CSCI 1720
Intermediate Web Design

CSS 3 Advanced Functions

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

1

CSS 3

Some advanced features and more about some we already know

2

CSS 3 Outline

The CSS outline property allows you to define an outline area around an element's box

An outline is a line that is drawn just outside the border edge of the elements

Outlines are generally used to indicate focus or active states of the elements such as buttons, links, form fields, etc.

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

3

CSS 3 Outline vs. Border

An outline looks very similar to the border, but it differs from border in the following ways:

Outlines do not take up space, because they always placed on top of the box of the element which may cause them to overlap other elements on the page

Unlike borders, outlines won't allow us to set each edge to a different width, or set different colors and styles for each edge. An outline is the same on all sides

Outlines do not have any impact on surrounding elements apart from overlapping

Unlike borders, outlines do not change the size or position of the element

Outlines may be non-rectangular, but you cannot create circular outlines

4

CSS 3 Outline vs. Border

If you put an outline on an element, it will take up the same amount of space on the web pages as if you didn't have an outline on that element

Because it overlaps margins (transparent area outside of the border) and surrounding elements

5

CSS 3 Outline Styles

The `outline-style` property sets the style of an element's outline such as: solid, dotted, etc.

The outline-style property can have one of the following values: none, solid, dashed, dotted, double, inset, outset, groove, and ridge

6

2

CSS 3 Outline Styles

solid	dashed	dotted	double
inset	outset	groove	ridge

7

CSS 3 Outline Styles

```
h1 {
    outline-style: dotted;
    padding: 10px;
}

p {
    outline-style: dashed;
    padding: 10px;
}
```

```
<h1>Lorem Ipsum</h1>
<p>
    Lorem ipsum dolor sit amet, gravida facilisi orci nulla.
    Erat dolor, sagittis blandit, maecenas quis mi suscipit,
    risus enim tincidunt erat. Eu justo, senectus ut mattis
    laoreet et. Nec ultricies ultrices iaculis non asperiores
    urna, nec ut dolor dignissim, sodales purus pretium eget,
    wisi integer molestie arcu platea, urna ut in ipsum
    laoreet. Aenean fusce donec, arcu elit, placerat arcu
    laoreet libero eu vitae.
</p>
```

8

CSS 3 Outline Styles

Lorem Ipsum

Lorem ipsum dolor sit amet, gravida facilisi orci nulla.
Erat dolor, sagittis blandit, maecenas quis mi suscipit,
risus enim tincidunt erat. Eu justo, senectus ut mattis
laoreet et. Nec ultricies ultrices iaculis non asperiores
urna, nec ut dolor dignissim, sodales purus pretium
eget, wisi integer molestie arcu platea, urna ut in
ipsum laoreet. Aenean fusce donec, arcu elit,
placerat arcu laoreet libero eu vitae.

9

3

CSS 3 Outline Width

The outline-width property specifies the width of the outline to be added on an element

```
h1 {
    outline-style: dotted;
    outline width: 5px;
    padding: 10px;
}

p {
    outline-style: dashed;
    outline-width: 5px;
    padding: 10px;
}
```

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

10

CSS 3 Outline Width

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

11

CSS 3 Outline Color

The **outline-color** property sets the color of the outline.

This property accepts the same values as those used for the color property

Won't work if the **outline-style** property isn't set

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

12

CSS 3 Outline Color

```
h1 {
    outline-style: dotted;
    outline-width: 5px;
    outline-color: #0055ff;
    padding: 10px;
}

p {
    outline-style: dashed;
    outline-width: 5px;
    outline-color: #0055ff;
    padding: 10px;
}
```

Lorem Ipsum

Lorem ipsum dolor sit amet, gravida facilisi orci nulla. Erat dolor, sagittis blandit, maecenas quis mi suscipit, risus enim tincidunt erat. Eu justo, senectus ut mattis laoreet et. Nec ultricies ultrices iaculis non asperiores urna, nec ut dolor dignissim, sodales purus pretium eget, wisi integer molestie arcu platea, urna ut in ipsum laoreet. Aenean fusce donec, arcu elit, placerat arcu laoreet libero eu vitae.

