Safari CSS Reference

Apple Applications > Safari

2008-11-19

Contents

5

Supported CSS Rules   119

7

CSS Property Functions   123

Document Revision History   131

Introduction to Safari CSS Reference

All Safari web browsers use the WebKit engine to display webpages. WebKit is an open source framework in Mac OS X that lets developers embed web browser functionality into applications. In providing this browser functionality, WebKit implements a number of extensions to HTML, CSS, and JavaScript. This document covers support of cascading style sheets (CSS) in WebKit.

You can use Cascading Style Sheets (CSS) in conjunction with HTML-based web content to fine-tune the style of the content. The goal of CSS is to separate the *structure* provided by HTML from the *style* provided by CSS. Taking style information out of the structure allows designers to independently tune a page's style for a variety of audiences and readers (such as desktops, hand-held devices like iPhone, and text-based browsers).

You should read this document if you are creating web content for any version of Safari or any other WebKit-based application such as Dashboard.

Organization of This Document

This document contains the following articles:

■ "Explanation of Terms" (page 11) explains terminology used in this reference.

■ "Supported CSS Properties" (page 15) describes all of the CSS properties supported by Safari and provides information about which web standards (as defined by the World Wide Web Consortium, or W3C) include those properties, where applicable.

■ "Supported CSS Rules" (page 119) describes CSS rules supported by Safari, including media rules, downloadable font rules, and so on.

■ "CSS Property Functions" (page 123) describes functions used by CSS transform properties.

Explanation of Terms

This reference uses CSS-specific terminology in its headings within a property description. This article describes these terms and explains their meanings and possible values.

Syntax

Syntax describes the syntax of a CSS property. If a property can have multiple forms, each form appears in its own line.

Types Allowed

Types Allowed contains information about what numeric types are allowed in a given property. This field is omitted from properties that do not have a single numeric type, such as properties that take multiple values or use nonnumeric constants exclusively.

The Types Allowed field does not present the complete story, however, because a property may also take additional types specific to its subproperties.

Constants

Constants contains a list of special nonnumeric values that you can assign to a specific property. For example, the `border-width` property can take the value `caption`.

The Constants field does not present the complete story, however, because a property may also take additional values specific to its subproperties.

Subproperties

Subproperties provides a list of properties that make up a larger property.

There are three basic types of properties: simple properties, convenience properties, and composite properties.

- Simple properties take a single value of a single type (or a single named constant, such as `thin`).
- Convenience properties (`border-width`, for example) combine multiple properties of the same type into a single value. Many also allow you to optionally set distinct values for each of the included properties.
- Composite properties (`border`, for example) take multiple values of different types.

Convenience properties, such as `border-width`, have related subproperties with finer granularity. For example, instead of setting the `border-width` property, you could set the `border-bottom-width`, `border-top-width`, `border-left-width`, and `border-right-width` properties to the same value and achieve the same result.

Because these convenience properties can be broken down into subproperties of the same basic type, any value that is legal for all of the subproperties is also legal for the convenience property as a whole if the property has a single-value form, and for the individual parts if the property has a multiple-value form. For example, the `border-width` property can accept the value `thin` even though it is listed only in related subproperties such as `border-bottom-width`. Similarly, you could use a multiple-value form, such as `border-width: thin thin thin thin`.

Composite properties also have related subproperties. For example, the second parameter in the `border` property is equivalent to the `border-width` property. Thus, any value that is appropriate for the `border-width` property is also appropriate for the width portion of the `border` composite property.

Similarly, the types allowed for a subproperty are also allowed for convenience properties and the appropriate portions of composite properties that contain them.

Availability

Availability gives the version of Safari in which the property first appeared.

Support Level

Support Level specifies the revision of the W3C standard in which a given property is defined, where applicable, and provides the overall status of the property for properties that are not part of a W3C standard. The possible values are:

- CSS 2.1—properties in CSS 2.1 and earlier revisions. These properties are fully supported across most major browsers, including Safari and other applications that use WebKit.

- Stable CSS 3—properties new in CSS version 3 but believed to be stable. Apple is committed to supporting these properties going forward and does not expect their syntax or names to change. Unlike more experimental CSS 3 properties, these properties are not prefixed by `-webkit-`, and many are supported by other browsers such as Internet Explorer or Firefox.

- Apple extension—properties defined by Apple. These properties are fully supported by WebKit and Safari. A few of these extensions, where noted, are specific to Dashboard widgets or Safari on iPhone. Some of these extensions may have been submitted to the W3C CSS working group for standardization, but are not currently part of the latest draft standard.

- Experimental CSS 3—properties new in CSS 3. The syntax for these properties may change, but because they are prefixed by `-webkit-`, Apple believes that the current syntax can be supported going forward. You still need to update these properties to the final CSS 3 syntax (as needed) and remove the `-webkit-` prefix if you need to use them on other browsers.

- Under development—new CSS 3 or Apple extension properties that are likely to change in syntax. Although it is OK to use these, support for these properties may change in incompatible ways in the future.

Properties may be upgraded to more stable categories as time passes, particularly as tags are added to relevant standards. Go to http://webkit.org/specs for current W3C proposals.

Supported CSS Properties

Safari and WebKit implement a large subset of the CSS 2.1 Specification defined by the World Wide Web Consortium (W3C), along with portions of the CSS 3 Specification. This reference describes the supported properties and provides Safari availability information. If a property is not listed here, it is not implemented by Safari and WebKit.

The CSS attributes in this article are divided according to the groups defined by the W3C CSS Specification:

- "Box Model" (page 15) describes properties specific to the bounding boxes of block elements, including borders, padding, and margins. Additional box-related properties specific to tables are described separately in "Tables" (page 98).

- "Visual Formatting Model" (page 40) describes properties that set the position and size of block elements.

- "Visual Effects" (page 50) describes properties that adjust the visual presentation of block elements, including overflow behavior, resizing behavior, visibility, animation, transforms, and transitions.

- "Generated Content, Automatic Numbering, and Lists" (page 66) describes properties that allow you to change the contents of an element, create automatically numbered sections and headings, and manipulate the style of list elements.

- "Paged Media" (page 70) describes properties associated with controlling appearance attributes specific to printed versions of a web page, such as page break behavior.

- "Colors and Backgrounds" (page 72) describes properties that control the background of block-level elements and the color of text content within elements.

- "Fonts" (page 78) describes properties specific to font selection for text within an element. It also describes properties used in downloadable font definitions.

- "Text" (page 83) describes properties specific to text styles, spacing, and automatic scrolling (marquee).

- "Tables" (page 98) describes layout and styling properties specific to table elements.

- "User Interface" (page 107) describes properties that relate to user interface elements in the browser, such as scrolling text areas, scroll bars, and so on. It also describes properties that are outside the scope of the page content, such as cursor style and the callout shown when you touch and hold a touch target such as a link on iPhone.

Box Model

border

Defines a variety of border properties for an element within one declaration.

Syntax

```
border: border_width  border_style  border_color;
```

Parameters

> *border_width*
>
>> The width of the border on all sides.
>
> *border_style*
>
>> The style of the border.
>
> *border_color*
>
>> The color of the border.

Subproperties
> border-color
> border-style
> border-width

Availability
> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
> CSS 2.1.

border-bottom

Defines a variety of properties for an element's bottom border within one declaration.

Syntax
> border-bottom: *border_width border_style border_color*;

Parameters

> *border_width*
>
>> The width of the bottom border.
>
> *border_style*
>
>> The style of the border.
>
> *border_color*
>
>> The color of the border.

Subproperties
> border-bottom-color
> border-bottom-style
> border-bottom-width

Availability
> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
> CSS 2.1.

border-bottom-color

Defines the color of the bottom border of an element.

Syntax

    border-bottom-color: *color*

Parameters

color

> The color of the bottom border.

Constants

    activeborder, activecaption, appworkspace, aqua, background, black, blue, buttonface,
    buttonhighlight, buttonshadow, buttontext, captiontext, fuchsia, gray, graytext, green,
    grey, highlight, highlighttext, inactiveborder, inactivecaption, inactivecaptiontext,
    infobackground, infotext, lime, maroon, match, menu, menutext, navy, olive, orange, purple,
    red, scrollbar, silver, teal, threeddarkshadow, threedface, threedhighlight,
    threedlightshadow, threedshadow, transparent, -webkit-activelink,
    -webkit-focus-ring-color, -webkit-link, -webkit-text, white, window, windowframe,
    windowtext, yellow

Discussion

> Changes to this property can be animated using transitions.

Availability

> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

border-bottom-style

Defines the style of the bottom border of an element.

Syntax

    border-bottom-style: *style*;

Parameters

style

> The style of the bottom border.

Constants

    dashed, dotted, double, groove, hidden, inset, none, outset, ridge, solid

Availability

> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

border-bottom-width

Defines the width of the bottom border of an element.

Syntax

```
border-bottom-width: width;
```

Parameters

width

The width of the bottom border.

Types Allowed

Length units

Constants

```
medium, thick, thin
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

border-color

Defines the color of an element's border.

Syntax

```
border-color: color;
```

Parameters

color

The color of the border.

Subproperties

```
border-bottom-color
border-left-color
border-right-color
border-top-color
```

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

border-left

Defines a variety of properties for an element's left border within one declaration.

Syntax

```
border-left: border_width  border_style  border_color
```

Parameters

border_width

> The width of the left border.

border_style

> The style of the left border.

border_color

> The color of the left border.

Subproperties

```
border-left-color
border-left-style
border-left-width
```

Availability

> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

border-left-color

Defines the color of the left border of an element.

Syntax

```
border-left-color: color;
```

Parameters

color

> The color of the left border.

Constants

```
activeborder, activecaption, appworkspace, aqua, background, black, blue, buttonface,
buttonhighlight, buttonshadow, buttontext, captiontext, fuchsia, gray, graytext, green,
grey, highlight, highlighttext, inactiveborder, inactivecaption, inactivecaptiontext,
infobackground, infotext, lime, maroon, match, menu, menutext, navy, olive, orange, purple,
red, scrollbar, silver, teal, threeddarkshadow, threedface, threedhighlight,
threedlightshadow, threedshadow, transparent, -webkit-activelink,
-webkit-focus-ring-color, -webkit-link, -webkit-text, white, window, windowframe,
windowtext, yellow
```

Discussion

> Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

border-left-style

Defines the style of the left border of an element.

Syntax

```
border-left-style: style;
```

Parameters

style

The style of the left border.

Constants

`dashed`, `dotted`, `double`, `groove`, `hidden`, `inset`, `none`, `outset`, `ridge`, `solid`

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

border-left-width

Defines the width of the left border of an element.

Syntax

```
border-left-width: width;
```

Parameters

width

The width of the left border.

Types Allowed

Length units

Constants

`medium`, `thick`, `thin`

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

border-right

Defines a variety of properties for an element's right border within one declaration.

Syntax

```
border-right: border_width border_style border_color;
```

Parameters

border_width

> The width of the right border.

border_style

> The style of the right border.

border_color

> The color of the right border.

Subproperties

```
border-right-color
border-right-style
border-right-width
```

Availability

> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

border-right-color

Defines the color of the right border of an element.

Syntax

```
border-right-color: color;
```

Parameters

color

> The color of the right border.

Constants

```
activeborder, activecaption, appworkspace, aqua, background, black, blue, buttonface,
buttonhighlight, buttonshadow, buttontext, captiontext, fuchsia, gray, graytext, green,
grey, highlight, highlighttext, inactiveborder, inactivecaption, inactivecaptiontext,
infobackground, infotext, lime, maroon, match, menu, menutext, navy, olive, orange, purple,
red, scrollbar, silver, teal, threeddarkshadow, threedface, threedhighlight,
threedlightshadow, threedshadow, transparent, -webkit-activelink,
-webkit-focus-ring-color, -webkit-link, -webkit-text, white, window, windowframe,
windowtext, yellow
```

Discussion

> Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

border-right-style

Defines the style of the right border of an element.

Syntax

```
border-right-style: style;
```

Parameters

style

The style of the right border.

Constants

```
dashed, dotted, double, groove, hidden, inset, none, outset, ridge, solid
```

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

border-right-width

Defines the width of the right border of an element.

Syntax

```
border-right-width: width;
```

Parameters

width

The width of the right border.

Types Allowed

Length units

Constants

```
medium, thick, thin
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

border-style

Defines the style for an element's border.

Syntax

```
border-style: style;
```

Parameters

style

> The style of the border.

Subproperties

```
border-bottom-style
border-left-style
border-right-style
border-top-style
```

Availability

> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

border-top

Defines a variety of properties for an element's top border within one declaration.

Syntax

```
border-top: border_width border_style border_color;
```

Parameters

border_width

> The width of the top border.

border_style

> The style of the top border.

border_color

> The color of the top border.

Subproperties

```
border-top-color
border-top-style
border-top-width
```

Availability

> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

border-top-color

Defines the color of the top border of an element.

Syntax

```
border-top-color: color;
```

Parameters

color

> The color of the top border.

Constants

```
activeborder, activecaption, appworkspace, aqua, background, black, blue, buttonface,
buttonhighlight, buttonshadow, buttontext, captiontext, fuchsia, gray, graytext, green,
grey, highlight, highlighttext, inactiveborder, inactivecaption, inactivecaptiontext,
infobackground, infotext, lime, maroon, match, menu, menutext, navy, olive, orange, purple,
red, scrollbar, silver, teal, threeddarkshadow, threedface, threedhighlight,
threedlightshadow, threedshadow, transparent, -webkit-activelink,
-webkit-focus-ring-color, -webkit-link, -webkit-text, white, window, windowframe,
windowtext, yellow
```

Discussion

> Changes to this property can be animated using transitions.

Availability

> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

border-top-style

Defines the style of the top border of an element.

Syntax

```
border-top-style: style;
```

Parameters

style

> The style of the top border.

Constants

```
dashed, dotted, double, groove, hidden, inset, none, outset, ridge, solid
```

Availability

> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

border-top-width

Defines the width of the top border of an element.

Syntax

```
border-top-width: width;
```

Parameters

width

> The width of the top border.

Types Allowed
Length units

Constants
`medium`, `thick`, `thin`

Discussion
Changes to this property can be animated using transitions.

Availability
Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
CSS 2.1.

border-width

Defines the width of the border of an element.

Syntax

```
border-width: width;
```

Parameters

width

> The width of the border.

Subproperties

```
border-bottom-width
border-left-width
border-right-width
border-top-width
```

Availability
Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
CSS 2.1.

margin

Defines the width of an element's outer-element margin.

