

# **Tutorial 8**

## **Enhancing a Web Site with Advanced CSS**

### HTML, CSS, and Dynamic HTML

5<sup>TH</sup> EDITION



# Objectives

---

- Create text and box shadows
- Work with IE filters
- Rotate an object using CSS3
- Create linear gradients
- Apply a border image
- Set the opacity of a page object

# Objectives

---

- Apply a style to a media device
- Create and apply print styles
- Define the visual viewport
- Create a media query
- Create styles for mobile devices in portrait and landscape mode

# Visual Effects with CSS3

The `text-shadow` property adds a drop shadow to a text string; the first parameter sets the shadow color; the next two parameters specify the horizontal and vertical offsets; and the last parameter sets the size of the shadow blur.

```
text-shadow:  
green 3px 4px 5px;
```

Stanis

Stanislav Dubček (1903 - 1981)

The Dubček Family  
Stanislav (1903 - 1981)  
Irena (1910 - 2005)  
Peter (1931 - 2002)  
Nadine (1932 - 2009)  
Robert (1934 - )  
Vera (1936 - )  
Karen (1939 - 2001)  
Jan (1944 - )

Family Album  
Family Tree  
Galleries  
Archival Records  
Timeline  
Articles  
Story Book  
Message Board

Extended Family

Links

Stanislav and Irena (1929)



<b>Born</b>	February 5, 1903 (Hnatince, Bohemia)
<b>Died</b>	September 20, 1981 (Rochester, MN)
<b>Married</b>	Irena (Torkia); June 2, 1929
<b>Children</b>	Peter, Nadine, Robert, Vera, Karen, Jan

Family portrait (1918)



Family portrait (1953)



At work (1947)



The `linear-gradient()` function creates a background color blending two or more colors in a straight line from an initial color to an ending color.

```
background: linear-gradient  
(rgb(51, 66, 51), white);
```

The `transform` property applies a 2D or 3D transformation to an object; the `rotate()` function rotates an object in two dimensions to a specified angle of rotation.

```
transform:  
rotate(30deg);
```

The `box-shadow` property adds a drop shadow to any element; the parameters match those for the text-shadow property.

```
box-shadow:  
gray 3px 4px 5px;
```

# Creating Drop Shadows with CSS3

---

- To create a text shadow, apply the style

```
text-shadow: color offsetX  
offsetY blur;
```

where `color` is the color of the shadow, `offsetX` and `offsetY` are the displacements of the shadow from the text in the horizontal and vertical directions, and `blur` is the size of the blurring effect.

# Creating Drop Shadows with CSS3

---

- To apply a drop shadow to a page element, use the style

```
box-shadow: [inset] color  
offsetX offsetY blur [spread];
```

where the optional `inset` keyword places the shadow within the object, and the optional `spread` value increases or decreases the size of the shadow relative to the size of the object.

# Creating Drop Shadows with CSS3

Figure 8-4 Adding text shadows to the h1 heading

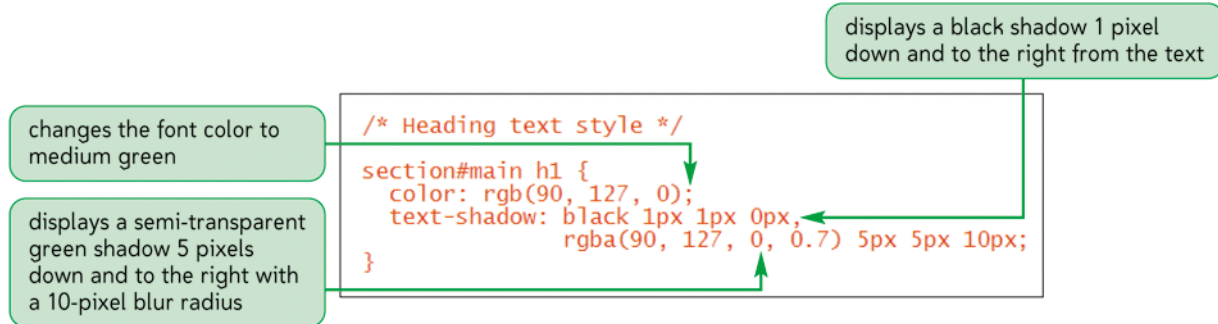
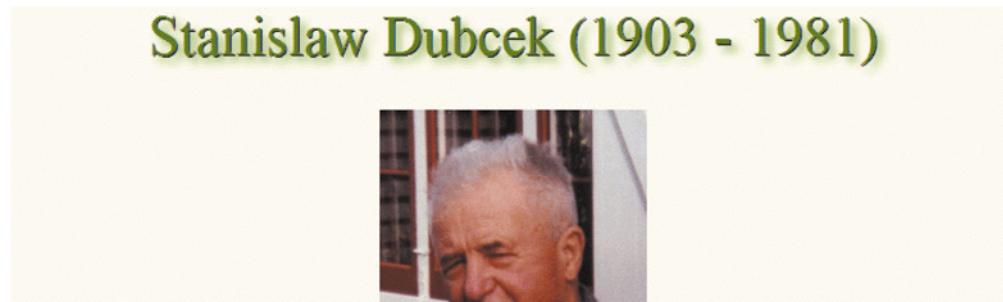