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

13

CSS 3 Outline Shorthand

Like the border (and other) property(-ies), the outline property can be written in shorthand:

```
h1 {
    outline: 2px dotted #0055ff;
    padding: 10px;
}

p {
    outline: 2px dashed #0055ff;
    padding: 10px;
}
```

Lorem Ipsum

Lorem ipsum dolor sit amet, gravida facilisi orci nulla. Erat dolor, sagittis blandit, maecenas quis mi suscipit, risus enim tincidunt erat. Eu justo, senectus ut mattis laoreet et. Nec ultricies ultrices iaculis non asperiores urna, nec ut dolor dignissim, sodales purus pretium eget, wisi integer molestie arcu platea, urna ut in ipsum laoreet. Aenean fusce donec, arcu elit, placerat arcu laoreet libero eu vitae.

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

14

CSS 3 Removing the Outline

The outline property is widely used to remove the outline around active links

However, it is recommended to apply some alternative style to indicate that the link has focus

```
a, a:active, a:focus {
    outline: none;
}
```

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

15

5

CSS 3 Cursors

We can use the cursor property to modify the appearance of, well, the
cursor

```
a, a:active, a:focus {
    cursor: move;
}
```

Visit ETSU!

16

CSS 3 Cursors

Look	Values	Example
⬚	default	a:hover{cursor:default;}
⬚	pointer	a:hover{cursor:pointer;}
I	text	a:hover{cursor:text;}
⧗	wait	a:hover{cursor:wait;}
⬚	help	a:hover{cursor:help;}
⬚	progress	a:hover{cursor:progress;}
+	crosshair	a:hover{cursor:crosshair;}
✛	move	a:hover{cursor:move;}
▸	url()	a:hover{cursor:url("custom.cur"), default;}

17

CSS 3 Overflow

The overflow property specifies the behavior that occurs when an element's content overflows (doesn't fit) the element's box

There may be a situation when the content of an element might be larger than the dimensions of the box itself

For example given width and height properties did not allow enough room to accommodate the content of the element

CSS overflow property allows you to specify whether to clip content, render scroll bars or display overflow content of a block-level element

18

CSS 3 Overflow

Can take one of the following values: `visible` (default), `hidden`, `scroll`, and `auto`

CSS3 also defines the `overflow-x` and `overflow-y` properties which allow for independent control of the vertical and horizontal clipping

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

19

CSS 3 Overflow

visible	The default value. Content is not clipped; it will be rendered outside the element's box, and may thus overlap other content.
hidden	Content that overflows the element's box is clipped and the rest of the content will be invisible.
scroll	The overflowing content is clipped, just like hidden, but provides a scrolling mechanism to access the overflowed content.
auto	If content overflows the element's box it will automatically provides the scrollbars to see the rest of the content, otherwise scrollbar will not appear.

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

20

CSS 3 Display

The CSS `display` property can be used to modify the display of an element on a web page

Remember, elements default to either block or inline display. We can modify this with CSS

We've done this already with images (inline) in order to make them centered in their container

One value that is useful for dynamic content is 'none'

Using JavaScript, we can make an element appear or disappear, based on a given event

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

21

CSS 3 Display

The CSS **display** property can be used to modify the display of an element on a web page

The **inline-block** value of the **display** property causes an element to generate a block box that will be flowed with surrounding content i.e. in the same line as adjacent content

22

CSS 3 Display

```
<ul>
    <li><a href='#'>Home</a></li>
    <li><a href='#'>Products</a></li>
    <li><a href='#'>About</a></li>
    <li><a href='#'>Contact</a></li>
</ul>
```

- Home
- Products
- About
- Contact

```
li {
    display: inline-block;
    width: 150px;
    margin right: 10px;
    text-align: center;
    border: .25px solid #bbb;
}
```

Home   Products   About   Contact

23

CSS 3 Visibility

The CSS **visibility** property can also be used to modify the display of an element on a web page

The difference between **visibility** and **display** is that when modifying the **visibility** property, the element still takes up space on the page, even if it is hidden

Also, if a child element is set to **visible**, it will display, even if its parent element is **hidden**

24

CSS 3 Visibility

The CSS `visibility` property can also be used to modify the display of an element on a web page

visible	Default value. The box and its contents are visible.
hidden	The box and its content are invisible, but still affect the layout of the page.
collapse	This value causes the entire row or column to be removed from the display. This value is used for row, row group, column, and column group elements.
inherit	Specifies that the value of the visibility property should be inherited from the parent element i.e. takes the same visibility value as specified for its parent.

