FRONT-END FORMATIONS

Level 4 - CSS3 Styles

TABLE OF CONTENTS

TABLE OF CONTENTS

BORDER RADIUS

The border-radius property applies rounded corners to borders.

BORDER RADIUS

The base .box element we'll be working with:

```
.box {
  background: grey;
  height: 50px;
  width: 200px;
}
```

BORDER RADIUS

The base .box element we'll be working with:

```
.box {
  background: grey;
  height: 50px;
  width: 200px;
}
```

BORDER RADIUS

Example usage of the border-radius property:

```
.box {
  border-top-left-radius: 15px;
  border-top-right-radius: 15px;
  border-bottom-right-radius: 15px;
  border-bottom-left-radius: 15px;
}
```

BORDER RADIUS

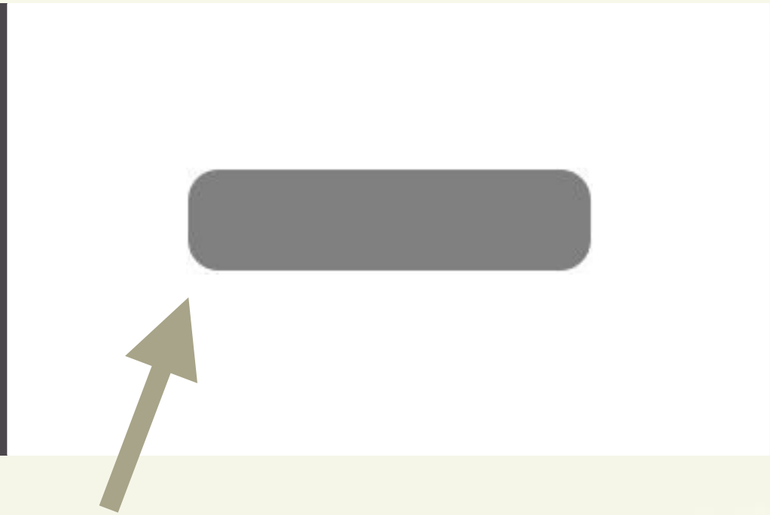
Example usage of the border-radius property:

```
.box {
  border-top-left-radius: 15px;
  border-top-right-radius: 15px;
  border-bottom-right-radius: 15px;
  border-bottom-left-radius: 15px;
}
```

BORDER RADIUS

Example usage of the border-radius property:

```
.box {
  border-top-left-radius: 15px;
  border-top-right-radius: 15px;
  border-bottom-right-radius: 15px;
  border-bottom-left-radius: 15px;
}
```



Applies a 15px rounded corner to our .box.

BORDER RADIUS

Example usage of the border-radius property:

```
.box {
  border-radius: 15px;
}
```

BORDER RADIUS

Example usage of the border-radius property:

```
.box {
  border-radius: 15px;
}
```

We can use the shorthand property to specify all sides at once.

BORDER RADIUS

Example usage of the border-radius property:

```
.box {
  border-radius: 4px 15px 12px 10px;
}
```

BORDER RADIUS

Example usage of the border-radius property:

```
.box {
  border-radius: 4px 15px 12px 10px;
}
```

we can specify each border-radius value individually, as well.

BORDER RADIUS

Example usage of the border-radius property:

```
.box {
  border-radius: 4px 15px 12px 10px;
}
```

BORDER RADIUS

Example usage of the border-radius property:

```
border-radius: <top left> <top right> <bottom right> <bottom left>
```

BORDER RADIUS

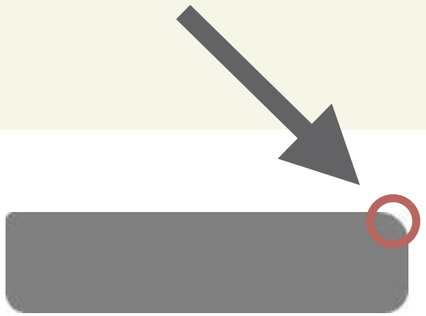
Example usage of the border-radius property:

```
border-radius: <top left> <top right> <bottom right> <bottom left>
```

BORDER RADIUS

Example usage of the border-radius property:

```
border-radius: <top left> <top right> <bottom right> <bottom left>
```



BORDER RADIUS

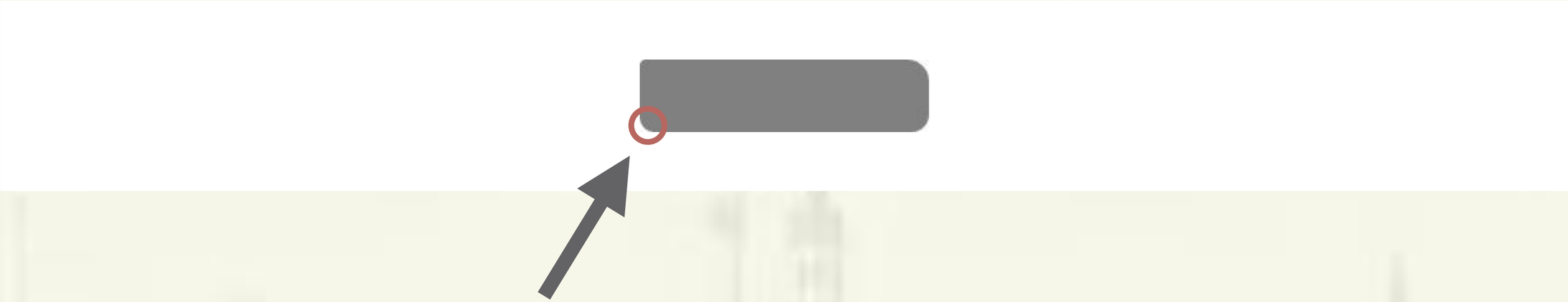
Example usage of the border-radius property:

```
border-radius: <top left> <top right> <bottom right> <bottom left>
```

BORDER RADIUS

Example usage of the border-radius property:

```
border-radius: <top left> <top right> <bottom right> <bottom left>
```



BORDER RADIUS

You can also specify the border-radius value in percentages.

BORDER RADIUS

Example usage of the border-radius property:

```
.box {
  border-radius: 50%;
}
```

BORDER RADIUS

Example usage of the border-radius property:

```
.box {
  border-radius: 50%;
}
```

TABLE OF CONTENTS

TABLE OF CONTENTS

BOX SHADOW

The box-shadow property specifies a shadow on an element.

BOX SHADOW

Example usage of the box-shadow property:

```
.box {
  box-shadow: 1px 2px 2px #000;
}
```

BOX SHADOW

Example usage of the box-shadow property:

```
.box {
  box-shadow: 1px 2px 2px #000;
}
```

BOX SHADOW

Example usage of the box-shadow property:

```
box-shadow: <inset> <offset-x> <offset-y> <blur-radius> <spread-radius> <color>
```

BOX SHADOW

Example usage of the box-shadow property:

```
box-shadow: <inset> <offset-x> <offset-y> <blur-radius> <spread-radius> <color>
```

?

If it is not specified (which is the default), a drop shadow is created, rather than an inset shadow.

BOX SHADOW

Example usage of the box-shadow property:

```
box-shadow: <inset> <offset-x> <offset-y> <blur-radius> <spread-radius> <color>
```

If it is not specified (which is the default), a drop shadow is created, rather than an inset shadow.

?

This icon denotes that the highlighted argument is optional.

BOX SHADOW

Example usage of the box-shadow property:

```
box-shadow: <inset> <offset-x> <offset-y> <blur-radius> <spread-radius> <color>
```

The shadow offset x value.

BOX SHADOW

Example usage of the box-shadow property:

```
box-shadow: <inset> <offset-x> <offset-y> <blur-radius> <spread-radius> <color>
```

The shadow offset y value.

BOX SHADOW

Example usage of the box-shadow property:

```
box-shadow: <inset> <offset-x> <offset-y> <blur-radius> <spread-radius> <color>
```

The blur-radius alters the blur amount of the shadow, causing it to become bigger *and* lighter (with a larger value).

BOX SHADOW

Example usage of the box-shadow property:

```
box-shadow: <inset> <offset-x> <offset-y> <blur-radius> <spread-radius> <color>
```

The spread-radius causes the shadow to expand *or* shrink.

BOX SHADOW

Example usage of the box-shadow property:

```
box-shadow: <inset> <offset-x> <offset-y> <blur-radius> <spread-radius> <color>
```

The color of the shadow.

BOX SHADOW

Example usage of the box-shadow property:

```
.box {
  box-shadow: 1px 2px 2px #000;
}
```

BOX SHADOW

Example usage of the box-shadow property:

```
.box {
  box-shadow: 1px 2px 2px #000;
}
```

No inset value is specified, so this is a drop shadow.

