**Padding %**

This can be achieved by giving the element height:0 and padding-bottom:30%.

In all browsers, when padding is specified in %, it's calculated relative to the parent element's width. This can be a very useful feature for Responsive Design.

[JSFiddle Demo](http://jsfiddle.net/Matt_Coughlin/agtyt/3/)

In the [demo](http://jsfiddle.net/Matt_Coughlin/agtyt/3/), the blue box at the top of the page is a single div. The height is responsive, and it can contain a variable amount of inline content without increasing the height. It tested fine in IE7/8/9/10, Firefox, Chrome, Safari, and Opera.

.content {

width: 100%;

height: 0;

padding-bottom: 30%;

}

...

<div class="content"></div>

**Padding % and absolute position**

If there are any problems with using a 0-height element, the [demo](http://jsfiddle.net/Matt_Coughlin/agtyt/3/) also has a green box that's implemented using a wrapper element and absolute position. Not sure if there are any advantages to this approach.

.wrapper {

position: relative;

width: 100%;

padding-bottom: 30%;

}

.wrapper .content {

position: absolute;

width: 100%;

height: 100%;

}

...

<div class="wrapper">

<div class="content"></div>

</div>