Figure 8-5 Heading text with an embossed effect



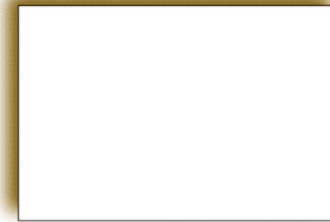
# Creating Drop Shadows with CSS3

Figure 8-6

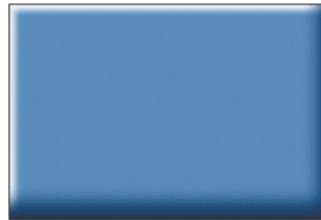
Examples of box shadows



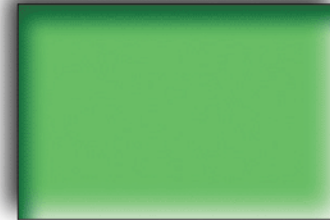
`box-shadow: 15px 15px;`



`box-shadow: rgb(127, 90, 0)`  
`-10px -10px 15px;`



`background-color: rgb(90, 141, 191);`  
`box-shadow: inset rgb(0, 51, 101)`  
`-10px -15px 15px,`  
`inset white 5px 5px 5px;`



`background-color: rgb(101, 191, 101);`  
`box-shadow: inset rgb(0, 101, 51)`  
`10px 15px 15px,`  
`inset white -5px -5px 25px,`  
`rgb(51, 51, 51) -10px -10px 20px;`



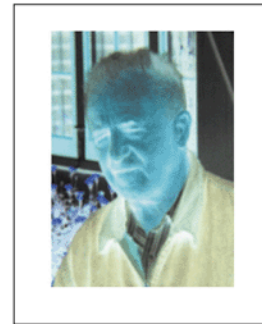
# Introducing Internet Explorer Filters

Figure 8-10

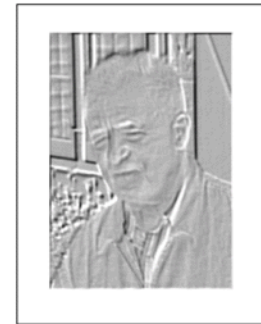
Examples of IE filters



Alpha(style=2)



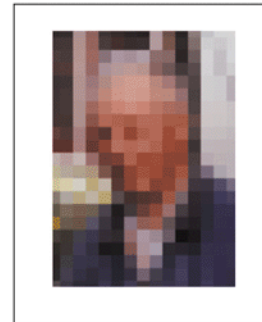
BasicImage(invert=1)



Emboss()



Engrave()



Pixelate(maxSquare=10)



Wave(freq=3, strength=10)

# Applying an Internet Explorer Filter

---

- To apply an Internet Explorer filter, add the style
  - `filter:`  
`progid:DXImageTransform.Microsoft.`  
`filter(param) ;`  
  
where `filter` is the name of an Internet Explorer visual effect, and `param` is the parameters that apply to that effect

# Applying an Internet Explorer Filter

---

- To apply an Internet Explorer filter in compliance with correct CSS syntax, add the following style:  
`-ms-filter:`  
`"progid:DXImageTransform.Microsoft`  
`t.filter(param) " ;`

# Rotating an Object

---

- Styles that modify the placement or orientation of a page object are organized under the transform style

`transform: effect(params);`

where `effect` is the transformation function that will be applied to the object, and `params` are any parameters required by the transformation

# Rotating an Object

Figure 8-14

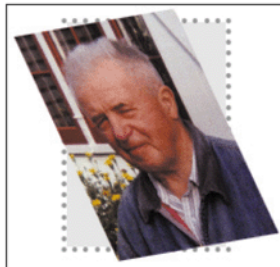
Examples of CSS3 transformation functions