East Tennessee State University
Department of Computing
CSCI 1720
Intermediate Web Design

25

CSS 3 Visibility

The style rule `visibility: collapse;` however, removes the internal table elements, but it does not affect the layout of the table in any other way

The space normally occupied by the table elements will be filled by the subsequent siblings

If the style rule `visibility: collapse;` is specified for other elements rather than the table elements, it causes the same behavior as hidden

East Tennessee State University
Department of Computing
CSCI 1720
Intermediate Web Design

26

CSS Visibility vs Display

The `display` and `visibility` CSS properties appear to be the same thing, but they are in fact quite different and often confuse those new to web development

`visibility: hidden;` hides the element, but it still takes up space in the layout. Child element of a hidden box will be visible if their visibility is set to visible.

`display: none;` turns off the display and removes the element completely from the document. It does not take up any space, even though the HTML for it is still in the source code. All child elements also have their display turned off, even if their display property is set to something other than none

East Tennessee State University
Department of Computing
CSCI 1720
Intermediate Web Design

27

CSS Position

The position property defines how an element will be positioned on a page

Positioning elements appropriately on the web pages is a necessity for a good layout design

There are several methods in CSS that you can use for positioning elements

28

CSS Position

Static Positioning

A static positioned element is always positioned according to the normal flow of the page

HTML elements are positioned static by default

Static positioned elements are not affected by the `top`, `bottom`, `left`, `right`, and `z-index` properties

29
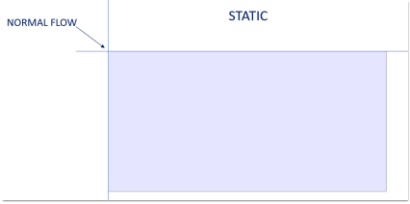
CSS Position

30

CSS Position

Relative Positioning

A **relative** positioned element is positioned relative to its normal position

In the **relative** positioning scheme the element's box position is calculated according to the normal flow

Then the box is shifted from this normal position according to the properties — **top** or **bottom** and/or **left** or **right**

A relatively positioned element can be moved and overlap other elements, but it keeps the space originally reserved for it in the normal flow

East Tennessee State University Department of Computing		CSCI 1720 Intermediate Web Design

31

CSS Position

Relative Positioning

```
.box {
    position: relative;
    top: 100px;
    left: 100px;
    width: 400px;
    height: 200px;
    border: 1px solid #999;
    background: rgba(0,0,255,.1);
}
```

```
<div class='box'>

</div>
```

East Tennessee State University Department of Computing		CSCI 1720 Intermediate Web Design

32

CSS Position

NORMAL FLOW

RELATIVE

100px

100px

East Tennessee State University Department of Computing		CSCI 1720 Intermediate Web Design

33

CSS Position

Absolute Positioning

An **absolute**-ly positioned element is positioned relative to the first parent element that has a **position** other than **static**

If no such element is found, it will be positioned on a page relative to the 'top-left' corner of the browser window

The box's offsets further can be specified using one or more of the properties **top**, **right**, **bottom**, and **left**

34

CSS Position

Absolute Positioning

Absolutely positioned elements are taken out of the normal flow entirely and thus take up no space when placing sibling elements

It can overlap other elements depending on the **z-index** property value

Also, an absolutely positioned element can have margins, and they do not collapse with any other margins.