Syntax
```
margin: value;
margin: margin_top margin_right margin_bottom margin_left
```

Parameters

value

> The width of the margin.

margin_top

> The width of the top margin.

margin_right

> The width of the right margin.

margin_bottom

> The width of the bottom margin.

margin_left

> The width of the left margin.

Subproperties
```
margin-bottom
margin-left
margin-right
margin-top
```

Availability
> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
> CSS 2.1.

margin-bottom

Defines the width of the bottom margin of an element.

Syntax
```
margin-bottom: value;
```

Parameters

value

> The width of the bottom margin.

Types Allowed
> Numbers as a percentage, length units

Constants
```
auto
```

Discussion
>Changes to this property can be animated using transitions.

Availability
>Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
>CSS 2.1.

margin-left

Defines the width of the left margin of an element.

Syntax
```
margin-left: value;
```

Parameters

value
>>The width of the left margin.

Types Allowed
>Numbers as a percentage, length units

Constants
```
auto
```

Discussion
>Changes to this property can be animated using transitions.

Availability
>Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
>CSS 2.1.

margin-right

Defines the width of the right margin of an element.

Syntax
```
margin-right: value;
```

Parameters

value
>>The width of the right margin.

Types Allowed
>Numbers as a percentage, length units

Constants
```
auto
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

margin-top

Defines the width of the top margin of an element.

Syntax

```
margin-top: value;
```

Parameters

value

The width of the top margin.

Types Allowed

Numbers as a percentage, length units

Constants

```
auto
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

padding

Defines the width of an element's inner-element padding.

Syntax

```
padding: value;
padding: padding_top padding_right padding_bottom padding_left
```

Parameters

value

The width of the padding on all sides.

padding_top

The width of the top padding.

padding_right

The width of the right padding.

padding_bottom

The width of the bottom padding.

padding_left

The width of the left padding.

Subproperties
```
padding-bottom
padding-left
padding-right
padding-top
```

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

padding-bottom

Defines the width of the bottom padding of an element.

Syntax
```
padding-bottom: value;
```

Parameters

value

The width of the bottom padding.

Types Allowed

Numbers as a percentage, length units

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

padding-left

Defines the width of the left padding of an element.

Syntax

```
padding-left: value;
```

Parameters

value

The width of the left padding.

Types Allowed

Numbers as a percentage, length units

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

padding-right

Defines the width of the right padding of an element.

Syntax

```
padding-right: value;
```

Parameters

value

The width of the right padding.

Types Allowed

Numbers as a percentage, length units

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

padding-top

Defines the width of the top padding of an element.

Syntax

```
padding-top: value;
```

Parameters

value

The width of the top padding.

Types Allowed

Numbers as a percentage, length units

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

-webkit-border-bottom-left-radius

Specifies that the bottom-left corner of a box be rounded with the specified radius.

Syntax

```
-webkit-border-bottom-left-radius: radius;
-webkit-border-bottom-left-radius: horizontal_radius vertical_radius;
```

Parameters

radius

The radius of the rounded corner.

horizontal_radius

The horizontal radius of the rounded corner.

vertical_radius

The vertical radius of the rounded corner.

Types Allowed

Length units

Subproperties

```
-webkit-border-bottom-right-radius
-webkit-border-radius
-webkit-border-top-left-radius
-webkit-border-top-right-radius
```

Discussion

This property takes either one or two parameters. If one parameter is specified, it controls both the horizontal and vertical radii of a quarter ellipse. If two parameters are specified, the first parameter normally represents the horizontal radius and the second parameter represents the remaining radius. (Compatibility note: In Internet Explorer, if writing-mode is specified as `tb-rl`, these parameters are reversed.)

Changes to this property can be animated using transitions.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level
    Experimental CSS 3.

-webkit-border-bottom-right-radius

Specifies that the bottom-right corner of a box be rounded with the specified radius.

Syntax
```
-webkit-border-bottom-right-radius: radius;
-webkit-border-bottom-right-radius: horizontal_radius vertical_radius;
```

Parameters

radius

    The radius of the rounded corner.

horizontal_radius

    The horizontal radius of the rounded corner.

vertical_radius

    The vertical radius of the rounded corner.

Types Allowed
    Length units

Subproperties
```
-webkit-border-bottom-left-radius
-webkit-border-radius
-webkit-border-top-left-radius
-webkit-border-top-right-radius
```

Discussion
    This property takes either one or two parameters. If one parameter is specified, it controls both the horizontal and vertical radii of a quarter ellipse. If two parameters are specified, the first parameter normally represents the horizontal radius and the second parameter represents the remaining radius. (Compatibility note: In Internet Explorer, if writing-mode is specified as `tb-rl`, these parameters are reversed.)

    Changes to this property can be animated using transitions.

Availability
    Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level
    Experimental CSS 3.

-webkit-border-image

Specifies an image as the border for a box.

Syntax
```
border-image: uri top right bottom left x_repeat y_repeat
border-image: uri top right bottom left / border x_repeat y_repeat
```

```
border-image: uri top right bottom left / top_border right_border bottom_border
 left_border x_repeat y_repeat
```

Parameters

uri

      The location of the image.

top

      The distance from the top edge of the image.

right

      The distance from the right edge of the image.

bottom

      The distance from the bottom edge of the image.

left

      The distance from the left edge of the image.

x_repeat

      The horizontal repeat style.

y_repeat

      The vertical repeat style.

border

      The width of the border on all sides.

top_border

      The width of the top border.

right_border

      The width of the right border.

bottom_border

      The width of the bottom border.

left_border

      The width of the left border.

Constants

`repeat`

`round`

`stretch`

Discussion

The specified image is cut into nine pieces according to the length values given. This property applies to any box, including inline elements, but does not apply to table cells if the `border-collapse` property is set to `collapse`.

The first five fields are required. The `uri` field contains the URI for the image. The four inset values that follow represent distances from the top, right, bottom, and left edges of the image. If no unit is specified, they represent actual pixels in the original image (assuming a raster image). If a unit (such

as `px`) is specified, they represent CSS units (which may or may not be the same thing). The values may also be specified as a percentage of the size of the image.

After the required fields, you can optionally include a slash (`/`) followed by a border width field or fields. You can specify all four border widths individually or specify a single value that applies to all four fields. If these values are not the same size as the inset values, the slices of the original image are scaled to fit.

Finally, you can specify a repeat style in each direction. These values affect how the top, bottom, left, right, and center portions are altered to fit the required dimensions, and can be any of the following: `repeat` (tiled), `stretch`, or `round` (the `round` style is like tiling, except that it stretches all nine pieces slightly so that there is no partial tile at the end).

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

Experimental CSS 3.

-webkit-border-radius

Specifies that the corners of a box be rounded with the specified radius.

Syntax

```
-webkit-border-radius: radius;
-webkit-border-radius: horizontal_radius vertical_radius;
```

Parameters

`radius`

The radius of the rounded corners.

`horizontal_radius`

The horizontal radius of the rounded corners.

`vertical_radius`

The vertical radius of the rounded corners.

Types Allowed

Length units

Subproperties

```
-webkit-border-bottom-left-radius
-webkit-border-bottom-right-radius
-webkit-border-top-left-radius
-webkit-border-top-right-radius
```

Discussion

This property takes either one or two parameters. If one parameter is specified, it controls both the horizontal and vertical radii of a quarter ellipse. If two parameters are specified, the first parameter normally represents the horizontal radius and the second parameter represents the remaining radius. (Compatibility note: In Internet Explorer, if writing-mode is specified as `tb-rl`, these parameters are reversed.)

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

Experimental CSS 3.

-webkit-border-top-left-radius

Specifies that the top-left corner of a box be rounded with the specified radius.

Syntax

```
-webkit-border-top-left-radius: radius;
-webkit-border-top-left-radius: horizontal_radius vertical_radius;
```

Parameters

`radius`

The radius of the rounded corner.

`horizontal_radius`

The horizontal radius of the rounded corner.

`vertical_radius`

The vertical radius of the rounded corner.

Types Allowed

Length units

Subproperties

```
-webkit-border-bottom-left-radius
-webkit-border-bottom-right-radius
-webkit-border-radius
-webkit-border-top-right-radius
```

Discussion

This property takes either one or two parameters. If one parameter is specified, it controls both the horizontal and vertical radii of a quarter ellipse. If two parameters are specified, the first parameter normally represents the horizontal radius and the second parameter represents the remaining radius. (Compatibility note: In Internet Explorer, if writing-mode is specified as `tb-rl`, these parameters are reversed.)

Changes to this property can be animated using transitions.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

Experimental CSS 3.

-webkit-border-top-right-radius

Specifies that the top-right corner of a box be rounded with the specified radius.

Syntax

```
-webkit-border-top-right-radius: radius;
-webkit-border-top-right-radius: horizontal_radius vertical_radius;
```

Parameters

radius

The radius of the rounded corner.

horizontal_radius

The horizontal radius of the rounded corner.

vertical_radius

The vertical radius of the rounded corner.

Types Allowed
Length units

Subproperties
`-webkit-border-bottom-left-radius`
`-webkit-border-bottom-right-radius`
`-webkit-border-radius`
`-webkit-border-top-left-radius`

Discussion

This property takes either one or two parameters. If one parameter is specified, it controls both the horizontal and vertical radii of a quarter ellipse. If two parameters are specified, the first parameter normally represents the horizontal radius and the second parameter represents the remaining radius. (Compatibility note: In Internet Explorer, if writing-mode is specified as `tb-rl`, these parameters are reversed.)

Changes to this property can be animated using transitions.

Availability
Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level
Experimental CSS 3.

-webkit-box-sizing

Specifies that the size of a box be measured according to either its content (default) or its total size including borders.

Syntax
`-webkit-box-sizing: `*sizing_model*`;`

Parameters

sizing_model

The model by which the size of the box is measured.

Constants
`border-box`, `content-box`

Availability
Available in Safari 3.0 and later. Available in iPhone OS 1.1 and later. (Called `box-sizing` in iPhone OS 1.0.)

Support Level

Experimental CSS 3.

-webkit-box-shadow

Applies a drop shadow effect to the border box of an object.

Syntax

`-webkit-box-shadow: hoff voff blur color;`

Parameters

`hoff`

The horizontal offset of the shadow.

`voff`

The vertical offset of the shadow.

`blur`

The blur radius of the shadow.

`color`

The color of the shadow.

Constants

`none`

Discussion

This property takes four parameters. The first two are horizontal and vertical offsets—down for horizontal, and to the right for vertical. The third value is a blur radius. The fourth value is the color of the shadow. Changes to this property can be animated using transitions.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Experimental CSS 3.

-webkit-margin-bottom-collapse

Specifies the behavior of an element's bottom margin if it is adjacent to an element with a margin. Elements can maintain their respective margins or share a single margin between them.

Syntax

`-webkit-margin-bottom-collapse: collapse_behavior;`

Parameters

`collapse_behavior`

The behavior of the bottom margin.

Constants

`collapse, discard, separate`

Discussion

This property allows you to emulate the behavior of some browsers in quirks mode where table cell margins are collapsed into the borders of vertically adjacent cells.

Availability

Available in Safari 3.0 and later. (Called `-khtml-margin-bottom-collapse` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Apple extension.

-webkit-margin-collapse

Specifies the behavior of an element's vertical margins if it is adjacent to an element with a margin. Elements can maintain their respective margins or share a single margin between them.

Syntax

`-webkit-margin-collapse:` *collapse_behavior*`;`

Parameters

collapse_behavior

The behavior of the vertical margins.

Subproperties

`-webkit-margin-bottom-collapse`
`-webkit-margin-top-collapse`

Discussion

This property allows you to emulate the behavior of some browsers in quirks mode where table cell margins are collapsed into the borders of vertically adjacent cells.

Availability

Available in Safari 3.0 and later. (Called `-khtml-magin-collapse` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Apple extension.

-webkit-margin-start

Provides the width of the starting margin.

Syntax

`-webkit-margin-start:` *width*`;`

Parameters

width

The width of the starting margin.

Types Allowed

Numbers as a percentage, length units

Constants

> `auto`

Discussion

> If the writing direction is left-to-right, this value overrides `margin-left`. If the writing direction is right-to-left, this value overrides `margin-right`.

Availability

> Available in Safari 3.0 and later. (Called it is `-khtml-margin-start` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

> Apple extension.

-webkit-margin-top-collapse

Specifies the behavior of an element's top margin if it is adjacent to an element with a margin. Elements can maintain their respective margins or share a single margin between them.

Syntax

> `-webkit-margin-top-collapse: `*`collapse_behavior`*`;`

Parameters

> *`collapse_behavior`*
>
> > The behavior of the top margin.

Constants

> `collapse`, `discard`, `separate`

Discussion

> This property allows you to emulate the behavior of some browsers in quirks mode where table cell margins are collapsed into the borders of vertically adjacent cells.

Availability

> Available in Safari 3.0 and later. (Called `-khtml-magin-top-collapse` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

> Apple extension.

-webkit-padding-start

Provides the width of the starting padding.

Syntax

> `-webkit-padding-start: `*`width`*`;`

Parameters

> *`width`*
>
> > The width of the starting padding.

Types Allowed

Numbers as a percentage, length units

Discussion

If the writing direction is left-to-right, this value overrides `padding-left`. If the writing direction is right-to-left, this value overrides `padding-right`.

Availability

Available in Safari 3.0 and later. (Called `-khtml-padding-start` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Apple extension.

Visual Formatting Model

bottom

Defines the location of the bottom edge of the element for both absolute and relative positioning.

Syntax

```
bottom: position;
```

Parameters

position

The location of the bottom edge of the element.