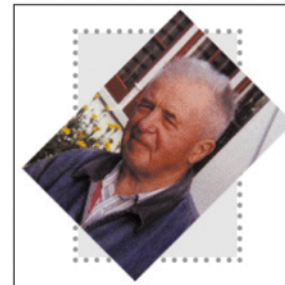
`transform: translate(10px, -15px);`



`transform: scale(0.8, 0.5);`



`transform: skew(25deg, -10deg);`



`transform: rotate(45deg);`

# Transformations in Three Dimensions

Figure 8-18

CSS3 3D transformation functions

Transform Function	Description
<code>translate3d(offX, offY, offZ)</code>	Moves the object offX pixels horizontally, offY pixels vertically, and offZ pixels along the z-axis
<code>translateZ(offZ)</code>	Moves the object offZ pixels along the z-axis
<code>rotate3d(x, y, z, angle)</code>	Rotates the object around the three-dimensional vector (x, y, z) at an angle of angle
<code>rotateZ(angle)</code>	Rotates the object around the z-axis at an angle of angle
<code>scale3d(x, y, z)</code>	Resizes the object by a factor of x horizontally, a factor of y vertically, and a factor of z along the z-axis
<code>scaleZ(z)</code>	Resizes the object by a factor of z along the z-axis
<code>perspective(p)</code>	Sets the size of the perspective effect to p
<code>matrix3d(n, n, ..., n)</code>	Applies a 3D transformation based on a matrix of 16 values

# Applying a Color Gradient

---

- To create a linear gradient, apply the function  
`linear-gradient(position ||  
angle, color-stop, color-stop, ...)`  
where `position` is the starting point of  
the gradient using the keywords `left`, `right`,  
`top`, and `bottom`; `angle` is the angle of the  
gradient; and `color-stop` is the position and  
color of each color (entered as `color  
position`).

# Applying a Color Gradient

---

- To create a radial gradient, apply the function  
`radial-gradient(center, shape  
size, color-stop, color-stop, ...)`  
where `center` is the position of the radial  
gradient's center, `shape` is the gradient's  
shape, `size` is the size of the gradient, `color`  
is the initial color at the center, and `color-`  
`stop` is a color and its position within the radial  
gradient.



# Applying a Color Gradient

---

- To create a linear gradient in WebKit, apply the function

```
-webkit-gradient(linear, start,  
stop, from(color), colorstop(percent,  
color), to(color))
```

where `start` provides the starting location of the gradient, `stop` defines the gradient's stopping location, `color` is a color value or color name, and the `color-stop()` function identifies the location and color of an intermediate color in the gradient.

# Applying a Color Gradient

---

- To create a radial gradient in WebKit, apply the function

```
-webkit-gradient(radial, inner-center, inner-radius, outer-center, outer-radius, from(color), color-stop(percent, color), to(color))
```

where `inner-center` and `outer-center` set the centers of the first and last color in the color list, respectively; `inner-radius` sets the endpoint of the first color; and `outer-radius` sets the starting point of the last color.

