

Vertical Rhythm on the web

Or how I almost **lost my mind** and **half a year**
coming to terms with vertical rhythm on the web

What is Vertical Rhythm?

Mauris ac neque metus. Phasellus pretium quis ligula vel varius. Donec ut
tristique libero. Cras erat tortor, malesuada id dapibus euismod, dapibus
vitae dolor. Integer nec massa bibendum, posuere erat ac, pretium nisl.
Integer mauris tortor, bibendum sed elit id, mollis luctus elit.

Lorem ipsum dolor sit amet

Mauris ac neque metus. Phasellus pretium quis ligula vel varius. Donec ut
tristique libero. Cras erat tortor, malesuada id dapibus euismod, dapibus
vitae dolor. Integer nec massa bibendum, posuere erat ac, pretium nisl.
Integer mauris tortor, bibendum sed elit id, mollis luctus elit.

VIVAMUS TEMPUS EGET URNA EU SUSCIPIT.

Phasellus mollis vehicula leo at interdum. In ac lacinia tortor. Nunc sit
amet neque et quam lobortis sollicitudin eu vitae metus. Proin ut maximus
dolor. Praesent in ex nunc. Curabitur accumsan blandit arcu. Praesent
quis suscipit velit, at Vestibulum blandit eget urna vel pretium.

Vivamus tempus eget	Lorem ipsum dolor sit amet
urna eu suscipit.	
Phasellus mollis	
vehicula leo at	
interdum. In ac lacinia	Mauris ac neque metus. Phasellus pretium quis ligula vel varius.
tortor. Nunc sit amet	Donec ut tristique libero. Cras erat tortor, malesuada id dapibus
neque et quam	euismod, dapibus vitae dolor. Integer nec massa bibendum,
lobortis sollicitudin eu	posuere erat ac, pretium nisl. Integer mauris tortor, bibendum sed
vitae metus. Proin ut	elit id, mollis luctus elit.
maximus dolor.	
Praesent in ex nunc.	VIVAMUS TEMPUS EGET URNA EU SUSCIPIT.
	Phasellus mollis vehicula leo at interdum. In ac lacinia tortor. Nunc
	sit amet neque et quam lobortis sollicitudin eu vitae metus. Proin
	ut maximus dolor. Praesent in ex nunc. Curabitur accumsan
	blandit arcu. Praesent quis suscipit velit, at Vestibulum blandit
	eget urna vel pretium.

Vertical rhythm could help us
annotate and review designs faster.

Search



Paperless[®]
POST

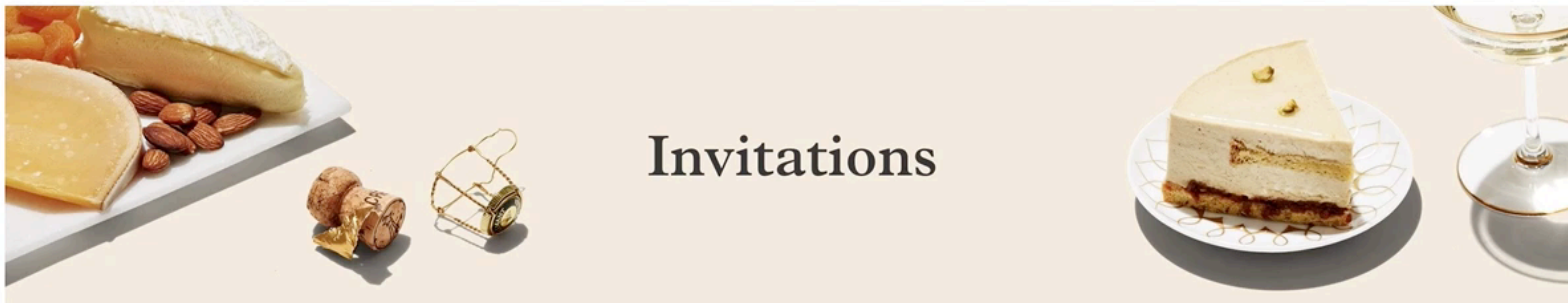
Log in | Sign up

ONLINE INVITATIONS

ONLINE CARDS

PAPER

WEDDING



Invitations

Birthday

Adult birthday

Kids' birthday

Milestone

Surprise party

[more](#)

