

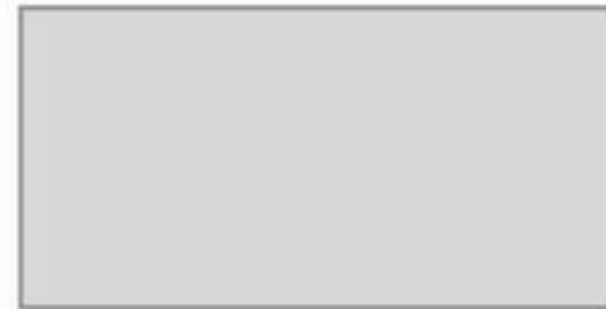
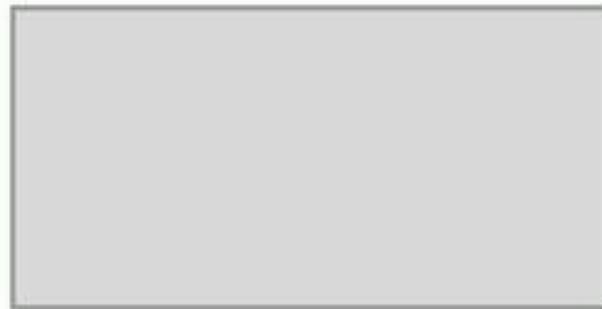
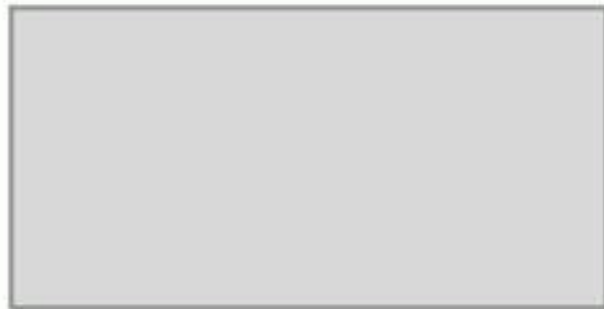
SO YOUR DESIGNER WANTS A MASTHEAD...

By [Chen Hui Jing](#) / [@hj_chen](#)

LOOKS FAMILIAR?

*Some (hopefully)
relevant text*

Be thankful they didn't ask for a carousel



Probably some more marketing copy for each of the 3 images above that will only get passed to you after you've built most of site, which is never the length of the lorem ipsum your designer used and hence he/she will ask you how come the columns cannot be the same height.

GET IMAGE RATIO



Ratio = Height / Width

$1057 / 2560 = 0.41289$ or 41.289%

SCENARIO #1

Designer wants the masthead to maintain its aspect ratio regardless of screen size.

OPTION 1: USE PADDING BOTTOM

```
div {  
  background-size: contain;  
  padding-bottom: 41.289%;  
}
```

contain tells the browser to always show the entire image, even if it means there is empty space left in the container.

padding-bottom value gives the container a height corresponding to the image ratio

OPTION 2: USE VIEWPORT UNITS

```
div {  
  background-size: cover;  
  height: 41.289vw;  
}
```

cover tells the browser to always cover the entire container, even if the sides have to be cut off.

height value gives the container a height corresponding to the image ratio, because of the relative unit viewport width.

OPTION 3: DO NOTHING

```
img {  
    max-width: 100%;  
}
```

If you use a content image, you don't need anything.

Maybe not entirely nothing. `max-width: 100%` makes sure the image doesn't overflow the container.

May occur if you're working with a CMS.

SCENARIO #2

Designer wants the masthead to have a minimum height (otherwise the magnificent image's focal point will end up too tiny to make sense).

OPTION 1: ADJUST WITH BACKGROUND-POSITION

```
div {  
  background-size: cover;  
  background-position: 75% 10%;  
  min-height: 480px;  
}
```

For background images applied on a container, just use **cover**.

Control the position of the "crop" based on percentage values along the x-axis and y-axis respectively.

OPTION 2: USE OBJECT-FIT

```
img {  
  width: 100%;  
  min-height: 480px;  
  object-fit: cover;  
  object-position: 75% 10%;  
}
```

`object-fit: cover` behaves similarly to
`background-size: cover`.

`object-position` behaves similarly to
`background-position`.

Again, may be relevant if you're working with a CMS.

HOW ABOUT TEXT?

If it was up to me...

(͡° ͜° ͡°) ͡° ͜° ͡°

USE POSITION: ABSOLUTE;

This removes the text from the normal document flow, so all the previous examples will work fine.

Just remember to set the **position: relative** property on the parent container.

USE CALC()

If your text is within the normal document flow, it takes up space in the **div**.

This extra space needs to be offset, using **calc()** is a good option.

```
h1 {  
  font-size: 5em;  
}  
.background {  
  background-size: cover;  
  height: calc(41.289vw - 5em);  
}
```

Need to also account for margins and padding, if any.

TO FIND OUT MORE...

- [Scaling background images](#)
- [background-position on MDN](#)
- [object-fit on MDN](#)
- [object-fit on CSS-Tricks](#)
- [object-position on MDN](#)

THE END

 <http://www.chenhuijing.com>

 @hj_chen

 @hj_chen

 @huijing