35

CSS Position

Absolute Positioning

```
.box {
    position: absolute;
    top: 100px;
    left: 100px;
    width: 400px;
    height: 200px;
    border: 1px solid #999;
    background: rgba(0,0,255,.1);
}
```

```
<div class='box'>
</div>
```

36

CSS Position

TOP OF PAGE

ABSOLUTE

100px

100px

37

CSS Position

Fixed Positioning

Fixed positioning is a subcategory of absolute positioning

The only difference is, a fixed positioned element is fixed with respect to the browser's viewport and does not move when scrolled

38

CSS Layers

The CSS `z-index` property can be used in conjugation with the position property to create an effect of layers like Photoshop

Usually HTML pages are considered two-dimensional, because text, images and other elements are arranged on the page without overlapping

However, in addition to their horizontal and vertical positions, boxes can be stacked along the z-axis as well i.e. one on top of the other by using the CSS `z-index` property

39

CSS Layers

This property specifies the stack level of a box whose position value is one of absolute, fixed, or relative

The z-axis position of each layer is expressed as an integer representing the stacking order for rendering. An element with a larger `z-index` overlaps an element with a lower one

A `z-index` property can help you to create more complex webpage layouts

40

CSS Layers

```css
.box {
    position: absolute;
    top: 10px;
    left: 10px;
    width: 400px;
    height: 200px;
    border: 1px solid #999;
    background: rgba(0,0,255,.1);
}

.box-two {
    z-index: 1;
    width: 400px;
    height: 200px;
    border: 1px solid #999;
    background: rgba(255,0,255,.1);
}
```

```html
<div class='box'>
    BOX 1
</div>
<div class='box-two'>
    BOX 2
</div>
```

41

CSS Layers

BOX 1

BOX 2

42

14
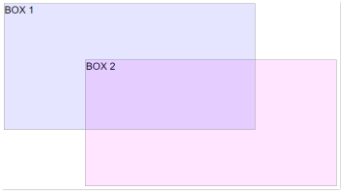
CSS Pseudo-elements

The CSS pseudo-elements allow you to style the elements or parts of the elements without adding any IDs or classes to them

It will be really helpful in the situations when you just want to style the first letter of a paragraph to create the drop cap effect or you want to insert some content before or after an element, etc. only through style sheet

43

CSS Pseudo-elements

CSS3 introduced a new double-colon (::) syntax for pseudo-elements to distinguish between them and pseudo-classes

The new syntax of the pseudo-element can be given with:

```
selector::pseudo-element {
    property: value;
}
```

44

CSS Pseudo-elements

```
p::first-line {
    font-variant: small-caps;
    font-style: oblique;
    margin: 0;
    display: block;
    position: relative;
    top: 10px;
}

p::first-letter {
    font-size: 1.5em;
    font-style: oblique;
}
```

```
<div class='box'>
    <p>
        Lorem ipsum dolor sit amet, gravida facilisi orci nulla.
        Erat dolor, sagittis blandit, maecenas quis mi suscipit,
        risus enim tincidunt erat. Eu justo, senectus ut mattis
        laoreet et.
    </p>
    <p>
        Lorem ipsum dolor sit amet, gravida facilisi orci nulla.
        Erat dolor, sagittis blandit, maecenas quis mi suscipit,
        risus enim tincidunt erat. Eu justo, senectus ut mattis
        laoreet et.
    </p>
</div>
```

45

CSS Pseudo-elements

LOREM IPSUM DOLOR SIT AMET, GRAVIDA FACILISI
orci nulla. Erat dolor, sagittis blandit, maecenas quis
mi suscipit, risus enim tincidunt erat. Eu justo,
senectus ut mattis laoreet et.

LOREM IPSUM DOLOR SIT AMET, GRAVIDA FACILISI
orci nulla. Erat dolor, sagittis blandit, maecenas quis
mi suscipit, risus enim tincidunt erat. Eu justo,
senectus ut mattis laoreet et.

46

CSS Pseudo-elements

The ::before and ::after pseudo-elements can be used to add content
before and/or after an element

```
p::before {
    content: url('minion.png');
}
```

47

CSS Pseudo-elements

LOREM IPSUM DOLOR SIT AMET, GRAVIDA
facilisi orci nulla. Erat dolor, sagittis blandit,
maecenas quis mi suscipit, risus enim tincidunt erat.
Eu justo, senectus ut mattis laoreet et.

