



Code4SP Training Materials Subchapter 3: CSS

WP3: Code4SP Training Materials

Prepared by: Social Hackers Academy



CITIZENS
IN POWER



Center for Social
Innovation



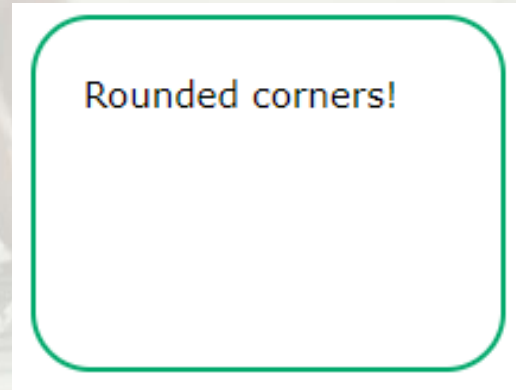
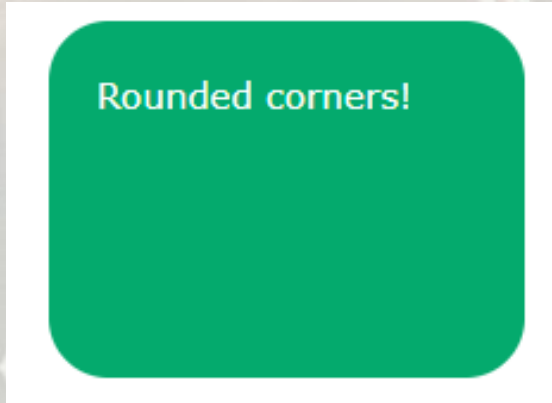
ZAUG
gGmbH



CSS Rounded Corners

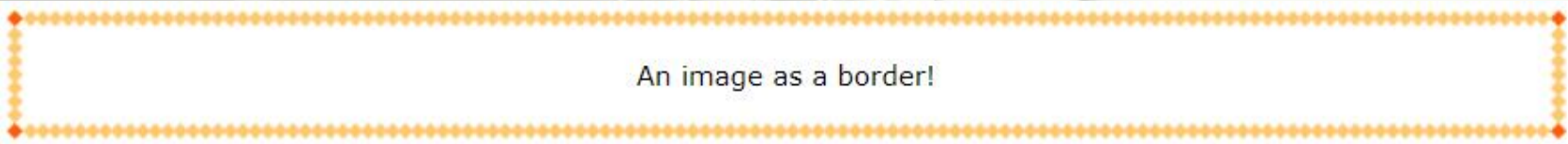
The border-radius property defines the radius of an element's corners.

Tip: This property allows you to add rounded corners to elements!



CSS Border Images

- The CSS border-image property allows you to specify an image to be used instead of the normal border around an element.
- The property has three parts:
 - The image to use as the border
 - Where to slice the image
 - Define whether the middle sections should be repeated or stretched

A white rectangular box with a decorative border of orange dots. The text "An image as a border!" is centered inside the box.

An image as a border!

CSS Backgrounds

- CSS allows you to add multiple background images for an element, through the background-image property
- The following example has two background images, the first image is a flower (aligned to the bottom and right) and the second image is a paper background (aligned to the top-left corner)

```
#example1 {  
  background-image: url(img_flwr.gif), url(paper.gif);  
  background-position: right bottom, left top;  
  background-repeat: no-repeat, repeat;  
}
```





CSS Colors

- CSS supports +140 color names, HEX values, RGB values, RGBA values, HSL values, HSLA values, and opacity.

```
rgba(255, 0, 0, 0.2);  
rgba(255, 0, 0, 0.4);  
rgba(255, 0, 0, 0.6);  
rgba(255, 0, 0, 0.8);
```

```
hsl(0, 100%, 30%);  
hsl(0, 100%, 50%);  
hsl(0, 100%, 70%);  
hsl(0, 100%, 90%);
```

```
hsla(0, 100%, 30%, 0.3);  
hsla(0, 100%, 50%, 0.3);  
hsla(0, 100%, 70%, 0.3);  
hsla(0, 100%, 90%, 0.3);
```

```
rgb(255, 0, 0);opacity:0.2;  
rgb(255, 0, 0);opacity:0.4;  
rgb(255, 0, 0);opacity:0.6;  
rgb(255, 0, 0);opacity:0.8;
```





CSS Color Keywords

This page will explain the transparent, currentcolor, and inherit keywords :

- The transparent keyword is used to make a color transparent. This is often used to make a transparent background color for an element.
- The currentcolor keyword is like a variable that holds the current value of the color property of an element.

