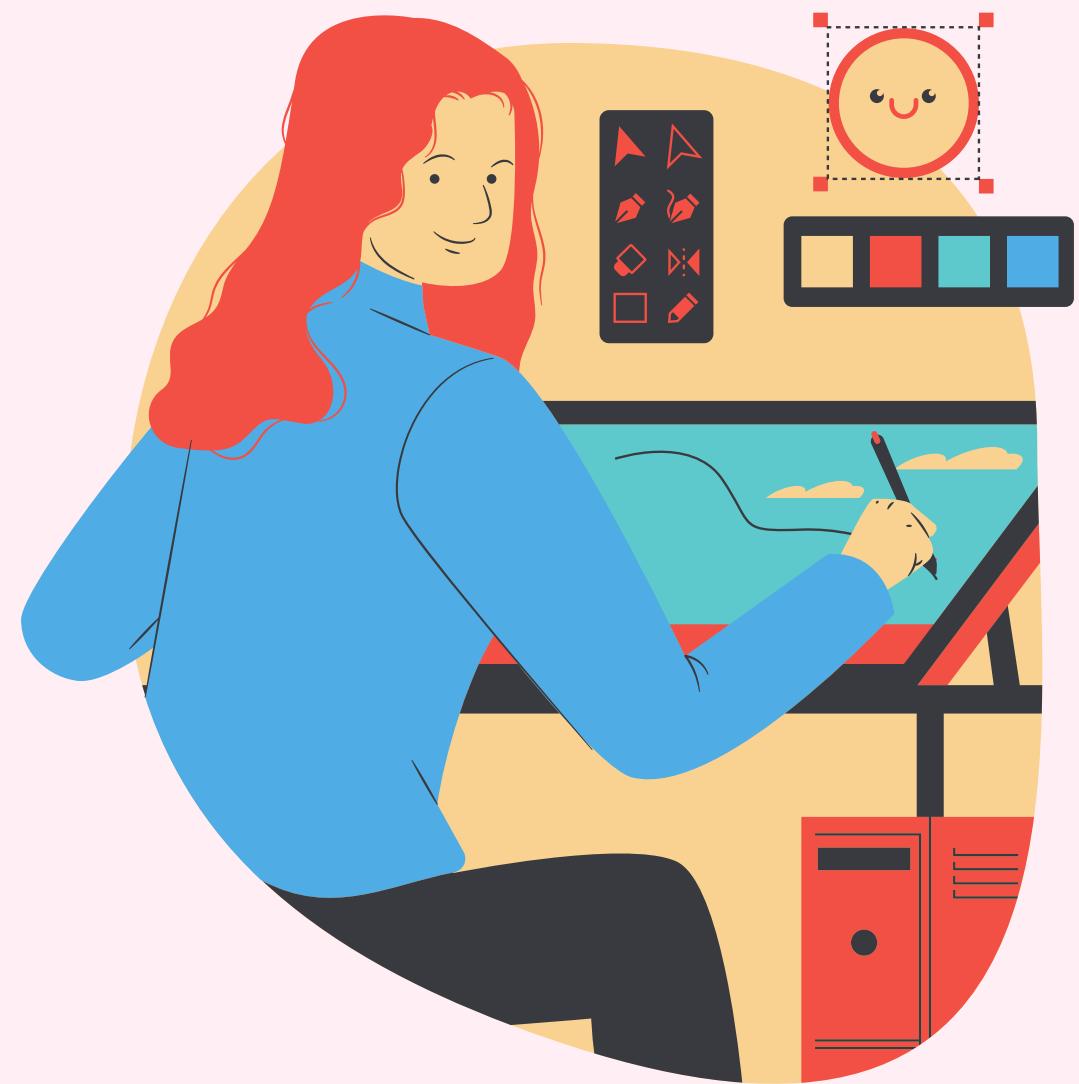


{ }

css

THE PURPLE CSS – adjectives
DINO HTML – nouns
DANCED JS – verbs





CSS

WHAT IS IT?

