1] Note: Do not add space b/w the property value and the unit

2] Colors:

~~RGB~~ ~~HEX~~ ~~HSL~~ ~~RGBA~~ ~~HSLA~~ } values

RGB — Color specified using formula → rgb (red, green, blue)

- Each parameter defines intensity of color b/w 0 to 255

eg:

```
<!DOCTYPE html>
<html>
<head>
<style>
/* rgb */
.eg1 {
  color: rgb (red, green, blue); /* By default */
/* wrong */ Syntax
.eg2 {
  color: rgb (0, 245, 7); /* value change for green */
.eg3 {
  color: rgb (245, 0, 0); /* Value change for red */
.eg4 {
  color: rgb (0, 150, 7); /* Value change for green with
                             different values. */
.eg5 {
  color: rgb (0, 0, 255); /* Value change for blue */
.eg6 {
  color: rgb (red, purple, silver); /* Value change ; Wrong */
                                    Everything Syntax
}
</style>
</head>
<body>
<div class = "eg_1">
<p> This is the world of colors eg1 </p>
</div>
<div class = "eg_2">
```

```
<p> This is the world of colors eg2 </p>
</div>
<div class="eg3">
  <p> This is the world of colors eg3 </p>
</div>
<div class="eg4">
  <p> This is the world of colors eg4 </p>
</div>
<div class="eg5">
  <p> This is the world of colors eg5 </p>
</div>
<div class="eg6">
  <p> This is the world of colors eg6 </p>
</div>
</body>
</html>
```

RGBA — Extension of RGB color values with an Alpha channel

— Alpha – Specifies the opacity for a color

— Syntax: rgba (red, green, blue, alpha)

— Alpha — It is b/w 0.0 (Fully Transparent) to 1.0 (Not Transparent at all)

Eg1:

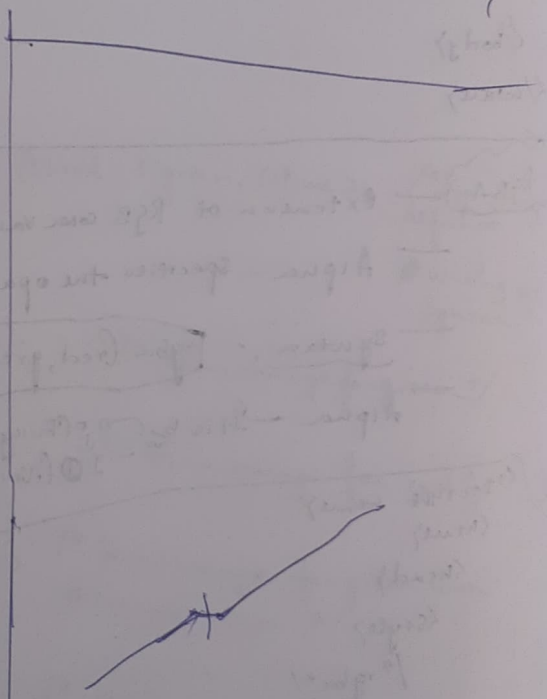
```
<!DOCTYPE html>
<html>
<head>
<style>
/* rgba */
.eg1 {
  color: rgba (red, green, blue, 0.0) /* By default; Wrong Syntax;
                                         alpha = 0.0 */
}
.eg2 {
  color: rgba (red, green, blue, 1.0); /* By default; Wrong Syntax;
                                          alpha = 1.0 */
}
.eg3 {
  color: rgba (0, 245, 7, 0.0); /* Value change for green & alpha = 0.0 */
}
.eg4 {
  color: rgba (0, 245, 7, 1.0); /* Value change for green & alpha = 1.0 */
}
```

1 ×
2 ×
3 ✓
4 ✓
5 ×
6 ×
7 ×

```
.eg5 {
    color: rgba (□red, □purple, □silver, 0.0);
}
                    /* Value change everything;
                       Wrong syntax
                       alpha = 0.0 */

.eg6 {
    color: rgba (□red, □purple, □silver, 1.0);
}
                    /* Value change everything;
                       Wrong Syntax;
                       alpha = 1.0 */

.eg7 {
    color: rgba (□red, □green, □blue, alpha)   /* By default;
                                                  Wrong Syntax;
                                                  alpha = alpha */
}
</style>
</head>
<body>
    <div class = "eg1" >
        <p> Tmi </p>
    </div>
    <div class = "eg2" >
        <p> Tmi </p>
    </div>
    <div class = "eg3" >
        <p> Tmi </p>
    </div>
    <div class = "eg4" >
        <p> Tmi </p>
    </div>
    <div class = "eg5" >
        <p> Tmi </p>
    </div>
    <div class = "eg6" >
        <p> Tmi </p>
    </div>
    <div class = "eg7" >
        <p> Tmi </p>
    </div>
</body>
</html>
```



52

Here → Specified with → #RRggBB
- RR - Red
- gg - green
- BB - Blue

Hexadecimal integer → Specify component of color.