This keyword can be useful if you want a specific color to be consistent in an element or a page.

- The inherit keyword specifies that a property should inherit its value from its parent element.

The inherit keyword can be used for any CSS property, and on any HTML element.





CSS Gradient

CSS gradients let you display smooth transitions between two or more specified colors.

CSS defines three types of gradients:

- Linear Gradients (goes down/up/left/right/diagonally)
- Radial Gradients (defined by their center)
- Conic Gradients (rotated around a center point)

A rectangular box with a horizontal gradient background transitioning from purple on the left to green on the right. The text 'Gradient Backgrounds' is centered in white.

Gradient Backgrounds





CSS Shadows

With CSS you can add shadow to text and to elements.

In these chapters you will learn about the following properties :

- Text-shadow
- Box-shadow

With CSS you can create shadow effects!



Shadows





CSS Text Effects

In this chapter you will learn about Text-overflow, word-wrap, word-break, writing-mode :

- The text-overflow property specifies how overflowed content that is not displayed should be signaled to the user.
- The CSS word-wrap property allows long words to be able to be broken and wrap onto the next line.
- The CSS word-break property specifies line breaking rules.
- The CSS writing-mode property specifies whether lines of text are laid out horizontally or vertically.





CSS Web Fonts

- Web fonts allow Web designers to use fonts that are not installed on the user's computer.
- When you have found/bought the font you wish to use, just include the font file on your web server, and it will be automatically downloaded to the user when needed.
- Your "own" fonts are defined within the CSS @font-face rule

The @font-face Rule

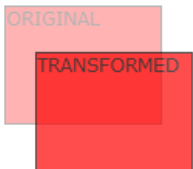
With CSS, websites can use **fonts other than the pre-selected "web-safe" fonts.**



CSS 2D transforms

CSS transforms allow you to move, rotate, scale, and skew elements.

The translate() Method



The rotate() Method



The scale() Method



CSS 3D transforms

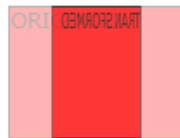
With the CSS transform property you can use the following 3D transformation methods:

- RotateX()
- RotateY()
- RotateZ()

The rotateX() Method



The rotateY() Method





CSS Transitions

- CSS transitions allows you to change property values smoothly, over a given duration.
- To create a transition effect, you must specify two things:
 - the CSS property you want to add an effect to
 - the duration of the effect
- The following example shows a 100px * 100px red <div> element. The <div> element has also specified a transition effect for the width property, with a duration of 2 seconds:

```
div {  
  width: 100px;  
  height: 100px;  
  background: red;  
  transition: width 2s;  
}
```





CSS Animations

- An animation lets an element gradually change from one style to another.
- You can change as many CSS properties you want, as many times as you want.
- To use CSS animation, you must first specify some keyframes for the animation.
- Keyframes hold what styles the element will have at certain times.
- These are the main the main animation's properties :

- `@keyframes`
- `animation-name`
- `animation-duration`
- `animation-delay`
- `animation-iteration-count`
- `animation-direction`
- `animation-timing-function`
- `animation-fill-mode`
- `animation`



CSS Tooltips

- A tooltip is often used to specify extra information about something when the user moves the mouse pointer over an element:

default top	blue top	green top	orange top	red top
tooltip top	tooltip top blue	tooltip top green	tooltip top orange	tooltip top red
tooltip bottom	tooltip bottom blue	tooltip bottom green	tooltip bottom orange	tooltip bottom red
default bottom	blue bottom	green bottom	orange bottom	red bottom