CSS is a language for describing how documents are presented visually - how they are arranged and styled.

WHAT DOES IT STAND FOR?

CSS stands for Cascading Style Sheets. We'll cover the "cascading" part in a bit; don't worry about it for now!

THERE'S A LOT!

CSS is very easy to get the hang of, but it can be intimidating because of how many properties we can manipulate.

CSS RULES

(almost) everything you do in CSS follows this basic pattern:

```
selector {  
    property: value;  
}
```

CSS RULES

Make all `<h1>` elements purple

```
h1 {  
    color: purple;  
}
```

CSS RULES

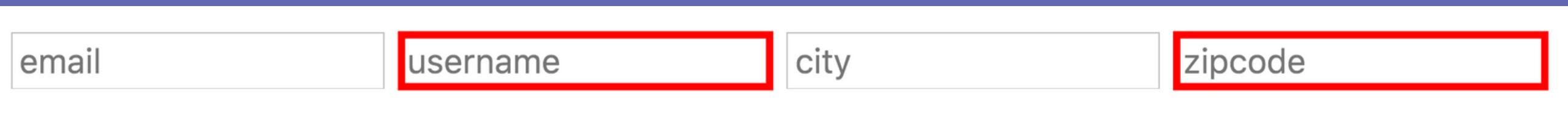
Make all image elements
100 pixels wide & 200 pixels tall

```
img {  
    width: 100px;  
    height: 200px;  
}
```

FANCIER!

Select every other text input
and give it a red border:

```
input[type="text"]:nth-of-type(2n){  
    border:2px solid red;  
}
```



email username city zipcode

So Many CSS Border Properties

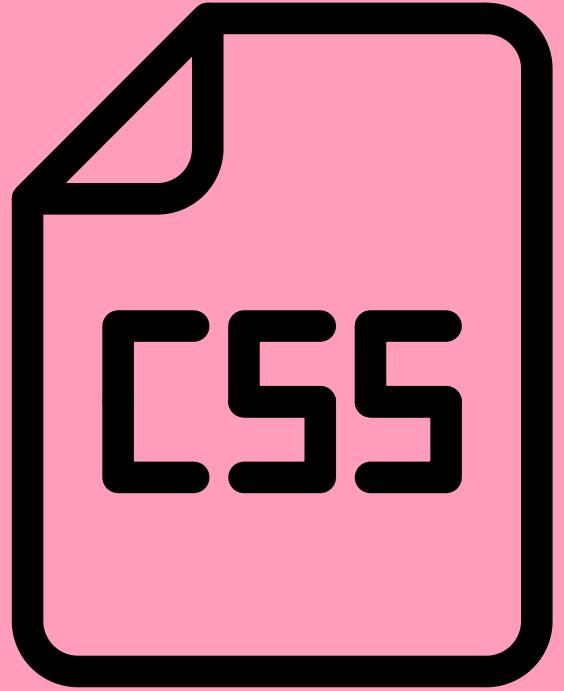


border border-blockborder-block-color border-block-end border-block-end-color border-block-end-style border-block-end-width border-block-start border-block-start-color border-block-start-style border-block-start-width border-block-style border-block-width border-bottom border-bottom-color border-bottom-left-radius border-bottom-right-radius border-bottom-style border-bottom width border-collapse border-color border-end-end-radiusborder-end-start-radiusborder-imageborder-image-outsetborder-image-repeatborder-image-sliceborder-image-sourceborder-image-widthborder-inlineborder-inline-colorborder-inline-endborder-inline-end-colorborder-inline-end-styleborder-inline-end-widthborder-inline-startborder-inline-start-colorborder-inline-start-styleborder-inline-start-widthborder-inline-styleborder-inline-widthborder-leftborder-left-colorborder-left-styleborder-left-widthborder-radiusborder-rightborder-right-colorborder-right-styleborder-right-widthborder-spacingborder-start-end-radiusborder-start-start-radiusborder-styleborder-topborder-top-colorborder-top-left-radiusborder-top-right-radiusborder-top-styleborder-top-widthborder-width









Including Styles

INLINE STYLES

You can write your styles directly inline on each element, but this is **NOT A GOOD IDEA** most of the time.