Baby and kids

Baby shower

Kids' birthday

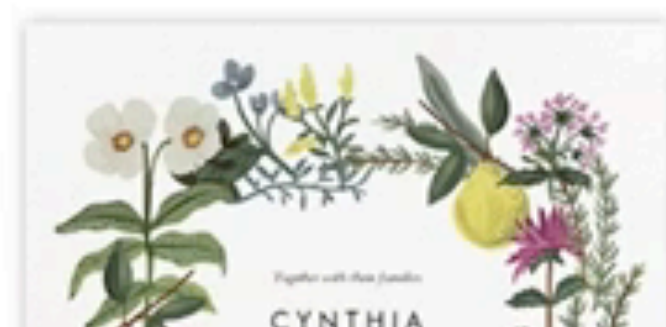
Online designs

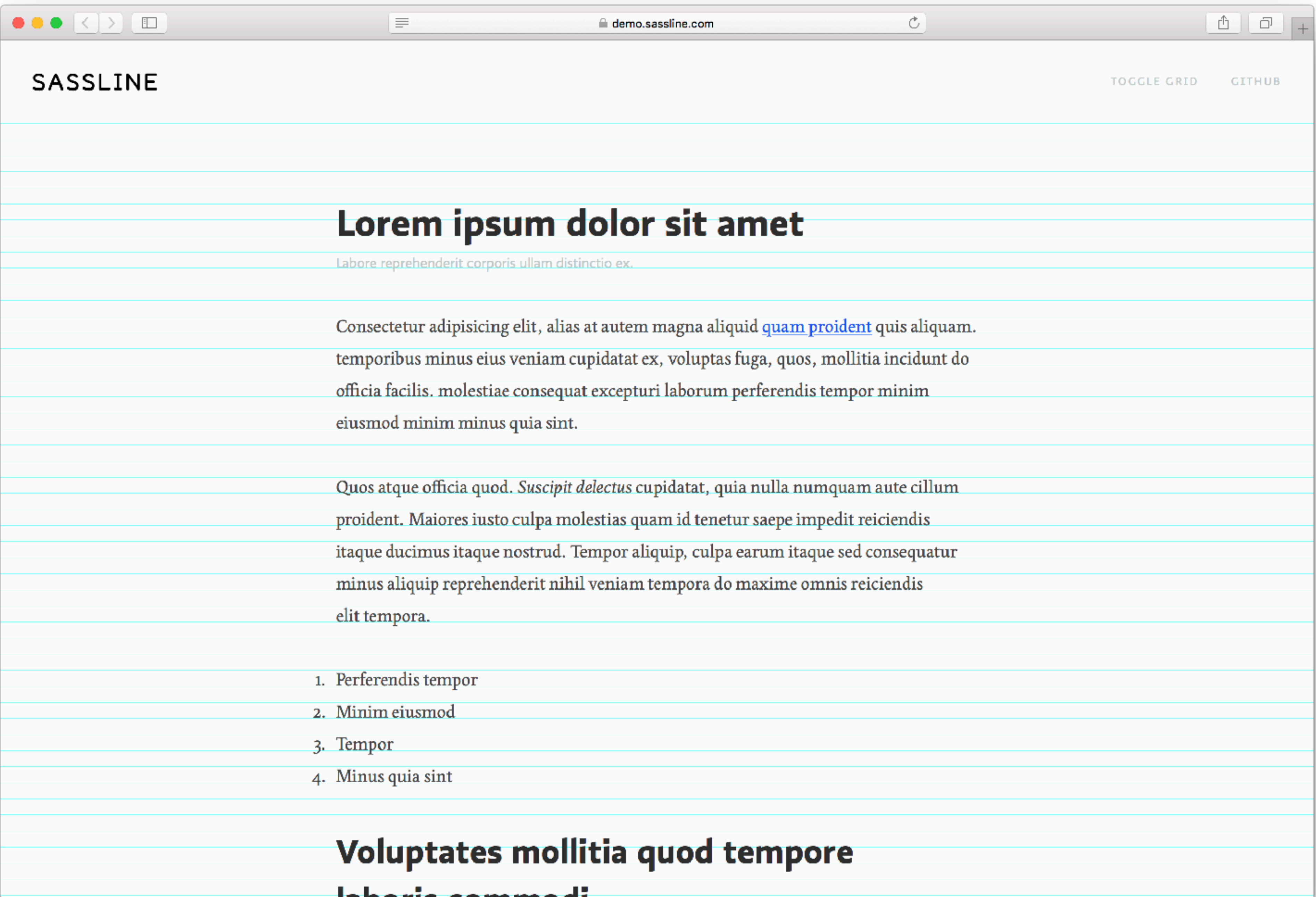
Paper designs

Sort by: Most Popular ▾

☐ Free ☐ Photo card [More Filters ▾](#)