# Applying a Color Gradient

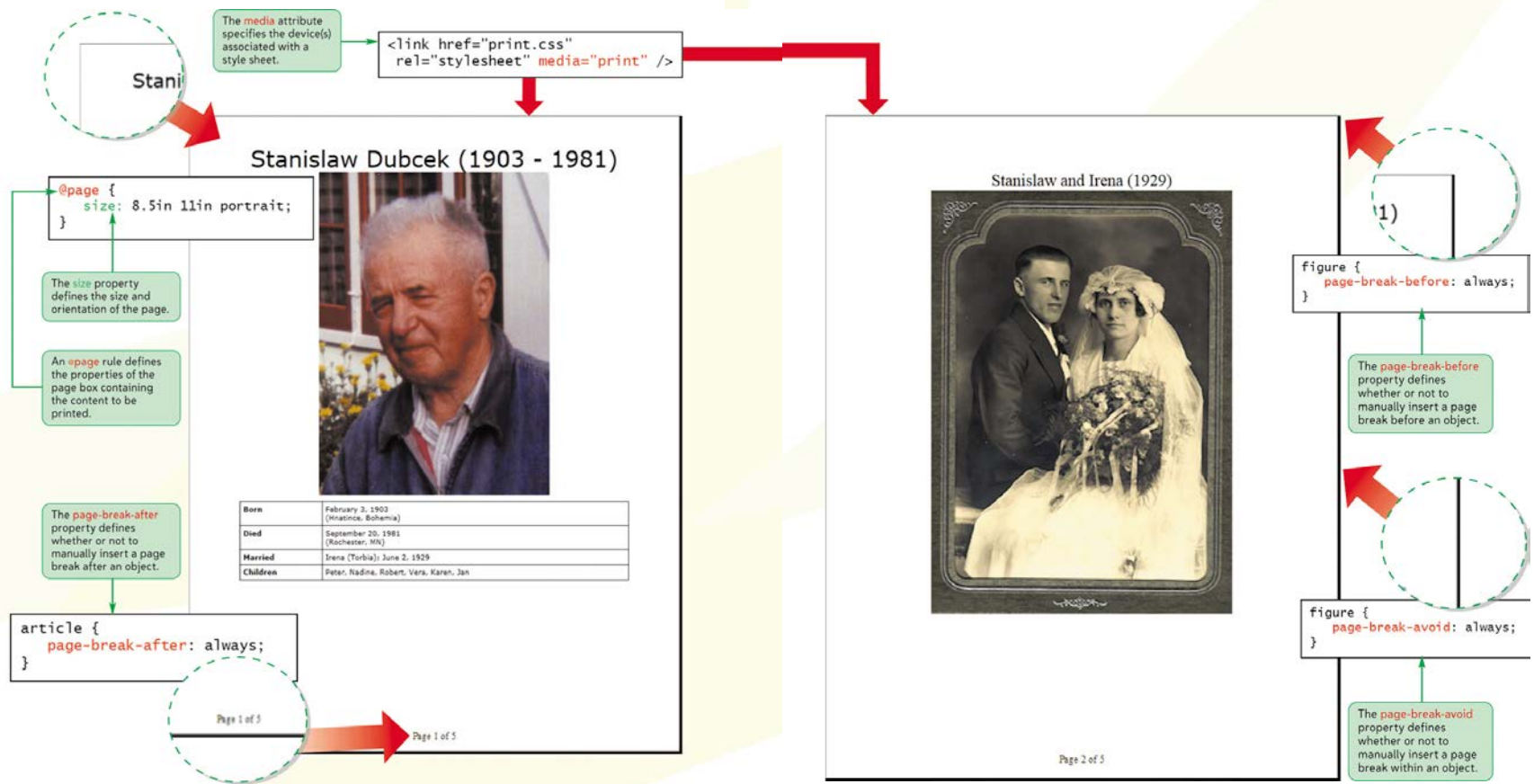
---

- To create a linear gradient in Internet Explorer, apply the filter

```
Gradient ( gradientType=type ,  
startColorStr=#rrggbb ,  
endColorStr=#rrggbb )
```

where `type` is either 0 for a vertical gradient or 1 for a horizontal gradient, `startColorStr` indicates the starting color, and `endColorStr` indicates the ending color

# Designing for Printed Media



# Applying a Border Image

---

- To use a graphic image as an element border, apply the style

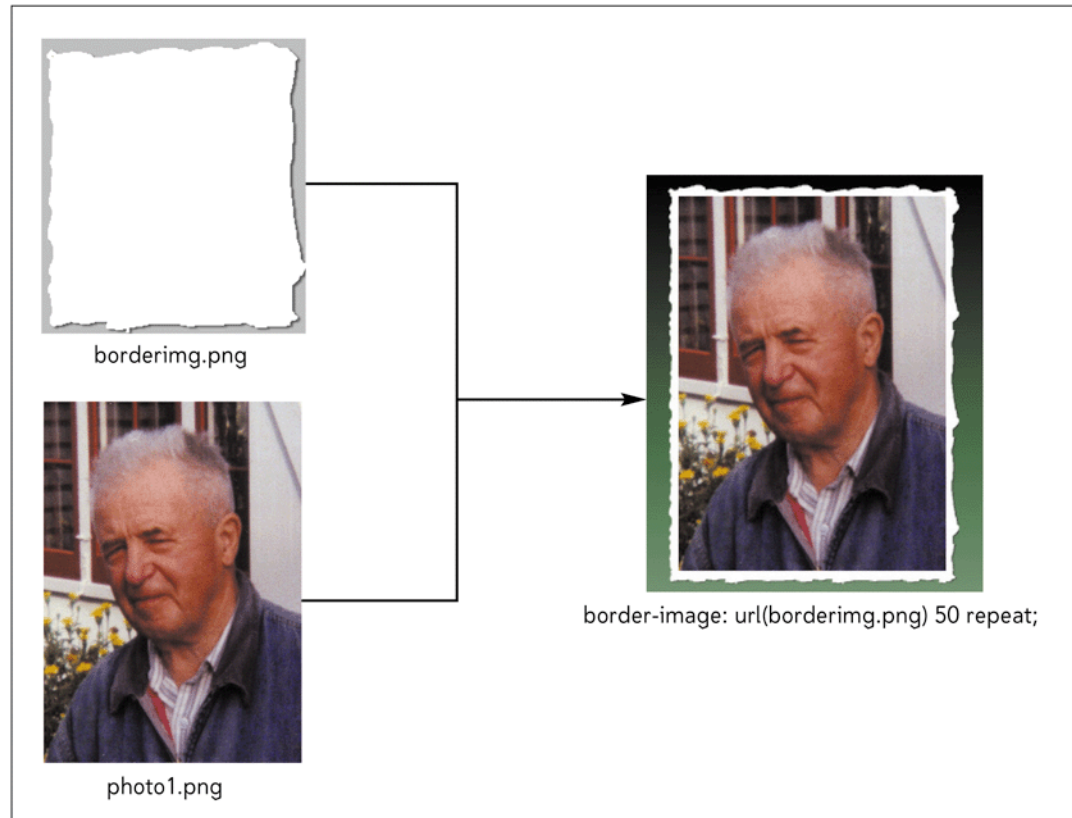
```
border-image: url(url) slice  
repeat;
```

