

HTML Practice 4B – CSS Examples

Introduction: The latest version of CSS is now available and is supported in the most current browsers. We will be looking at some of the new features of this version using Firefox.

We will first look at the W3 Schools Website on CSS and try out some of the newer properties. Be sure to use a current version of Firefox if possible.

Step 1. Go to the <https://www.w3schools.com/cssref/default.asp> site.

Let's spend some time looking at some of the newer features in CSS. This tutorial allows us to see and try out some of these features.

From the left navigation, we will look at a CSS Property, see its structure and possible values and then use the "Try it Yourself" feature to practice the property. When making changes, click the [Run](#) button to see your changes.

Step 2. Under CSS Properties in the left navigation, click [border](#). Notice in the browser support, there is pretty good coverage for all these properties except for earlier versions of Internet Explorer.

Under CSS Properties, scroll down and click on [border-radius](#). The border-radius can provide just one parameter to determine the horizontal and vertical distance of the radius of all corners. You can provide two values, three values, or four values as shown in the examples. **Try it Yourself:** Take a look at the code that is displaying the border-radius. Values closer to 0px gives corners that are more square.

Change the border-radius to be 0. [Run](#).

Change the border-radius it to be: 40px 100px/100px 40px. [Run](#).

Finally, change the border-radius to be: 40px 0 40px 0. [Run](#).

Step 3. Go back to the CSS Properties menu and look at [box-shadow](#). You can add a shadow to text by using the [text-shadow](#) property or to a box by using the [box-shadow](#) property.

For Firefox browsers after 3.5 you can just use the [box-shadow](#) property.

In its simplest use, you only specify the horizontal shadow and the vertical shadow with the actual value expressed in pixels. **Try it Yourself:** Take a look at the code that is displaying in the See Result window. The first pixel value moves the shadow to the left and the second pixel value moves the shadow above. Experiment with the [text-shadow](#) property and the [box-shadow](#) property.

perspective-origin

pointer-events

position

quotes

resize

right

scroll-behavior

tab-size

table-layout

text-align

text-align-last

text-decoration

text-decoration-color

text-decoration-line

text-decoration-style

text-indent

text-justify

text-overflow

text-shadow

CSS text-shadow Property

[< Previous](#)

[Complete CSS Reference](#)

Example

Basic text-shadow:

```
h1 {
  text-shadow: 2px 2px #ff0000;
}
```

[Try it Yourself >](#)

More "Try it Yourself" examples below.

border-left-style

border-left-width

border-radius

border-right

border-right-color

border-right-style

border-right-width

border-spacing

border-style

border-top

border-top-color

border-top-left-radius

border-top-right-radius

border-top-style

border-top-width

border-width

bottom

box-decoration-break

box-shadow

CSS box-shadow Property

[< Previous](#)

[Complete CSS Reference](#)

Example

Add shadows to different <div> elements:

```
#example1 {
  box-shadow: 5px 10px;
}

#example2 {
  box-shadow: 5px 10px #888888;
}
```

[Try it Yourself >](#)

Step 4. Go back to the CSS Properties menu and look at [border-image](#). This property is not supported by earlier versions of Internet Explorer.

The [border-image](#) property allows you to provide an image for the border that can be rounded, repeated, or stretched.

The border-image property also needs the border-width set to provide it a width.

Think of your image as being divided into 9 parts (sliced). The corner parts are obvious.

When we give the slice values we leave off the px. The diamond dimensions in the border source image are 50 by 50.

Try it Yourself: Take a look at the code that is displaying the border-image. Try the other [border-image](#) examples as well.

background-blend-mode

background-clip

background-color

background-image

background-origin

background-position

background-repeat

background-size

border

border-bottom

border-bottom-color

border-bottom-left-radius

border-bottom-right-radius

border-bottom-style

border-bottom-width

border-collapse

border-color

border-image

border-image-outset

border-image-repeat

border-image-slice

border-image-source

border-image-width

border-left

Example

Different slice values completely changes the look of the border:

```
#borderimg1 {
  border: 10px solid transparent;
  padding: 15px;
  border-image: url(border.png) 50 round;
}

#borderimg2 {
  border: 10px solid transparent;
  padding: 15px;
  border-image: url(border.png) 20% round;
}

#borderimg3 {
  border: 10px solid transparent;
  padding: 15px;
  border-image: url(border.png) 30% round;
}
```

Try it Yourself »

Step 5. Next we will look at [background](#). There is good support for these properties in the most current browsers. The [background-size](#) allows you to manage the size of a background image as it is displayed on the page, but does not actually resize the image file.