BOX SHADOW

Example usage of the box-shadow property:

```
.box {
  box-shadow: 1px 2px 2px #000;
}
```

A 1px offset-x value.

BOX SHADOW

Example usage of the box-shadow property:

```
.box {
  box-shadow: 1px 2px 2px #000;
}
```

A 2px offset-y value.

BOX SHADOW

Example usage of the box-shadow property:

```
.box {
  box-shadow: 1px 2px 2px #000;
}
```

A 2px blur-radius.

BOX SHADOW

Example usage of the box-shadow property:

```
.box {
  box-shadow: 1px 2px 2px #000;
}
```

No spread-radius value is specified, so the drop shadow is the same size as the element.

BOX SHADOW

Example usage of the box-shadow property:

```
.box {
  box-shadow: 1px 2px 2px #000;
}
```

The drop shadow color is black.

BOX SHADOW

Example usage of the box-shadow property:

```
.box {
  box-shadow: 1px 2px 2px #000;
}
```

BOX SHADOW

What if we wanted the blur-radius value to instead be the spread-radius?

BOX SHADOW

Example usage of the box-shadow property:

```
box-shadow: 1px 2px 2px #000;
```

If we wanted this 2px to be the spread-radius instead, we'd need to specify 0 as the blur-radius first.

BOX SHADOW

Example usage of the box-shadow property:

```
box-shadow: 1px 2px 0 2px #000;
```

BOX SHADOW

Example usage of the box-shadow property:

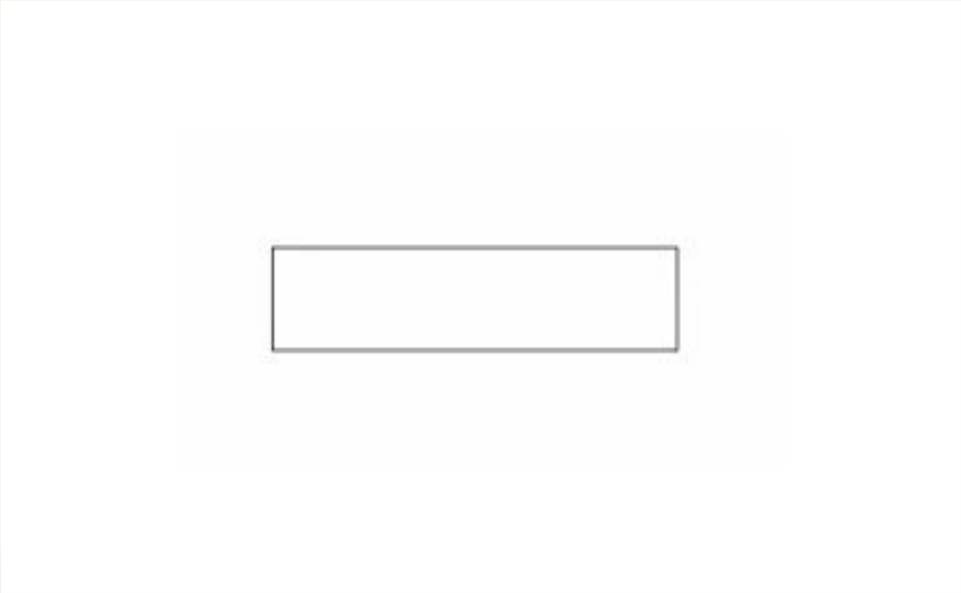
```
box-shadow: 1px 2px 0 2px #000;
```

2px is now the spread-radius.

BOX SHADOW

Example of the blur-radius and spread-radius properties:

blur-radius



spread-radius

BOX SHADOW

You can specify multiple box-shadows via a comma-separated list:

```
.box {
  box-shadow:
    1px 1px 2px #000,

}
```

BOX SHADOW

You can specify multiple box-shadows via a comma-separated list:

```
.box {
  box-shadow:
    1px 1px 2px #000,

}
```

our first box-shadow.

BOX SHADOW

You can specify multiple box-shadows via a comma-separated list:

```
.box {
  box-shadow:
    1px 1px 2px #000,
    inset 1px 1px 2px blue;
}
```

BOX SHADOW

You can specify multiple box-shadows via a comma-separated list:

```
.box {
  box-shadow:
    1px 1px 2px #000,
    inset 1px 1px 2px blue;
}
```

our second box-shadow.

BOX SHADOW

You can specify multiple box-shadows via a comma-separated list:

```
.box {
  box-shadow:
    1px 1px 2px #000,
    inset 1px 1px 2px blue;
}
```

BOX SHADOW

You can also specify negative values:

```
.box {
  box-shadow: -1px -2px 2px #000;
}
```

BOX SHADOW

You can also specify negative values:

```
.box {
  box-shadow: -1px -2px 2px #000;
}
```

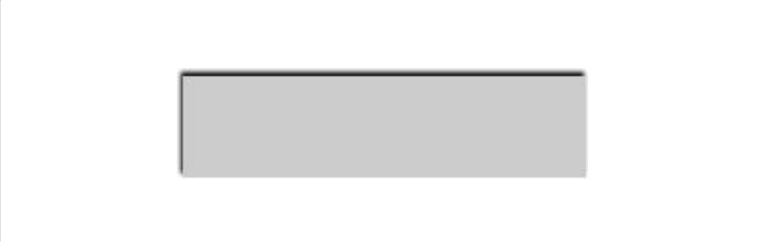


TABLE OF CONTENTS

TABLE OF CONTENTS

TEXT SHADOW

The text-shadow property is very similar to box-shadow, but it applies the shadow to text, as the name implies.

TEXT SHADOW

Example usage of the text-shadow property:

```
<h1>I have a shadow!</h1>
```

TEXT SHADOW

Example usage of the text-shadow property:

```
h1 {
  text-shadow: 1px 2px 2px #000;
}
```

TEXT SHADOW

Example usage of the text-shadow property:

```
h1 {
  text-shadow: 1px 2px 2px #000;
}
```

I have a shadow!

TEXT SHADOW

Example usage of the text-shadow property:

```
text-shadow: <offset-x> <offset-y> <blur-radius> <color>
```

TEXT SHADOW

Example usage of the text-shadow property:

```
text-shadow: <offset-x> <offset-y> <blur-radius> <color>
```

The shadow offset x value.

TEXT SHADOW

Example usage of the text-shadow property:

```
text-shadow: <offset-x> <offset-y> <blur-radius> <color>
```

The shadow offset y value.

TEXT SHADOW

Example usage of the text-shadow property:

```
text-shadow: <offset-x> <offset-y> <blur-radius> <color>
```

The blur-radius alters the blur amount of the shadow, causing it to become bigger *and* lighter (with a larger value).

TEXT SHADOW

Example usage of the text-shadow property:

```
text-shadow: <offset-x> <offset-y> <blur-radius> <color>
```

The color of the shadow.

TEXT SHADOW

Example usage of the text-shadow property:

```
h1 {
  text-shadow: 1px 2px 2px #000;
}
```

TEXT SHADOW

Example usage of the text-shadow property:

```
h1 {
  text-shadow: 1px 2px 2px #000;
}
```

A 1px offset-x value.

TEXT SHADOW

Example usage of the text-shadow property:

```
h1 {
  text-shadow: 1px 2px 2px #000;
}
```

A 2px offset-y value.

TEXT SHADOW

Example usage of the text-shadow property:

```
h1 {
  text-shadow: 1px 2px 2px #000;
}
```

A 2px blur-radius.

TEXT SHADOW

Example usage of the text-shadow property:

```
h1 {
  text-shadow: 1px 2px 2px #000;
}
```

The drop shadow color is black.

TEXT SHADOW

Example usage of the text-shadow property:

```
h1 {
  text-shadow: 1px 2px 2px #000;
}
```

I have a shadow!

TABLE OF CONTENTS

TABLE OF CONTENTS

BOX SIZING

The box-sizing property is used to change the default CSS box model, which is used to calculate widths and heights of given elements.