Where
- rr (red)
- gg (green)
- bb (blue)

→ hexadecimal values → b/w 00 - ff = 0 - 255

eg:

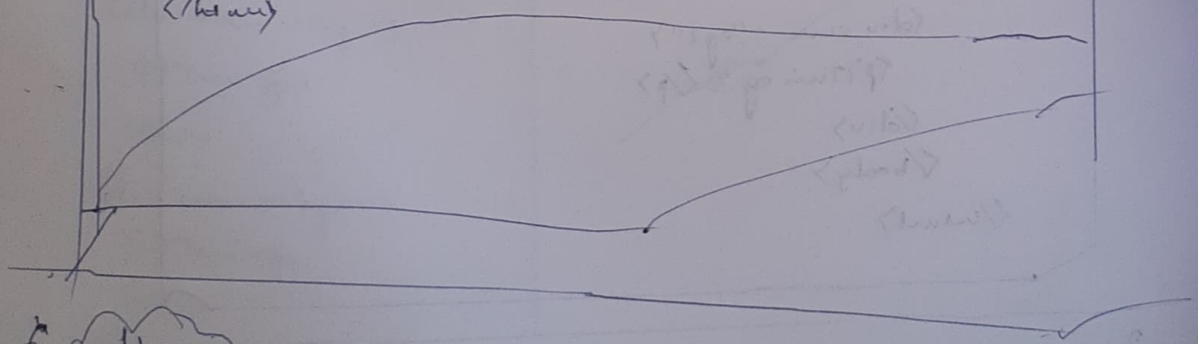
```
<!DOCTYPE html>
<html>
  <head>
    <style>
      .eg1{
        color: hex(#926565); /*wrong syntax*/
      }
      .eg2{
        color: #926565;
      }
      .eg3{
        color: #9265ff;
      }
    </style>
  </head>
  <body>
    <div class="eg1">
      <p> This eg1 </p>
    </div>
    <div class="eg2">
      <p> This eg2 </p>
    </div>
    <div class="eg3">
      <p> This eg3 </p>
    </div>
  </body>
</html>
```

3 digit Hex value

- Shorthand for some 6 digit hex code
- It has form → #rgb ; values b/w 0 & f.

88 Eg:-

```
<!DOCTYPE html>
<html>
<head>
<style>
.eg1 {
  color: hex(#ffcc99);   /*wrong syntax*/
}
.eg2 {
  color: #fc9;
}
.eg3 {
  color: hex(#fc9);   /*wrong syntax*/
}
</style>
</head>
<body>
<div class="eg1">
  <p> This eg1 </p>
</div>
<div class="eg2">
  <p> This eg2 </p>
</div>
<div class="eg3">
  <p> This eg 2 </p>
</div>
</body>
</html>
```



→ HSL → In the form : hsl (H, S, L)

- H → Hue
- S → Saturation
- L → Lightness

Sl no	About	Detail	What it is
1	Hue	degree on Color wheel From 0 to 360	0 → Red 120 → green 240 → Blue
2	Saturation	percentage value	0% → Shade of grey 100% → Full color
3	Lightness	percentage	0% → Black 50% → neither Light or Dark 100% → white

69

```
<!DOCTYPE html>
<html>
<head>
<style>
.eg1{
color: hsl(hue, saturation, lightness); /* default value; Error Syntax */
}
.eg2{
color: hsl(0,0%,0%) /* hue=0 saturation=0% lightness=0%; */
}
.eg3{
color: hsl(120,0%,0%) /* hue=120 saturation=0% lightness=0%; */
}
.eg4{
color: hsl(240,0%,0%) /* hue=240 saturation=0% lightness =0%. */
}
.eg5{
color: hsl(360,0%,0%); /* hue=360 saturation=0% lightness =0%.*/
}
.eg6{
color: hsl(120, 100%,80%); /* hue=120 Saturation =100% lightness =20%.
Try to change the lightness value between 0 to 100%. */
</style>
</head>
<body>
<p class="eg1">This is p1</p> <br>
<p class="eg2">This is p2</p> <br>
<p class="eg3">This is p3</p> <br>
<p class="eg4">This is p4</p> <br>
<p class="eg5">This is p5</p> <br>
<p class="eg6">This is p6</p>
</body>
</html>
```

(9)

↳ HSLA ← Extension of HSL color values with alpha channel.

← form: hsla(hue, saturation, lightness, alpha)

↳ Similar to rgba.

Eg:

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      .eg1{                          ← Wrong
        color: hsla(hue, saturation, lightness, alpha);
      }
      .eg2{
        color: hsla(0, 0%, 0%, 0.0);   /* Invisible */
      }
      .eg3{
        color: hsla(0, 0%, 0%, 0.5);
      }
      .eg4{
        color: hsla(120, 100%, 50%, 0.5);
      }
    </style>
  </head>
  <body>
    <p class="eg1">This is p1</p> <br>   <!-- default hsla -->
    <p class="eg2">This is p2</p> <br>
    <p class="eg3">This is p3</p> <br>
    <p class="eg4">This is p4</p>
  </body>
</html>
```