

lesson4.2

Go to file

Code

About

This branch is 5 commits ahead, 4 commits behind master.

No description, website, or topics provided.

Readme

Activity

8 stars

7 watching

9 forks

Report repository

Releases

No releases published

Packages

No packages published

Contributors2





spxds

donovanh Donovan Hutchinson

Languages

CSS93.2%

HTML6.8%

	guyroutledge Add start code samples ...	on Feb 6, 2017	6
	04/start Add start code samples		6 years ago
	CSS_Animations_hando... uploading handouts and updating readme		7 years ago
	README.md uploading handouts and updating readme		7 years ago

README.md



In this lesson, we'll go through each of the examples of timing functions and see them in action. Let's begin with the HTML:

```
<html>
  <head>
    <title>Lesson 04 - Cubic-bezier</title>
    <link href='./stylesheets/defaults.css' rel='stylesheet'>
    <link href='./stylesheets/02-cubic-bezier.css' rel='stylesheet'>
  </head>
  <body>
    <h2>Cubic Bezier curves</h2>
    <h3>linear (<a href='http://cubic-bezier.com/#0,0,1,1'>source</a></h3>
    <div class='demo linear'></div>
    <h3>ease-in (<a href='http://cubic-bezier.com/#1,0,1,1'>source</a></h3>
    <div class='demo ease-in'></div>
    <h3>ease-out (<a href='http://cubic-bezier.com/#0,1,1,1'>source</a></h3>
    <div class='demo ease-out'></div>
    <h3>ease-in-out (<a href='http://cubic-bezier.com/#1,0,0,1'>source</a></div>
    <div class='demo ease-in-out'></div>
    <h3>Custom (<a href='http://cubic-bezier.com/#0,.9,.5,1.4'>source</a></h3>
    <div class='demo custom-1'></div>
    <h3>Custom (<a href='http://cubic-bezier.com/#.3,1.5,.7,-0.5'>source</a></div>
    <div class='demo custom-2'></div>
  </body>
</html>
```

Each demo element has a class that relates to the timing function we'll apply.

Now open the CSS file and start with linear timing function by setting the animation-timing-function to "linear":

```
.linear {
  -webkit-animation-timing-function: linear;
  animation-timing-function: linear;
}
```

Then, "ease-out":

```
.ease-out {
  -webkit-animation-timing-function: ease-out;
```

```
    animation-timing-function: ease-out;
}
```

Next, we'll add in "ease-in-out", this is the combination of both the ease-in and ease-out timing functions, which shows a starting slowly and also slowing down toward the end:

```
.ease-in-out {
  -webkit-animation-timing-function: ease-in-out;
  animation-timing-function: ease-in-out;
}
```



Next let's add a cubic-bezier:

```
.cubic-bezier {
  -webkit-animation-timing-function: cubic-bezier(.5,-0.5,.5,1.5);
  animation-timing-function: cubic-bezier(.5,-0.5,.5,1.5);
}
```



Lastly we'll set the "steps" timing function:

```
.steps {
  -webkit-animation-timing-function: steps(3);
  animation-timing-function: steps(3);
}
```



It's interesting to note that the animation duration in each of these examples is the same, about two seconds each direction. Each circle starts and stops at the same time, yet the feel of each animation is very different.

Let's round off the discussion of timing functions with an exploration of the cubic-bezier timing function. In this example we will be using the same HTML CSS as before. But instead of the built in timing functions, we'll use cubic-bezier curve to create our own. Open a tool I'd like to use to create cubic-bezier curves, the website cubic-bezier.com. There are many tools around for creating cubic-bezier curves but this is one of my favorite. It's just a handy way to create the numbers we'll use in our CSS.

On the left, we have a graph with a red and blue handle, these are the cubic-bezier points. The graph shows how an animation progresses over time, with time being the x-axis and progression being the y-axis. The shape on the black line is the speed at which the animation proceeds. The steeper this is, the faster the animation moves. Pressing "go" beside the "Preview and Compare" heading allows us to see how this compares to other animation timing functions.

Going back to our CSS, create various timing functions:

```
linear {
  -webkit-animation-timing-function: cubic-bezier(0,0,1,1);
  animation-timing-function: cubic-bezier(0,0,1,1);
}

.ease-in {
  -webkit-animation-timing-function: cubic-bezier(1,0,1,1);
  animation-timing-function: cubic-bezier(1,0,1,1);
}

.ease-out {
  -webkit-animation-timing-function: cubic-bezier(0,1,1,1);
  animation-timing-function: cubic-bezier(0,1,1,1);
}

.ease-in-out {
  -webkit-animation-timing-function: cubic-bezier(1,0,0,1);
  animation-timing-function: cubic-bezier(1,0,0,1);
}

.custom-1 {
  -webkit-animation-timing-function: cubic-bezier(0,.9,.5,1.4);
  animation-timing-function: cubic-bezier(0,.9,.5,1.4);
}
```



```
.custom-2 {  
  -webkit-animation-timing-function: cubic-bezier(.3,1.5,.7,-0.5);  
  animation-timing-function: cubic-bezier(.3,1.5,.7,-0.5);  
}
```

I urge you to experiment with it. See what different curves do to your animations, and find ways they can make your animation stand out from the crowd.