BOX MODEL REFRESHER

- The CSS box model references the design and layout of given HTML elements
- Each HTML element is a “box,” which consists of margins, borders, padding, and the content of the element
- The “box model” refers to how those properties are calculated in conjunction with one another in order to set the element’s dimensions

BOX MODEL REFRESHER

The content of the box is where the actual content, the text and images, is located:

CONTENT

BOX MODEL REFRESHER

The padding clears the area around the content:

BOX MODEL REFRESHER

The border goes around the padding and content:

BOX MODEL REFRESHER

The margin clears the area around the border:

BOX MODEL REFRESHER

Calculating the width of the .box:

```
.box {
  border: 2px solid black;
  margin: 20px;
  padding: 10px;
  width: 300px;
}
```

BOX MODEL REFRESHER

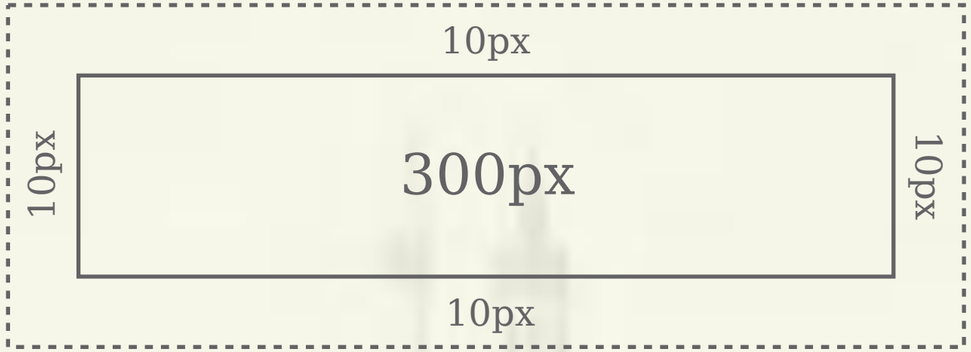
Calculating the width of the .box:

```
.box { width: 300px; }
```

BOX MODEL REFRESHER

Calculating the width of the .box:

```
.box { padding: 10px; }
```



BOX MODEL REFRESHER

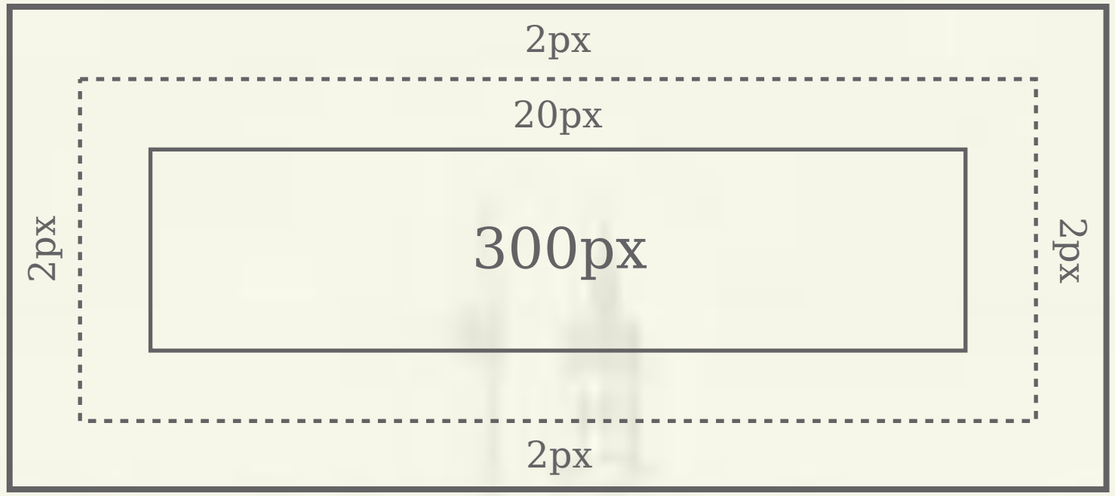
Calculating the width of the .box:

```
.box { padding: 10px; }
```

BOX MODEL REFRESHER

Calculating the width of the .box:

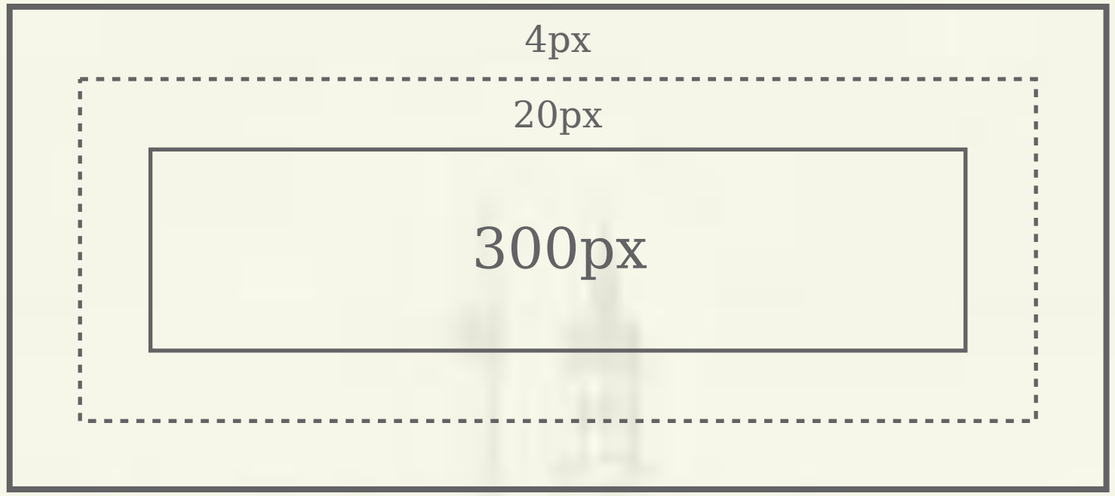
```
.box { border: 2px solid black; }
```



BOX MODEL REFRESHER

Calculating the width of the .box:

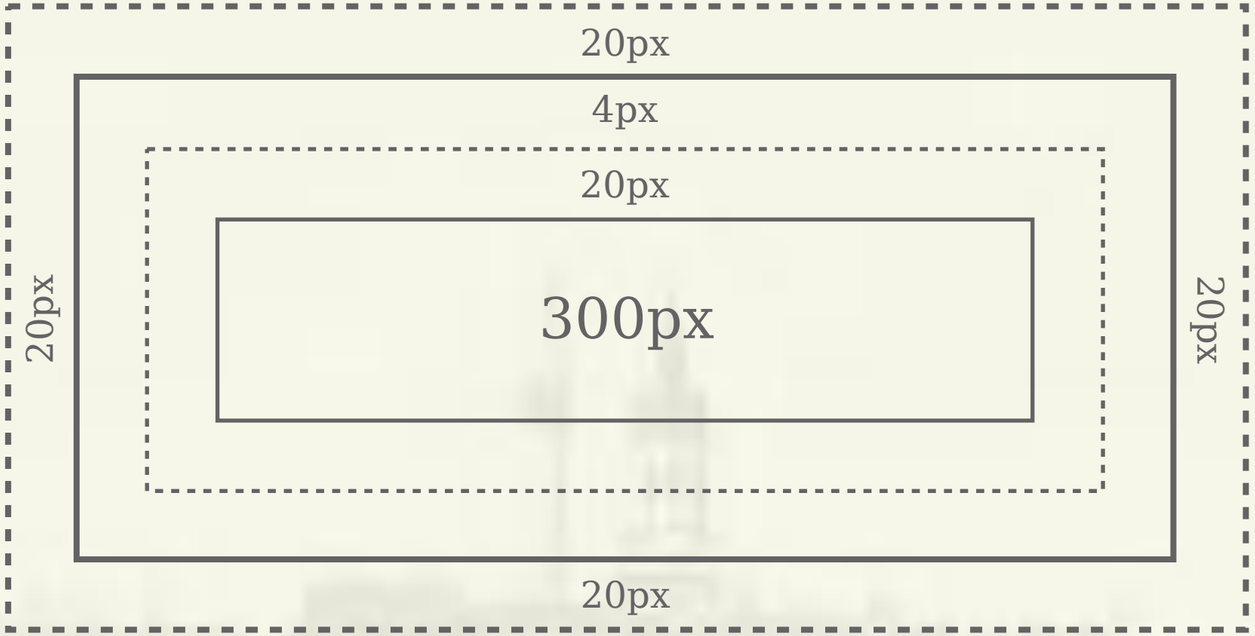
```
.box { border: 2px solid black; }
```



BOX MODEL REFRESHER

Calculating the width of the .box:

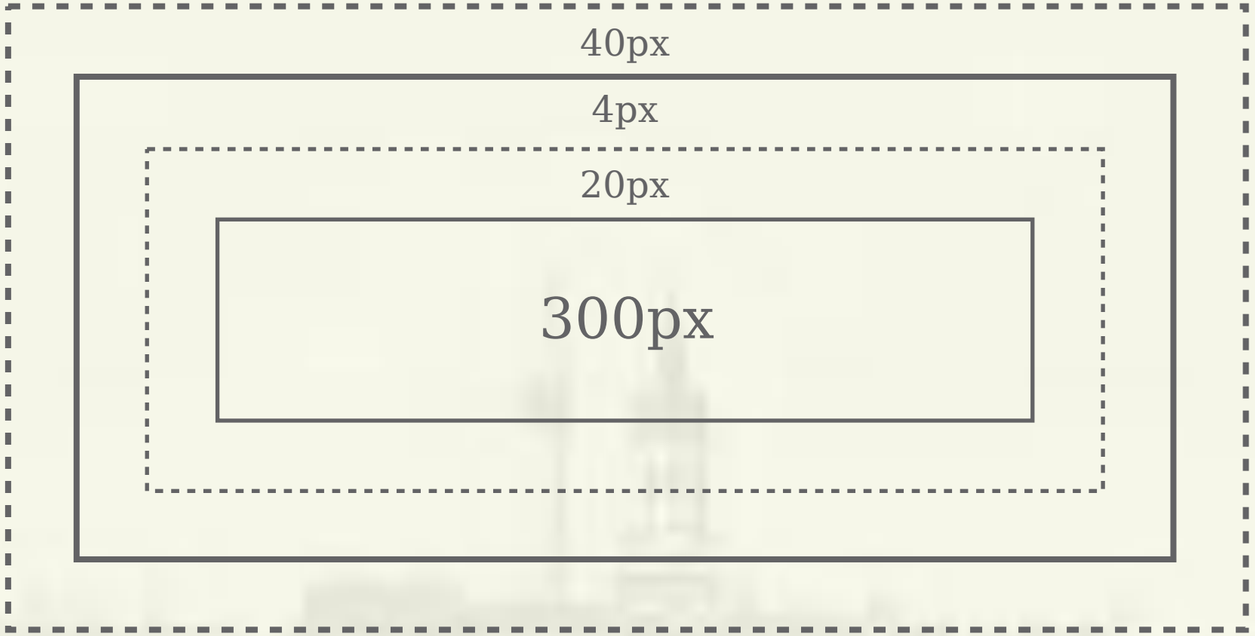
```
.box { margin: 20px; }
```



BOX MODEL REFRESHER

Calculating the width of the .box:

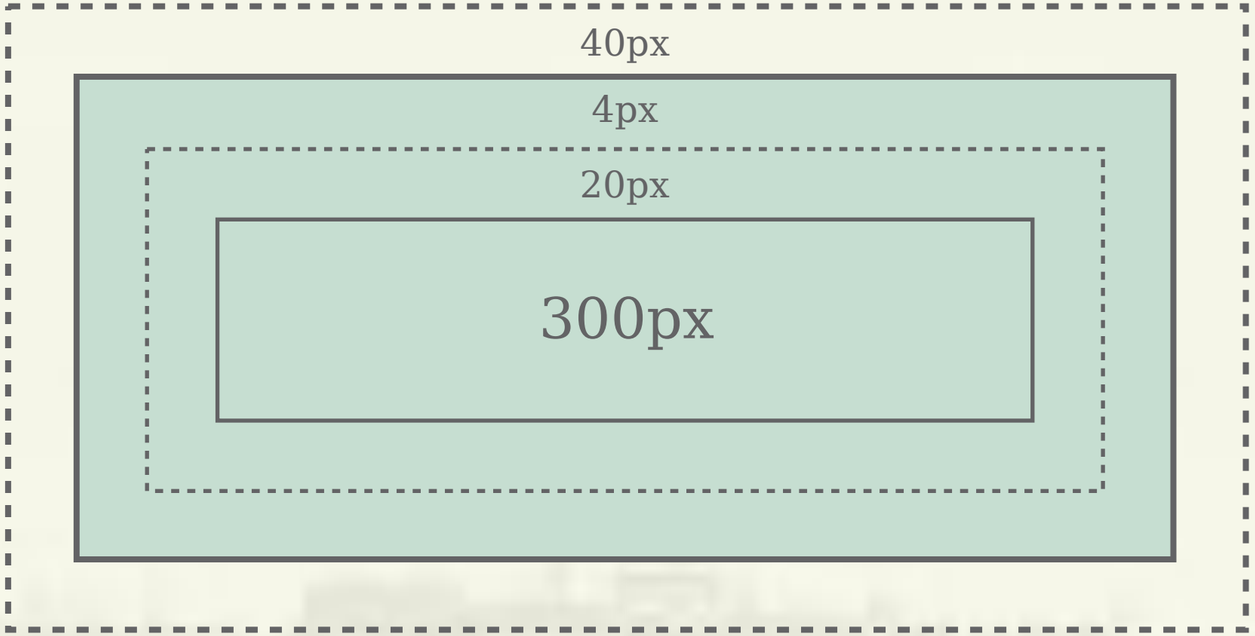
```
.box { margin: 20px; }
```



BOX MODEL REFRESHER

Calculating the width of the .box:

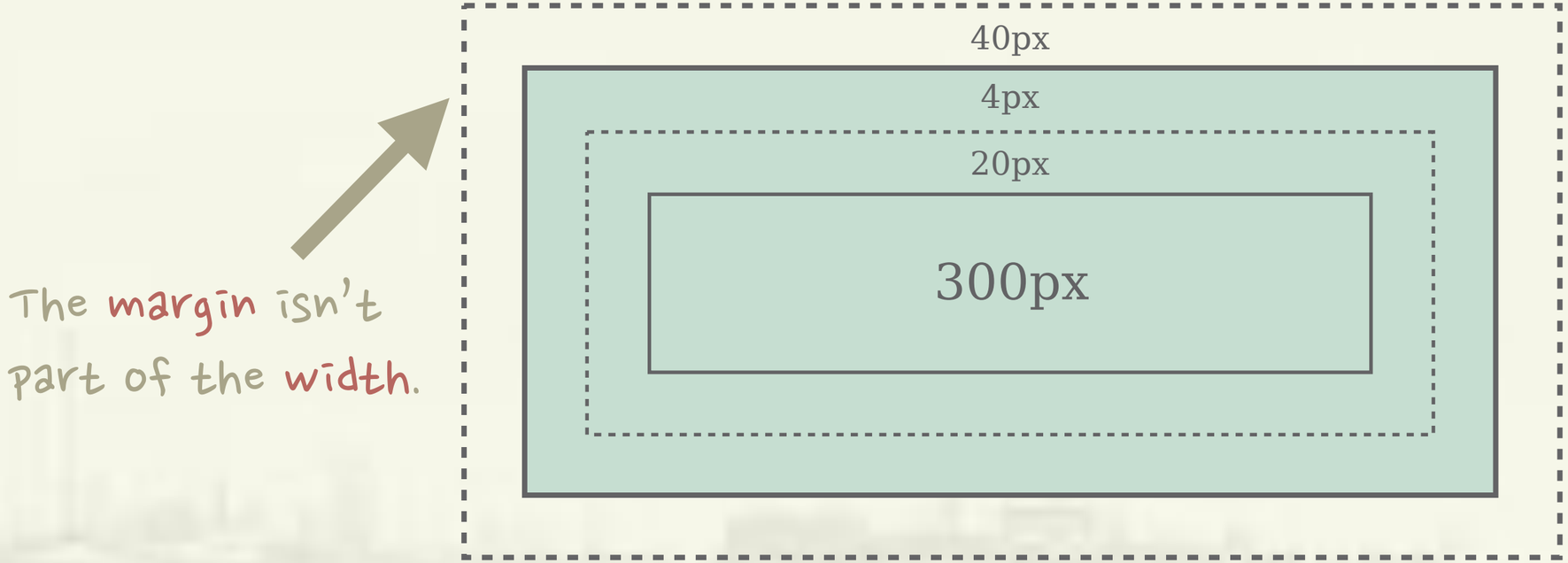
```
.box { margin: 20px; }
```



BOX MODEL REFRESHER

Calculating the width of the .box:

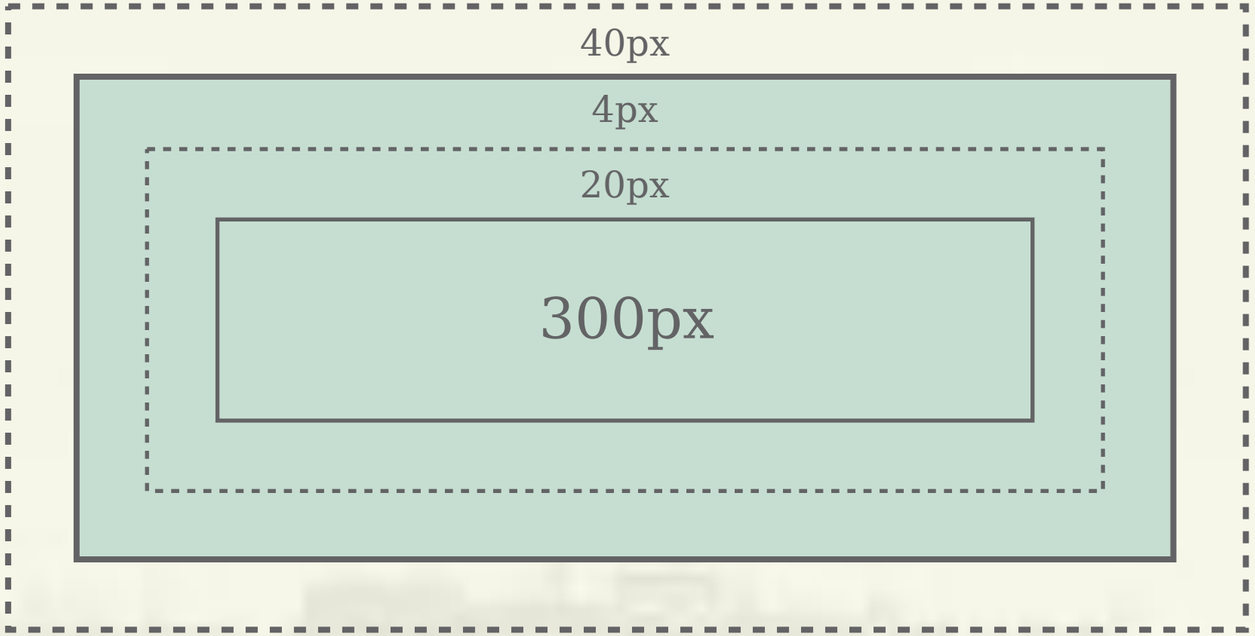
```
.box { margin: 20px; }
```



BOX MODEL REFRESHER

Calculating the width of the .box:

300 + 20 + 4 = 324px



BOX SIZING

The box-sizing property is used to change the default CSS box model, which is used to calculate widths and heights of given elements.

BOX SIZING

There are three different values for box-sizing:

- content-box
- padding-box
- border-box

CONTENT-BOX

This is the default value. The width and height are measured by including *only* the content, but *not* the border, margin, or padding.

PADDING-BOX

The width and height include the padding, but *do not* include the border *or* margin.

PADDING-BOX

Calculating the width of the .box:

```
.box {
  box-sizing: padding-box;
  border: 2px solid black;
  margin: 20px;
  padding: 10px;
  width: 300px;
}
```

PADDING-BOX

Calculating the width of the .box:

```
.box {
  box-sizing: padding-box;
  border: 2px solid black;
  margin: 20px;
  padding: 10px;
  width: 300px;
}
```

The padding has been included in the width (content) area, so they are treated as one region.

PADDING-BOX

Calculating the width of the .box:

```
.box {
  box-sizing: padding-box;
  border: 2px solid black;
  margin: 20px;
  padding: 10px;
  width: 300px;
}
```

304px

BORDER-BOX

The width and height include the padding and border, but *not* the margin.

BORDER-BOX

Calculating the width of the .box:

```
.box {
  box-sizing: border-box;
  border: 2px solid black;
  margin: 20px;
  padding: 10px;
  width: 300px;
}
```

BORDER-BOX

Calculating the width of the .box:

```
.box {
  box-sizing: border-box;
  border: 2px solid black;
  margin: 20px;
  padding: 10px;
  width: 300px;
}
```

The padding and border has been included in the width (content) area, so they are treated as one region.

BORDER-BOX

Calculating the width of the .box:

```
.box {
  box-sizing: border-box;
  border: 2px solid black;
  margin: 20px;
  padding: 10px;
  width: 300px;
}
```

300px

FRONT-END FORMATIONS

TABLE OF CONTENTS

TABLE OF CONTENTS

MULTIPLE BACKGROUNDS

CSS3 allows you to apply multiple backgrounds to an element. They are stacked in the order in which you specify them.

MULTIPLE BACKGROUNDS

First, specify your background-images in a comma-delimited list:

```
.element {
  background-image: url(bg1.png), url(bg2.png);



}
```

MULTIPLE BACKGROUNDS

Then specify the background-position for each, in order:

```
.element {
  background-image: url(bg1.png), url(bg2.png);
  background-position: top left, center right;

}
```

MULTIPLE BACKGROUNDS

Finally, specify the background-repeat for each:

```
.element {
  background-image: url(bg1.png), url(bg2.png);
  background-position: top left, center right;
  background-repeat: no-repeat, no-repeat;
}
```

MULTIPLE BACKGROUNDS

MULTIPLE BACKGROUNDS

The first background-image we specified.

The second background-image we specified.

MULTIPLE BACKGROUNDS

You can also use the shorthand background:

```
.element {
  background:
    url(bg1.png) top left no-repeat,

}
```

MULTIPLE BACKGROUNDS

You can also use the shorthand background:

```
.element {
  background:
    url(bg1.png) top left no-repeat,
    url(bg2.png) center right no-repeat;
}
```

TABLE OF CONTENTS

TABLE OF CONTENTS

COLOR

CSS3 provides multiple ways to work with color:

- RGBa
- HSLa

RGBa

RGB represents the three additive primary colors, red, green, and blue. In CSS3, we can also pass the alpha value (the “a” in RGBa), which represents the opacity of a color.

RGBa

Example usage of rgba:

```
.element {
  color: rgba(0, 0, 0, 0.75);
}
```

RGBa

Example usage of rgba:

```
.element {
  color: rgba(0, 0, 0, 0.75);
}
```

Here we're specifying a 75% opaque black color value.

RGBa

Example usage of rgba:

```
.element {
  color: rgba(0, 0, 0, 0.75);
}
```

Here we're specifying "0 0 0" as the RGB value, which is black.

RGBa

Example usage of rgba:

```
.element {
  color: rgba(0, 0, 0, 0.75);
}
```

Here we're specifying a "0.75" alpha value, which is 75% opaque.

HSLa

CSS3 also adds HSLa (Hue, Saturation, Lightness). In addition to providing the hue, saturation, and lightness values, you can specify the alpha value for the opacity of the color.

HSLa

Example usage of hsla:

```
.element {
  color: hsla(240, 100%, 50%, 0.75);
}
```

HSLa

Example usage of hsla:

```
.element {
  color: hsla(240, 100%, 50%, 0.75);
}
```

The hue value.

HSLa

Example usage of hsla:

```
.element {
  color: hsla(240, 100%, 50%, 0.75);
}
```

The saturation value.

HSLa

Example usage of hsla:

```
.element {
  color: hsla(240, 100%, 50%, 0.75);
}
```

The lightness value.

HSLa

Example usage of hsla:

```
.element {
  color: hsla(240, 100%, 50%, 0.75);
}
```

The alpha value.

HSLa + RGBa

HSLa is more intuitive than RGBa, and it's much easier to make color adjustments on the fly.

HSLa + RGBa

HSLa is more intuitive than RGBa, and it's much easier to make color adjustments on the fly.

However, use whichever color utility you prefer.

TABLE OF CONTENTS

TABLE OF CONTENTS

OPACITY

CSS3 allows you to specify the opacity of an element using the opacity property.

OPACITY

Example usage of the opacity property:

```
.element {
  opacity: 0.45;
}
```

OPACITY

Example usage of the opacity property:

```
.element {
  opacity: 0.45;
}
```

Here we're specifying a "0.45" opacity value, which is 45% opaque.

OPACITY

Example output of the opacity property:

OPACITY

Example output of the opacity property:

OPACITY

Opacity on an element affects all elements that are nested inside.

OPACITY

Example of the opacity property with nested elements:

```
<div class="element">
  <h2>Hello.</h2>
</div>
```

```
.element {
  background: url(bg.jpg) center no-repeat;
  opacity: 0.45;
}
```

OPACITY

Example output of the opacity property with nested elements:

OPACITY

Example output of the opacity property with nested elements:

OPACITY

Example output of the opacity property with nested elements:

The 0.45 opacity on the element affects the text as well as the image.

TABLE OF CONTENTS

TABLE OF CONTENTS

GRADIENTS

CSS3 provides the ability to create gradients, smooth transitions between two or more colors.

GRADIENTS

There are two types of gradients that browsers support:

- Linear gradients
- Radial gradients

LINEAR GRADIENT

To create a linear gradient, we need to specify the starting point, the ending point, and optional stop-color points.

LINEAR GRADIENT

Example usage of a linear-gradient:

```
.element {
  background: linear-gradient(to bottom, red, yellow);
}
```

LINEAR GRADIENT

Example usage of a linear-gradient:

```
.element {
  background: linear-gradient(to bottom, red, yellow);
}
```

LINEAR GRADIENT

Example usage of a linear-gradient:

```
linear-gradient(<angle> to <side-or-corner>, <color-stop>s)
```

LINEAR GRADIENT

Example usage of a linear-gradient:

```
linear-gradient(<angle> to <side-or-corner>, <color-stop>s)
```

?