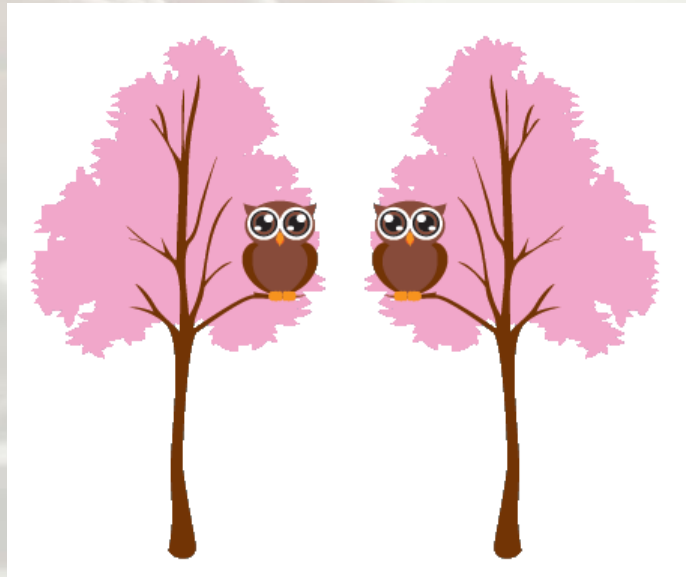
CSS Style images

- Using CSS to style images allows you to uniformly specify how images should appear across your website with only a few rulesets : add a border, and change the shape, and size of the image, ...



CSS image reflection

- The box-reflect property is used to create an image reflection.
- The value of the The box-reflect property can be : below, above, left, or right



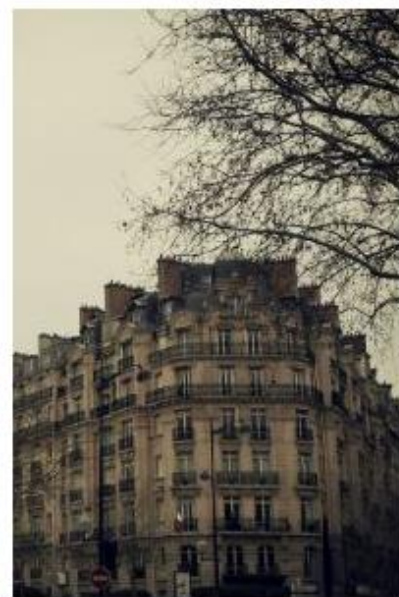
CSS Object-fit

- The CSS object-fit property is used to specify how an `` or `<video>` should be resized to fit its container.
- This property tells the content to fill the container in a variety of ways; such as "preserve that aspect ratio" or "stretch up and take up as much space as possible".
- Look at the following image from Paris. This image is 400 pixels wide and 300 pixels high:



CSS Object-position

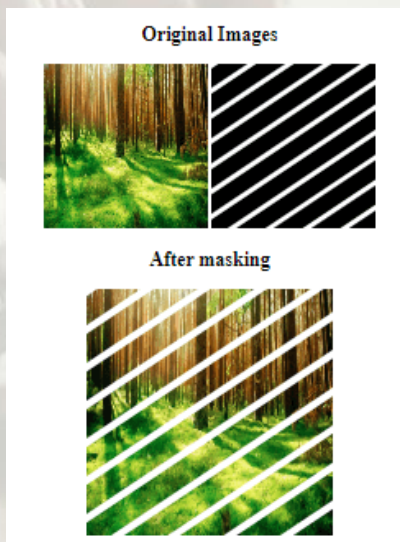
- Let's say that the part of the image that is shown, is not positioned as we want. To position the image, we will use the object-position property
- Here we will use the object-position property property to position the image so that the great old building is in center:



CSS Masking

- With CSS masking you create a mask layer to place over an element to partially or fully hide portions of the element.
- The CSS mask-image property specifies a mask layer image.

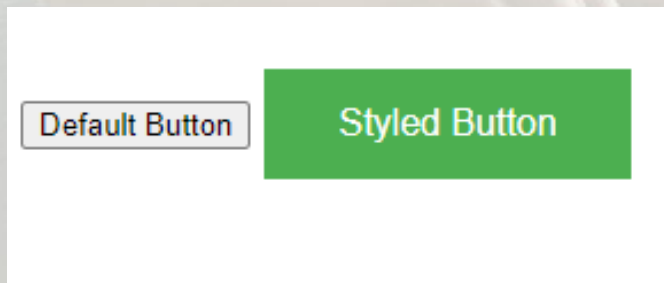
The mask layer image can be a PNG image, an SVG image, a CSS gradient, or an SVG `<mask>`element.