THE <STYLE> ELEMENT

You can write your styles inside of a `<style>` element. This is easy, but it makes it impossible to share styles between documents. **Not recommended either.**

EXTERNAL STYLESHEET

Write your styles in a `.css` file, and then include the using a `<link>` in the head of your html document. **Recommended!**

<link>



```
<head>
  <title>Forms Demo</title>
  <link rel="stylesheet" href="my_styles.css">
</head>
```



css colors



**NAMED
COLORS**

Hotpink

Mediumorchid

Firebrick

Darkkhaki

Gold

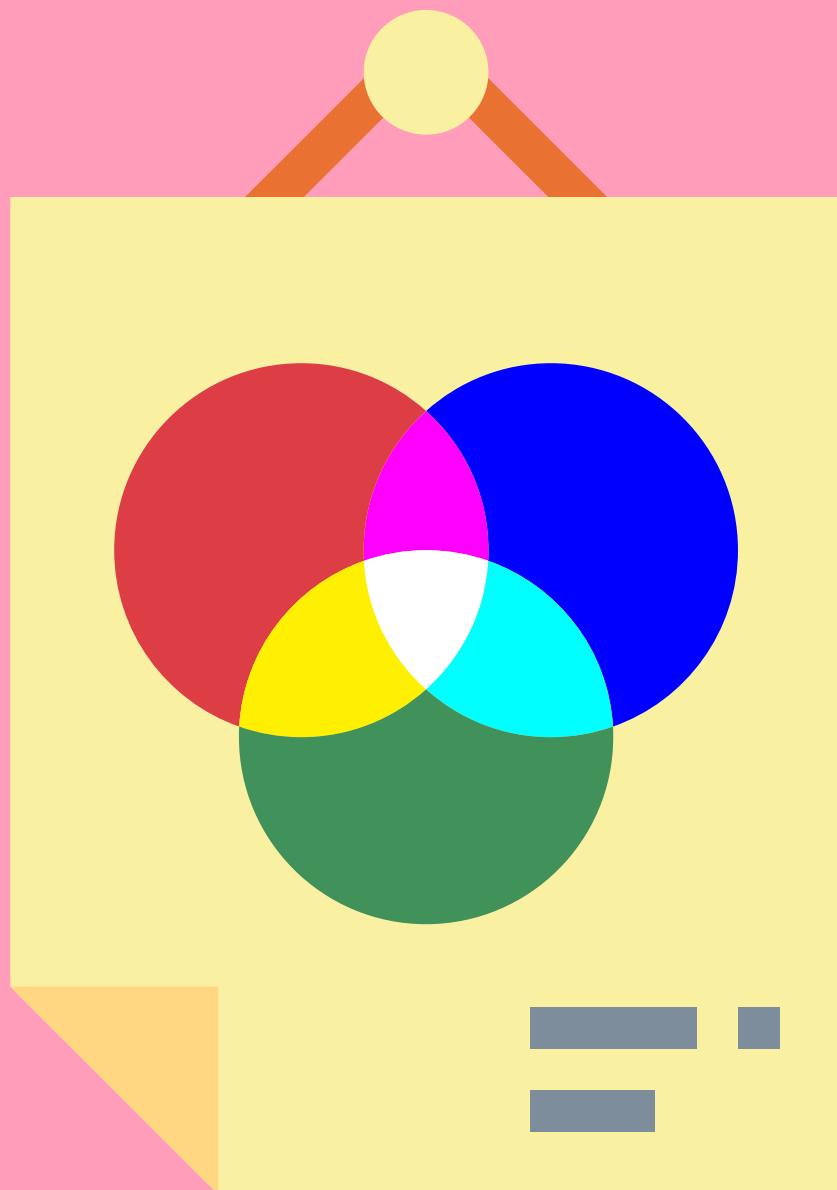
MediumAquamarine

Lightskyblue

Tomato

A typical computer can display
~16,000,000
different colors





RGB

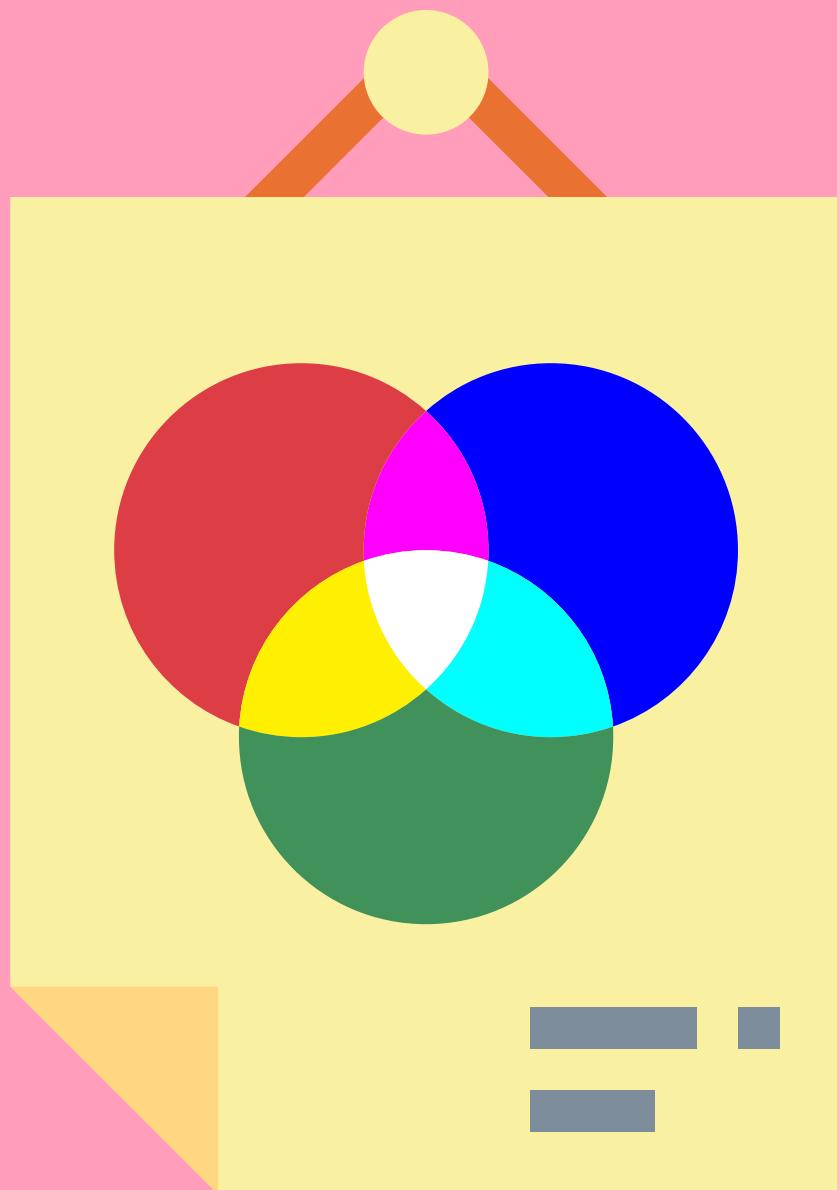
- Red, Green, Blue Channels
- Each channel ranges from 0-255

rgb(255,0,0)

rgb(0,0,255)

rgb(173, 20, 219)

rgb(0,0,0)



Hex

- Still red, green, and blue channels
- Each ranges from 0-255 BUT represented with hexadecimal