We can specify the direction through an angle or a keyword.

LINEAR GRADIENT

Example usage of a linear-gradient:

```
linear-gradient(<angle> to <side-or-corner>, <color-stop>s)
```

The angle is generally a degree (e.g. 45deg).

LINEAR GRADIENT

Example usage of a linear-gradient:

```
linear-gradient(<angle> to <side-or-corner>, <color-stop>s)
```

The side-or-corner consists of two keywords:

Horizontal: left or right

Vertical: top or bottom

LINEAR GRADIENT

Example usage of a linear-gradient:

```
linear-gradient(<angle> to <side-or-corner>, <color-stop>s)
```

The color-stops consists of a color and an optional stop position, which can be either a percentage or length.

LINEAR GRADIENT

Example usage of a linear-gradient:

```
.element {
  background: linear-gradient(to bottom, red, yellow);
}
```

LINEAR GRADIENT

Example usage of a linear-gradient:

```
.element {
  background: linear-gradient(to bottom, red, yellow);
}
```

No angle is specified.

LINEAR GRADIENT

Example usage of a linear-gradient:

```
.element {
  background: linear-gradient(to bottom, red, yellow);
}
```

The side-or-corner is bottom, which makes the gradient go from the top to the bottom.

KEYWORDS

to top	→	0deg
to bottom	→	180deg
to right	→	270deg
to left	→	90deg

LINEAR GRADIENT

Example usage of a linear-gradient:

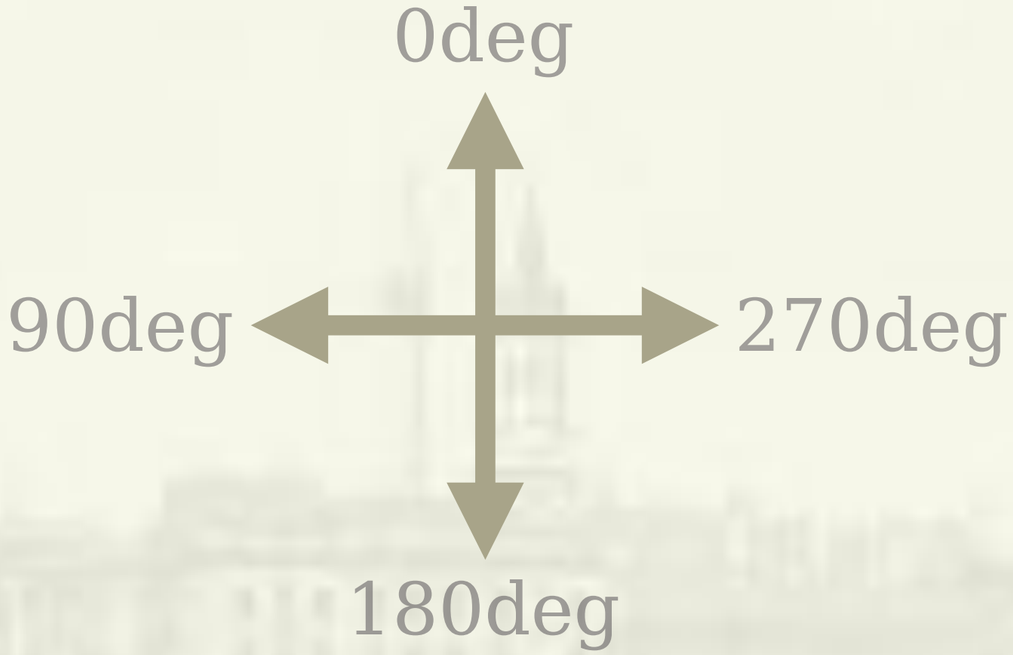
```
.element {
  background: linear-gradient(to bottom, red, yellow);
}
```

This is translated into 180deg.

LINEAR GRADIENT

Example usage of a linear-gradient:

```
.element {
  background: linear-gradient(to bottom, red, yellow);
}
```



LINEAR GRADIENT

Example usage of a linear-gradient:

```
.element {
  background: linear-gradient(to bottom, red, yellow);
}
```

The top of the gradient starts at red.

LINEAR GRADIENT

Example usage of a linear-gradient:

```
.element {
  background: linear-gradient(to bottom, red, yellow);
}
```

The gradient ends at yellow at the bottom.

RADIAL GRADIENT

A radial gradient, unlike a linear gradient, creates a gradient that extends from an origin, the center of the element, extending outward in a circular or elliptical shape.

RADIAL GRADIENT

A radial-gradient consists of:

- The center
- The ending shape contour and position
- Color stops

RADIAL GRADIENT

Example usage of a radial-gradient in its simplest form:

```
.element {
  background:
    radial-gradient(aqua, blue);
}
```

RADIAL GRADIENT

Example usage of a radial-gradient in its simplest form:

```
.element {
  background:
    radial-gradient(aqua, blue);
}
```

RADIAL GRADIENT

Example usage of a radial-gradient in its simplest form:

```
.element {
  background:
    radial-gradient(aqua, blue);
}
```

This creates a two-color elliptical gradient that radiates from the center by default.

RADIAL GRADIENT

Example usage of a radial-gradient:

```
radial-gradient(<shape> <size> at <position>, <color-stop>s)
```

RADIAL GRADIENT

Example usage of a radial-gradient:

```
radial-gradient(<shape> <size> at <position>, <color-stop>s)
```

Specify the shape or size of the gradient.

RADIAL GRADIENT

Example usage of a radial-gradient:

```
radial-gradient(<shape> <size> at <position>, <color-stop>s)
```

The shape of the gradient; circle or ellipse. The default is ellipse.

RADIAL GRADIENT

Example usage of a radial-gradient:

```
radial-gradient(<shape> <size> at <position>, <color-stop>s)
```

The size of the gradient, which consist of keywords.

RADIAL GRADIENT

Example usage of a radial-gra

```
radial-gradient( <shape> <size>                                  )
```

The size of the gradient,
which can

KEYWORDS

closest-side

closest-corner

farthest-side

farthest-corner

KEYWORDS

closest-side

closest-corner

farthest-side

farthest-corner

The default value.

RADIAL GRADIENT

Example usage of a radial-gradient:

```
radial-gradient(<shape> <size> at <position>, <color-stop>s)
```

The size can also be a length or percentage.

RADIAL GRADIENT

Example usage of a radial-gradient:

```
radial-gradient(<shape> <size> at <position>, <color-stop>s)
```

Same as background-position. Default is center.

RADIAL GRADIENT

Example usage of a radial-gradient:

```
radial-gradient(<shape> <size> at <position>, <color-stop>s)
```

The color-stops consists of a color and an optional stop position, which can be either a percentage or length.

RADIAL GRADIENT

Example usage of a radial-gradient:

```
.element {
  background: radial-gradient(circle at top left, aqua, blue);
}
```

RADIAL GRADIENT

Example usage of a radial-gradient:

```
.element {
  background: radial-gradient(circle at top left, aqua, blue);
}
```

The shape of the gradient is circle, rather than ellipse.

RADIAL GRADIENT

Example usage of a radial-gradient:

```
.element {
  background: radial-gradient(circle at top left, aqua, blue);
}
```

The position of the gradient is top left.

RADIAL GRADIENT

Example usage of a radial-gradient:

```
.element {
  background: radial-gradient(circle at top left, aqua, blue);
}
```

The first color-stop is aqua.

RADIAL GRADIENT

Example usage of a radial-gradient:

```
.element {
  background: radial-gradient(circle at top left, aqua, blue);
}
```

The last color-stop is blue.

RADIAL GRADIENT

Example output of the radial-gradient:

FRONT-END FORMATIONS

FRONT-END FORMATIONS

Level 5 - Fonts & Interactions

TABLE OF CONTENTS

TABLE OF CONTENTS

FONT FACE

Using @font-face, we have the ability to provide online fonts for use on our websites.

FONT FACE

Example usage of @font-face:

```
@font-face {



}
```

FONT FACE

We specify the font-family, which is what we'll use to call the font:

```
@font-face {
  font-family: 'OpenSansRegular';




}
```

FONT FACE

We add the location of the font files through the src property:

```
@font-face {
  font-family: 'OpenSansRegular';
  src: url('OpenSansRegular-webfont.eot');



}
```

FONT FACE

We add the location of the font files through the src property:

```
@font-face {
  font-family: 'OpenSansRegular';
  src: url('OpenSansRegular-webfont.eot');



}
```

we'll have to specify multiple font types, which can be added as additional url()'s to the files.