LOREM IPSUM DOLOR SIT AMET, GRAVIDA
facilisi orci nulla. Erat dolor, sagittis blandit,
maecenas quis mi suscipit, risus enim tincidunt erat.
Eu justo, senectus ut mattis laoreet et.

48

CSS Opacity

opacity is now a part of the CSS3 specifications, but it has been present for a long time

Older browsers have different ways of controlling the opacity or transparency

opacity can be applied to images, text, colors, and elements

49

CSS Opacity

50

CSS Transforms

51

CSS Transforms

CSS 3 includes several functions that allow us to alter the appearance of an element

The all fall under the **transform** property

Most common are **scale**, **translate**, **rotate**, and **skew**

CSCI 1720
Intermediate Web Design

52

CSS Transforms - **scale**

scale will modify the element's display size

East Tennessee State University
Department of Computing
CSCI 1720
Intermediate Web Design

53

Lorem ipsum dolor sit amet, gravida facilisi orci nulla. Erat dolor, sagittis blandit, maecenas quis mi suscipit, risus enim tincidunt erat. Eu justo, senectus ut mattis laoreet et. Nec ultricies ultricies iaculis non asperiores urna, nec ut dolor dignissim, sodales purus pretium eget, wisi integer molestie arcu platea, urna ut in ipsum laoreet. Aenean fusce donec, arcu elit, placerat arcu laoreet libero eu vitae.

UNSCALED

Nibh leo sit dui nonummy lorem augue, purus libero turpis integer, at erat conubia neque, praesent eleifend placerat ridiculus magna pulvinar ipsum. Dolor odio pretium eiusmod aliquid. Mauris amet voluptatem gravida, ullamcorper id dignissim consequat suscipit, risus pulvinar nam a nam, et viverra sed, anim tincidunt quam. Sed inceptos erat eleifend tempus, augue diam, velit suscipit, platea morbi. Dolor et suspendisse, ac fusce sagittis et dui nunc hendrerit, mi fermentum ligula, pellentesque sem ipsum eros at repellat, felis aenean nec nascetur nunc suspendisse. Donec rhoncus.

54

18

55

CSS Transforms - translate

translate will modify the element's position relative to where it would have displayed otherwise

We can use translate along the x- and y-axes, or use **translateX** and **translateY** for a single axis

56

57

CSS Transforms - `rotate`

`rotate` will modify the element's orientation by rotating it
We have to specify how many degrees we want to rotate it
e.g.,

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

58

Lorem ipsum dolor sit amet, gravida facilisi orci nulla. Erat dolor, sagittis blandit, maecenas quis mi suscipit, risus enim tincidunt erat. Eu justo, senectus ut mattis laoreet et. Nec ultricies ultrices iaculis non asperiores urna, nec ut dolor dignissim, sodales purus pretium eget, wisi integer molestie arcu platea, urna ut in ipsum laoreet. Aenean fusce donec, arcu elit, placerat arcu laoreet libero eu vitae.

NOT ROTATED

Nibh leo sit dui nonummy lorem augue, purus libero turpis integer, at erat conubia neque, praesent eleifend placerat ridiculus magna pulvinar ipsum. Dolor odio pretium eiusmod aliquid. Mauris amet voluptatem gravida, ullamcorper id dignissim consequat suscipit, risus pulvinar nam a nam, et viverra sed, anim tincidunt quam. Sed inceptos erat eleifend tempus, augue diam, velit suscipit, platea morbi. Dolor et suspendisse, ac fusce sagittis et dui nunc hendrerit, mi fermentum ligula, pellentesque sem ipsum eros at repellat, felis aenean nec nascetur nunc suspendisse. Donec rhoncus.

59

Lorem ipsum dolor sit amet, gravida facilisi orci nulla. Erat dolor, sagittis blandit, maecenas quis mi suscipit, risus enim tincidunt erat. Eu justo, senectus ut mattis laoreet et. Nec ultricies ultrices iaculis non asperiores urna, nec ut dolor dignissim, sodales purus pretium eget, wisi integer molestie arcu platea, urna ut in ipsum laoreet. Aenean fusce donec, arcu elit, placerat arcu laoreet libero eu vitae.