where `url` is the source of the border image file, `slice` is the size of the slice cut from the border image file, and `repeat` indicates whether the border slices should be stretched to cover the object's four sides or tiled

# Applying a Border Image

Figure 8-32

Torn paper border image around the main photo



# Creating Semi-Transparent Objects

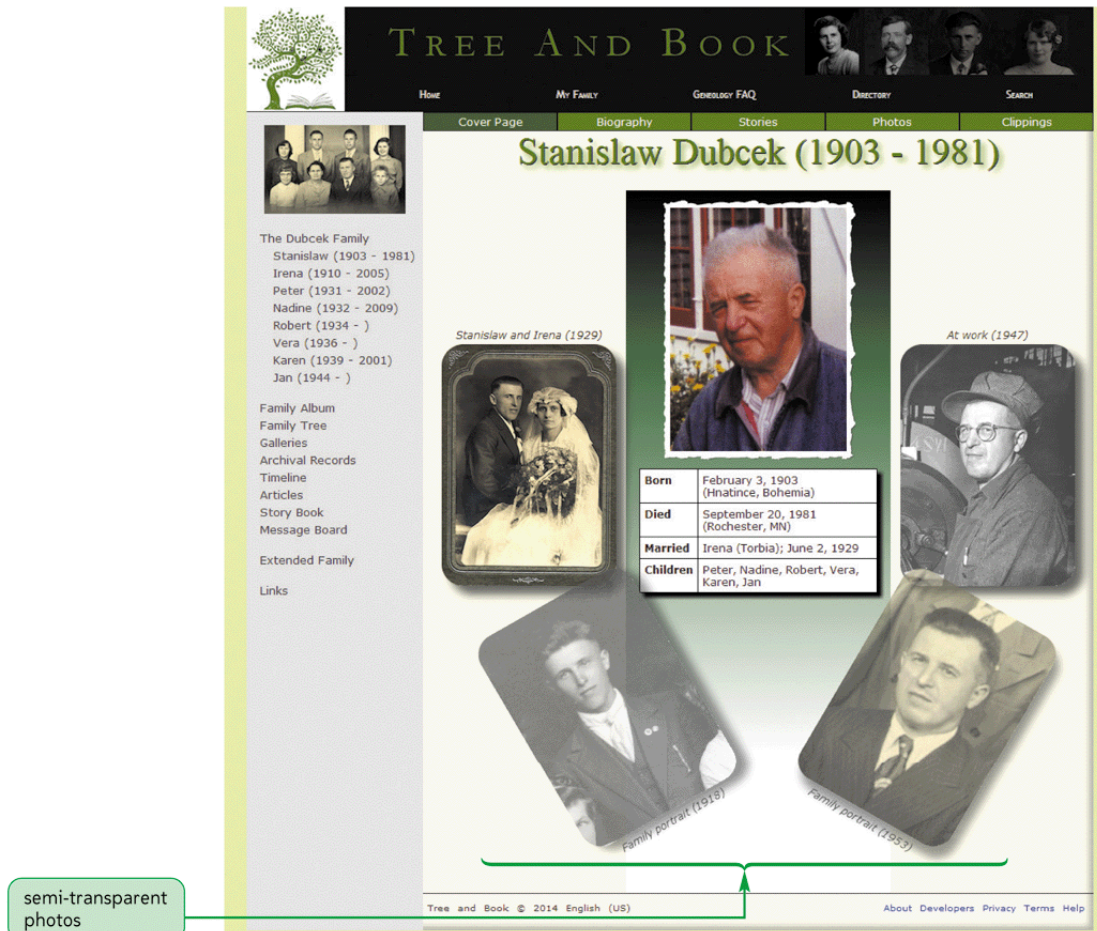
---

- To create a semi-transparent object, apply the style `opacity: value;`  
where `value` ranges from 0 (completely transparent) up to 1 (completely opaque).
- To create a semi-transparent object under Internet Explorer version 8 or earlier, apply the filter `Alpha ( opacity=value )`  
where `value` ranges from 0 (completely transparent) up to 100 (completely opaque).