Decimal

0,1,2,3,4,

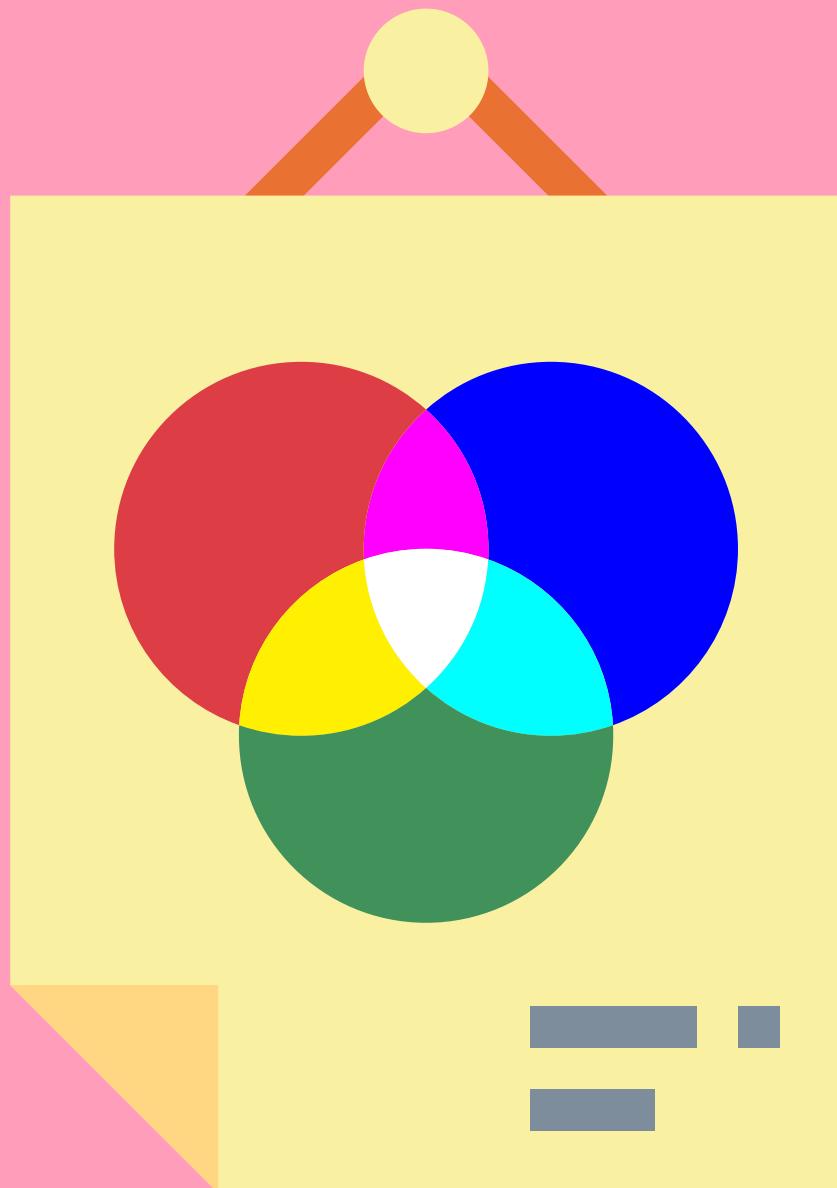
5,6,7,8,9

Hexadecimal

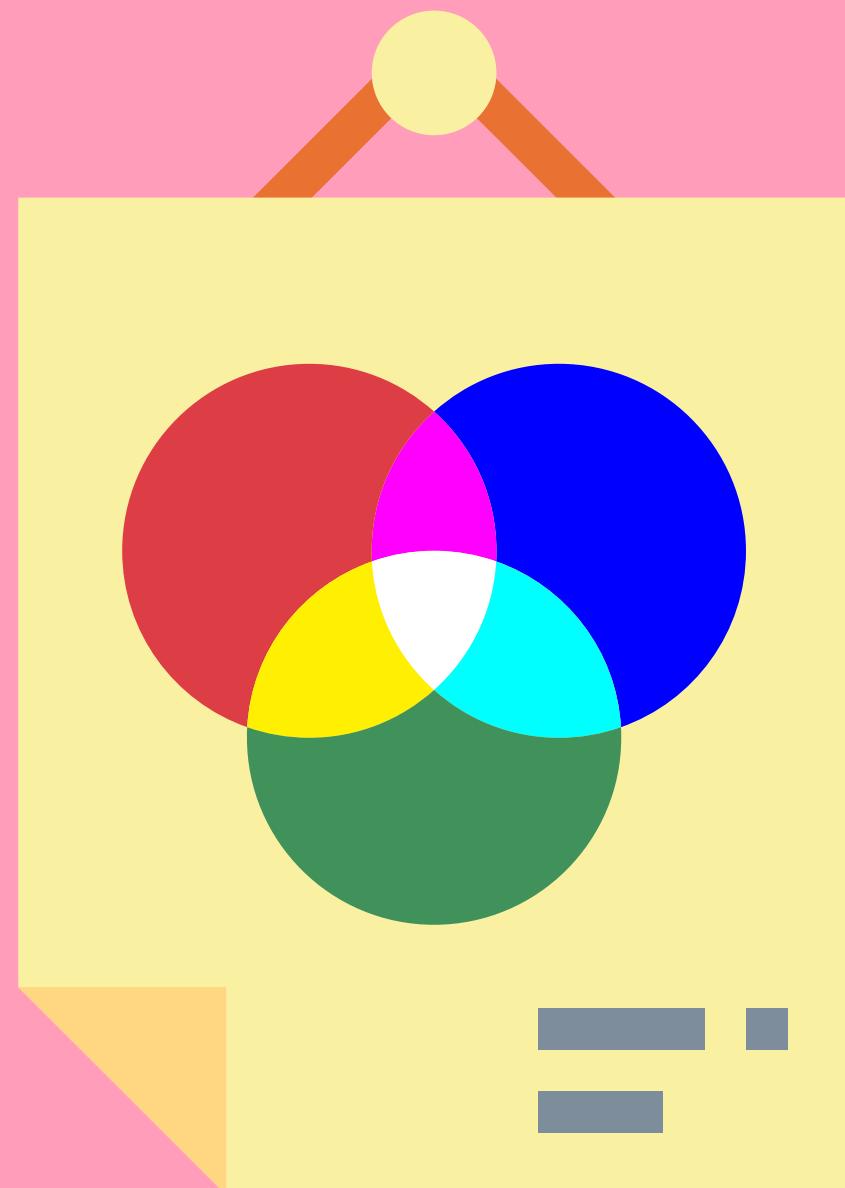
0, 1, 2, 3, 4,

5, 6, 7, 8, 9,

A, B, C, D, E, F



#fffff00
red green blue



#0f5679

red

green

blue



CSS Text Properties

- **text-align**
- **font-weight**
- **text-decoration**
- **line-height**
- **letter-spacing**



**FONT
SIZE**





Relative

- EM
- REM
- VH
- VW
- %
- AND MORE!

