根据她的以往经验，这个网站的 **HTML** 和 **CSS** 是完全在她的能力范围以内，于是我帮助她完成了这个效果。显示 **nav** 被裁减的效果是一个不简单的任务，我看到图片的第一反应是创建一个相匹配的背景被部分裁剪的图像，然后把它设置为一个 **after** 元素。问题是，至少要解决响应式问题，并且响应式并不完全可控的。

### **了解CSS属性：clip-path**

**clip-path** 是工作草案的一部分，它是一个通过屏蔽和裁减来隐藏元素的一部分的工具。尽管 **clip-path** 没有被主流的浏览器支持（包括IE和Firefox），但在 **webkit**浏览器中，它仍然是一个实现时尚效果的小工具。

注意，在现代浏览器中需要使用 **-webkit-** 前缀。

**clip-path** 简单的工作原理是提供一系列的 **X** 和 **Y** 值来创建路径。当使用这些值创建一条完整路径时，就会把图像按照路径内部的尺寸进行裁剪。

利用 **clip-path** ，我们可以创建圆形、椭圆和多边形等不同的形状，创造力是唯一的限制。

### **一个简单的三角形裁减**

***[View the code on codepen](http://codepen.io/drewminns/pen/emeVLz" \t "http://www.tuicool.com/articles/_blank)*** 对元素简单地运用 **clip-path** 就能实现上面的效果：

1. **.clipClass {**
2. **-webkit-clip-path: polygon(0 100%, 50% 0, 100% 100%);**
3. **}**

### **逐步分析**

很像定位属性，我们需要考虑 **X** 值和 **Y** 值。 **X:0** 和 **Y:0** 表示从元素的左上角开始，并从左上角开始移动。 **X:100%** 指的是元素右边， **Y:100%** 指的是元素底部。

对于上面创建的路径，实际是创建了如下的点：

1. **x: 0, y:100%**
2. **x: 50%, y: 0**
3. **x: 100%, y: 100%**

这个简单路径开始于左下角，水平移动50%，并到达顶部位置，然后又水平移动到100%的位置，垂直向下回到底部，到达第三个坐标点。三角形就出来了。

### **形状**

在上面的示例中，我们使用 **polygon** 来创建一个图形，并通过多对用逗号（ **,** ）分开的 **X** 值和 **Y** 值定义了一个路径。然后，我们可以通过取不同的值来创建不同的图形。

#### **圆**

***[View the code on codepen](http://codepen.io/drewminns/pen/azVYmq" \t "http://www.tuicool.com/articles/_blank)*** 为了创建圆形，需要给 **circle** 传入三个值：圆心的坐标（X值和Y值）以及半径。当定义圆的半径时，我们可以用 **at** 关键字来定义圆心坐标。

1. **.clipClass {**
2. **-webkit-clip-path: circle(50% at 50% 50%);**
3. **}**

#### **椭圆**

***[View this code on codepen](http://codepen.io/drewminns/pen/qEVoaK" \t "http://www.tuicool.com/articles/_blank)*** 很多时候，你不需要一个简单的圆，而是一个椭圆。

为了实现椭圆，需要给 **ellipse** 提供4个值：椭圆的 **x** 轴半径、 **y** 轴半径、定位椭圆位置的 **x** 坐标和 **y** 坐标，后面两个值用 **at** 关键字和前面两个值分开。

1. **.clipClass {**
2. **-webkit-clip-path: ellipse(30% 20% at 50% 50%);**
3. **}**

#### **插图**

( *在老版本的chrome中有错误* )



***[View the code on codepen](http://codepen.io/drewminns/pen/myqxrg" \t "http://www.tuicool.com/articles/_blank)*** 因为多边形边缘锐利，所以它可能不是你想要的东西，你想创建的是圆角矩形，所以我们来看看Inset的值。Inset使用四个值（对应“ ****上 右 下 左**** ”的顺序）来设置 ****圆角半径**** 。

1. **.clipClass {**
2. **-webkit-clip-path: inset(25% 0 25% 0 round 0 25% 0 25%);**
3. **}**

上面的各个值分别对应为：

1. **inset(<top> <right> <bottom> <left> round <top-radius> <right-radius> <bottom-radius> <left-radius>)**

其简写形式：

1. **.clipClass {**
2. **-webkit-clip-path: inset(25% 0 round 0 25%);**
3. **}**

#### **快速参考**

* Circle: circle(radius at x-axis y-axis)
* Ellipse: ellipse(x-rad y-rad at x-axis y-axis)
* Polygon: polygon(x-axis y-axis, x-axis y-axis, … )
* Inset: inset(top right bottom left round top-radius right-radius bottom-radius left-radius)

### **创建形状**

正如你看到的，原型和圆角图形被少数几个值限制了,因而 **Polygons** 是创建复杂图形的最好选择。 **Polygons** 能定义多组点，允许我们用各种方式去裁图形。

#### **Comic Textbox**

***[View the code on codepen](http://codepen.io/drewminns/pen/WbXzbb" \t "http://www.tuicool.com/articles/_blank)***

1. **.clipClass {**
2. **-webkit-clip-path: polygon(0% 0%, 100% 0%, 100% 75%, 75% 75%, 75% 100%, 50% 75%, 0% 75%);**
3. **}**

#### **Star**

***[View the code on codepen](http://codepen.io/drewminns/pen/zxPWxd" \t "http://www.tuicool.com/articles/_blank)***

1. **.clipClass {**
2. **-webkit-clip-path: polygon(50% 0%, 63% 38%, 100% 38%, 69% 59%, 82% 100%, 50% 75%, 18% 100%, 31% 59%, 0 38%, 37% 38%);**
3. **}**

### **动画**

现在我们已经了解了各种图形和怎么创建它们，那怎么利用这些图形创建我们想要的效果呢？

给形状应用一个 **hover** ，并用过渡属性来创建平滑的效果。但是需要记住，我们创建的初始默认状态，必须同所有的 **hover** 状态都使用相同的坐标系。

***[View the code on codepen](http://codepen.io/drewminns/pen/VYrXvG" \t "http://www.tuicool.com/articles/_blank)***

1. **.animateClass {**
2. **-webkit-clip-path: polygon(20% 0%, 0% 0%, 0% 50%, 0% 80%, 0% 100%, 50% 100%, 80% 100%, 100% 100%, 100% 50%, 100% 0, 80% 0, 50% 0);**
3. **}**
4. **.animateClass:hover {**
5. **-webkit-clip-path: polygon(50% 0%, 0% 20%, 30% 50%, 0% 80%, 20% 100%, 50%70%, 80% 100%, 100% 80%, 70% 50%, 100% 20%, 80% 0%, 50% 30%);**
6. **}**