4276 results





Actual solution: tiled image



STYLEGUIDE

Overview

Typography

Colors

Grid

UI ELEMENTS

CSS MIXINS

Typography

Paperless Post uses 2 font families in 5 weights and a scale of standard typographic sizes to achieve visual clarity and harmony across our applications. Our font sizes here are specified in pixels, but the css uses the more accessible and flexible `em`.

IN THIS SECTION

- > [Typographic scale](#)
- > [Font families](#)
- > [Text styles](#)

Typographic scale

Paperless Post's type styles use the fewest number of type styles needed to achieve visual hierarchy. Our available type sizes are based on a custom, interval-based scale. This list represents the full list of available sizes. It doesn't make any claims to which styles, weights and sizes should be used together. For this usage, refer to our UI elements.

**Problem #1: The web doesn't
understand baselines.**

The web centers text in its line-height

Mauris ac neque metus. Phasellus pretium quis ligula vel varius. Donec ut
tristique libero. Cras erat tortor, malesuada id dapibus euismod, dapibus
vitae dolor. Integer nec massa bibendum, posuere erat ac, pretium nisl.
Integer mauris tortor, bibendum sed elit id, mollis luctus elit.

Basehold.it

A QUICK, PAINLESS, JAVASCRIPT-FREE BASELINE OVERLAY.

Baseline grids are super helpful for web designers, allowing us to maintain vertical rhythm on our designs. But short of a few Javascript bookmarklets, there isn't really a simple way to get a baseline grid overlay on our designs.

Enter **Basehold.it**

Using Basehold.it couldn't be easier. Simply add it like any other stylesheet:

```
<link rel="stylesheet" href="//basehold.it/24">
```

Just like that. Easy, right? That'll get you a 24px baseline grid overlay, just like the one you can see on this page. For more information, or to look at the code (and maybe even contribute some code yourself), check out [the repo on Github](#).

ADDITIONAL OPTIONS

You can also specify a custom colour using Hex values, RGB values, or RGBA values, like so:

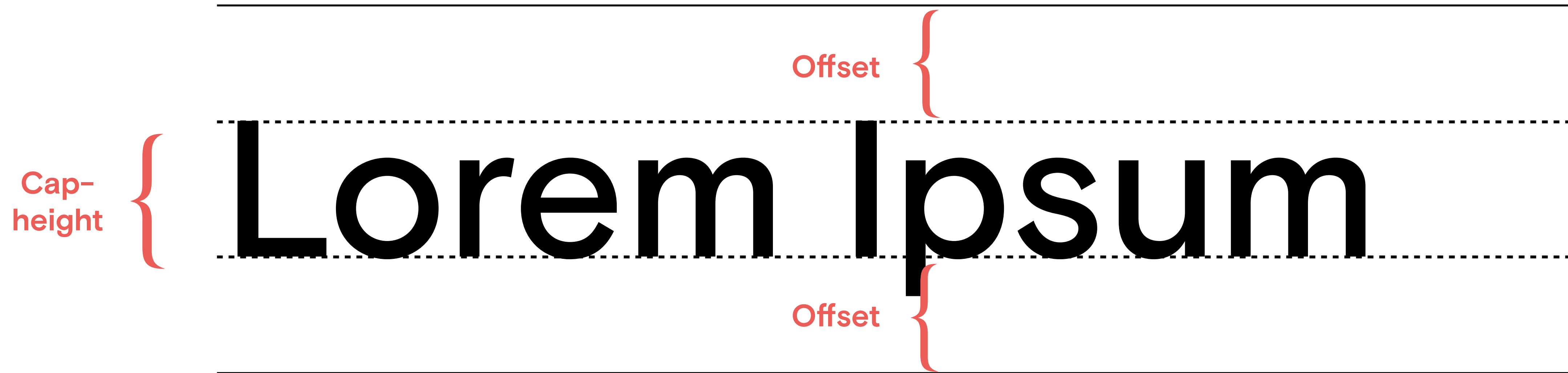
**Lorem ipsum I can't effing
work like this it's so
unsatisfying.**

**Lorem ipsum
dolor sit amet**

I almost gave up.

