

Lesson 5 Extra Challenge 1

IMAGES WITH CSS TRANSITIONS

In this lab we'll experiment with hover effects on images.

THE HTML

1. Create a new project called HoverImages
2. Add html document called index.html with the usual html, head, and body tags.
3. Add the images file to the same folder as your index.html page
4. Add the following between the opening and closing <body> tags.

```
<div class="container">

    <h1>Original Hover Effects <span>with CSS3 Transitions and Animations</span></h1>

    <div class="main">
        <div class="view view-first">
            
            <div class="mask">
                <h2>Hover Style #1</h2>
                <p>A wonderful serenity has taken possession of my entire soul,
like these sweet mornings of spring which I enjoy with my whole heart.</p>
                <a href="#" class="info">Read More</a>
            </div>
        </div>
    </div>
</div>
```

Load the html in a browser and view results. You should see the image and the text below.

THE CSS

1. Create a new CSS page called style.css and save it to the css folder within your main folder (same as index.html)
2. we're going to start styling our image panel and text panel.
3. Define a style for the view and the main container:

```
.main {
    position: relative;
```

```

        width: 680px;
        margin: 0 auto;
    }
    .view {
        width: 300px;
        height: 200px;
        margin: 10px;
        display: inline-block;
        border: 10px solid #fff;
        overflow: hidden;
        position: relative;
        text-align: center;
        box-shadow: 1px 1px 2px #e6e6e6;
        cursor: default;
    }
}

```

Load html in browser and view results. Where has our text gone ?

As we have added **overflow: hidden** to the css any content that extends beyond the height or width of the containing unit will not appear. As our image is 300x200 in dimension it occupies the entire space of the element. Remove this **overflow: hidden** property and see what happens.

Replace the **overflow: hidden** property.

4. We want to reposition the text panel so that it is positioned over the image when display. To do this we make it absolutely positioned relative to the parent div (.view), which has been relatively positioned. We give it the same dimension so that it fits perfectly over the image.

```

.view .mask {
    width: 300px;
    height: 200px;
    position: absolute;
    overflow: hidden;
    top: 0;
}

```

```
left: 0  
}
```

5. Now we want to style the text elements

```
.view h2 {  
  
    text-transform: uppercase;  
  
    color: #fff;  
  
    text-align: center;  
  
    position: relative;  
  
    font-size: 17px;  
  
    padding: 10px;  
  
    background: rgba(0, 0, 0, 0.8);  
  
    margin: 20px 0 0 0  
  
}  
  
.view p {  
  
    font-family: Georgia, serif;  
  
    font-style: italic;  
  
    font-size: 12px;  
  
    position: relative;  
  
    color: #fff;  
  
    padding: 10px 20px 20px;  
  
    text-align: center  
  
}
```

```

.view a.info {

    display: inline-block;

    text-decoration: none;

    padding: 7px 14px;

    background: #000;

    color: #fff;

    text-transform: uppercase;

    box-shadow: 0 0 1px #000

}

```

Refresh browser and observe changes.

Finally add a box shadow when we hover over the link element

6. Now we want to add our effects. We will use the .view-first class to add our CSS3 effects.
7. First make the text panel disappear (using the .mask class) and add a transition which will take effect when we hover over the element and change the opacity property to 1 (visible).

```

.view-first .mask {

    opacity: 0; /* make it disappear */
    background-color: rgba(219,127,8, 0.7); /*give it a color */
    transition: all 0.4s ease-in-out;

}

```

Next we translate the header text upwards by 100 px . When we hover over the element we will translate it back to its original position

```

.view-first h2 {

    transform: translateY(-100px);
    transition: all 0.2s ease-in-out;

}

```

We do the same to the paragraph text, but in the downward direction

```
.view-first p {  
    transform: translateY(100px);  
    transition: all 0.2s linear;  
}
```

8. How we add our transition effect on the hover event

```
.view-first:hover img {  
    transform: scale(1.1);  
}  
.view-first:hover .mask {  
    opacity: 1; /* when opacity changes transition kicks  
in*/  
}  
.view-first:hover h2,  
.view-first:hover p,  
.view-first:hover a.info {  
    opacity: 1; /* causes transition */  
    transform: translateY(0px); /*revert back to 0  
position*/  
}  
.view-first:hover p {  
    transition-delay: 0.1s;  
}  
.view-first:hover a.info {  
    transition-delay: 0.2s;  
}
```

The link element <a> has 0 opacity and a transition (we want to fade it in on hover but with a delay) , experiment with the transition delay above to make it appear later .

9. The above challenge is based on a tutorial available at <http://tympanus.net/codrops/2011/11/02/original-hover-effects-with-css3/> there are several other effects you can experiment with. To add a second panel add

```
<div class="view view-second">
  
  <div class="mask"></div>
  <div class="content">
    <h2>Hover Style #2</h2>
    <p>Some description</p>
    <a href="#" class="info">Read More</a>
  </div>
</div>
```

inside the main <div> and continue with the tutorial. One change if you want to use this example add

```
.view .content
```

so that

```
.view .mask {
  width: 300px;
  height: 200px;
  position: absolute;
  overflow: hidden;
  top: 0;
  left: 0
}
```

becomes

```
.view .mask, .view .content {
  width: 300px;
  height: 200px;
  position: absolute;
```

```
overflow: hidden;  
top: 0;  
left: 0  
}
```