FONT FACE

We specify the font-style:

```
@font-face {
  font-family: 'OpenSansRegular';
  src: url('OpenSansRegular-webfont.eot');
  font-style: normal;


}
```

FONT FACE

We specify the font-weight:

```
@font-face {
  font-family: 'OpenSansRegular';
  src: url('OpenSansRegular-webfont.eot');
  font-style: normal;
  font-weight: normal;
}
```

FONT FACE

Using @font-face in our stylesheet:

```
h1 {
  font-family: 'OpenSansRegular';
}
```

FONT FACE

Using @font-face in our stylesheet:

```
h1 {
  font-family: 'OpenSansRegular';
}
```

we specify the font-family as the same one established in the @font-face call.

FONT FACE

With @font-face fonts, just like any other font declaration, we'll want to add fallback fonts.

FONT FACE

Using @font-face in our stylesheet with fallbacks:

```
h1 {
  font-family: 'OpenSansRegular', Helvetica, Arial, sans-serif;
}
```

Provide fallback fonts here,
as you normally would.

FONT FACE

Using varying weights with @font-face:

```
@font-face {
  font-family: 'OpenSansBold';
  src: url('OpenSansBold-webfont.eot');
  font-style: normal;
  font-weight: normal;
}
```

FONT FACE

Using varying weights with @font-face:

```
@font-face {
  font-family: 'OpenSansBold';
  src: url('OpenSansBold-webfont.eot');
  font-style: normal;
  font-weight: normal;
}
```

we're using a bold font family of 'openSansBold'.

FONT FACE

Using varying weights with @font-face:

```
h1 {
  font-family: 'OpenSansBold';
}
```

we use the bold version by changing the font-family.

FONT FACE

We can alter the @font-face call in order to use the font-weight and font-style properties as usual.

FONT FACE

Using varying weights with @font-face:

```
@font-face {
  font-family: 'OpenSansBold';
  src: url('OpenSansBold-webfont.eot');
  font-style: normal;
  font-weight: normal;
}
```

FONT FACE

Using varying weights with @font-face:

```
@font-face {
  font-family: 'OpenSansRegular';
  src: url('OpenSansBold-webfont.eot');
  font-style: normal;
  font-weight: normal;
}
```

we can instead change the font-family to the same name as the regular weight version.

FONT FACE

Using varying weights with @font-face:

```
@font-face {
  font-family: 'OpenSansRegular';
  src: url('OpenSansBold-webfont.eot');
  font-style: normal;
  font-weight: normal;
}
```

We keep the src url() the same in order to include the bold font weight.

FONT FACE

Using varying weights with @font-face:

```
@font-face {
  font-family: 'OpenSansRegular';
  src: url('OpenSansBold-webfont.eot');
  font-style: normal;
  font-weight: bold;
}
```

We change the font-weight to bold.

FONT FACE

Using varying weights with @font-face:

```
h1 {
  font-weight: bold;
}
```

we use the bold version by changing the font-weight instead of the font-family.

TABLE OF CONTENTS

TABLE OF CONTENTS

TRANSFORM

Using the transform property in CSS3, we can translate, rotate, scale, and skew elements in CSS.

TRANSLATE

You can create a 2D translation using transform:

```
.element {
  transform: translate(20px, 30px);
}
```

TRANSLATE

You can create a 2D translation using transform:

```
.element {
  transform: translate(20px, 30px);
}
```

Translate the .element 20px to the right (x-axis).

TRANSLATE

You can create a 2D translation using transform:

```
.element {
  transform: translate(20px, 30px);
}
```

Translate the .element 30px down (y-axis).

TRANSLATE

Example output of the transform translate:

TRANSLATE

Example usage of a 2D translation using transform:

```
translate(<tx>, <ty>)
```

TRANSLATE

Example usage of a 2D translation using transform:

```
translate(<tx>, <ty>)
```

A <transition-value> for the x-axis, which can be either a length or percentage.

TRANSLATE

Example usage of a 2D translation using transform:

```
translate(<tx>, <ty>)
```

A <transition-value> for the y-axis, which can be either a length or percentage. If not specified, the value is 0.

TRANSLATE

You can use translateX and translateY to translate the x and y values individually:

```
.element {
  transform: translateX(20px);
}
```

```
.element {
  transform: translateY(30px);
}
```

TRANSLATE

You can use translateX and translateY to translate the x and y values individually:

```
translateX(<tx>)
```

```
translateY(<ty>)
```

ROTATE

With rotate, you can rotate an element clockwise around its origin by the specified angle.

ROTATE

Example usage of rotate:

```
.element {
  transform: rotate(45deg);
}
```

ROTATE

Example usage of rotate:

```
.element {
  transform: rotate(45deg);
}
```

The element is rotated 45 degrees.

ROTATE

Example output of the transform rotate:

SCALE

With scale, you can do a 2D scale by a specified unitless number:

```
.element {
  transform: scale(1.2);
}
```

SCALE

With scale, you can do a 2D scale by a specified unitless number:

```
.element {
  transform: scale(1.2);
}
```

The element is scaled to the unitless number, 1.2.

SCALE

Example output of the transform scale:

SCALE

Exampled usage of scale:

```
scale(<sx>, <sy>)
```

SCALE

Exampled usage of scale:

```
scale(<sx>, <sy>)
```

A unitless number for the x-axis.

SCALE

Exampled usage of scale:

```
scale(<sx>, <sy>)
```

A unitless number for the y-axis. If not specified, it defaults to the value of <sx>.

SCALE

You can use scaleX and scaleY to translate the x and y values individually:

```
.element {
  transform: scaleX(1.2);
}
```

```
.element {
  transform: scaleY(0.3);
}
```

SCALE

You can use scaleX and scaleY to scale the x and y values individually:

```
scaleX(<sx>)
```

```
scaleY(<sy>)
```

SKEW

With skew, an element is skewed around the x or y axis by the angle specified.

SKEW

Example usage of skewX:

```
.element {
  transform: skewX(-25deg);
}
```

SKEW

Example usage of skewX:

```
.element {
  transform: skewX(-25deg);
}
```

The element is skewed -25 degrees along the x-axis.

SKEW

Example output of the transform skewX:

SKEW

Example usage of skewX:

```
skewX(<ax>)
```

SKEW

Example usage of skewX:

```
skewX(<ax>)
```

An <angle> for the x-axis.

SKEW

Example usage of skewY:

```
skewY(<ay>)
```

SKEW

Example usage of skewY:

```
skewY(<ay>)
```

An <angle> for the y-axis.

SKEW

Example usage of skewX and skewY:

```
.element {
  transform: skewX(25deg);
}
```

```
.element {
  transform: skewY(-85deg);
}
```

SKEW

Example output of the transform skewX and skewY:

TABLE OF CONTENTS

TABLE OF CONTENTS

TRANSITION

CSS3 provides transitions, which allow you to transition between two states of a specified element.

TRANSITION

Example usage of transition:

```
.element {
  background-color: black;

}
```

TRANSITION

Example usage of transition:

```
.element {
  background-color: black;

}
```

```
.element:hover {
  background-color: blue;
}
```

TRANSITION

Example usage of transition:

```
.element {
  background-color: black;
  transition: background-color 0.2s ease-in-out;
}
```

```
.element:hover {
  background-color: blue;
}
```

TRANSITION

Example usage of transition:

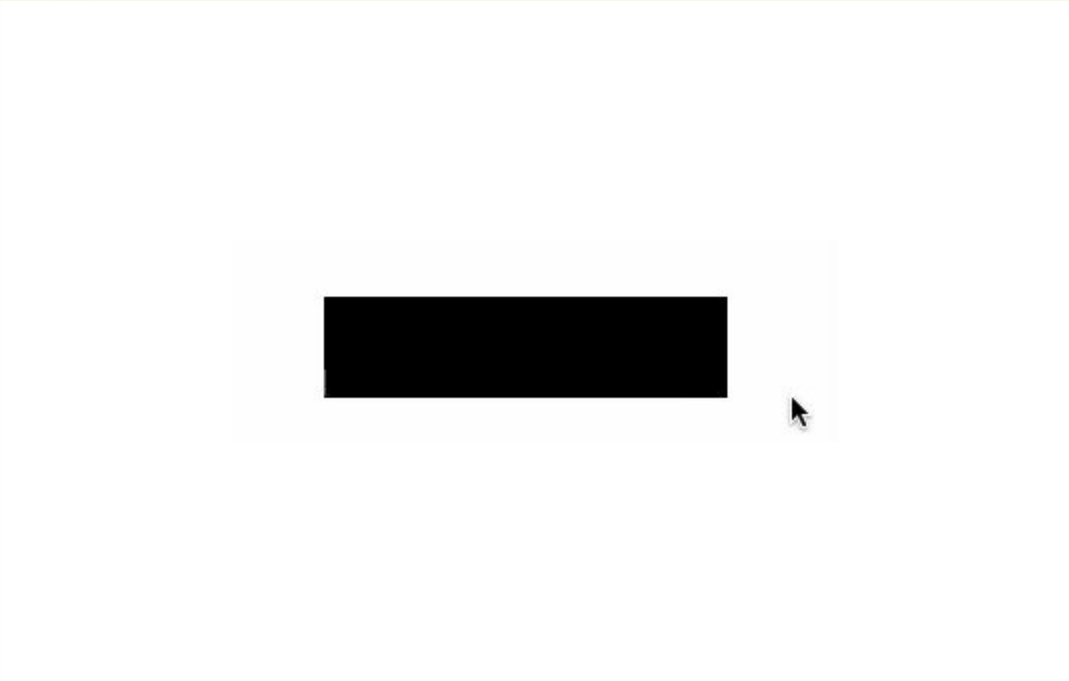
```
.element {
  background-color: black;
  transition: background-color 0.2s ease-in-out;
}
```

```
.element:hover {
  background-color: blue;
}
```

The background-color transitions from black to blue over the period of 0.2 seconds.

