Generating Random using SASS

> We are not using prefixes in the SCSS code. During the build process we'll add prefixes where necessaary using Autoprefixer.

I've been looking for projects to work in and saw [this piece](#) from Dudley Storey's The New Code. This made me think about whether we can create a similar effect just with CSS transitions and animations.

In doing so we'll explore some newer features of SASS like the new `random()` function and work with viewport measurements. I'm not trying to fully duplicate the project from The New Code but see how much of it is doable without bringing the Web Animationn API.

SASS `random()` function

SASS had had a random function since version 3.3. We can use this to generate random values and, what I think it more useful for this project is to create a random value between a negative number and a positive one. This way we can randomly rotate an object. An example of an image roated between -45 and 45 degrees may look like this:

```scss
.roated-image {
  transform: rotate(random (45) -  90 + deg);
}
```

We can also randomize other aspects of we place the image on screen.

```scss
$rotation: (random(45) - 46 + 'deg');

@keyframes reveal {
  to {
```

```scss
    opacity: 1;
    transform:rotate($rotation);
  }
}

.shuffle {
  min-height: 100vh;
  position: relative;
}
.shuffle img {
  opacity: 0;
  transform: scale(.8);
}

.reveal img {
  animation: reveal 1s 1s forwards;
}
```

The images rotate but they all rotate by the same amount. This happens because I've coded the animation in such a way that it'll use the same value for all the items that match the class.

This would be easier to work with the Web Annimations API but, as I said earlier, I want to see if it's possible to create similar effects with just CSS Keyframe animations so it's back to the drawing board.

For a second iteration I think we need to give each image in the page a different rootationn value. In SASS we can build functions to manage the animation and a for loop to iterate through the items in a list.