# Creating Semi-Transparent Objects

Figure 8-34 The final Tree and Book page as viewed in Google Chrome



sabri deniz kizil/Shutterstock.com  
okicoki/Shutterstock.com



# Working with Different Media Devices

---

- To create a style sheet for a specific media device, add the attribute

`media = "devices"`

to either the `link` element or the `style` element, where `devices` is one or more of `braille`, `embossed`, `handheld`, `print`, `projection`, `screen`, `speech`, `tty`, `tv`, or `all`. If you don't specify a media device, the style sheet applies to all devices. Multiple media types should be entered in a comma-separated list

# Working with Different Media Devices

---

- To create a style for specific media from within a style sheet, add the rule

`@media devices {styles}`

where *styles* is the style rules that are applied to the different page elements displayed by those media devices

# Using Print Styles

---

- To define a page box for a printout that indicates the page size, margins, and orientation, use the style rule

```
@page {styles}
```

where `styles` is the styles that define the page.

# Using Print Styles

---

- To set the page size and orientation, use the style property

```
size: width height  
orientation;
```

where `width` and `height` are the width and height of the page, and `orientation` is the orientation of the page (`portrait` or `landscape`).

# Setting Page Breaks

---

- To insert a page break before an element, use the style property

`page-break-before: type;`

where `type` is to always place a page break, `avoid` to never place a page break, `left` to force a page break where the succeeding page will be a left page, `right` to force a page break where the succeeding page will be a right page, `auto` to allow browsers to determine whether or not to insert a page break, or `inherit` to inherit the page break style of the parent element.

# Setting Page Breaks

---

- To insert a page break after an element, use the property

`page-break-after: type;`

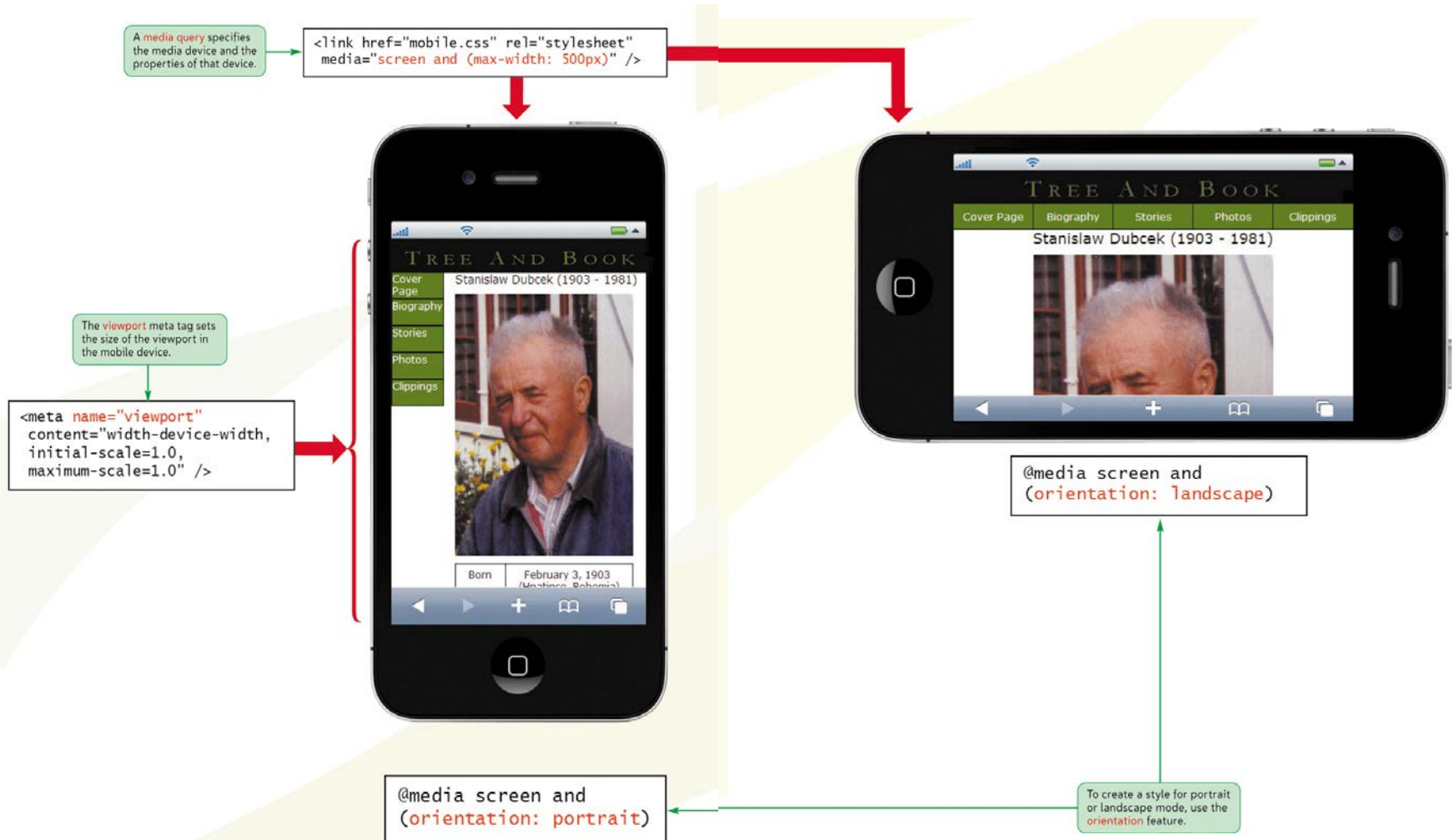
where `type` has the same values as the `page-break-before` style.