CSS Buttons

- There are lots of properties to style a button in CSS
- This is an example :



```
.button {  
  background-color: #4CAF50;  
  border: none;  
  color: white;  
  padding: 15px 32px;  
  text-align: center;  
  text-decoration: none;  
  font-size: 16px;  
  margin: 4px 2px;  
}
```





CSS Pagination

If you have a website with lots of pages, you may wish to add some sort of pagination to each page and these are some examples :





CSS Multiple Columns

The CSS multi-column layout allows easy definition of multiple columns of text - just like in newspapers:

Daily Ping

Lorem ipsum

dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud

exercitation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio

dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum.



CSS User Interface

In this chapter you will learn about the following CSS user interface properties:

- The `resize` property specifies if (and how) an element should be resizable by the user.

This div element is resizable by the user!

To resize: Click and drag the bottom right corner of this div element.

- The `outline-offset` property adds space between an outline and the edge or border of an element.

This div has an outline 15px outside the border edge.



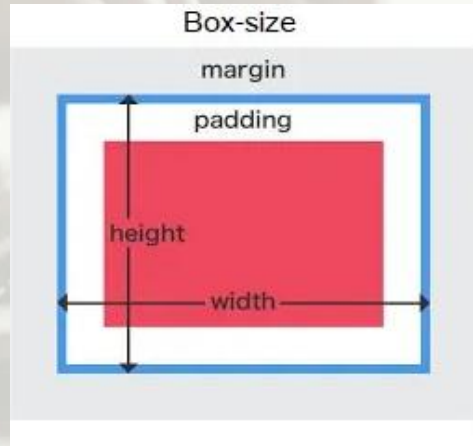
CSS Variables

- The `var()` function is used to insert the value of a CSS variable.
- CSS variables have access to the DOM, which means that you can create variables with local or global scope, change the variables with JavaScript, and change the variables based on media queries.
- A good way to use CSS variables is when it comes to the colors of your design. Instead of copy and paste the same colors over and over again, you can place them in variables.



CSS Box sizing

- The box-sizing property in CSS defines how the user should calculate the total width and height of an element i.e padding and borders, are to be included or not.





CSS Media Queries

Media queries in CSS3 extended the CSS2 media types idea: Instead of looking for a type of device, they look at the capability of the device.

Media queries can be used to check many things, such as:

- width and height of the viewport
- width and height of the device
- orientation (is the tablet/phone in landscape or portrait mode?)
- resolution

Using media queries are a popular technique for delivering a tailored style sheet to desktops, laptops, tablets, and mobile phones (such as iPhone and Android phones).

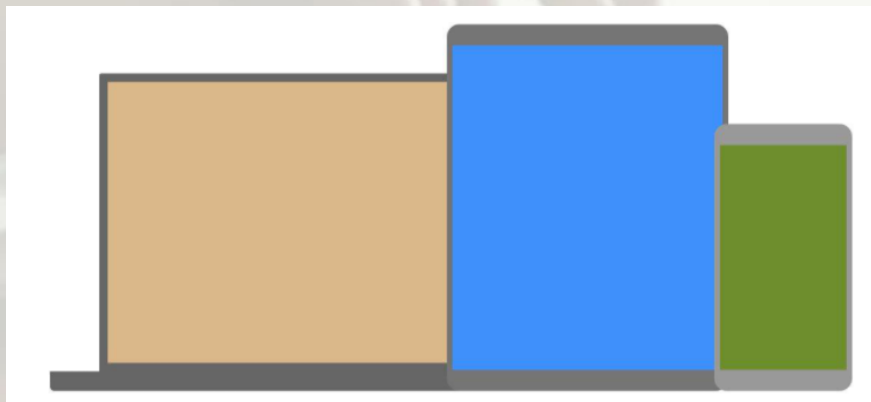




CSS MQ examples

Let us look at some more examples of using media queries.

Media queries are a popular technique for delivering a tailored style sheet to different devices. To demonstrate a simple example, we can change the background color for different devices:





CSS Flexbox

The Flexible Box Layout Module, makes it easier to design flexible responsive layout structure without using float or positioning.