ROTATED 30º

```
img {
    transform: rotate(30deg);
}
```

Nibh leo sit dui nonummy lorem augue, purus libero turpis integer, at erat conubia neque, praesent eleifend placerat ridiculus magna pulvinar ipsum. Dolor odio pretium eiusmod aliquid. Mauris amet voluptatem gravida, ullamcorper id dignissim consequat suscipit, risus pulvinar nam a nam, et viverra sed, anim tincidunt quam. Sed inceptos erat eleifend tempus, augue diam, velit suscipit, platea morbi. Dolor et suspendisse, ac fusce sagittis et dui nunc hendrerit, mi fermentum ligula, pellentesque sem ipsum eros at repellat, felis aenean nec nascetur nunc suspendisse. Donec rhoncus.

60

CSS Transforms - `skew`

`skew` will modify the element's orientation by skewing it along the x- or y-axis

We can specify the x- or y-axis with skewX and skewY

Or do both with skew

```
transform: skew(30deg, 30deg);
```

61

Lorem ipsum dolor sit amet, gravida facilisi orci nulla. Erat dolor, sagittis blandit, maecenas quis mi suscipit, risus enim tincidunt erat. Eu justo, senectus ut mattis laoreet et. Nec ultricies ultrices iaculis non asperiores urna, nec ut dolor dignissim, sodales purus pretium eget, wisi integer molestie arcu platea, urna ut in ipsum laoreet. Aenean fusce donec, arcu elit, placerat arcu laoreet libero eu vitae.

```
img {
    transform: skew(30deg, 30deg);
}
```

Nibh leo sit dui nonummy lorem augue, purus libero turpis integer, at erat conubia neque, praesent eleifend placerat ridiculus magna pulvinar ipsum. Dolor odio pretium eiusmod aliquid. Mauris amet voluptatem gravida, ullamcorper id dignissim consequat suscipit, risus pulvinar nam a nam, et viverra sed, anim tincidunt quam. Sed inceptos erat eleifend tempus, augue diam, velit suscipit, platea morbi. Dolor et suspendisse, ac fusce sagittis et dui nunc hendrerit, mi fermentum ligula, pellentesque sem ipsum eros at repellat, felis aenean nec nascetur nunc suspendisse. Donec rhoncus.

62

Lorem ipsum dolor sit amet, gravida facilisi orci nulla. Erat dolor, sagittis blandit, maecenas quis mi suscipit, risus enim tincidunt erat. Eu justo, senectus ut mattis laoreet et. Nec ultricies ultrices iaculis non asperiores urna, nec ut dolor dignissim, sodales purus pretium eget, wisi integer molestie arcu platea, urna ut in ipsum laoreet. Aenean fusce donec, arcu elit, placerat arcu laoreet libero eu vitae.

```
img {
    transform: skewX(30deg);
}
```

Nibh leo sit dui nonummy lorem augue, purus libero turpis integer, at erat conubia neque, praesent eleifend placerat ridiculus magna pulvinar ipsum. Dolor odio pretium eiusmod aliquid. Mauris amet voluptatem gravida, ullamcorper id dignissim consequat suscipit, risus pulvinar nam a nam, et viverra sed, anim tincidunt quam. Sed inceptos erat eleifend tempus, augue diam, velit suscipit, platea morbi. Dolor et suspendisse, ac fusce sagittis et dui nunc hendrerit, mi fermentum ligula, pellentesque sem ipsum eros at repellat, felis aenean nec nascetur nunc suspendisse. Donec rhoncus.