Types Allowed

Numbers as a percentage, length units

Constants

```
auto
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later

Support Level

CSS 2.1.

clear

Defines the sides of an element on which no floating elements are permitted to be displayed.

Syntax

```
clear: value;
```

Parameters

value

The sides of the element on which no floating elements can be displayed.

Constants
`both`, `left`, `none`, `right`

Availability
Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later

Support Level
CSS 2.1.

direction

Sets the direction in which text is rendered.

Syntax
`direction: value;`

Parameters

value

The direction of the text.

Constants
`ltr`, `rtl`

Availability
Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later

Support Level
CSS 2.1.

display

Defines how an element is displayed onscreen.

Syntax
`display: mode;`

Parameters

mode

The display mode.

Constants
`block`, `compact`, `inline`, `inline-block`, `inline-table`, `list-item`, `none`, `run-in`, `table`, `table-caption`, `table-cell`, `table-column`, `table-column-group`, `table-footer-group`, `table-header-group`, `table-row`, `table-row-group`, `-webkit-box`, `-webkit-inline-box`

Availability
Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later

Support Level
>   CSS 2.1.

float

Indicates whether an element (often a graphic) should be pulled out of the normal text flow and floated toward a particular horizontal position within its enclosing element.

Syntax
>   `float: ` *`position`*`;`

Parameters

> *`position`*

>>   The position for the element to be floated toward.

Constants
>   `center,left,none,right`

Discussion

>   If `float` is set to `none`, the element is displayed inline wherever it appears within the text flow.

>   If `float` is set to a positional value, the element is laid out as it normally would be within the flow, then is moved as far as possible towards the specified position. If an element is vertically positioned such that it would run into another element that is part of the same float, it stops at the point of contact. Thus, in effect, this causes these floating elements to stack up at the specified horizontal position.

>   If the width of a series of stacked floating elements exceeds the width of the enclosing box, further elements wrap to a new row. You can force an element to always wrap to a new row by setting the `clear` property on that element. (See "clear" (page 40) for more information.)

> **Note:** With the exception of elements with intrinsic width (an `img` tag, for example), you should always set the width property on floating elements to ensure consistent behavior across browsers.

Availability
>   Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later

Support Level
>   CSS 2.1.

height

Defines the height of a structural element.

Syntax
>   `height: ` *`value`*`;`

Parameters

> *`value`*

>>   The height of the element.

Types Allowed

Numbers as a percentage, length units, nonnegative values

Constants

`auto`, `intrinsic`, `min-intrinsic`

Discussion

This property has no effect on inline elements. Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later

Support Level

CSS 2.1.

left

Defines the location of the left edge of the element for both absolute and relative positioning.

Syntax

`left: position;`

Parameters

position

The location of the left edge of the element.

Types Allowed

Numbers as a percentage, length units

Constants

`auto`

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later

Support Level

CSS 2.1.

line-height

Defines the vertical interline spacing of lines within the text of an element.

Syntax

`line-height: height;`

Parameters

height

The interline spacing value.

Types Allowed
> Floating-point numbers, Numbers as a percentage, length units

Constants
> `normal`

Discussion
> Changes to this property can be animated using transitions.

Availability
> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later

Support Level
> CSS 2.1.

max-height

Defines the maximum height of a structural element.

Syntax
> `max-height: height;`

Parameters

> `height`
>> The maximum height.

Constants
> `intrinsic,min-intrinsic,none`

Availability
> Available in Safari 1.3 and later. (Positioned elements require Safari 2.0.2 and later.) Available in iPhone OS 1.0 and later.

Support Level
> CSS 2.1.

max-width

Defines the maximum width of a structural element.

Syntax
> `max-width: width;`

Parameters

> `width`
>> The maximum width.

Constants
> `intrinsic,min-intrinsic,none`

Availability
> Available in Safari 1.0 and later. (Positioned elements require Safari 2.0.2 and later.) Available in iPhone OS 1.0 and later.

Support Level
 CSS 2.1.

min-height

Defines the minimum height of a structural element.

Syntax
 `min-height:` *`height`*`;`

Parameters

 `height`

  The minimum height.

Types Allowed
 Numbers as a percentage, length units, nonnegative values

Constants
 `intrinsic,min-intrinsic`

Availability
 Available in Safari 1.3 and later. (Positioned elements require Safari 2.0.2 and later.) Available in iPhone OS 1.0 and later.

Support Level
 CSS 2.1.

min-width

Defines the minimum width of a structural element.

Syntax
 `min-width:` *`width`*`;`

Parameters

 `width`

  The minimum width.

Types Allowed
 Numbers as a percentage, length units, nonnegative values

Constants
 `intrinsic,min-intrinsic`

Availability
 Available in Safari 1.0 and later. (Positioned elements require Safari 2.0.2 and later.) Available in iPhone OS 1.0 and later.

Support Level
 CSS 2.1.

opacity

Specifies how to blend the offscreen rendering into the current composite rendering.

Syntax

```
opacity: value;
```

Parameters

value

The opacity.

Types Allowed
Floating-point numbers

Discussion
Changes to this property can be animated using transitions.

Availability
Available in Safari 2.0 and later. (Called `-khtml-opacity` in Safari 1.1.) Available in iPhone OS 1.0 and later.

Support Level
CSS 2.1.

position

Specifies how an element is positioned.

Syntax

```
position: positioning_model;
```

Parameters

positioning_model

The positioning model for the element.

Constants
`absolute`, `fixed`, `relative`, `static`

Discussion
This property affects the behavior of positional properties such as `float` and `left`/`right`/`top`/`bottom`.

Availability
Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
CSS 2.1.

right

Defines the location of the right edge of the element for both absolute and relative positioning.

Syntax

```
right: position;
```

Parameters

position

   The location of the right edge of the element.

Types Allowed

Numbers as a percentage, length units

Constants

```
auto
```

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Discussion

Changes to this property can be animated using transitions.

Support Level

CSS 2.1.

top

Defines the location of the top edge of the element for both absolute and relative positioning.

Syntax

```
top: position;
```

Parameters

position

   The opacity.

Types Allowed

The location of the top edge of the element.

Constants

```
auto
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

unicode-bidi

Defines the Unicode bidirectional text algorithm used to display text.

Syntax

```
unicode-bidi: algorithm;
```

Parameters

algorithm

The bidirectional text algorithm.

Constants

`bidi-override`, `embed`, `normal`

Discussion

This property *must* be set if you intend to change the direction of inline text.

Availability

Available in Safari 1.3 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

vertical-align

Defines the vertical alignment of text in an element.

Syntax

```
vertical-align: position;
```

Parameters

position

The vertical alignment of the text.

Types Allowed

Numbers as a percentage, length units

Constants

`baseline`, `bottom`, `middle`, `sub`, `super`, `text-bottom`, `text-top`, `top`, `-webkit-baseline-middle`

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

width

Defines the width of a structural element.

Syntax

```
width: value;
```

Parameters

value

The width of the element.

Types Allowed

Numbers as a percentage, length units, nonnegative values

Constants

`auto, intrinsic, min-intrinsic`

Discussion

This property has no effect on inline elements. Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

z-index

Overrides the default stacking order of elements.

Syntax

`z-index: distance;`

Parameters

distance

The z-index of the element.

Constants

`auto`

Discussion

Formally, the `z-index` property sets the height of an element above the drawing plane (in pixels). Its primary use is to override the default stacking order of elements.

By default, elements are stacked in the order in which they appear within the DOM tree; later elements appear on top of earlier elements. If you set a `z-index` value for an element, that element is displayed on top of all elements with a lower `z-index` value, underneath all elements with a higher `z-index` value, and stacked according to its position in the DOM tree relative to all elements with the same `z-index` value.

By default, elements are assigned a `z-index` value of `auto`, which is equivalent to zero (0).

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

Visual Effects

clip

Defines the clipping region.

Syntax

```
clip: shape;
```

Parameters

shape

> The clipping region.

Constants
```
auto
```

Discussion

> A clipping region is the portion of an element in which its content will be rendered. The default is to render content within the entire element size.
>
> If you do not use the constant `auto`, the value should be in the form of a supported shape (currently limited to `rect`).
>
> For example, `clip: rect(3px 20px 5px 8px);` defines a rectangular clip region with a top edge 3 pixels from the top of the element, a right edge 20 pixels from the left edge of the element, a bottom border 5 pixels from the top of the element, and a left border 8 pixels from the left edge of the element.

Availability

> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

overflow

Defines the treatment of content that overflows the element's bounds.

Syntax

```
overflow: behavior;
```

Parameters

behavior

> The overflow behavior.

Subproperties
```
overflow-x
overflow-y
```

Discussion

> This property allows you to choose the behavior for content that overflows the element bounds, such as providing scroll bars or hiding the overflowed content.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

overflow-x

Defines the treatment of content that overflows the element's horizontal bounds.

Syntax

```
overflow-x: behavior;
```

Parameters

behavior

The overflow behavior.

Constants

`auto`, `hidden`, `overlay`, `scroll`, `visible`, `-webkit-marquee`

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

Stable CSS 3.

overflow-y

Defines the treatment of content that overflows the element's vertical bounds.

Syntax

```
overflow-y: behavior;
```

Parameters

behavior

The overflow behavior.

Constants

`auto`, `hidden`, `overlay`, `scroll`, `visible`, `-webkit-marquee`

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

Stable CSS 3.

resize

Specifies the directions in which resizing is allowed.

Syntax

```
resize: direction;
```

Parameters

direction

> The directions in which resizing is allowed.

Constants

```
auto, both, horizontal, none, vertical
```

Availability

> Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> Stable CSS 3.

visibility

Defines whether or not an element is visible onscreen.

Syntax

```
visibility: value;
```

Constants

```
collapse, hidden, visible
```

Discussion

> Note that elements made invisible using this property still take up space onscreen. Changes to this property can be animated using transitions.

Availability

> Available in Safari 1.0 and later. (All supported except for `collapse`.) Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

-webkit-animation

Combines common animation properties into a single property.

Syntax

```
-webkit-animation: name duration timing_function delay iteration_count direction
[, ... ];
```

Parameters

name

See "-webkit-animation-name" (page 55) for details.

duration

See "-webkit-animation-duration" (page 54) for details.

timing_function

See "-webkit-animation-timing-function" (page 56) for details.

delay

See "-webkit-animation-delay" (page 53) for details.

iteration-count

See "-webkit-animation-iteration-count" (page 54) for details.

direction

See "-webkit-animation-direction" (page 54) for details.

Discussion

Refer to the respective property for details of each property and default values.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-animation-delay

Defines when an animation starts.

Syntax

```
-webkit-animation-delay: time [, ...];
```

Parameters

time

The time to begin executing an animation after it is applied. If `0`, the animation executes as soon as it is applied. If positive, it specifies an offset from the moment the animation is applied, and the animation delays execution by that offset. If negative, the animation executes the moment the property changes but appears to begin at the specified negative offset—that is, begins part-way through the animation. Nonzero values must specify a unit: `s` for seconds, `ms` for milliseconds. The default value is `0`.

Discussion

This property allows an animation to begin execution some time after it is applied.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-animation-direction

Determines whether the animation should play in reverse on alternate iterations.

Syntax

```
-webkit-animation-direction: direction [, ...]
```

Parameters

direction

> The direction to play. The default value is `normal`.

Constants

`normal`

> Play each iteration of the animation in the forward direction.

`alternate`

> Play even-numbered iterations of the animation in the forward direction and odd-numbered iterations in the reverse direction.

> When an animation is played in reverse, the timing functions are also reversed. For example, when played in reverse, an ease-in animation appears as an ease-out animation.

Availability
Available in iPhone OS 2.0 and later.

Support Level
Apple extension.

-webkit-animation-duration

Specifies the length of time that an animation takes to complete one iteration.

Syntax

```
-webkit-animation-duration: time [, ...]
```

Parameters

time

> The duration of an animation. If `0`, the animation iteration is immediate (there is no animation). A negative value is treated as `0`. The default value is `0`.

Availability
Available in iPhone OS 2.0 and later.

Support Level
Apple extension.

-webkit-animation-iteration-count

Specifies the number of times an animation iterates.

Syntax

    -webkit-animation-iteration-count: *number* [, ...]

Parameters

number

> The number of iterations. If `1`, the animation plays from beginning to end once. A value of `infinite` causes the animation to repeat forever. Noninteger values cause the animation to end partway through an interation. Negative values are invalid. The default value is `1`.

Constants

`infinite`

> Repeats the animation forever.

Discussion

This property is often used with a `-webkit-animation-direction` property set to `alternate`, which causes the animation to play in reverse on alternate iterations.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-animation-name

Specifies the name of an animation.

Syntax

    -webkit-animation-name: *name* [, ...]

Parameters

name

> The name of the animation.

> The name is used to select the `-webkit-keyframe` at-rule that provides the keyframes and property values for the animation. If the name does not match any `-webkit-keyframe` at-rule, there are no properties to be animated and the animation is not executed. See "@-webkit-keyframes" (page 121) for a description of this rule.

> If "`none`", no animation is executed even if there is a `-webkit-keyframe` at-rule with that name. Setting this property to "`none`" explicitly disables animations.

> The default value is "`"`".

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-animation-play-state

Determines whether the animation is running or paused.

Syntax

```
-webkit-animation-play-state: play_state [, ...]
```

Parameters

play_state

> The state of an animation.

Constants

`running`

> Plays the animation.

`paused`

> Pauses the animation.

Discussion

A running animation can be paused by setting this property to `paused`. Set this property to `running` to continue running a paused animation. A paused animation continues to display the current value of the animation in a static state. When a paused animation is resumed, it restarts from the current value, not from the beginning of the animation.

The default value is `running`.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-animation-timing-function

Defines how an animation progresses between keyframes.

Syntax

```
-webkit-animation-timing-function: function [, ...]
```

Parameters

function

> The function to apply between keyframes. The default value is `ease`.

Constants

`ease`

> Equivalent to `cubic-bezier(0.25, 0.1, 0.25, 1.0)`.

`linear`

> Equivalent to `cubic-bezier(0.0, 0.0, 1.0, 1.0)`.

`ease-in`

> Equivalent to `cubic-bezier(0.42, 0, 1.0, 1.0)`.

`ease-out`

> Equivalent to `cubic-bezier(0, 0, 0.58, 1.0)`.

`ease-in-out`

> Equivalent to `cubic-bezier(0.42, 0, 0.58, 1.0)`.

Discussion

The timing function is specified using a cubic Bezier curve. Use the constants to specify preset points of the curve or the `cubic-bezier` function to specify your own points. See `cubic-bezier` for a description of the parameters for this function. See "-webkit-animation-timing-function and -webkit-transition-timing-function Functions" (page 123) for additional information about timing functions.

This property applies between keyframes, not over the entire animation. For example, for an `ease-in-out` timing function, an animation eases in at the start of the keyframe and eases out at the end of the keyframe. A `-webkit-animation-timing-function` defined within a keyframe block applies to that keyframe; otherwise, the timing function specified for the animation is used.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-backface-visibility

Determines whether or not a transformed element is visible when it is not facing the screen.

Syntax

```
-webkit-backface-visibility: visibility;
```

Parameters

visibility

> Determines whether or not the back face of a transformed element is visible. The default value is `visible`.