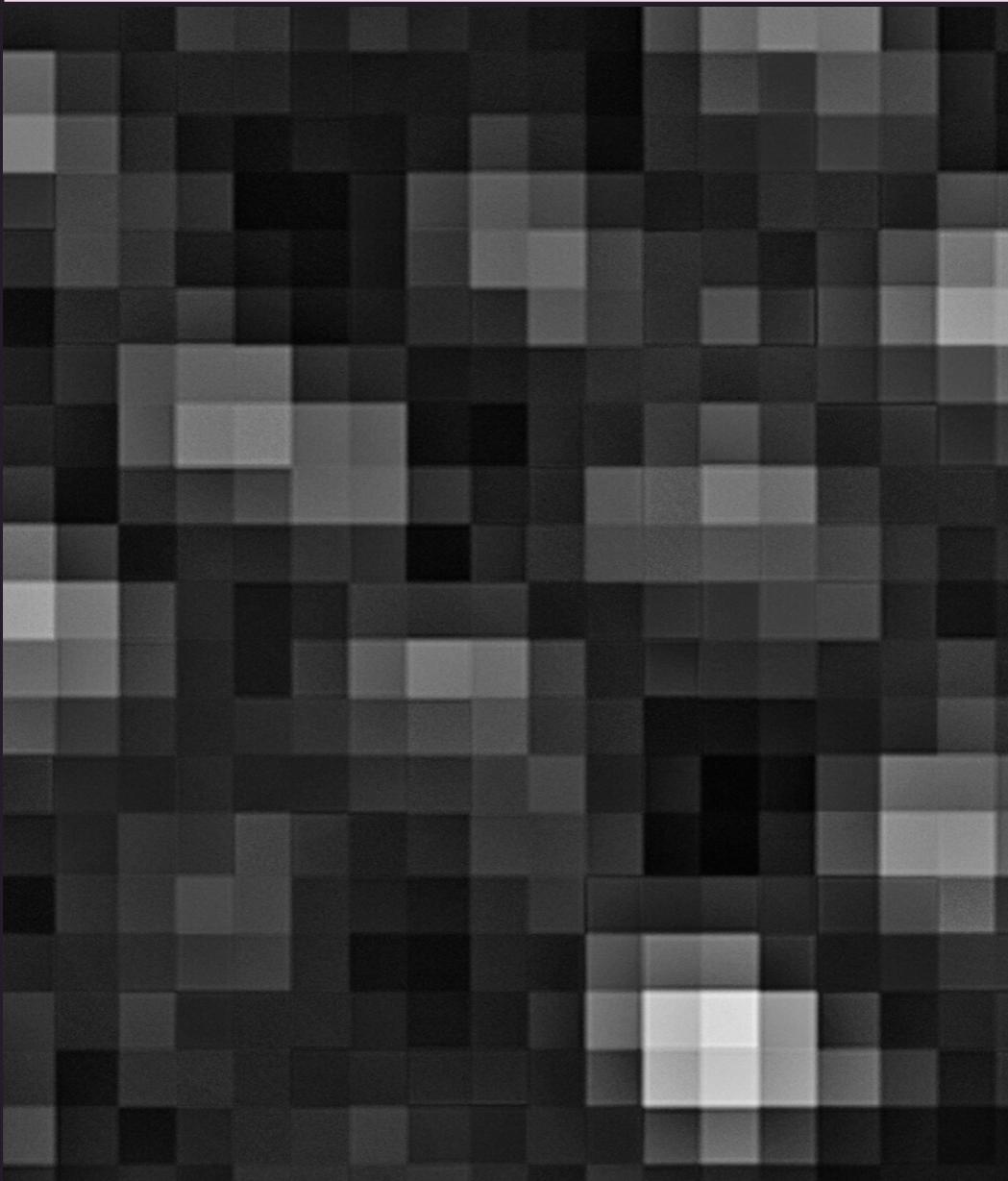
Absolute

- PX
- PT
- CM
- IN
- MM

FONT FAMILY



Absolute Units



**PX - BY FAR THE MOST
COMMONLY USED ABSOLUTE UNIT**

1px does not necessarily equal the width
of exactly one pixel!

Not recommended for responsive websites.

em



EM'S ARE RELATIVE UNITS

With font-size, 1em equals the font-size of the parent. 2em's is twice the font-size of the parent, etc.

With other properties, 1em is equal to the computed font-size of the element itself.

rem



ROOT EMS

Relative to the **root html element's** font-size. Often easier to work with.

If the root font-size is 20px, 1 rem is always 20px, 2rem is always 40px, etc.