- To apply a page break inside an element, use the property

`page-break-inside: type;`

where `type` is `auto`, `inherit`, or `avoid`.

# Designing for the Mobile Web



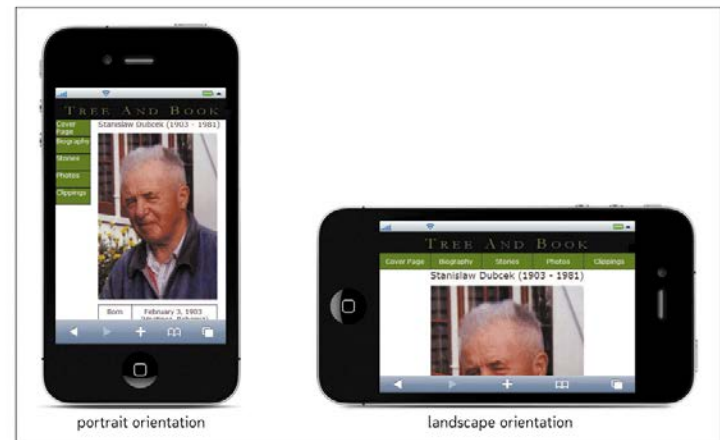
# Designing for the Mobile Web

Figure 8-46 The Tree and Book page as it appears on a mobile device



sabri deniz kizil/Shutterstock.com  
okicoki/Shutterstock.com

Figure 8-47 Preview of Kevin's proposed mobile design



portrait orientation

landscape orientation



# Testing a Mobile Design

**Figure 8-48** Mobile SDKs, emulators, and simulators

Emulator and Simulator	Description	URL
Adobe Device Central	A suite of emulators for a variety of mobile devices	<a href="http://www.adobe.com/products/devicecentral.html">www.adobe.com/products/devicecentral.html</a>
Android SDK	SDK for Android development (Mac OS X, Windows, Linux)	<a href="http://developer.android.com/sdk">developer.android.com/sdk</a>
Firefox Mobile Emulator	Emulator for mobile version of Firefox, also known as Fennec, developed for Nokia and Android	<a href="http://developer.mozilla.org/En/Mobile">developer.mozilla.org/En/Mobile</a>
iPhone SDK	SDK and emulator for the iPhone (Mac OS X only)	<a href="http://developer.apple.com/">developer.apple.com/</a>
iPhoney	iPhone simulator that allows developers to test their Web sites on a 320 × 480 pixel screen	<a href="http://www.marketcircle.com/iphoney/">www.marketcircle.com/iphoney/</a>
iPad Emulator	Online emulator for iPad developers	<a href="http://www.ipad-emulator.com">www.ipad-emulator.com</a>
MobiOne Studio	iPhone emulator for Windows (free trial period with option to purchase)	<a href="http://www.genuitec.com/mobile/">www.genuitec.com/mobile/</a>
Opera Mini Simulator	Java applet that simulates the Opera mobile browser	<a href="http://www.opera.com/mini/demo">www.opera.com/mini/demo</a>
Opera Mobile Emulator	Emulator for mobile version of Opera	<a href="http://www.opera.com/developer/tools">www.opera.com/developer/tools</a>
HP webOS SDK	Emulator for webOS devices	<a href="http://developer.palm.com">developer.palm.com</a>
Windows Phone Emulator	Developer tools for the Windows Phone	<a href="http://create.msdn.com/en-US/">create.msdn.com/en-US/</a>

# Configuring the Viewport

- Mobile devices can display pages written for the larger screens found on desktop computers
- The contents of a Web page are displayed within a window known as the **viewport**
  - **Visual viewport**
  - **Layout viewport**

Figure 8-50 Adding the viewport meta element

```
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width,
                             initial-scale=1.0, maximum-scale=1.0" />

<title>Tree and Book</title>
<script src="modernizr-1.5.js"></script>
```

# Introducing Media Queries

---

- To create a media query for loading a style sheet, add the media attribute

```
media = "devices and|or  
(features)"
```

to the `link` element, where `devices` is a list of media devices, and `features` is a list of display features and their values as found on those devices

# Introducing Media Queries

---

- To apply a media query to a collection of style rules, apply the `@media` rule

```
@media devices and|or (features) {  
    styles  
}
```

in your style sheet, where `styles` is those styles applied to the specified devices and features.

