

## Project 6 – A More Robust Image Rotator

In the previous class, we built an image rotator. But it had a severe limitation. Maybe you didn't notice it in class.

### Examine the Current Page

1. Load **rotate-different-sizes.htm** in your browser and notice the image rotator work exactly as it did in class. That's because the images were all the same size.
2. Open **rotate-different-sizes.htm** in your text editor and locate the four lines of `<img>` tags:

```

```

```

```

```

```

```

```

3. In each of the tags, delete the width and height attributes:

```

```

```

```

```

```

```

```

4. Open the CSS file in your text editor and delete both the width and height declarations from the first rule.

```
#photoShow {  
    height: 445px;  
    width: 445px;  
    border: 3px double #333;  
    padding: 10px;  
    background: #ccc;  
}
```

5. Save your changes and reload the page in your browser.
6. Now that the images are different sizes, the rotator no longer works as intended.
7. So...

## Fix It

Here's what should happen, in no particular order:

- Do not modify any more CSS.
- Do not modify any more HTML.
- Only function calls should appear in the `document.ready` function. No anonymous functions allowed in there.
- Edit the existing `rotateImages()` function to make it work properly.
- Create a second function to initialize the images. (Whatever you think needs to happen on page load.) You may want to set this new function to fire first since it initializes the page.
- Only the 'top' image, Andy Timmons, should be visible on page load. The containing DIV should be set to match the dimensions of the Andy Timmons image.
- As Andy Timmons fades out and the next image fades in, the DIV container should animate to match the size of the new image. The DIV should continue to animate its dimensions to accommodate each image.
- Change the speed of the image rotation to rotate every 4 seconds instead of every 2 seconds.
- Change the speed of the fade in from 1 second to 2 seconds.

## Things You May Need That Are Not In The Chapter But Were Covered In Class

- `$(object).width()` gets the width as an integer
- `$(object).width(someVar + 'px')` sets the width and appends 'px' as the unit of measure
- `$(object).height()` gets the height as an integer
- `$(object).height(someVar + 'px')` sets the height and appends 'px' as the unit of measure
- `$(object).next()` gets the next sibling after the specified object