Cap-height { Lorem Ipsum

Basis has a cap-height ratio of **.75**



$$\text{Offset} = \frac{\text{line-height} - \text{cap-height}}{2}$$

Offset



Lorem Ipsum

Offset = 6px

\$cap-height-ratio: .75;

@mixin baseline(\$font-size, \$line-height)

\$cap-height: \$font-size * \$cap-height-ratio;

\$top-offset: (\$line-height - (\$cap-height)) / 2;

font-size: \$font-size + px;

line-height: \$line-height + px;

padding-top: \$top-offset + px;

}

\$cap-height-ratio: **.75**

@mixin baseline(**16**, **24**)

\$cap-height: **16** * **.75** ; **= 12**

\$top-offset: (**24** - (**6**)) / 2; **= 6**

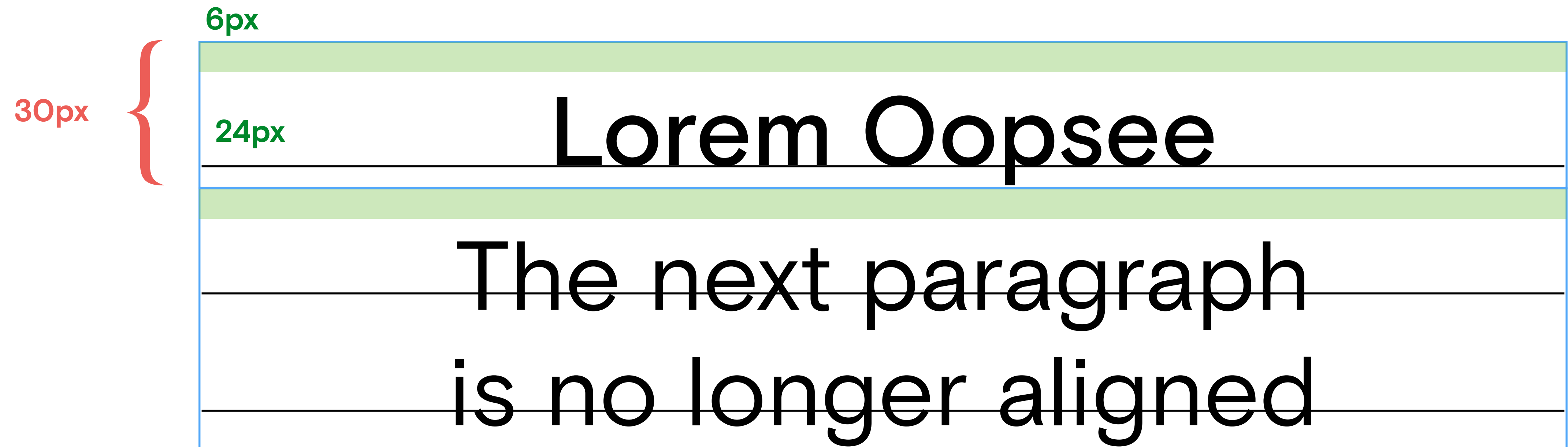
font-size: **16** px;

line-height: **24** + px;

padding-top: **6** px;

}

Lorem Ipsum



\$cap-height-ratio: .72;

\$rhythm-unit: 24;

@mixin baseline(\$font-size, \$line-height)

 \$cap-height: \$font-size * \$cap-height-ratio;

 \$top-offset: (\$line-height - (\$cap-height)) / 2;

 font-size: \$font-size + px;