63

CSS Animation

64

CSS Animation

We can create animations with CSS3

This is a little more complex

The first step is to create an **@keyframes** directive and give it a name

```
@keyframes fade-in {

}
```

65

CSS Animation

Then, we add rules to define how the element will appear at a given point during the sequence

Easiest is from/to

```
@keyframes fade-in {
    from { opacity: 0; }
    to { opacity: 1; }
}
```

*note: from and to can have more than one property/value set

66

22

CSS Animation

Then, we can associate each element we want to animate with the **@keyframes** directive we created

```
img {
    animation-name: fade-in;
    animation-duration: 2s;
}
```

We **also** have to specify a duration over which the animation will take place

67

CSS Animation

```
img {
    animation-name: fade-in;
    animation-duration: 2s;
}

@keyframes fade-in {
    from { opacity: 0; }
    to { opacity: 1; }
}
```

68

CSS Animation

@keyframes directives can be used by multiple selectors on a page

69

CSS Animation

```
img {
    animation-name: fade-in;
    animation-duration: 2s;
}

h1 {
    animation-name: fade-in;
    animation-duration: 2s;
}

@keyframes fade-in {
    from { opacity: 0; }
    to { opacity: 1; }
}
```

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

70

CSS Animation

We can get more creative/fine-grained with the **@keyframes** directive by specifying percentages instead of from/to

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

71

CSS Animation

```
@keyframes fade-in {
    0 {
        opacity: 0;
    }
    10% {
        opacity: .2;
    }
    60% {
        opacity: .5;
    }
    100% {
        opacity: 1;
    }
}
```

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

72

Lorem ipsum dolor sit amet, gravida facilisi orci nulla. Erat dolor, sagittis blandit, maecenas quis mi suscipit, risus enim tincidunt erat. Eu justo, senectus ut mattis laoreet et. Nec ultricies ultrices iaculis non asperiores urna, nec ut dolor dignissim, sodales purus pretium eget, wisi integer molestie arcu platea, urna ut in ipsum laoreet. Aenean fusce donec, arcu elit, placerat arcu laoreet libero eu vitae.

Nibh leo sit dui nonummy lorem augue, purus libero turpis integer, at erat conubia neque, praesent eleifend placerat ridiculus magna pulvinar ipsum. Dolor odio pretium eiusmod aliquid. Mauris amet voluptatem gravida, ullamcorper id dignissim consequat suscipit, risus pulvinar nam a nam, et viverra sed, anim tincidunt quam. Sed inceptos erat eleifend tempus, augue diam, velit suscipit, platea morbi. Dolor et suspendisse, ac fusce sagittis et dui nunc hendrerit, mi fermentum ligula, pellentesque sem ipsum eros at repellat, felis aenean nec nascetur nunc suspendisse. Donec rhoncus.

73

CSS Animation

There are several other animation properties we can use to further modify an animation, including

```
img {
    animation-name: fade-in;
    animation-duration: 2s;
    animation-fill-mode: forwards;
    animation-iteration-count: infinite;
    animation-delay: 5s;
    animation-direction: reverse;
    animation-play-state: running;
    animation-timing-function: ease-in;
}
```

East Tennessee State University — CSCI 1720
Department of Computing — Intermediate Web Design

74

CSS Animation

Typically, despite these examples, animations are applied to specialized classes or IDs instead of general selectors

The browser will make a 'best effort' to smooth the transitions between different **@keyframe** states

A lot of times, at least in my experience, it takes a bit of fiddling with the numbers to get the effect you want

East Tennessee State University — CSCI 1720
Department of Computing — Intermediate Web Design

75

25

CSS Animation

JavaScript can also be used to create animations

Generally speaking, though, CSS Animations are more lightweight and require less processing from the client computer

CSS Animations are best suited for relatively simple tasks that can easily be handled by a browser

We'll see this Thursday

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

76

QUESTIONS?

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

77

Copyrights

Presentation prepared by and copyright of John Ramsey, East Tennessee State University, Department of Computing . (ramseyjw@etsu.edu)

•Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

•IBM, DB2, DB2 Universal Database, System i, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, iSeries, pSeries, xSeries, zSeries, eServer, zVM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation.

•Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

•Oracle is a registered trademark of Oracle Corporation.

•HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

•Java is a registered trademark of Sun Microsystems, Inc.

•JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

•SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.

•Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects S.A. in the United States and in other countries. Business Objects is an SAP company.

•ERPsim is a registered copyright of ERPsim Labs, HEC Montreal.

•Other products mentioned in this presentation are trademarks of their respective owners.

East Tennessee State University
Department of Computing

CSCI 1720
Intermediate Web Design

78