

Front End Fundamentals II

Lea Ann Bradford for Girl Develop It



Fonts and More CSS

Google Fonts

- The world of HTML has progressed beyond Times New Roman and Arial
- Google has hundreds of free, open-source fonts that have been optimized for the web, and ready for us to use!
- The service runs on Google's servers which are fast, reliable and tested. Google provides this service free of charge.
- <https://fonts.google.com/>

Let's Develop It

Let's add Google fonts to our page

- <https://fonts.google.com/>
- Select a font and style that you like
- Click 'Select this style' for the styles you like
- Click 'View your selected families' in the top right of the page
- Choose <link> and copy the code. Add to your <head> tag and your CSS file.

```
body {  
    font-family: 'Lato', sans-serif;  
}  
h1, h2, h3, h4, h5, h6 {  
    font-family: 'Bree Serif', serif;  
}
```

RGBa Colors and Opacity

- In CSS, not only we can choose colors with
- hexadecimal #0000FF or rgb(0, 0, 255) value, but we can also control the opacity of a color
 - rgba - controls the opacity of a color - rgba(0, 0, 255, .5)
 - We can also use CSS to control the opacity of an element

RGBa Colors and Opacity

`rgba(255, 0, 0, 1);`

`rgba(255, 0, 0, .8);`

`rgba(255, 0, 0, .6);`

`rgba(255, 0, 0, .4);`

`rgba(255, 0, 0, .2);`

`rgba(255, 0, 0, .1);`

`opacity: 1;`

`opacity: .8;`

`opacity: .6;`

`opacity: .4;`

`opacity: .2;`

`opacity: .1;`

RGBa Colors and Opacity

- color property using (red, green, blue, opacity)
- opacity is a decimal value from 0 to 1
 - 0 - not visible
 - 1 - fully visible

```
.example {  
    background: rgba(250, 250, 250, 0.8);  
}
```

RGBa Colors and Opacity

- controls the opacity of an element (and it's decendents)
- opacity is a decimal value from 0 to 1
 - 0 - not visible
 - 1 - fully visible

```
.example {  
    opacity: 0.8;  
}
```


Let's Develop It

Let's add some opacity to our page

- Change some colors to have alpha transparency or opacity.
- Use <http://hex2rgba.devoth.com/> to convert HEX to RGBA.
- Try making the navbar 80% opaque.

Shadow Effects

- In CSS, you can add shadow effects to text and to elements
 - text-shadow applies the effect to text
 - box-shadow applies the effect to elements
- Both box-shadow and text shadow have the same values

The text "Text Shadow" is displayed in a bold, white, sans-serif font. It has a prominent blue and purple gradient shadow that is offset to the right and bottom, giving it a 3D appearance as if it is floating above the background.

Text Shadow

Word of the Day

be·nev·o·lent

adjective

well meaning and kindly.

"a benevolent smile"

[LEARN MORE](#)

Shadow Effects

- The simplest shadow only specifies the horizontal and vertical shadow

```
h2 {  
    text-shadow: 2px 2px;  
}  
  
.card {  
    box-shadow: 5px 5px;  
}
```

Text Shadow

This element has a box-shadow applied

Shadow Effects

- You can also add color to your text effect

```
h2 {  
    text-shadow: 2px 2px red;  
}  
  
.card {  
    box-shadow: 5px 5px #8a80c3;  
}
```

Text Shadow

This element has a box-shadow applied

Shadow Effects

- You can add a blur to your shadow effect

```
h2 {  
    text-shadow: 2px 2px 3px red;  
}  
  
.card {  
    box-shadow: 5px 5px 5px #8a80c3;  
}
```

Text Shadow

This element has a box-shadow applied

Shadow Effects

- You can also add more than one shadow to the same text element
- Just separate each of them with commas

```
h2 {  
    text-shadow: 2px 2px 3px red, -2px -2px 3px green, 0 0 15px black;  
}
```



Text Shadow

Let's Develop It

Let's add some box shadow to our page

- Remember: Be subtle!
- Add a text shadow
- Add a box shadow to your navigation

CSS Gradients

- CSS gradients let you display smooth transitions between two or more specified colors.
- There are two different types of gradients:
 - Linear
 - Radial