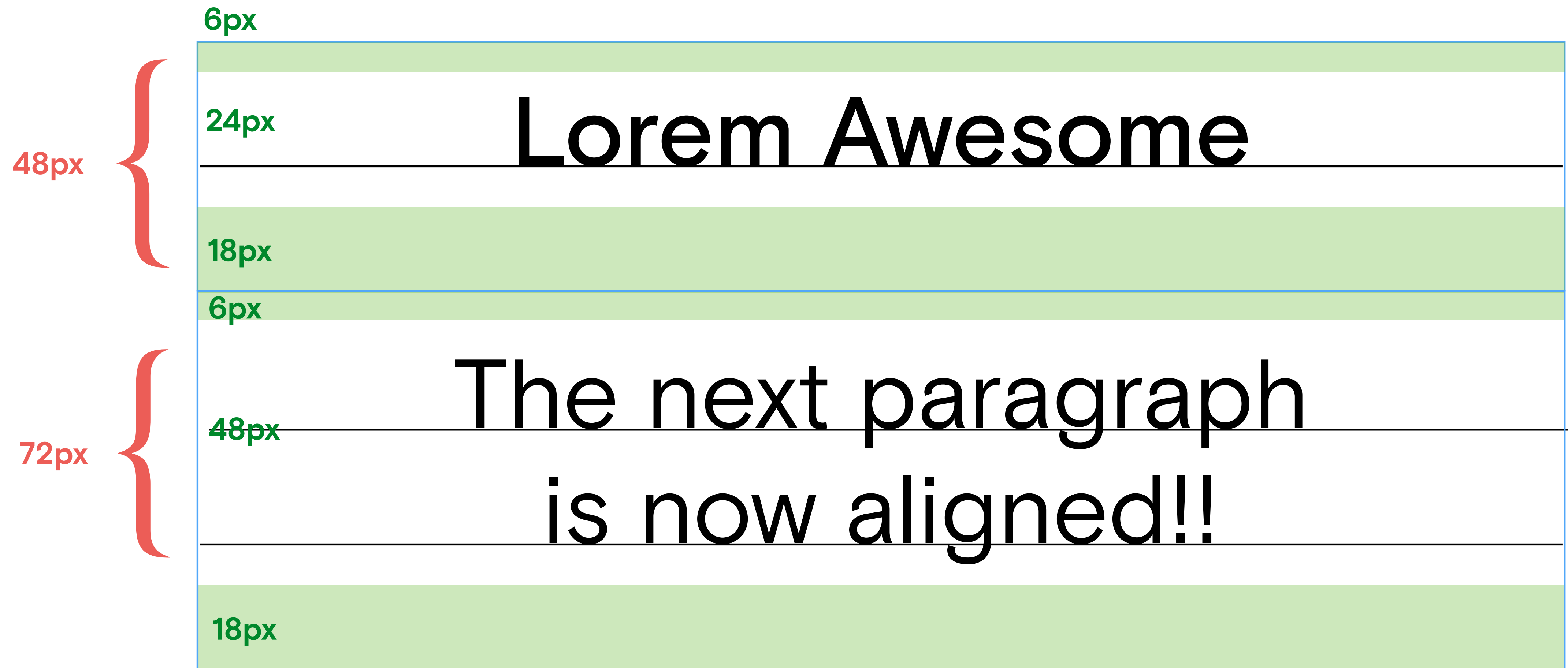
 line-height: \$line-height + px;

 padding-top: \$top-offset + px;

padding-bottom: \$rhythm-unit - \$top-offset + px;

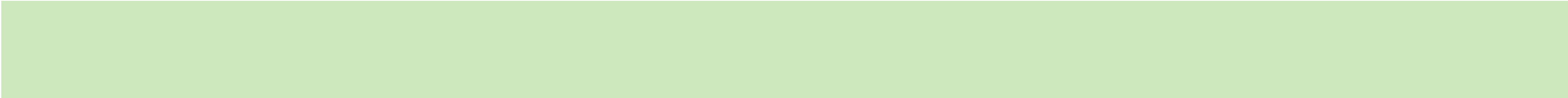
}

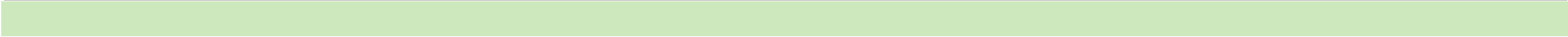
$$24 - 6 = 18\text{px}$$





Lorem Awesome





We can even add margins &
buttons and they will align!