Constants

`visible`

> The element is always visible even when it is not facing the screen.

`hidden`

> The element is invisible if it is not facing the screen.

Discussion

Use this property to specify whether or not an element is visible when it is not facing the screen. For example, if the identity transform is set, an element faces the screen; otherwise, it may face away from the screen. For example, applying a rotation about y of 180 degrees in the absence of any other transforms causes an element to face away from the screen.

This property is useful when you place two elements back to back, as you would do to create a playing card. Without this property, the front and back elements could at times switch places during an animation to flip the card. Another example is creating a box out of six elements whose outside and inside faces can be viewed. This is useful when creating the backdrop for a three-dimensional stage.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-perspective

Gives depth to a scene, causing elements farther away from the viewer to appear smaller.

Syntax

```
-webkit-perspective: value;
```

Parameters

value

The distance in pixels from the viewer's position to the z=0 plane. The default value is `none`.

Constants

`none`

No perspective transform is applied.

Discussion

The `-webkit-perspective` property applies the same transform as the `perspective(<number>)` transform function, except that it applies only to the children of the element, not to the transform on the element itself.

The use of this property with any value other than `none` establishes a stacking context. It also establishes a containing block (somewhat similar to `position:relative`), just as the `-webkit-transform` property does.

This transform alters the effect of other transforms. In the absence of additional transforms, this transform has no effect.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-perspective-origin

Sets the origin of the `-webkit-perspective` property described in "-webkit-perspective" (page 58).

Syntax

`-webkit-perspective-origin:` *`posx posy`*`;`

Parameters

posx

The x-origin as a percentage or value.

posy

The y-origin as a percentage or value.

Constants

`top`

Sets the y-origin to the top of the element's border box.

`center`

Sets the x or y origin to the center of the element's border box. If this constant appears before `left` or `right`, specifies the y-origin. If it appears after `top` or `bottom`, specifies the x-origin. If appears alone, centers both the x and y origin.

`bottom`

Sets the y-origin to the bottom of the element's border box.

`left`

Sets the x-origin to the left side of the border box.

`right`

Sets the x-origin to the right side of the border box.

Discussion

This property effectively sets the x and y position at which the viewer appears to be looking at the children of the element. The default value is `50%` for both x and y coordinates.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transform

Specifies transformations to be applied to an element.

Syntax

`-webkit-transform:` *`function`* `... ;`

Parameters

function

A transform function. Possible values are described in "-webkit-transform Functions" (page 123).

Constants

`none`

Discussion

The `-webkit-transform` property specifies a list of transformations, separated by whitespace, to be applied to an element, such as rotation, scaling, and so on.

The set of transform functions is similar to those allowed by SVG, although there are additional functions to support 3D transformations. If multiple transforms are applied, the transform is generated by performing a matrix concatenation of each transform in the list.

For example, the following `div` element is rotated 45 degrees clockwise:

```
<div style="width: 12em; margin-top: 5em;
            -webkit-transform: rotate(45deg)">...</div>
```

If a list of transforms is provided, the net effect is as if each transform is specified separately in the order provided.

The default value is `none` (no transforms applied).

Changes to this property can be animated using transitions.

Availability

Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transform-origin

Sets the origin for the "-webkit-transform" (page 59) property.

Syntax

```
-webkit-transform-origin: posx
-webkit-transform-origin: posx posy
```

Parameters

posx

The x origin as a percentage or value.

posy

The y origin as a percentage or value.

Constants

`top`

Sets the y origin to the top of the element's border box.

`center`

Sets the x or y origin to the center of the element's border box. If this constant appears before `left` or `right`, specifies the y origin. If this constant appears after `top` or `bottom`, specifies the x origin. If it appears alone, centers both the x and y origin.

`bottom`

Sets the y origin to the bottom of the element's border box.

`left`

Sets the x origin to the left side of the border box.

`right`

Sets the x origin to the right side of the border box.

Discussion

The `-webkit-transform-origin` property establishes the origin for transforms applied to an element with respect to its border box.

The values may be expressed either as a CSS length unit or as a percentage of the element's size. For example, a value of `50% 50%` causes transformations to occur around the element's center. Changing the origin to `100% 0%` causes transformation to occur around the top-right corner of the element. The default value is `50% 50%`.

If only one argument is provided, it is interpreted as the horizontal position.

Availability

Available in Safari 3.1 and Later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transform-origin-x

The x coordinate of the origin for transforms applied to an element with respect to its border box.

Syntax

`-webkit-transform-origin-x: posx`

Parameters

`posx`

The x origin as a percentage or value.

Discussion

The values may be expressed either as a CSS length unit or as a percentage of the element's size. For example, a value of `50% 50%` causes transformations to occur around the element's center. Changing the origin to `100% 0%` causes transformation to occur around the top-right corner of the element. The default value is `50% 50%`.

Changes to this property can be animated using transitions.

Availability

Available in Safari 3.1 and Later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transform-origin-y

The y coordinate of the origin for transforms applied to an element with respect to its border box.

Syntax

```
-webkit-transform-origin-y: posy
```

Parameters

`posy`

The y origin as a percentage or value.

Discussion

The values may be expressed either as a CSS length unit or as a percentage of the element's size. For example, a value of `50% 50%` causes transformations to occur around the element's center. Changing the origin to `100% 0%` causes transformation to occur around the top-right corner of the element. The default value is `50% 50%`.

Changes to this property can be animated using transitions.

Availability

Available in Safari 3.1 and Later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transform-origin-z

The z coordinate of the origin for transforms applied to an element with respect to its border box.

Syntax

```
-webkit-transform-origin-z: posz
```

Parameters

`posz`

The z origin as a percentage or value.

Discussion

The values may be expressed either as a CSS length unit or as a percentage of the element's size. For example, a value of `50% 50%` causes transformations to occur around the element's center. Changing the origin to `100% 0%` causes transformation to occur around the top-right corner of the element. The default value is `50% 50%`.

Changes to this property can be animated using transitions.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transform-style

Defines how nested, transformed elements are rendered in 3D space.

Syntax

`-webkit-transform-style: `*`style`*`;`

Parameters

style

The transform style.

Constants

`flat`

Flatten all children of this element into the 2D plane.

`preserve-3d`

Preserve the 3D perspective.

Discussion

If `-webkit-transform-style` is `flat`, all children of this element are rendered flattened into the 2D plane of the element. Therefore, rotating the element about the x or y axes causes children positioned at positive or negative z positions to appear on the element's plane, rather than in front of or behind it. If `-webkit-transform-style` is `preserve-3d`, this flattening is not performed, so children maintain their position in 3D space.

This flattening takes place at each element, so preserving a hierarchy of elements in 3D space requires that each ancestor in the hierarchy have the value `preserve-3d` for `-webkit-transform-style`. But `-webkit-transform-style` affects only an element's children; the leaf nodes in a hierarchy do not require the `preserve-3d` style.

The default value is `flat`.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transition

Combines
`-webkit-transition-delay`,`-webkit-transition-duration`,`-webkit-transition-property`,and `-webkit-transition-timing-function` into a single property.

Syntax

`-webkit-transition: `*`property duration timing_function delay`*` [, ...]`

Parameters

property

    See "-webkit-transition-property" (page 65) for details.

duration

    See "-webkit-transition-duration" (page 64) for details.

timing_function

    See "-webkit-transition-timing-function" (page 65) for details.

delay

    See "-webkit-transition-delay" (page 64) for details.

Availability

Available in Safari 3.1 and Later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transition-delay

Defines when the transition starts.

Syntax

```
-webkit-transition-delay: time [, ...]
```

Parameters

time

The time to begin executing a transition after it is applied. If 0, the transition executes as soon as the property changes. Otherwise, the value specifies an offset from the moment the property changes, and the transition delays execution by that offset. If the value is negative, the transition executes the moment the property changes but appears to begin at the specified negative offset—that is, begins part-way through the transition. Nonzero values must specify a unit: s for seconds, ms for milliseconds. Negative values are invalid. The default value is 0.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transition-duration

Defines how long the transition from the old value to the new value should take.

Syntax

```
-webkit-transition-duration: time [, ...]
```

Parameters

time

> If 0, the transition is immediate (there is no animation). A negative value is treated as 0. Nonzero values must specify a unit: s for seconds, ms for milliseconds. Negative values are invalid. The default value is 0.

Availability

Available in Safari 3.1 and Later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transition-property

Specifies the name of the CSS property to which the transition is applied.

Syntax

```
-webkit-transition-property: name;
```

Parameters

name

> The name of the transition. You can list multiple properties. Property names should be bare, unquoted names. The default value is all.

Constants

> No transition specified.

all

> The default transition name.

Availability

Available in Safari 3.1 and Later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-transition-timing-function

Specifies how the intermediate values used during a transition are calculated.

Syntax

```
-webkit-transition-timing-function: timing_function [, ...]
```

Parameters

timing_function

> The timing function.

Constants

`ease`

> **Equivalent to** `cubic-bezier(0.25, 0.1, 0.25, 1.0).`

`linear`

> **Equivalent to** `cubic-bezier(0.0, 0.0, 1.0, 1.0).`

`ease-in`

> **Equivalent to** `cubic-bezier(0.42, 0, 1.0, 1.0).`

`ease-out`

> **Equivalent to** `cubic-bezier(0, 0, 0.58, 1.0).`

`ease-in-out`

> **Equivalent to** `cubic-bezier(0.42, 0, 0.58, 1.0).`

Discussion

This property allows for a transition to change speed over its duration. These effects, commonly called easing functions, are mathematical functions that produce a smooth curve.

The timing function is specified using a cubic Bezier curve. Use the constants to specify preset points of the curve or the `cubic-bezier` function to specify your own points. See `cubic-bezier` for a description of the parameters for this function.

The timing function takes as its input the current elapsed percentage of the transition duration and outputs a percentage that determines how close the transition is to its goal state.

The default value is `ease`.

Availability

Available in Safari 3.1 and Later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

Generated Content, Automatic Numbering, and Lists

content

Embeds an arbitrary batch of content (such as a movie or a specially formatted string) to be embedded alongside a CSS property.

Syntax

`content: value;`

Parameters

`value`

> The location of the content.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

counter-increment

Increments a numerical counter for auto-numbering.

Syntax

```
counter-increment: counter_name increment_by;
```

Parameters

counter_name

The name of the counter.

increment_by

The amount by which the counter increments.

Constants

Discussion

This property is commonly used in conjunction with the "content" (page 66) property to create section numbers or other auto-numbered containers. For example:

```
<style>
    p#top {
        counter-reset: section;
    }

    h1:before
    {
        content: "Section " counter(section) " ";
        counter-increment: section 1;
    }
</style>

<p id="top">This resets the counter.</p>

<H1>First section</H1>
<H1>Next section</H1>
```

This snippet inserts "Section 1:" at the beginning of the first heading, "Section 2:" at the beginning of the second, and so on.

> **Important:** You *must* use the "counter-reset" (page 68) property to reset the counter on some element that appears in the DOM tree prior to the first element where you use counter-increment on that counter. Otherwise, this call increments a nonexistent counter and all of your sections will be numbered "Section 1".

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

counter-reset

Resets a counter used by the "counter-increment" (page 67) property and the `counter` function.

Syntax

        counter-reset: *counter_name*

Parameters

counter_name

> The name of the counter.

Constants
        none
Discussion
> For an example of this property, see the documentation for "counter-increment" (page 67).

Availability
> Available in Safari 3.0 and later.

Support Level
> CSS 2.1.

list-style

Defines the display style for a list and list elements.

Syntax
        list-style: *type position image*;

Parameters

type

> The type of list.

position

> The position of the list marker.

image

> The location of an image to be used as the list marker.

Subproperties
        list-style-image
        list-style-position
        list-style-type

Discussion
> As with most composite properties, all arguments are optional.

Availability
> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
> CSS 2.1.

list-style-image

Defines an image to use as the opening symbol of a list element.

Syntax

```
list-style-image: value
```

Parameters

value

> The location of the image.

Availability
> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
> CSS 2.1.

list-style-position

Defines the position of the marker of a list element.

Syntax

```
list-style-position: value
```

Parameters

value

> The position of the marker.

Constants

```
inside
```

> The marker is placed inside the text. Wrapping text appears directly below the marker.

```
outside
```

> The text of the list item is indented from the marker.

Availability
> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
> CSS 2.1.

list-style-type

Defines the type of marker of a list element.

Syntax

```
list-style-type: value
```

Parameters

value

> The type of marker.

Constants

armenian, circle, cjk-ideographic, decimal, decimal-leading-zero, disc, georgian, hebrew, hiragana, hiragana-iroha, katakana, katakana-iroha, lower-alpha, lower-greek, lower-latin, lower-roman, none, square, upper-alpha, upper-latin, upper-roman

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

Paged Media

orphans

Defines the minimum number of lines in a paragraph that must be left at the bottom of a page (before a page break).

Syntax

orphans: *number_of_lines*

Parameters

number_of_lines

> The number of lines.

Types Allowed

Integers

Availability

Available in Safari 1.3 and later.

Support Level

CSS 2.1.

page-break-after

Defines the page break behavior following an element's definition.

Syntax

page-break-after: *behavior*

Parameters

behavior

> The page break behavior.

Constants
       `always, auto, avoid, left, right`

Availability
       Safari 1.2 and later.

Support Level
       CSS 2.1.

page-break-before

Defines the page break behavior before an element's definition.

Syntax
       `page-break-before: behavior`

Parameters

       `behavior`

               The page break behavior.

Constants
       `always, auto, avoid, left, right`

Availability
       Safari 1.2 and later.

Support Level
       CSS 2.1.

page-break-inside

Defines the page break behavior within an element.

Syntax
       `page-break-inside: behavior`

Parameters

       `behavior`

               The page break behavior.

Constants
       `auto, avoid`

Availability
       Safari 1.3 and later.

Support Level
       CSS 2.1.

widows

Defines the minimum number of lines in a paragraph that must be left at the top of a page (after a page break).

Syntax

```
widows: number_of_lines
```

Parameters

number_of_lines

> The number of lines.

Types Allowed
> Integers

Availability
> Safari 1.3 and later.

Support Level
> CSS 2.1.

Colors and Backgrounds

background

Defines a variety of background properties within one declaration.