Try it Yourself: Take a look at the code that is displaying the background-size. Go back to CSS Backgrounds.

animation-name

animation-play-state

animation-timing-function

backface-visibility

background

background-attachment

background-blend-mode

background-clip

background-color

background-image

background-origin

background-position

background-repeat

background-size

border

border-bottom

border-bottom-color

border-bottom-left-radius

Example

Specify the size of a background image with "auto" and in pixels:

```
#example1 {
  background: url(mountain.jpg);
  background-repeat: no-repeat;
  background-size: auto;
}

#example2 {
  background: url(mountain.jpg);
  background-repeat: no-repeat;
  background-size: 300px 100px;
}
```

Try it Yourself »

Step 6. Next, go to the [word-wrap](#) property. The browser support is good for text-overflow, word-wrap, and wordbreak.

Let's take a look at [word-wrap](#). This allows very long words or URLs that could not fit to be wrapped within the containing element.

Try it Yourself: Take a look at the code for the word-wrap property. Make this change to the word-wrap: normal; [Run](#).

transform-origin
transform-style
transition
transition-delay
transition-duration
transition-property
transition-timing-function
unicode-bidi
user-select
vertical-align
visibility
white-space
width
word-break
word-spacing
word-wrap
writing-mode

Example

Allow long words to be able to break and wrap onto the next line:

```
div {  
  word-wrap: break-word;  
}
```

Try it Yourself »

Definition and Usage

The **word-wrap** property allows long words to be able to be broken and w

Step 7. Next, go to the [@font-face](#) property. Now with CSS, we can make use of fonts that may not be available on the user's computer with the [@font-face](#) property.

If there is a problem embedding your font, always provide other fonts in your font-family list to act as fall back fonts.

See the Intro to [CSS slideshow](#) in Canvas for help on converting a TTF file to an EOT file needed for IE. We will also change the order of the font src urls from that shown in this tutorial. **Try it yourself** Example: Take a look at the code for the @font-face rule.

Notice first the new font is given a name in the font-family property. This could be any name, but should probably be the name of the font. Names with more than one word should be in “ ”.

HTML CSS JAVASCRIPT SQL PHP BOOTSTRAP HOW TO PYTHON W3.CSS JQUERY

counter-increment
counter-reset
cursor
direction
display
empty-cells
filter
flex
flex-basis
flex-direction
flex-flow
flex-grow
flex-shrink
flex-wrap
float
font
@font-face
font-family
font-kerning
font-size
font-size-adjust

10 Gig Speed In Orem

UTOPIA Fiber Is Available
Speeds Up To 10 Gig Now

CSS @font-face Rule

< Previous

Complete CSS Reference

Example

Specify a font named "myFirstFont", and specify the URL where it can be found:

```
@font-face {  
  font-family: myFirstFont;  
  src: url(sansation_light.woff);  
}
```

Try it Yourself »

Website with Embedded Fonts:

Download the CSS3Fonts Zipped file in Canvas and Extract the folder.

If using both a TTF and an EOT file...

- First list src: url of the EOT file without the format attribute.
- Next, do src: local ('font name') to see if the font is already on the local machine. Then separated with a comma, do the url of the TTF file with the format ('truetype').

In this CSS3Fonts Web site, notice the [fonts folder](#) contains some [TTF](#) and [EOT](#) font files. The fonts used in this example were free fonts with an open license. You can see the documentation for these fonts in the text files. If using font files, always include licensing and use guideline documentation on your site.

In the embedded CSS on the [CSS3Fonts.html](#) page, you can see the @font-face with the suggested ordering. View the CSS3fonts.html page in both IE and Firefox. The embedded font should show up in both. Change the font to be the [Darkh4.eot](#) and [Darkh4.ttf](#). Then view the page again in both browsers.

Conclusion: As CSS continues to evolve, these newer CSS properties will become more prevalent. Make sure to always include some fallback styling for browsers where the feature is not supported and then test your pages in these browsers. Hopefully you enjoyed this exploration. The only submission for this assignment will be the CSS3Fonts folder.

Submission: Upload your CSS3Fonts folder to your Practices folder on Yoda. Please provide the Internet URL to the CSS3Fonts.html page in the Canvas.