BUTTON

```
@mixin baseline($font-size, $line-height) {  
  
    font-size: $font-size + px;  
    line-height: $line-height + px;  
  
    $top-offset: ($line-height - ($cap-height-ratio * $font-size)) / 2;  
  
    //top-offset values between 8 and 16  
    @if ($top-offset > $rhythm-unit and $top-offset < ($rhythm-unit*2)) {  
        $padding-top: $top-offset - $rhythm-unit;  
        padding-top: $padding-top + px;  
        padding-bottom: ($rhythm-unit - $padding-top) + px;  
    }  
    //top-offset values greater than 16  
    @else if $top-offset > (2 * $rhythm-unit) {  
        $padding-top: ($top-offset - ($rhythm-unit*2));  
        padding-top: $padding-top + px;  
        padding-bottom: ($rhythm-unit - $padding-top) + px;  
    }  
    //top-offset values less than 8  
    @else {  
        padding-top: $top-offset + px;  
        padding-bottom: ($rhythm-unit - $top-offset) + px;  
    }  
}
```

Heading text

This is a longer paragraph with wrapping text that is offering explanation of a feature.



Allow access to camera.

CANCEL

SAVE

Sketch works like the web

**Lorem ipsum dolor sit amet, consectetur
adipiscing elit. Sed vitae nisl eleifend, tincidunt
dolor at, sagittis mi. Vivamus quis erat purus.
Fusce volutpat scelerisque tellus.**

Lorem ipsum

dolor sit

ahhhhhhhhhhhhh

Lorem ipsum

dolor sit

ahhhhhhhhhhhhh

Lorem ipsum

dolor sit

ahhhhhhhhhhhhhhh

18px

BUTTON

Lorem ipsum

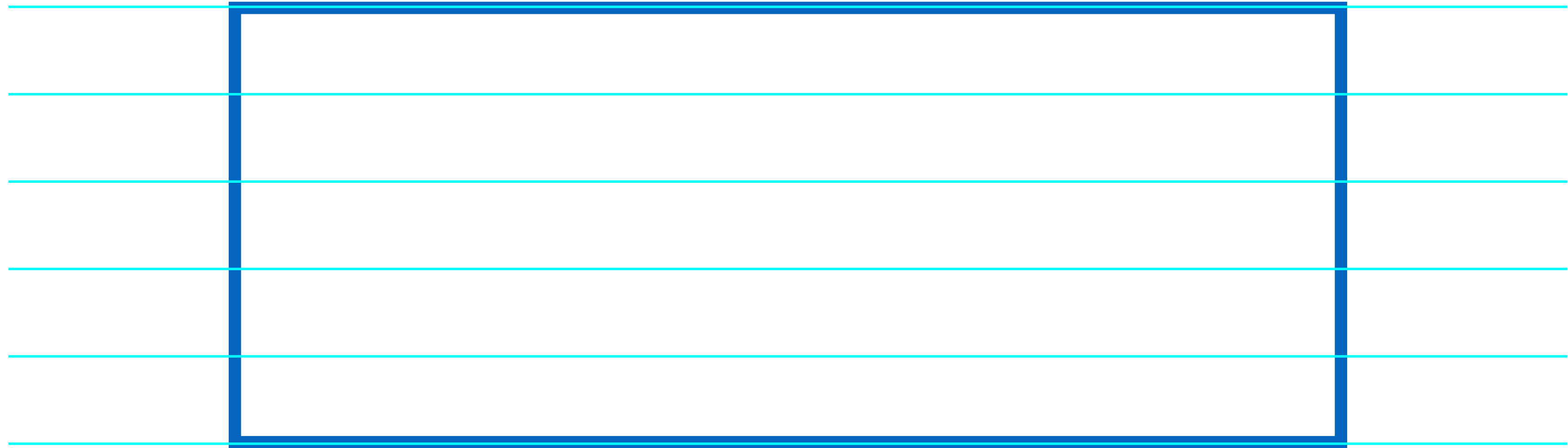
dolor sit

ahhhhhhhhhhhhh

Also just look at this

```
.dek {  
  font-size: 14px;  
  line-height: 24px;  
  padding-top: 6.75px;  
  padding-bottom: 17.25px;  
  margin-bottom: 24px;  
}
```

Let's say we're building a button



And it has text



BUTTON

CENTERED


BASELINED

Rules

- OBJECTS not TEXT align to the baseline grid.
- A TEXT also aligns, but it's container aligns, not it's baseline.