Syntax

```
background: background_color background_image background_repeat
background_attachment background_position;
```

Parameters

background_color

> The background color.

background_image

> The location of the background image.

background_repeat

> The repeating behavior of the background image.

background_attachment

> If `fixed`, the background image does not move when the page scrolls; if `scroll`, the image moves when the page scrolls.

background_position

> The position of the background image.

Discussion
> As with most composite properties, all arguments are optional.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

background-attachment

Defines the scrolling or fixed nature of the page background.

Syntax

```
background-attachment: behavior
```

Parameters

background_attachment

If `fixed`, the background image does not move when the page scrolls; if `scroll`, the image moves when the page scrolls.

Constants

`scroll`, `fixed`

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

background-color

Defines an element's background color.

Syntax

```
background-color: color
```

Parameters

color

The background color.

Constants

`activeborder`, `activecaption`, `appworkspace`, `aqua`, `background`, `black`, `blue`, `buttonface`, `buttonhighlight`, `buttonshadow`, `buttontext`, `captiontext`, `fuchsia`, `gray`, `graytext`, `green`, `grey`, `highlight`, `highlighttext`, `inactiveborder`, `inactivecaption`, `inactivecaptiontext`, `infobackground`, `infotext`, `lime`, `maroon`, `match`, `menu`, `menutext`, `navy`, `olive`, `orange`, `purple`, `red`, `scrollbar`, `silver`, `teal`, `threeddarkshadow`, `threedface`, `threedhighlight`, `threedlightshadow`, `threedshadow`, `transparent`, `-webkit-activelink`, `-webkit-focus-ring-color`, `-webkit-link`, `-webkit-text`, `white`, `window`, `windowframe`, `windowtext`, `yellow`

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

background-image

Defines an element's background image.

Syntax

```
background-image: value
```

Parameters

value

The location of the image.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

background-position

Defines the origin of a background image.

Syntax

```
background-position: xpos
background-position: xpos ypos
```

Parameters

xpos

The x-coordinate of the origin of the background image.

ypos

The y-coordinate of the origin of the background image.

Discussion

Position can be specified in terms of pixels or percentages of the viewport width or using the keywords `top`, `left`, `center`, `right`, or `bottom`.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

background-position-x

Defines the x-coordinate of the origin of a background image.

Syntax
```
background-position-x: value
```

Parameters

value

The x-coordinate of the origin of the background image.

Availability
Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
Apple extension.

background-position-y

Defines the y-coordinate of the origin of a background image.

Syntax
```
background-position-y: value
```

Parameters

value

The y-coordinate of the origin of the background image.

Availability
Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level
Apple extension.

background-repeat

Defines the repeating qualities of the background image.

Syntax
```
background-repeat: value
```

Parameters

value

The repeating behavior of the background image.

Discussion
This property controls whether tiling of an element's background image should occur in the x direction, the y direction, both, or neither.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

color

Defines the color of the text of an element.

Syntax

```
color: value
```

Parameters

value

The color. Colors can be specified with a constants, an RGB value, or a hexadecimal value.

Constants

```
activeborder, activecaption, appworkspace, aqua, background, black, blue, buttonface,
buttonhighlight, buttonshadow, buttontext, captiontext, fuchsia, gray, graytext, green,
grey, highlight, highlighttext, inactiveborder, inactivecaption, inactivecaptiontext,
infobackground, infotext, lime, maroon, match, menu, menutext, navy, olive, orange, purple,
red, scrollbar, silver, teal, threeddarkshadow, threedface, threedhighlight,
threedlightshadow, threedshadow, transparent, -webkit-activelink,
-webkit-focus-ring-color, -webkit-link, -webkit-text, white, window, windowframe,
windowtext, yellow
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

-webkit-background-clip

Specifies whether the background should extend into the border of a box.

Syntax

```
-webkit-background-clip: behavior
```

Parameters

behavior

The clipping behavior of the background.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

Experimental CSS 3.

-webkit-background-composite

Sets a compositing style for background images and colors.

Syntax

`-webkit-background-composite:` *compositing_style*

Parameters

compositing_style

The compositing style of the background.

Discussion

The default value is `border`, which means that the background extends into the border area. Specifying a value of `padding` limits the background so that it extends only into the padding area enclosed by the border.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

Under development.

-webkit-background-origin

Determines where the background-position property is anchored.

Syntax

`-webkit-background-origin:` *origin*

Parameters

origin

The origin of the background position.

Discussion

The background position can be anchored at the upper-left corner of the border, the upper-left corner of the padding area inside the border, or the upper-left corner of the content inside the padding area.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

Experimental CSS 3.

-webkit-background-size

Overrides the size of a background image.

Syntax
```
-webkit-background-size: length
-webkit-background-size: length_x length_y
```

Parameters

length

> The width and height of the background image.

length_x

> The width of the background image.

length_y

> The height of the background image.

Availability
Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level
Experimental CSS 3.

Fonts

font

Defines a variety of properties for an element's text font within one declaration.

Syntax
```
font: font_style font_variant font_weight font_size / line_height
      font_family
font: ui_style
```

Parameters

font_style

The style of the font.

font_variant

The variant of the font.

font_weight

The weight, or boldness, of the font.

font_size

The size of the font.

line_height

The distance between lines.

font_family

The family of the font.

ui_style

The user interface style to replicate.

Constants

`caption`, `icon`, `menu`, `message-box`, `small-caption`, `status-bar`, `-webkit-control`, `-webkit-mini-control`, `-webkit-small-control`

Discussion

In addition to declaring a font style explicitly by characteristics, you can also specify a user interface style using constants such as `caption`. These constants represent the default font style for the specified user interface element, and as such, their specific values are dependent on the browser, the operating system, and user configuration options.

Using the `font` property resets all related font properties that are not explicitly specified.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

font-family

Defines a list of fonts for element styling or downloadable font definitions.

Syntax

`font-family:` *family* `[, ...]`

Parameters

family

The family of the font.

Discussion

The font-family property has two different meanings, depending on context.

In the context of an element style, it defines a font to use for text within an element. Because not all computers have the same fonts available, this property to specify multiple acceptable fonts in descending order of preference. In addition, contants such as `serif` or `sans-serif` provide generic fallback fonts in case a browser does not have any of the listed fonts available.

In the context of a downloadable font definition, this property provides the name of the font that the font definition describes. In this form, you may specify multiple family names for the font, but generally only a single family name (optionally, specify that it should match against generic font names like `serif`).

For more information about downloadable font definitions, see "@font-face" (page 119).

Availability

Available in Safari 1.0 and later. Downloadable fonts supported in Safari 3.1 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

font-size

Defines the font size for the text in an element or in a downloadable font definition.

Syntax

```
font-size: value
```

Parameters

value

The size of the font.

Types Allowed

Numbers as a percentage, length units

Constants

`large`, `larger`, `medium`, `small`, `smaller`, `-webkit-xxx-large`, `x-large`, `x-small`, `xx-large`, `xx-small`

Discussion

For more information about downloadable font definitions, see "@font-face" (page 119).

Availability

Available in Safari 1.0 and later. Downloadable fonts supported in Safari 3.1 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

font-style

Defines the font style for the text in an element or a downloadable font definition.

Syntax

```
font-style: value
```

Parameters

value

    The style of the font.

Constants

```
italic,normal,oblique
```

Availability

Available in Safari 1.0 and later. Downloadable fonts supported in Safari 3.1 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

font-variant

Defines special font properties for the text in an element or for a downloadable font definition.

Syntax

```
font-variant: value
```

Parameters

value

    The variant of the font.

Constants

```
normal,small-caps
```

Discussion

For more information about downloadable font definitions, see "@font-face" (page 119).

Availability

Available in Safari 1.0 and later. (The value `small-caps` is not supported.) Downloadable fonts supported in Safari 3.1 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

font-weight

Defines the font weight of the text in an element or for a downloadable font definition.

Syntax

```
font-weight: value
```

Parameters

value

The weight, or boldness, of the font.

Types Allowed

Integers, nonnegative values

Constants

`100`, `200`, `300`, `400`, `500`, `600`, `700`, `800`, `900`, `bold`, `bolder`, `lighter`, `normal`

Discussion

For more information about downloadable font definitions, see "@font-face" (page 119).

Availability

Available in Safari 1.0 and later. Downloadable fonts supported in Safari 3.1 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

src

Provides a list of locations for a downloadable font definition.

Syntax

```
src: local("Times New Roman"),
    URL(http://...) format("truetype"),
    URL(http://...), ...
```

Discussion

This property takes a comma-delimited list of font locations which may be locally installed font family names or HTTP URLs.

For more information about downloadable font definitions, see "@font-face" (page 119).

Availability

Available in Safari 3.1 and later.

Support Level

CSS 3.

unicode-range

Describes the unicode characters supported by a downloadable font definition.

Syntax

```
unicode-range: range [, ...];
unicode-range: start_character-end_character [, ...];
```

Parameters

range

    The range of supported characters.

start_character

    The first character in a range of supported characters.

end_character

    The last character in a range of supported characters.

Discussion

This property takes a comma-delimited list of Unicode character ranges. There are two supported formats: singleton ranges and pair ranges.

A singleton range is in the form `U+xxxx` where `xxxx` is a hexadecimal number. For example, the range `U+2150` indicates that Unicode character `0x2150` is supported. Leading zeroes may be omitted, so `U+300` is the same as `U+0300`. The following snippet shows a singleton range: `unicode-range: U+2150;`

A singleton range may also contain wildcards in the form of a question-mark character. For example, `U+36??` contains two wildcard characters. This range matches any value in which the first two digits are `36`, without regard to the value for the last two digits. The following snippet shows a wildcard range that represents the Unicode characters 0x2160 through 0x216f, inclusive: `unicode-range: U+216?;`

A pair range is in the form of a hyphen-separated pair of hexadecimal values in the form `U+xxxx-yyyy` where `xxxx` and `yyyy` are hexadecimal numbers. For example, the following pair range represents the Unicode characters from `0x2164` through `0x2156`, inclusive: `unicode-range: U+2154-2156;`

For more information about downloadable font definitions, see "@font-face" (page 119).

Availability

Available in Safari 3.1 and later.

Support Level

CSS 3.

Text

letter-spacing

Defines the horizontal interletter spacing of characters within the text of an element.

Syntax

    `letter-spacing:` *length*

Parameters

length

    The size of the character spacing.

Types Allowed

Length units

Constants

```
normal
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

text-align

Defines the alignment for inline content within an element.

Syntax

```
text-align: position
```

Parameters

position

The inline content alignment.

Constants

```
center, end, justify, left, right, start, -webkit-auto, -webkit-center, -webkit-left,
-webkit-right
```

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

text-decoration

Defines special styling for text, such as underlines.

Syntax

```
text-decoration: style
```

Parameters

style

The type of decoration.

Constants

```
line-through, none, overline, underline
```

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

text-indent

Defines the amount to indent the first line of text within an element.

Syntax

```
text-indent: length;
```

Parameters

length

The amount to indent.

Types Allowed

Numbers as a percentage, length units

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

text-overflow

Controls overflow of non-wrapped text.

Syntax

```
text-overflow: behavior;
```

Parameters

behavior

The overflow behavior.

Constants

```
clip, ellipsis
```

Discussion

This property controls how Safari displays text that exceeds the specified width of the enclosing paragraph if the "overflow" (page 50) property is set to `hidden` and style rules or `nowrap` tags prevent the text from wrapping (or if a single word is too long to fit by itself).

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 3.

text-shadow

Defines a variety of properties for an element's text shadow within one declaration.

Syntax

```
text-shadow: color x_offset y_offset blur_radius
```

Parameters

`color`

> The color of the shadow.

`x_offset`

> The horizontal offset of the shadow.

`y_offset`

> The vertical offset of the shadow.

`blur_radius`

> The blur radius of the shadow.

Constants

`none`

Discussion

Although the CSS specification allows it, multiple shadows are not supported in Safari. Changes to this property can be animated using transitions.

Availability

Available in Safari 1.1 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

text-transform

Defines a capitalization transformation for the text in an element.

Syntax

`text-transform: transformation`

Parameters

`transformation`

> The capitalization transformation.

Constants

`capitalize`, `lowercase`, `none`, `uppercase`

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

white-space

Defines how whitespace characters in an element are handled onscreen.

Syntax

`white-space: policy`

Parameters

policy

The policy for displaying whitespace in the element.

Constants

`normal`, `nowrap`, `pre`, `pre-line`, `pre-wrap`

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

word-break

Specifies the level of strictness when breaking lines of text in ideographic languages such as Chinese, Japanese, and Korean.

Syntax

`word-break: ` *strictness*

Parameters

strictness

The level of strictness.

Constants

`break-all`, `break-word`, `normal`

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Stable CSS 3.

word-spacing

Defines the amount of space between words.

Syntax

`word-spacing: ` *length*

Parameters

length

The amount of spacing.

Types Allowed

Length units

Constants

`normal`

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

word-wrap

Specifies word-splitting behavior for wrapping lines that are too long for the enclosing box and contain no spaces.

Syntax

```
word-wrap: behavior
```

Parameters

```
behavior
```

The wrapping behavior.

Constants

```
break-word, normal
```

Availability

Available in Safari 2.0 and later. Available in iPhone OS 1.0 and later.

Support Level

Stable CSS 3.

-webkit-marquee

Defines properties for showing content as though displayed on an electronic marquee sign.

Syntax

```
-webkit-marquee: direction increment repetition style speed
```

Parameters

> `direction`
>
>> The direction of the marquee.
>
> `increment`
>
>> The distance the marquee moves in each increment
>
> `repetition`
>
>> The number of times the marquee repeats.
>
> `style`
>
>> The style of the marquee's motion.
>
> `speed`
>
>> The scroll or slide speed of the marquee.

Subproperties

> `-webkit-marquee-direction`
> `-webkit-marquee-increment`
> `-webkit-marquee-repetition`
> `-webkit-marquee-speed`
> `-webkit-marquee-style`

Availability

> Available in Safari 3.0 and later. (Called `-khtml-marquee` in Safari 2.0.) Available in iPhone OS 1.0.

Support Level

> Under development.

-webkit-marquee-direction

Specifies the direction of motion for a marquee box.

Syntax

> `-webkit-marquee-direction:` *direction*

Parameters

> `direction`
>
>> The direction of the marquee.

Constants

> `ahead`, `auto`, `backwards`, `down`, `forwards`, `left`, `reverse`, `right`, `up`

Availability

> Available in Safari 3.0 and later. (Called `-khtml-marquee-direction` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

> Under development.