CSS Gradients

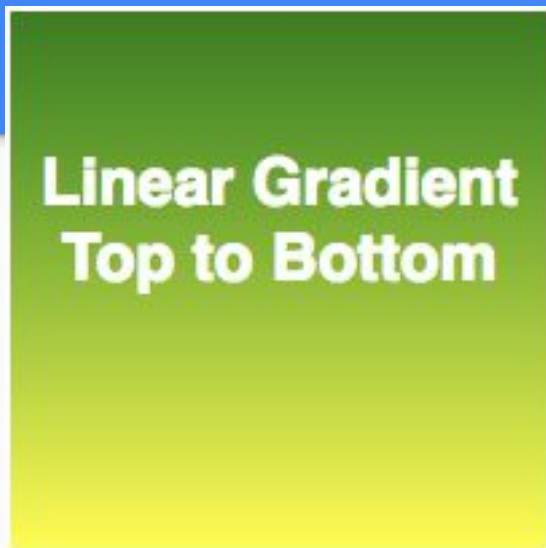
- To create a linear gradient, you must create at least 2 color stops
- This is what the syntax looks like

```
.gradient {  
    background: linear-gradient(direction, color-stop1, color-stop2, ...);  
}
```

CSS Gradients

- Top to bottom is the default gradient

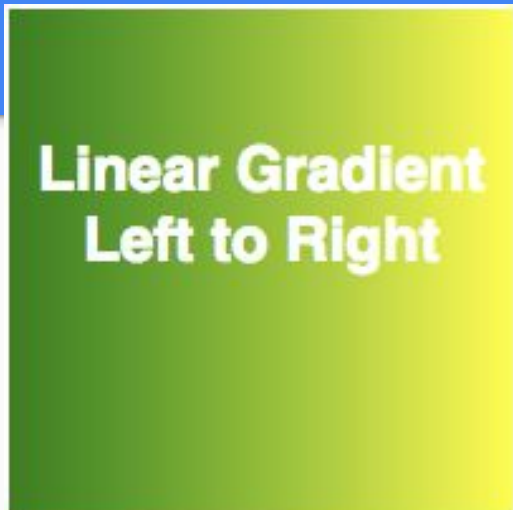
```
.gradient {  
    background: linear-gradient(green, yellow);  
}
```



CSS Gradients

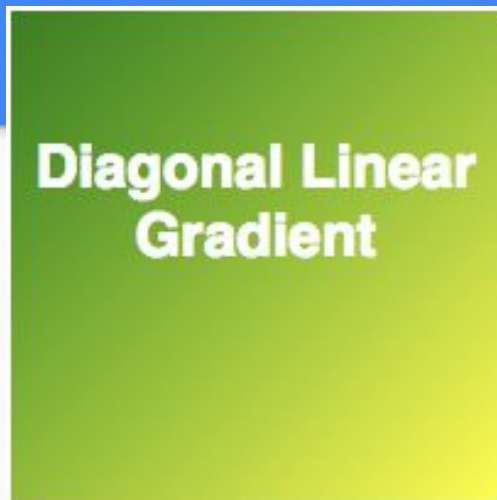
- This is an example of left-to-right gradient

```
.gradient {  
    linear-gradient(to right, green, yellow);  
}
```



CSS Gradients

- This is an example of a diagonal gradient



```
.gradient {  
  linear-gradient(to bottom right, green, yellow);  
}
```

CSS Gradients

- You can use multiple color stops on a gradient

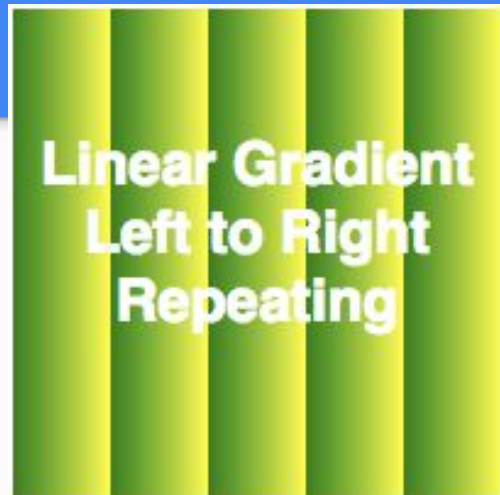


```
.gradient {  
    linear-gradient(to right, blue, green, yellow);  
}
```

CSS Gradients

- You can create a repeating gradient by using `repeating-linear-gradient`

```
.gradient {  
  background: repeating-linear-gradient(to right, green, yellow 20%);  
}
```



CSS Gradients

- Radial gradients are defined by their center
- You must have at least two color stops
- This is what the syntax looks like

```
.gradient {  
    background: radial-gradient(shape size at position, start-color, ..., last-color);  
}
```

CSS Gradients

- You can create a simple radial gradient like this



```
.gradient {  
    background: radial-gradient(yellow, green) ;  
}
```


CSS Gradients

- You can create a repeating radial gradient



```
.gradient {  
    background: repeating-radial-gradient(yellow, green 25%);  
}
```

CSS Gradients

- CSS gradients are a pain to do from scratch
- That's why people have made things like the Ultimate CSS Gradient Generator to make our lives easier!