Remember this?

Lorem ipsum I can't effing
work like this it's so
unsatisfying.

Oh well, turn off the grid.

Hover to make sure all is well

FORM LABEL

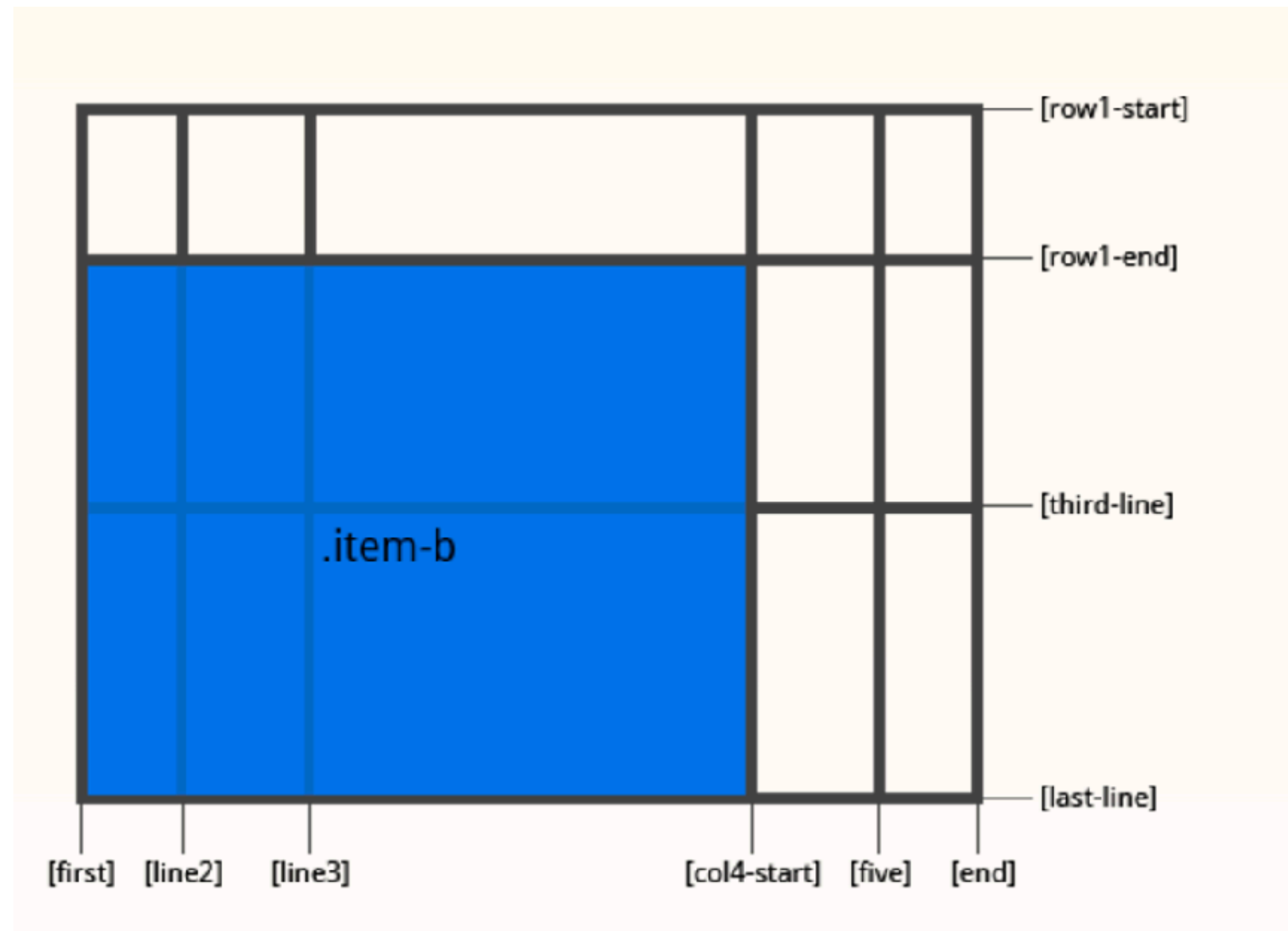
Form text

24px

FORM LABEL

Form text

The future?



<https://css-tricks.com/snippets/css/complete-guide-grid/>

Thanks