-webkit-marquee-increment

Defines the distance the marquee moves in each increment.

Syntax

```
-webkit-marquee-increment: distance
```

Parameters

distance

> The distance the marquee moves in each increment

Types Allowed
> Numbers as a percentage, length units

Constants
```
large, medium, small
```

Availability
> Available in Safari 3.0 and later. (Called `-khtml-marquee-increment` in Safari 2.0.) Available in iPhone 1.0 and later.

Support Level
> Under development.

-webkit-marquee-repetition

Specifies the number of times a marquee box repeats (or `infinite`).

Syntax

```
-webkit-marquee-repetition: iterations
```

Parameters

iterations

> The number of times the marquee repeats.

Types Allowed
> Integers, nonnegative values

Constants
```
infinite
```

Discussion
> The marquee box will repeat infinitely if `infinite` is specified.

Availability
> Available in Safari 3.0 and later. (Called `-khtml-marquee-repetition` in Safari 2.0.) Available in iPhone OS 1.0.

Support Level
> Under development.

-webkit-marquee-speed

Defines the scroll or slide speed of a marquee box.

Syntax

```
-webkit-marquee-speed: speed
-webkit-marquee-speed: distance / time
```

Parameters

speed

> The scroll or slide speed of the marquee.

distance

> The distance term in the speed equation.

time

> The time term in the speed equation.

Types Allowed

> Integers, time units, nonnegative values

Constants

```
fast, normal, slow
```

Discussion

> This property can either take one speed parameter (`slow`, for example) or a measure of distance and a measure of time separated by a slash (`/`).

Availability

> Available in Safari 3.0 and later. (Called `-khtml-marquee-speed` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

> Under development.

-webkit-marquee-style

Specifies the style of marquee motion.

Syntax

```
-webkit-marquee-style: style
```

Parameters

style

> The style of the marquee's motion.

Constants

```
alternate, none, scroll, slide
```

Discussion

> The values `scroll` and `slide` both cause the content to start outside the box and move into the box, but if the value `scroll` is specified, the content stops moving once the last content is visible.

The value `alternate` causes the content to shift back and forth within the box in the specified direction.

Availability

Available in Safari 3.0 and later. (Called `-khtml-marquee-style` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Under development.

-webkit-text-fill-color

Specifies a fill color for text.

Syntax

```
-webkit-fill-color: color
```

Parameters

color

The fill color. Colors can be specified with a constant, an RGB value, or a hexadecimal value.

Constants

`activeborder`, `activecaption`, `appworkspace`, `aqua`, `background`, `black`, `blue`, `buttonface`, `buttonhighlight`, `buttonshadow`, `buttontext`, `captiontext`, `fuchsia`, `gray`, `graytext`, `green`, `grey`, `highlight`, `highlighttext`, `inactiveborder`, `inactivecaption`, `inactivecaptiontext`, `infobackground`, `infotext`, `lime`, `maroon`, `match`, `menu`, `menutext`, `navy`, `olive`, `orange`, `purple`, `red`, `scrollbar`, `silver`, `teal`, `threeddarkshadow`, `threedface`, `threedhighlight`, `threedlightshadow`, `threedshadow`, `transparent`, `-webkit-activelink`, `-webkit-focus-ring-color`, `-webkit-link`, `-webkit-text`, `white`, `window`, `windowframe`, `windowtext`, `yellow`

Discussion

If not specified, the color specified by the `color` property is used. `-webkit-fill-color` is commonly used in combination with `-webkit-text-stroke`. Changes to this property can be animated using transitions.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-text-security

Specifies the shape to use in place of letters in a password input field.

Syntax

```
-webkit-text-security: shape
```

Parameters

shape

>> The shape to use in place of letters.

Constants
> `circle`, `disc`, `none`, `square`

Availability
> Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level
> Apple extension.

-webkit-text-size-adjust

Specifies a size adjustment for displaying text content in Safari on iPhone.

Syntax
> `-webkit-text-size-adjust:` *percentage*

Parameters

percentage

>> The size at which to display text in Safari on iPhone.

Constants
> `auto`, `none`

Availability
> Available in iPhone OS 1.0 and later.

Support Level
> Apple extension—iPhone OS only.

-webkit-text-stroke

Specifies the width and color of the outline (stroke) of text.

Syntax
> `-webkit-text-stroke:` *width color*

Parameters

width

>> The width of the stroke.

color

>> The color of the stroke.

Subproperties
> `-webkit-text-stroke-color`
> `-webkit-text-stroke-width`

Discussion

This property is commonly used in combination with `-webkit-text-fill-color`.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-text-stroke-color

Specifies the color of the outline (stroke) of text.

Syntax

`-webkit-text-stroke-color:` *color*

Parameters

color

The color of the stroke.

Constants

`activeborder`, `activecaption`, `appworkspace`, `aqua`, `background`, `black`, `blue`, `buttonface`, `buttonhighlight`, `buttonshadow`, `buttontext`, `captiontext`, `fuchsia`, `gray`, `graytext`, `green`, `grey`, `highlight`, `highlighttext`, `inactiveborder`, `inactivecaption`, `inactivecaptiontext`, `infobackground`, `infotext`, `lime`, `maroon`, `match`, `menu`, `menutext`, `navy`, `olive`, `orange`, `purple`, `red`, `scrollbar`, `silver`, `teal`, `threeddarkshadow`, `threedface`, `threedhighlight`, `threedlightshadow`, `threedshadow`, `transparent`, `-webkit-activelink`, `-webkit-focus-ring-color`, `-webkit-link`, `-webkit-text`, `white`, `window`, `windowframe`, `windowtext`, `yellow`

Discussion

If not specified, the color specified by the `color` property is used. `-webkit-text-stroke-color` is commonly used in combination with `-webkit-text-fill-color`. Changes to this property can be animated using transitions.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-text-stroke-width

Specifies the width for the text outline.

Syntax

`-webkit-text-stroke-width:` *width*

Parameters

width

The width of the stroke.

Types Allowed

Length units

Constants

`medium`, `thick`, `thin`

Discussion

This property is significant only in combination with `-webkit-text-stroke-color`.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-line-break

Specifies line-breaking rules for CJK (Chinese, Japanese, and Korean) text.

Syntax

`-webkit-line-break:` *setting*

Parameters

setting

The line-breaking setting.

Constants

`after-white-space`, `normal`

Availability

Available in Safari 3.0 and later. (Called `-khtml-line-break` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Apple extension.

-webkit-appearance

Changes the appearance of buttons and other controls to resemble native controls.

Syntax

`-webkit-appearance:` *appearance*

Parameters

appearance

The appearance of the control.

Constants

`button`, `button-bevel`, `caret`, `checkbox`, `listbox`, `listitem`, `media-fullscreen-button`, `media-mute-button`, `media-play-button`, `media-seek-back-button`, `media-seek-forward-button`, `media-slider`, `media-sliderthumb`, `menulist`, `menulist-button`, `menulist-text`, `menulist-textfield`, `none`, `push-button`, `radio`,

```
scrollbarbutton-down, scrollbarbutton-left, scrollbarbutton-right,
scrollbarbutton-up, scrollbargripper-horizontal, scrollbargripper-vertical,
scrollbarthumb-horizontal, scrollbarthumb-vertical, scrollbartrack-horizontal,
scrollbartrack-vertical, searchfield, searchfield-cancel-button,
searchfield-decoration, searchfield-results-button,
searchfield-results-decoration, slider-horizontal, slider-vertical,
sliderthumb-horizontal, sliderthumb-vertical, square-button, textarea, textfield
```

Availability

Available in Safari 3.0 and later. Available in iPhone OS 1.0 and later.

Support Level

Experimental CSS 3.

-webkit-nbsp-mode

Defines the behavior of nonbreaking spaces within text.

Syntax

`-webkit-nbsp-mode: behavior;`

Parameters

`behavior`

The behavior of nonbreaking spaces.

Constants

`normal`, `space`

Discussion

If this property is set to `space`, nonbreaking spaces in the element's contents are treated like normal spaces. If it is set to `normal`, nonbreaking spaces are treated as nonbreaking.

Availability

Available in Safari 3.0 and later. (Called `-khtml-nbsp-mode` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Apple extension.

-webkit-rtl-ordering

Overrides ordering defaults for right-to-left content.

Syntax

`-webkit-rtl-ordering: order;`

Parameters

`order`

The order of the content.

Constants

`logical, visual`

Discussion

There are two ways to order content that mixes left-to-right and right-to-left text: `logical`, in which the raw content is in mixed order (requiring a bidirectional renderer), and `visual`, in which the right-to-left content is encoded in reverse order so that the entire line of text can be rendered from left to right in a unidirectional fashion. The distinction between these two character orders is normally handled automatically as a side effect of character set. This property allows you to override whether the browser should treat the content as being in logical or visual order.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Apple extension.

-webkit-user-drag

Specifies that an entire element should be draggable instead of its contents.

Syntax

`-webkit-user-drag: behavior;`

Parameters

behavior

    The dragging behavior of the element.

Constants

`auto, element, none`

Availability

Available in Safari 3.0 and later. (Called `-khtml-user-drag` in Safari 2.0.)

Support Level

Apple extension.

-webkit-user-modify

Determines whether a user can edit the content of an element.

Syntax

`-webkit-user-modify: policy;`

Parameters

policy

    The user modification policy.

Constants

`read-only, read-write, read-write-plaintext-only`

Discussion

This is closely related to the `contentEditable` attribute.

Availability

Safari 3.0 and later. (Called `-khtml-user-modify` in Safari 2.0.)

Support Level

Apple extension.

-webkit-user-select

Determines whether a user can select the content of an element.

Syntax

`-webkit-user-select:` *policy*`;`

Parameters

policy

The user selection policy.

Constants

`auto`, `none`, `text`

Availability

Available in Safari 3.0 and later. (Called `-khtml-user-select` in Safari 2.0.)

Support Level

Apple extension.

Tables

border-collapse

Defines the model of an element's border.

Syntax

`border-collapse:` *behavior*`;`

Constants

`collapse`, `separate`

Availability

Available in Safari 1.2 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

border-spacing

Defines the spacing between an element's border and the content within.

Syntax

```
border-spacing: length;
```

Parameters

length

> The size of the spacing.

Subproperties

```
-webkit-border-horizontal-spacing
-webkit-border-vertical-spacing
```

Availability

> Available in Safari 1.2 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

caption-side

Defines the side of a table on which its caption appears.

Syntax

```
caption-side: side;
```

Parameters

side

> The side of the table that will have a caption.

Constants

```
bottom, left, right, top
```

Availability

> Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

> CSS 2.1.

empty-cells

Sets the border behavior for cells with no content.

Syntax

```
empty-cells: behavior;
```

Parameters

behavior

The behavior for cells with no content.

Constants

hide, show

Availability

Available in Safari 1.2 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

table-layout

Specifies whether to use automatic or fixed table layout.

Syntax

table-layout: *behavior*;

Parameters

behavior

If auto, layout is determined by all cells in the table; if fixed, layout is determined by the first row of content only.

Constants

auto, fixed

Discussion

Automatic table layout, specified by the value auto, is the default table layout behavior. In this mode, the table layout is calculated based on the contents of every cell in every row of the table.

Fixed table layout, specified by the value fixed, is a faster (but more restrictive) layout behavior. In this layout mode, the layout of the table is calculated based only on the first row of tabular content (not including any heading rows). This mode allows the layout to be calculated much earlier in the page load process and greatly simplifies the calculations, but can cause content in later rows to overflow the table's boundaries.

Availability

Available in Safari 1.0 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

-webkit-border-horizontal-spacing

Defines the spacing between the horizontal portion of an element's border and the content within.

Syntax

-webkit-border-horizontal-spacing: *value*;

Parameters

value

The amount of horizontal spacing.

Types Allowed

Length units, nonnegative values

Discussion

Equivalent to the horizontal portion of the border-spacing property. Changes to this property can be animated using transitions.

Availability

Available in Safari 3.0 and later. (Called `-khtml-border-horizontal-spacing` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Apple extension.

-webkit-border-vertical-spacing

Defines the spacing between the vertical portion of an element's border and the content within.

Syntax

`-webkit-border-vertical-spacing:` *value*`;`

Parameters

value

The amount of vertical spacing.

Types Allowed

Length units, nonnegative values

Discussion

Equivalent to the vertical portion of the border-spacing property. Changes to this property can be animated using transitions.

Availability

Available in Safari 3.0 and later. (Called `-khtml-border-vertical-spacing` in Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Apple extension.

-webkit-column-break-after

Determines whether a column break can and should occur after an element in a multicolumn flow layout.

Syntax

`-webkit-column-break-after:` *policy*`;`

Parameters

policy

The column break policy.

Constants

`always`, `auto`, `avoid`, `left`, `right`

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Under development.

-webkit-column-break-before

Determines whether a column break can and should occur before an element in a multicolumn flow layout.

Syntax

`-webkit-column-break-before:` *policy*`;`

Parameters

policy

The column break policy.

Constants

`always`, `auto`, `avoid`, `left`, `right`

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Under development.

-webkit-column-break-inside

Determines whether a column break should be avoided within the bounds of an element in a multicolumn flow layout.

Syntax

`-webkit-column-break-inside:` *policy*`;`

Parameters

policy

The column break policy.

Constants

`auto`, `avoid`

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level
Under development.

-webkit-column-count

Specifies the number of columns desired in a multicolumn flow.

Syntax
`-webkit-column-count:` *number_of_columns*`;`

Parameters

number_of_columns

The number of columns in the multicolumn flow.

Types Allowed
Integers, nonnegative values

Constants
`auto`

Discussion
Changes to this property can be animated using transitions.

Availability
Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level
Under development.

-webkit-column-gap

Specifies the space between columns in a multicolumn flow.

Syntax
`-webkit-column-gap:` *width*`;`

Parameters

width

The width of the gap.

Types Allowed
Length units

Constants
`normal`

Discussion
Changes to this property can be animated using transitions.

Availability
Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Under development.

-webkit-column-rule

Specifies the color, style, and width of the column rule.

Syntax

```
-webkit-column-rule: width style color;
```

Parameters

width

The width of the column rule.

style

The style of the column rule.

color

The color of the column rule.

Subproperties

```
-webkit-column-rule-color
-webkit-column-rule-style
-webkit-column-rule-width
```

Discussion

The column rule appears in the middle of the column gap in a multicolumn flow layout.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Under development.

-webkit-column-rule-color

Specifies the color of the column rule.