TRANSITION

Example output of the transition:



TRANSITION

Example usage of the shorthand transition property:

```
transition: <property> <duration> <timing-function> <delay>
```

TRANSITION

Example usage of the shorthand transition property:

```
transition: <property> <duration> <timing-function> <delay>
```

The CSS property you want to transition.

TRANSITION

Example usage of the shorthand transition property:

```
transition: <property> <duration> <timing-function> <delay>
```

The duration of the transition. The default value is 0s, or 0 seconds.

TRANSITION

Example usage of the shorthand transition property:

```
transition: <property> <duration> <timing-function> <delay>
```

The timing of the transition itself.

TRANSITION

Example usage of the shorthand transition property:

```
transition: <property> <duration> <timing-function> <delay>
```

The timing of the transition itself

TIMING-FUNCTIONS

- ease
- ease-in
- ease-in-out
- linear
- cubic-bezier
- step-start
- step-end
- steps()

TRANSITION

Example usage of the shorthand transition property:

```
transition: <property> <duration> <timing-function> <delay>
```

?

The amount of time to wait between the change that is being requested on a specific property, and the start of the transition.

TRANSITION

You can set the transition values individually, as well:

```
.element {



}
```

TRANSITION

You can set the transition values individually, as well:

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




}
```

TRANSITION

You can set the transition values individually, as well:

```
.element {
  transition-property: background-color;
  transition-duration: 0.2s;



}
```

TRANSITION

You can set the transition values individually, as well:

```
.element {
  transition-property: background-color;
  transition-duration: 0.2s;
  transition-timing-function: ease-in-out;

}
```

TRANSITION

You can set the transition values individually, as well:

```
.element {
  transition-property: background-color;
  transition-duration: 0.2s;
  transition-timing-function: ease-in-out;
  transition-delay: 0.1s;
}
```

TRANSITION

Using all as the transition-property, we can transition multiple properties at once.

TRANSITION

Example usage of transition using the all property:

```
.element {
  background-color: black;
  color: white;

}
```

```
.element:hover {
  background-color: grey;
  color: black;
}
```

TRANSITION

Example usage of transition using the all property:

```
.element {
  background-color: black;
  color: white;
  transition: all 0.2s ease-in-out;
}
```

```
.element:hover {
  background-color: grey;
  color: black;
}
```

TRANSITION

Example usage of transition using the all property:

```
.element {
  background-color: black;
  color: white;
  transition: all 0.2s ease-in-out;
}
```

```
.element:hover {
  background-color: grey;
  color: black;
}
```

The all property will transition both the background-color AND the color properties.

TRANSITION

Example output of the transition using the all property:

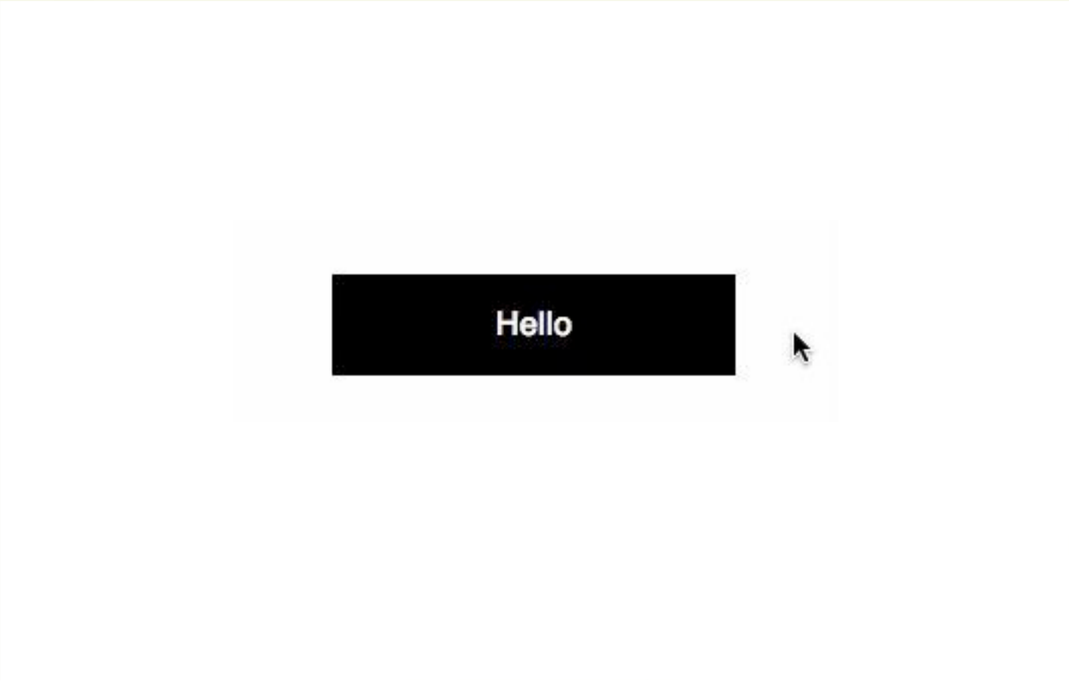


TABLE OF CONTENTS

TABLE OF CONTENTS

PROGRESSIVE ENHANCEMENT

The term “progressive enhancement” refers to the use of newer features that add to the experience in modern browsers that support those features, but doesn’t detract from the experience in older browsers.

PROGRESSIVE ENHANCEMENT

Example of progressive enhancement:

```
.element {
  background: #ccc;
  border-radius: 10px;
  box-shadow: 0 1px 1px rgba(0, 0, 0, 0.75);
}
```

PROGRESSIVE ENHANCEMENT

Example of progressive enhancement:

```
.element {
  background: #ccc;
  border-radius: 10px;
  box-shadow: 0 1px 1px rgba(0, 0, 0, 0.75);
}
```

PROGRESSIVE ENHANCEMENT

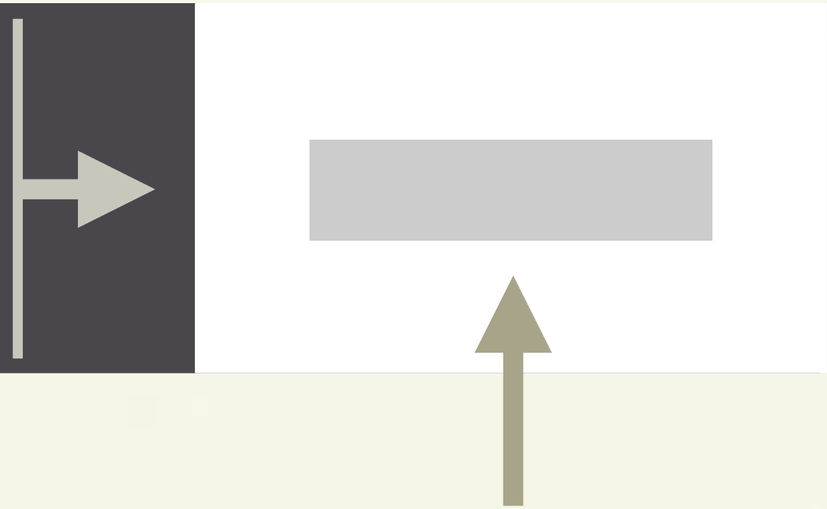
Example of progressive enhancement:

```
.element {
  background: #ccc;
  border-radius: 10px;
  box-shadow: 0 1px 1px rgba(0, 0, 0, 0.75);
}
```

PROGRESSIVE ENHANCEMENT

Example of progressive enhancement:

```
.element {
  background: #ccc;
  border-radius: 10px;
  box-shadow: 0 1px 1px rgba(0, 0, 0, 0.75);
}
```



If the border-radius and box-shadow properties aren't supported, we still get a usable design.

FRONT-END FORMATIONS