<http://www.colorzilla.com/gradient-editor/>

Let's Develop It

Let's add some gradients to our page

- Remember: Be subtle!
- Add a linear gradient
- Add a radial gradient
- **<http://www.colorzilla.com/gradient-editor/>**

CSS Transitions

- CSS transitions allows you to change property values smoothly (from one value to another), over a given duration.
- Create some awesome effects without any JavaScript

CSS Transitions

- To create a transition, you must specify two things
 - 1. The property you want to apply the effect to
 - 2. The duration
- Here we are targeting the width of the element
- The duration is 2 sec

```
<div id="transition"><h4>Transition</h4></div>
#transition {
  width: 100px;
  height: 100px;
  background: red;
  -webkit-transition: width 2s linear 1s;
  transition: width 2s;
}
#transition:hover {
  width: 300px;
}
```

Diagram illustrating CSS transitions with annotations:

- TARGET PROPERTY:** Points to `width: 100px;` in the initial state and `width: 300px;` in the hover state.
- DURATION:** Points to `2s` in the `transition: width 2s;` rule.

CSS Transitions

Transition Triggers:

- Hover
- Mouse click
- Focus state
- Active state
- Changes to the element

CSS Transitions

Transition Properties:

- transition-property
- transition-duration
- transition-delay
- transition-timing-function

CSS Transitions

transition-property:

```
.transition {  
    transition-property: background-color;  
}
```


CSS Transitions

transition-property:

Some examples in action: <http://leaverou.github.io/animatable/>

Full list of properties:

<https://www.w3.org/TR/css-transitions-1/#animatable-css>

CSS Transitions

transition-duration

the number of seconds
or milliseconds a
transition animation
should take to complete

```
.transition {  
    transition-duration: 2s;  
}
```

CSS Transitions

transition-delay

delay transitions from the moment a trigger happens

```
.transition {  
    transition-delay: 1s;  
}
```

CSS Transitions

Transition-timing-function

specifies the speed curve of the transition effect

- **ease** - specifies a transition effect with a slow start, then fast, then end slowly (this is default)
- **linear** - specifies a transition effect with the same speed from start to end
- **ease-in** - specifies a transition effect with a slow start
- **ease-out** - specifies a transition effect with a slow end
- **ease-in-out** - specifies a transition effect with a slow start and end
- **cubic-bezier(n,n,n,n)** - lets you define your own values in a cubic-bezier function

CSS Transitions

Too much to remember?

Someone has created a tool for us to use:

- **<http://css3generator.com/>**

Let's Develop It

Let's add some transitions to our page

- Remember: Be subtle!
- Pick a transition property and apply it to an element.
- Hint: The transition will only work if it has a pseudo class, like :hover
- Need an idea?
 - Try adding transition to the hover background color on your navigation
 - Try adding a border radius transition on hover

CSS Transform

- CSS transforms allow you to translate, rotate, scale, and skew elements
- A transformation is an effect that lets an element change shape, size and position

CSS Transform

Scale

scales the element

This example increases the <div> element to be two times of its original width, and three times of its original height:

```
div .scale {  
    -ms-transform: scale(2, 3); /* IE 9 */  
    -webkit-transform: scale(2, 3); /* Safari */  
    transform: scale(2, 3);  
}
```


CSS Transform

Rotate

rotates the element

This example rotates the <div> element clockwise with 20 degrees:

```
div .rotate {  
    -ms-transform: rotate(20deg); /* IE 9 */  
    -webkit-transform: rotate(20deg); /* Safari */  
    transform: rotate(20deg);  
}
```

CSS Transform

Translate

move element using x and y coordinates

This example moves the <div> element 50 pixels to the right, and 100 pixels down from its current position:

```
div .translate {  
    -ms-transform: translate(50px, 100px); /* IE 9 */  
    -webkit-transform: translate(50px, 100px); /* Safari */  
    transform: translate(50px, 100px);  
}
```

Let's Develop It

Let's try adding transform to some elements

- Pick a transform property and apply it to an element.
- Hint: The transforms, like transitions, will only work if the element has a pseudo class, like :hover
- Need an idea? Try adding scale to the hover on images
- Extra Credit: Use it with transition to make the transform smoother.

Next Steps

Code Snippets:

- We have created a folder with code snippets with examples of the items we covered today
- There are basic examples as well as some fun ideas for you to try in your projects

Sharing Time!

Questions?