Syntax

```
-webkit-column-rule-color: color;
```

Parameters

color

The color of the column rule.

Constants

```
activeborder, activecaption, appworkspace, aqua, background, black, blue, buttonface,
buttonhighlight, buttonshadow, buttontext, captiontext, fuchsia, gray, graytext, green,
grey, highlight, highlighttext, inactiveborder, inactivecaption, inactivecaptiontext,
infobackground, infotext, lime, maroon, match, menu, menutext, navy, olive, orange, purple,
```

```
red, scrollbar, silver, teal, threeddarkshadow, threedface, threehighlight,
threedlightshadow, threedshadow, transparent, -webkit-activelink,
-webkit-focus-ring-color, -webkit-link, -webkit-text, white, window, windowframe,
windowtext, yellow
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Under development.

-webkit-column-rule-style

Specifies the style of the column rule.

Syntax

```
-webkit-column-rule-style: style;
```

Parameters

style

The style of the column rule.

Constants

```
dashed, dotted, double, groove, hidden, inset, none, outset, ridge, solid
```

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Under development.

-webkit-column-rule-width

Specifies the width of the column rule.

Syntax

```
-webkit-column-rule-width: width;
```

Parameters

width

The width of the column rule.

Types Allowed

Length units

Constants

```
medium, thick, thin
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Under development.

-webkit-column-width

Specifies the width of the column in a multicolumn flow.

Syntax

```
-webkit-column-width: width;
```

Parameters

width

The width of the column.

Types Allowed

Length units

Constants

```
auto
```

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level

Under development.

-webkit-columns

A composite property that specifies the width and number of columns in a multicolumn flow layout.

Syntax

```
-webkit-columns: width count
```

Parameters

width

The width of each column.

count

The number of columns.

Subproperties

```
-webkit-column-count
-webkit-column-width
```

Availability

Available in Safari 3.0 and later. Available in iPhone OS 2.0 and later.

Support Level
Under development.

User Interface

cursor

Defines the cursor to display onscreen when the pointer is over an element.

Syntax
```
cursor: style;
```

Parameters

style

The type of cursor.

Constants
```
alias, all-scroll, auto, cell, col-resize, context-menu, copy, crosshair, default,
e-resize, ew-resize, hand, help, move, n-resize, ne-resize, nesw-resize, no-drop, none,
not-allowed, ns-resize, nw-resize, nwse-resize, pointer, progress, row-resize, s-resize,
se-resize, sw-resize, text, vertical-text, w-resize, wait, -webkit-zoom-in,
-webkit-zoom-out
```

Discussion
Although the CSS specification allows it, Safari does not support custom cursors.

Availability
Available in Safari 1.2 and later.

Support Level
CSS 2.1.

outline

Defines a variety of properties for an element's outline (drawn outside the element's border) within one declaration.

Syntax
```
outline: color style width;
```

Parameters

color

The color of the outline.

style

The style of the outline.

width

The width of the outline.

Subproperties

outline-color
outline-style
outline-width

Availability

Available in Safari 1.2 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

outline-color

Defines the color of an element's outline.

Syntax

outline-color: *color*;

Parameters

color

The color of the outline.

Constants

invert, -webkit-focus-ring-color

Discussion

Changes to this property can be animated using transitions.

Availability

Available in Safari 1.2 and later. Available in iPhone OS 1.0 and later.

Support Level

CSS 2.1.

outline-offset

Defines the offset of an element's outline from its border.

Syntax

outline-offset: *length*;

Parameters

length

The size of the offset.

Types Allowed
Length units

Discussion
Changes to this property can be animated using transitions.

Availability
Available in Safari 1.2 and later. Available in iPhone OS 1.0 and later.

Support Level
CSS 2.1.

outline-style

Defines the style of an element's outline.

Syntax
outline-style: *value*;

Parameters

value

The style of the outline.

Constants
auto

Availability
Available in Safari 1.2 and later. Available in iPhone OS 1.0 and later.

Support Level
CSS 2.1.

outline-width

Defines the width of an element's outline.

Syntax
outline-width: *value*;

Parameters

value

The width of the outline.

Types Allowed
Length units

Constants

    `medium`, `thick`, `thin`

Discussion

    Changes to this property can be animated using transitions.

Availability

    Available in Safari 1.2 and later. Available in iPhone OS 1.0 and later.

Support Level

    CSS 2.1.

-webkit-box-align

Specifies the alignment of nested elements within an outer flexible box element.

Syntax

    `-webkit-box-align:` *`alignment`*`;`

Parameters

    `alignment`

        The alignment of nested elements.

Constants

    `baseline`, `center`, `end`, `start`, `stretch`

Discussion

    This property specifies the horizontal alignment if the box direction is vertical, and vice versa. This property applies only to flexible box layouts. For more information about flexible boxes, see http://www.w3.org/TR/css3-layout/.

Availability

    Available in Safari 3.0 and later. (Called `-khtml-box-align` in Safari 1.1 through Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

    Under development.

-webkit-box-direction

Specifies the direction in which child elements of a flexible box element are laid out.

Syntax

    `-webkit-box-direction:` *`layout_direction`*`;`

Parameters

    `layout_direction`

        The layout direction.

Constants

    `normal`, `reverse`

Discussion

This applies only to flexible box layouts. For more information about flexible boxes, see http://www.w3.org/TR/css3-layout/.

Availability

Available in Safari 3.0 and later. (Called `-khtml-box-direction` in Safari 1.1 through Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Under development.

-webkit-box-flex

Specifies an element's flexibility.

Syntax

`-webkit-box-flex: ` *`flex_value`*`;`

Parameters

`flex_value`

The flexibility of the element.

Types Allowed

Floating-point numbers

Discussion

Flexible elements can stretch or shrink to fit the size of the bounding box of their parent element. The amount of stretching or shrinkage of an element is determined by its flex value relative to the flex values of other elements within the same parent element.

This property applies only to flexible box layouts. For more information about flexible boxes, see http://www.w3.org/TR/css3-layout/.

Availability

Available in Safari 3.0 and later. (Called `-khtml-box-flex` in Safari 1.1 through Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Under development.

-webkit-box-flex-group

Specifies groups of dynamically resizing elements that are adjusted to be the same size.

Syntax

`-webkit-box-flex-group: ` *`group_number`*`;`

Parameters

`group_number`

The group number of the flexible element.

Types Allowed

Integers, nonnegative values

Discussion

During size adjustment of flex boxes, any boxes with the same group number are adjusted to be the same size.

This property applies only to flexible box layouts. For more information about flexible boxes, see http://www.w3.org/TR/css3-layout/.

Availability

Available in Safari 3.0 and later. (Called `-khtml-box-flex-group` in Safari 1.1 through Safari 2.0.) Available in iPhone OS 1.0 and later. Safari 3.0 and later.

Support Level

Under development.

-webkit-box-lines

Specifies whether a flexible box should contain multiple lines of content.

Syntax

`-webkit-box-lines:` *behavior*`;`

Parameters

behavior

If `multiple`, the flexible box can contain multiple lines of content; if `single`, only one line is allowed.

Constants

`multiple`, `single`

Discussion

This property applies only to flexible box layouts. For more information about flexible boxes, see http://www.w3.org/TR/css3-layout/.

Availability

Available in Safari 3.0 and later. (Called `-khtml-box-lines` in Safari 1.1 through Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Under development.

-webkit-box-ordinal-group

Specifies a rough ordering of elements in a flexible box.

Syntax

`-webkit-box-ordinal-group:` *group_number*`;`

Parameters

> *group_number*
>
>> The ordinal group number of the element.

Types Allowed
Integers, nonnegative values

Discussion
Elements with lower ordinal group values are displayed first.

This property applies only to flexible box layouts. For more information about flexible boxes, see http://www.w3.org/TR/css3-layout/.

Availability
Available in Safari 3.0 and later. (Called `-khtml-box-ordinal-group` in Safari 1.1 through Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level
Under development.

-webkit-box-orient

Specifies the layout of elements nested within a flexible box element.

Syntax
`-webkit-box-orient: ` *orientation*`;`

Parameters

> *orientation*
>
>> The orientation of elements nested in the flexible box.

Constants
`block-axis`, `horizontal`, `inline-axis`, `vertical`

Discussion
This property applies only to flexible box layouts. For more information about flexible boxes, see http://www.w3.org/TR/css3-layout/.

Availability
Available in Safari 3.0 and later. (Called `-khtml-box-orient` in Safari 1.1 through Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level
Under development.

-webkit-box-pack

Specifies alignment of child elements within the current element in the direction of orientation.

Syntax
`-webkit-box-pack: ` *alignment*`;`

Parameters

alignment

The alignment of child elements.

Constants
center, end, justify, start

Discussion

For elements whose children are aligned horizontally, a packing value of start indicates left alignment with extra space towards the right side, a value of end indicates right alignment with extra space to the left, a value of center indicates center alignment with extra space split evenly on either side, and a value of justify indicates that the outer elements should be aligned on the left and right, with space added evenly between the elements.

Similarly, for elements whose children are aligned vertically, a value of start indicates that the elements should be aligned to the top, a value of end indicates that the elements should be aligned to the bottom, and so on.

This property is similar to "-webkit-box-align" (page 110), which specifies alignment in the opposite direction from the direction of orientation.

This property applies only to flexible box layouts. For more information about flexible boxes, see http://www.w3.org/TR/css3-layout/.

Availability

Available in Safari 3.0 and later. (Called -khtml-box-pack in Safari 1.1 through Safari 2.0.) Available in iPhone OS 1.0 and later.

Support Level

Under development.

-webkit-touch-callout

Disables the default callout shown when you touch and hold a touch target.

Syntax
-webkit-touch-callout: *behavior*;

Parameters

behavior

The touch callout behavior.

Discussion

On iPhone OS, when you touch and hold a touch target such as a link, Safari displays a callout containing information about the link. This property allows you to disable that callout.

The current allowable values are *none* and inherit.

Availability

Available in iPhone OS 2.0 and later.

Support Level

Apple extension—iPhone OS only.

-webkit-dashboard-region

Specifies the behavior of regions in a Dashboard widget.

Syntax

```
-webkit-dashboard-region:
        dashboard-region( ... )
        [...]
```

Constants

```
none
```

Discussion

This property is described in more detail in Declaring Control Regions in *Dashboard Programming Topics*.

Availability

Available in Safari 3.0 and later. (Called `-apple-dashboard-region` in Safari 2.0.)

Support Level

Apple extension—Dashboard only.

-webkit-tap-highlight-color

Overrides the highlight color shown when the user taps a link or a JavaScript clickable element in Safari on iPhone.

Syntax

```
-webkit-tap-highlight-color: color;
```

Parameters

color

> The tapped link color.

Discussion

This property obeys the alpha value, if specified. If you don't specify an alpha value, Safari on iPhone applies a default alpha value to the color. To disable tap highlighting, set the alpha value to `0` (invisible). If you set the alpha value to `1.0` (opaque), the element is not visible when tapped.

Availability

Available in iPhone OS 1.1.1 and later.

Support Level

Apple extension—iPhone OS only.

Additional Unsupported Properties

WebKit provides partial support for a number of properties that are not supported for developer use. This list may include:

Properties designed for Apple internal use, such as properties specific to the way Mail and other applications use WebKit.

Properties that are in a very early stage of development and are not really usable yet.

Properties that are used within WebKit itself and cannot be parsed in a CSS file.

Properties that are parsed for historical reasons, but that are not actually used.

Because these properties are unsupported, they are not documented in detail. However, they are listed here so that if you find them in the source code, in test cases, and so on, you will be able to determine their status.

WebKit-Specific Unsupported Properties

```
-webkit-border-fit
-webkit-font-size-delta
-webkit-highlight
-webkit-line-clamp
-webkit-match-nearest-mail-blockquote-color
-webkit-text-decorations-in-effect
-webkit-transition-repeat-count
```

Unsupported Properties from the CSS Specification

`font-size-adjust`—Describes the font aspect ratio to preserve proportionality in the event of font substitution. Unsupported CSS 2 property; removed in CSS 2.1; reintroduced in CSS 3.

`font-stretch`—Selects a normal, condensed, or extended variant of a font in an element or describes availability of these variants in a font definition. Declared in CSS 2.1/CSS 3.

`marker-offset`—Sets the offset of a marker (abullet in a bulleted list, for example). Unsupported CSS 2 property; removed in CSS 2.1.

`marks`—Sets what type of crop marks to use on paged media. Unsupported CSS 2 property; removed in CSS 2.1.

`page`—Used for named page support. Unsupported CSS 2 property; removed in CSS 2.1.

`quotes`—Sets the quotation mark characters used for nested <q> tags.

`size`—Sets page dimensions for paged media. Unsupported CSS 2 property; removed in CSS 2.1.

`speak-header`—Sets whether a browser should speak the contents of the corresponding table heading cell before speaking the contents of each cell. Unsupported CSS 2 aural media property. Aural media deprecated in CSS 2.1. Property reintroduced in CSS 3

`text-line-through`—Composite property describing overstrike color, style, and mode. Declared in CSS 3.

`text-line-through-color`—Describes color for overstrike. Declared in CSS 3.

`text-line-through-mode`—Describes the mode for overstrike. Declared in CSS 3.

`text-line-through-style`—Describes the style for overstrike. Declared in CSS 3.

`text-line-through-width`—Describes the width for overstrike. Declared in CSS 3.

`text-overline`—Composite property describing overline color, style, mode, and width(like underline, but above the text). Declared in CSS 3.

`text-overline-color`—Describes the color of overline (like underline, but above the text). Declared in CSS 3.

`text-overline-mode`—Describes the mode of overline (like underline, but above the text). Declared in CSS 3.

`text-overline-style`—Describes the style of overline (like underline, but above the text). Declared in CSS 3.

`text-overline-width`—Describes the width of overline (like underline, but above the text). Declared in CSS 3.

`text-underline`—Composite property describing underline color, style, mode, and width. Declared in CSS 3.

`text-underline-color`—Describes the color of underline. Declared in CSS 3.

`text-underline-mode`—Describes the mode of underline. Declared in CSS 3.

`text-underline-style`—Describes the style of underline. Declared in CSS 3.

`text-underline-width`—Describes the width of underline. Declared in CSS 3.

Unsupported Properties Specific to Other Browsers

`scrollbar-3dlight-color`—Microsoft Internet Explorer property.

`scrollbar-arrow-color`—Microsoft Internet Explorer property.