- To import a style sheet based on a media query, apply the following `@import` rule:

```
@import url(url) devices and|or  
(features)
```

# Introducing Media Queries

---

- To associate a style sheet with screen devices that are less than or equal to a specific width, use the query

`media = "screen and (max-width: value)"`

where `value` is the maximum allowable width of the screen's viewport

- To associate a style sheet with screen devices that are greater than or equal to a specific width, use the query

`media = "screen and (min-width: value)"`

where `value` is the minimum allowable width of the screen's viewport.

---

# Introducing Media Queries

---

- To associate a style sheet with screen devices that fall within a range of screen widths, use the following query:

```
media = "screen and (min-width:  
value and max-width: value)"
```

- To associate a style sheet with screen devices in portrait or landscape mode, use the query

```
media = "screen and (orientation:  
type)"
```

where `type` is either `portrait` or `landscape`.

# Creating a Mobile Style Sheet

Figure 8-55

Initial mobile styles

```
/* Hide page elements that will not be displayed */
header h1, header nav, section nav.vertical, footer {
    display: none;
}

/* Header styles */
header {
    background: rgb(18, 15, 12) url(tblogo_sm.png) center center no-repeat;

    -o-background-size: contain;
    -moz-background-size: contain;
    -webkit-background-size: contain;
    background-size: contain;

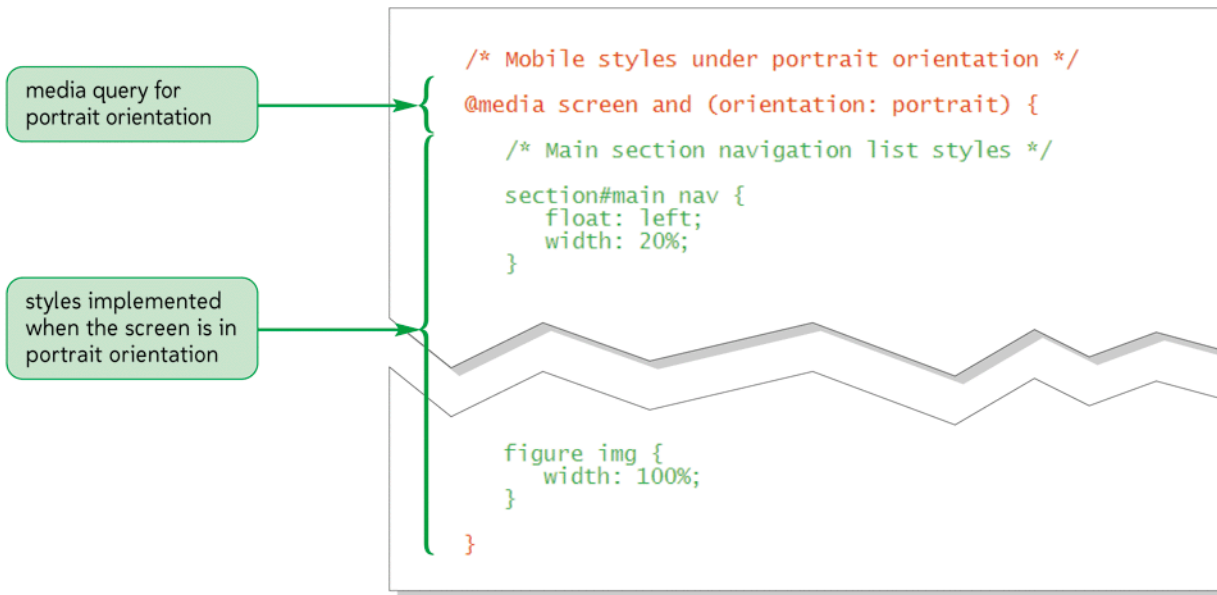
    height: 50px;
    width: 100%;
}

/* Figure box caption styles */
figure figcaption {
    font-size: 12px;
    text-align: center;
}
```

# Designing for Portrait Orientation

Figure 8-57

Inserting an @media rule for screens in portrait orientation





# Designing for Landscape Orientation

Figure 8-59

Inserting an @media rule for screens in landscape orientation

