Browser Capabilities

Designing for consistent appearance

Browsers Differ

- Even though browsers are moving to a consistent implementation of HTML, they differ in display and adherence.
- It is your responsibility to make sure your page works for a wide audience.

Handling Stylistic Differences

- “Easiest” way to eliminate browser differences is to use a default style sheet
- Default style sheets reset all of the values for the page
- Will make your page look worse!

Handling Unsupported Properties

- Not all browsers support all HTML5 tags
- Not all browsers support all CSS3 properties
- Browser prefixes (or vendor prefixes) provide a quick fix for handling unsupported CSS3 options.

Browser Prefixes

- -webkit-: Android, Chrome, iOS, Safari
- -moz-: Firefox
- -ms-: Internet Explorer
- -o-: Opera

Often Unsupported Properties

- column-count
- border-radius**
- gradient
- Sites such as http://caniuse.com/ will tell you when you need to use prefixes

Automated Ways to include Prefixes

- For now, add the prefixes by hand
- There are ways to automate the addition of prefixes
  - Editor add-ons (You have most of the control)
  - Use outside programs to dynamically add appropriate prefix based on browser

Transforms

Transforms

- Provide option for changing the appearance of elements
  - Two-dimensional
  - Three-dimensional

2D Transform Options

- Options
  - translate
  - rotate
  - scale
  - skew
  - matrix

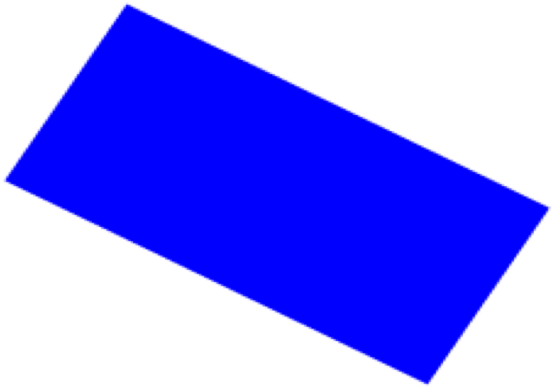
translate

- transform:translate(x, y);
  - move x pixels to the left/right and y pixel up/down

    transform:translate(100, 75);

rotate

- transform:rotate(deg);
  - Rotate/”spin” the element a certain number of degrees
- transform:rotate(30deg);

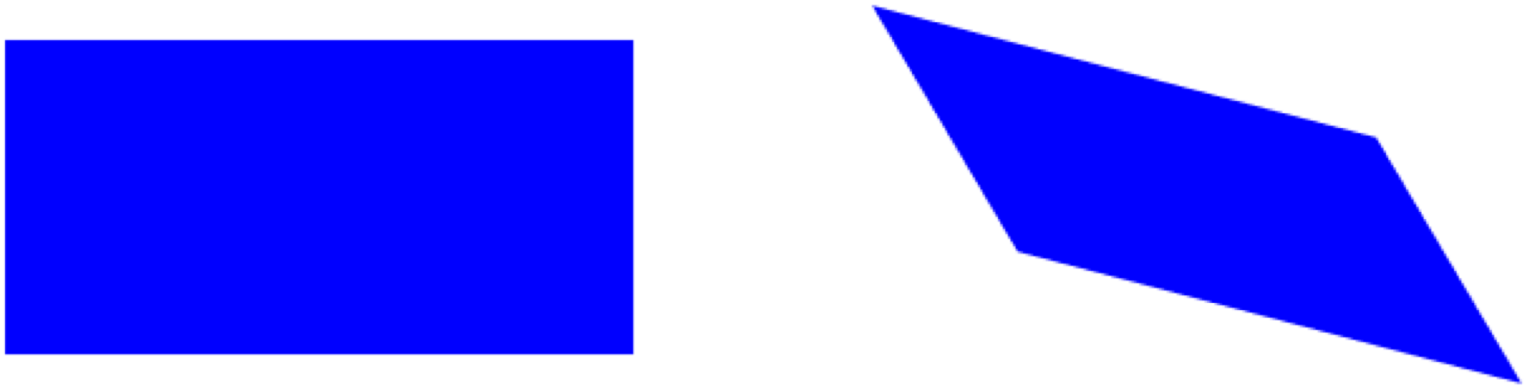


scale

- transform:scale(width, height);
  - Change the width and height of the element
- transform:scale(2,3);

skew

- transform:skew(x-angle, y-angle);
  - Rotate the element a certain number of degrees along the x and y axis
- transform:skew(30deg, 15deg)



matrix

- matrix() - combines all of the 2D transform methods into one

3D rotate

- You can rotate along the x, y, or z dimension along a given degree
  - transform:rotateY(deg)
  - transform:rotateX(deg)
  - transform:rotateZ(deg)
  - transform:rotate3d(x, y, z)

Others

- 3D scale
- 3D translate

Review

- Transforms are one more way to modify the look of your page.
- Often combined with state changes
- Will typically require browser prefixes.

Transitions

Transitions

- When elements transition from one state to another, you can alter their appearance
  - If you hover over the link, change the color
  - If an image comes into focus, change the size,

The Properties

- transition-property
  - What is it you want to change? (size, color, position, etc.)
- transition-duration
  - How long should each transition last?
- transition-timing
  - Should it be a smooth transition (linear)? Or different?
- transition-delay
  - How long should the wait be before the transition begins?

Setting up

1. Define your element
2. Choose the elements for transition
3. Define the new values
    - You must combine this step with a pseudo-class

Example (CSS3-transitions)

```
div {
    color: #000000;
    background: #2db34a;
    line-height: 200px;
    text-align: center;
    width: 250px;
    height:200px;
    border-radius: 6px;
}
```

Using Shorthands

- If you have multiple properties transitioning, you can use shorthand:

```
transition:background .2s linear,
border-radius 1s ease-in 1s;
```

Examples

- https://codepen.io/ColleenEMc/pen/zvPpRL
- https://codepen.io/ColleenEMc/pen/YyEYem?editors=1100

Acknowledgements

- Initial Development: Colleen van Lent , University of Michigan School of Information