`scrollbar-darkshadow-color`—Microsoft Internet Explorer property.

`scrollbar-face-color`—Microsoft Internet Explorer property.

`scrollbar-highlight-color`—Microsoft Internet Explorer property.

`scrollbar-shadow-color`—Microsoft Internet Explorer property.

`scrollbar-track-color`—Microsoft Internet Explorer property.

Supported CSS Rules

This chapter describes selected CSS rules supported by Safari. This is not intended to be an exhaustive list. The CSS rules described here are limited to rules that are either new or are not broadly supported by other browsers (including some specific to WebKit).

@font-face

Enables the use of downloadable web fonts (among other things). A downloadable web font definition looks like this:

```
@font-face {
    font-family: "MyFamilyname", cursive [, ...];
    font-style: normal [, ...];
    font-variant: normal[, ...];
    font-weight: bold[, ...];
    font-stretch: condensed[, ...]; /* Not supported */
    font-size: 12pt;[, ...] /* Not supported */
    src: local("Font Family Name"),
        url(http://..../fontfile.ttf) format("truetype"),
        url(http://..../fontfile.ttf) [, ...];
}
```

The only required properties are `font-family` and `src`. For each of these properties, you can specify either a single value or a comma-separated list containing multiple values.

In the `src` property, you can specify any number of local font family names and any number of URLs (provided that you include at least one local name or URL, of course).

For each URL, you can also specify a format hint if desired. This hint is intended to help the browser avoid downloading fonts in formats that it does not support.

The remaining properties tell the browser how to choose between multiple variants in the same font family. For example:

```
<style><!--
@font-face {
        font-family: Geo;
        font-style: normal;
        src: url(fonts/geo_sans_light/GeosansLight.ttf);
}

@font-face {
        font-family: Geo;
        font-style: oblique;
        src: url(fonts/geo_sans_light/GeosansLight-Oblique.ttf);
}
```

```
.ingeo {
        font-family: Geo;
}
--></style>
<div class='ingeo'>This is a test.</div>
```

Because the `font-style` property is specified for both font definitions, the browser uses the first entry for normal text and the second entry for oblique text, and thus, the text "This is a test." is displayed normally (vertically). If you remove these properties, the last font definition is used and the text appears in an oblique font (slanted).

For a list of specific values allowed for each of these properties, see the description for the property in question.

Availability
>Safari 3.1 and later. iPhone OS 1.1.4 and earlier. Deprecated in iPhone OS 2.0.

Support Level
>CSS 3

@media

Specifies CSS properties specific to a given output medium. For example, you might have styles specific to print media. Here is the syntax for these queries:

```
@media print {
    div.chapternumber {
        /* A new chapter should begin at the top of a
           printed page, slightly below normal text.
         */
        page-break-before: always;
        margin-top: .25 in;
    }
}
```

Safari supports the following media types:

Media Type	Description
all	Applies to all devices, regardless of medium. Equivalent to listing no media type.
print	Applies only to printed copies of the document.
screen	Applies only to content displayed on a screen.

Note: The `aural` media type is deprecated in CSS 2.1. The CSS 2.1 specification reserves the `speech` media type, but does not define which properties do or do not apply to it.

Availability
>Safari 1.0 and later. iPhone OS 1.0 and later.

Support Level
>CSS 3

@-webkit-keyframes

Specifies keyframes for CSS visual effect animation properties.

Syntax

```
keyframes-rule: '@-webkit-keyframes' [ IDENT | STRING ] '{' keyframes-blocks
'}';
keyframes-blocks: [ keyframe-selectors block ]* ;
keyframe-selectors: [ 'from' | 'to' | PERCENTAGE ] [ ',' [ 'from' | 'to' |
PERCENTAGE ] ]*;
```

Discussion

The `@-webkit-keyframes` keyword is followed by the name of the target animation and a set of style rules following the syntax above. You set the animation name using the `-webkit-animation-name` property.

The keyframes selector consists of a list of percentage values or the keywords `from` or `to`. The selector is used to specify the percentage along the duration of the animation or transition that the keyframes represent. The keyframes are specified by the block of property values declared for the selector. The keyword `from` is equivalent to the value `0`. The keyword `to` is equivalent to the value `100`.

The keyframe declaration consists of properties and values. Properties that are not animating are ignored in this rule, with the exception of the `-webkit-animation-timing-function` property.

This rule is the last rule encountered in sorted rules order that matches the name of the transition. This rule does not cascade; therefore, an animation never derives keyframes from more than one `@-webkit-keyframes` rule.

All of the values in selectors are sorted in increasing order by time. If there are any duplicates, the last keyframe specified inside the `@-webkit-keyframes` rule is used to provide the keyframe information for that time. There is no cascading within a `@-webkit-keyframes` rule if multiple keyframes specify the same keyframe selector value.

For example, the following `@-webkit-keyframes` rule contains keyframes for a transition or animation named "wobble." In the first keyframe, shown at the beginning of the animation cycle, the `left` value of the animation is 100 pixels. After 40% of the animation duration, the value of `left` is 150 pixels. After 60% of the animation duration, the `left` value is 75 pixels. At the end of the animation cycle, the `left` value returns to 100 pixels.

```
@-webkit-keyframes 'wobble' {

    0 {
      left: 100px;
    }

    40% {
      left: 150px;
    }

    60% {
      left: 75px;
    }

    100% {
      left: 100px;
    }
```

```
    }
```

You can also use a keyframes rule to set a timing function to animate or transition from one keyframe to another. You set the timing function for a keyframe within its block using the `-webkit-animation-timing-function` property.

For example, the following `@-webkit-keyframes` rule defines keyframes for a transition or animation named "bounce." Between the first and second keyframes—between 0 and 25%—an `ease-out` timing function is used. Between the second and third keyframes—between 25% and 50%—an `ease-in` timing function is used. As a result, the element moves up the page by 50 pixels, slowing down as it reaches its highest point, then speeds up as it falls back to 100 pixels. The second half of the animation behaves in a similar manner, but moves the element only 25 pixels up the page.

```
@-webkit-keyframes 'bounce' {

    from {
      top: 100px;
      -webkit-animation-timing-function: ease-out;
    }

    25% {
      top: 50px;
      -webkit-animation-timing-function: ease-in;
    }

    50% {
      top: 100px;
      -webkit-animation-timing-function: ease-out;
    }

    75% {
      top: 75px;
      -webkit-animation-timing-function: ease-in;
    }

    to {
      top: 100px;
    }

    }
```

Availability

iPhone OS 2.0 and later.

Support Level

Apple extension

CSS Property Functions

This chapter describes the functions you can use with supported CSS properties.

Visual Effects Timing Functions

cubic-bezier

Specifies a cubic Bézier curve.

```
cubic-bezier(P1x,P1y,P2x,P2y)
```

Parameters

> `P1x, P1y`
>
>> First point in the Bézier curve.
>
> `P2x, P2y`
>
>> Second point in the Bézier curve.

Discussion

> A cubic Bézier curve is defined by four control points, `P0` through `P3`. `P0` and `P3` are always set to `(0,0)` and `(1,1)`. This function is used to set the values for the points in between, points `P1` and `P2`. Each point is specified by both an x and y value.

Availability

> Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

Visual Effects Transform Functions

matrix

Specifies a 2D transformation in the form of a transformation matrix of six values.

```
matrix(m11, m12, m21, m22, tX, tY)
```

Parameters

> `m11, m12, m21, m22`
>
>> Elements of a 2 x 2 matrix in column-major order.

1,1	2,1
1,2	2,2

tX, tY

> The x and y translation elements.

Discussion

> Passing `matrix(a,b,c,d,e,f)` is equivalent to applying the transformation matrix `[a b c d e f]`.

Availability

> Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

matrix3d

Specifies a 3D transformation as a 4 x 4 matrix.

```
matrix3d(m00, m01, m02, m03, m10, m11, m12, m13, m20, m21, m22, m23, m30, m31, m31, m33)
```

Parameters

> *m00, m01, m02, m03, m10, m11, m12, m13, m20, m21, m22, m23, m30, m31, m31, m33*
>
> > Defines a 4 x 4 homogeneous matrix of 16 values in column-major order (`0,0`; `0,1`; `0,2`; …).

0,0	1,0	2,0	3,0
0,1	1,1	2,1	3,1
0,2	1,2	2,2	3,2
0,3	1,3	2,3	3,3

Availability

> Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

perspective

Specifies a perspective projection matrix.

```
perspective(depth)
```

Parameters

> *depth*
>
> > The distance, in pixels, of the z=0 plane from the viewer.

Discussion

This matrix maps a viewing cube onto a pyramid whose base is infinitely far away from the viewer and whose peak represents the viewer's position.

The viewable area is the region bounded by the four edges of the viewport (the portion of the browser window used for rendering the webpage between the viewer's position and a point at a distance of infinity from the viewer).

Lower values for this property give a more flattened pyramid and therefore a more pronounced perspective effect. A value of 1000 pixels gives a moderate amount of foreshortening, and a value of 200 pixels gives an extreme amount.

Availability

Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

rotate

Specifies a 2D rotation around the origin of the element.

```
rotate(angle)
```

Parameters

angle

The rotation angle. The angle may be specified using `deg`, `rad` or `grad` units.

Discussion

The rotation operation corresponds to the matrix `[cos(angle) sin(angle) -sin(angle) cos(angle) 0 0]`. The origin of the element is specified using the `-webkit-transform-origin` property.

Availability

Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

rotate3d

Specifies a clockwise 3D rotation.

```
rotate3d(x, y, z, angle)
```

Parameters

x, y, z

The `[x,y,z]` direction vector for the rotation.

If the direction vector is not of unit length, it will be normalized. If the direction vector cannot be normalized, such as `[0, 0, 0]`, the rotation will not be applied.

angle

The rotation angle. The angle may be specified using `deg`, `rad` or `grad` units.

Availability

Available in iPhone OS 2.0 and later.

rotateX

Specifies a clockwise rotation by the given angle about the x-axis.

```
rotateX(angle)
```

Parameters

angle

The rotation angle. The angle may be specified using `deg`, `rad` or `grad` units.

Availability
Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

rotateY

Specifies a clockwise rotation by the given angle about the y-axis.

```
rotateY(angle)
```

Parameters

angle

The rotation angle. The angle may be specified using `deg`, `rad` or `grad` units.

Availability
Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

rotateZ

Specifies a clockwise rotation by the given angle about the z-axis.

```
rotateZ(angle)
```

Parameters

angle

The angle of the rotation. The angle may be specified using `deg`, `rad` or `grad` units.

Availability
Available in iPhone OS 2.0 and later.

scale

Specifies a 2D scale operation.

```
scale(scaleX [, scaleY])
```

Parameters

scaleX

The scaling factor to apply in the x direction.

scaleY

The scaling factor to apply in the y direction. If not specified, defaults to *scaleX*.

Availability
Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

scale3d

Specifies a 3D scale operation.

```
scale3d(scaleX, scaleY, scaleZ)
```

Parameters

scaleX

The scaling factor to apply in the x direction.

scaleY

The scaling factor to apply in the y direction.

scaleZ

The scaling factor to apply in the z direction.

Availability
Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

scaleX

Scales in the x direction.

```
scaleX(sx)
```

Parameters

sx

The scaling factor to apply to the x direction.

Discussion
This function specifies a scale operation using the `[sx, 1, 1]` scaling vector.

Availability
Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

scaleY

Scales in the y direction.

```
scaleY(sy)
```

Parameters

sy

The scaling factor to apply to the y direction.

Discussion

This function specifies a scale operation using the `[1,sy,1]` scaling vector.

Availability

Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

scaleZ

Scales in the z direction.

scaleZ(sz)

Parameters

sz

The scaling factor to apply to the z direction.

Discussion

This function specifies a scale operation using the `[1,1,sz]` scaling vector.

Availability

Available in iPhone OS 2.0 and later.

skewX

Specifies a skew transformation along the x-axis by the given angle.

```
skewX(angle)
```

Parameters

angle

The angle of the skew. The angle may be specified using `deg`, `rad` or `grad` units.

Availability

Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

skewY

Specifies a skew transformation along the x-axis by the given angle.

```
skewY(angle)
```

Parameters

angle

> The angle of the skew. The angle may be specified using `deg`, `rad` or `grad` units.

Availability
> Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

translate

Specifies a 2D translation vector.

```
translate(deltaX [, deltaY])
```

Parameters

deltaX

> The number of units to translate along the x-axis. This value may be a percentage or a length.

deltaY

> The number of units to translate along the y-axis. If not specified, the y translation defaults to `0`. This value may be a percentage or a length.

Availability
> Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

translate3d

Specifies a 3D translation vector.

```
translate3d(deltaX, deltaY, deltaZ)
```

Parameters

deltaX

> The number of units to translate along the x-axis. This value may be a percentage or a length.

deltaY

> The number of units to translate along the y-axis. This value may be a percentage or a length.

deltaZ

> The number of units to translate along the z-axis. This value may be a percentage or a length.

Availability
Available in iPhone OS 2.0 and later.

translateX

Specifies a translation in the x direction.

```
translateX(deltaX)
```

Parameters

deltaX

The number of units to translate along the x-axis. This value may be a percentage or a length.

Availability
Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

translateY

Specifies a translation in the y direction.

```
translateY(deltaY)
```

Parameters

deltaY

The number of units to translate along the y-axis. This value may be a percentage or a length.

Availability
Available in Safari 3.1 and later. Available in iPhone OS 2.0 and later.

translateZ

Specifies a translation in the z direction.

```
translateZ(deltaZ)
```

Parameters

deltaZ

The number of units to translate along the z-axis. This value may be a percentage or a length.

Availability
Available in iPhone OS 2.0 and later.

Document Revision History

This table describes the changes to *Safari CSS Reference*.

Date	Notes
2008-11-19	Added animatable and iPhone OS availability information to CSS properties.
2008-10-15	Fixed typographical errors, improved consistency.
2008-09-09	Added iPhone OS 2.0 visual effects properties.
2008-07-11	Updated for iPhone 2.0.
2008-06-09	Updated for Safari 3.1.
2008-01-15	Made changes to terminology.
2007-12-11	Incorporated additional information about iPhone-specific properties.
2007-10-31	Added property-specific availability information and made minor edits throughout.
2007-09-07	Added availability, types, and special values and reformatted the content.
2007-06-06	Updated documentation to reflect CSS support added in Safari and Web Kit 1.3.
2005-06-04	New document that describes the CSS properties supported by Safari and the Web Kit.

131