PART 3

Intro + Bonus

CSS Typography

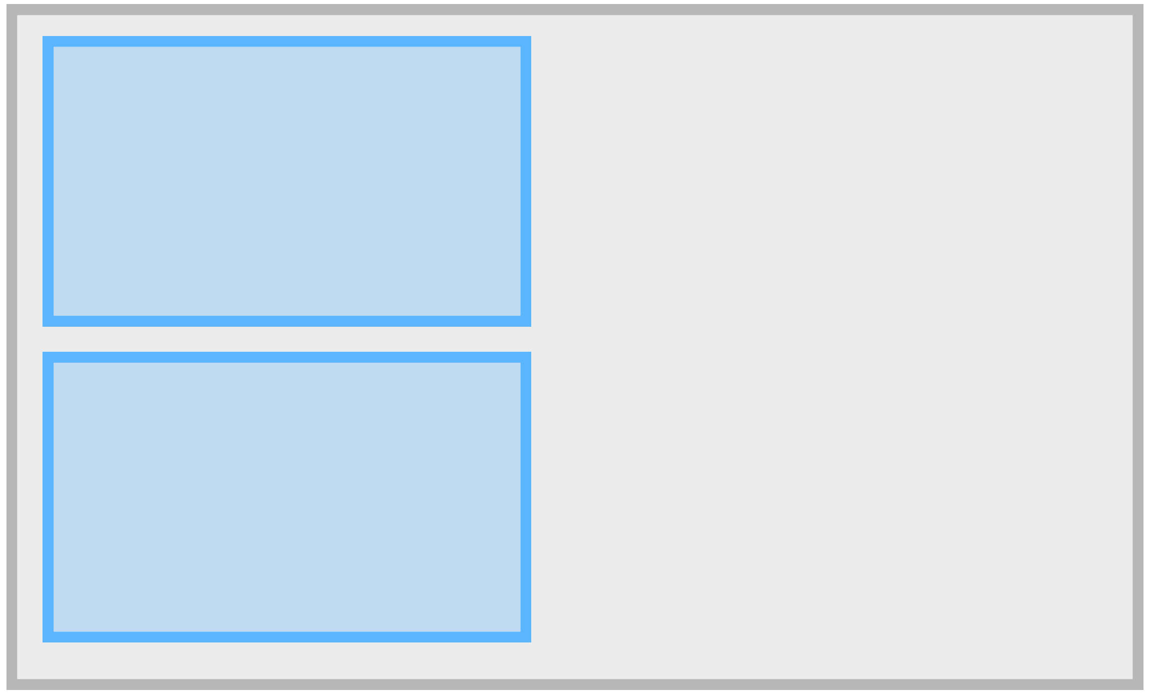
UI Elements

Quick revision + Bonus

Quick revision + Bonus

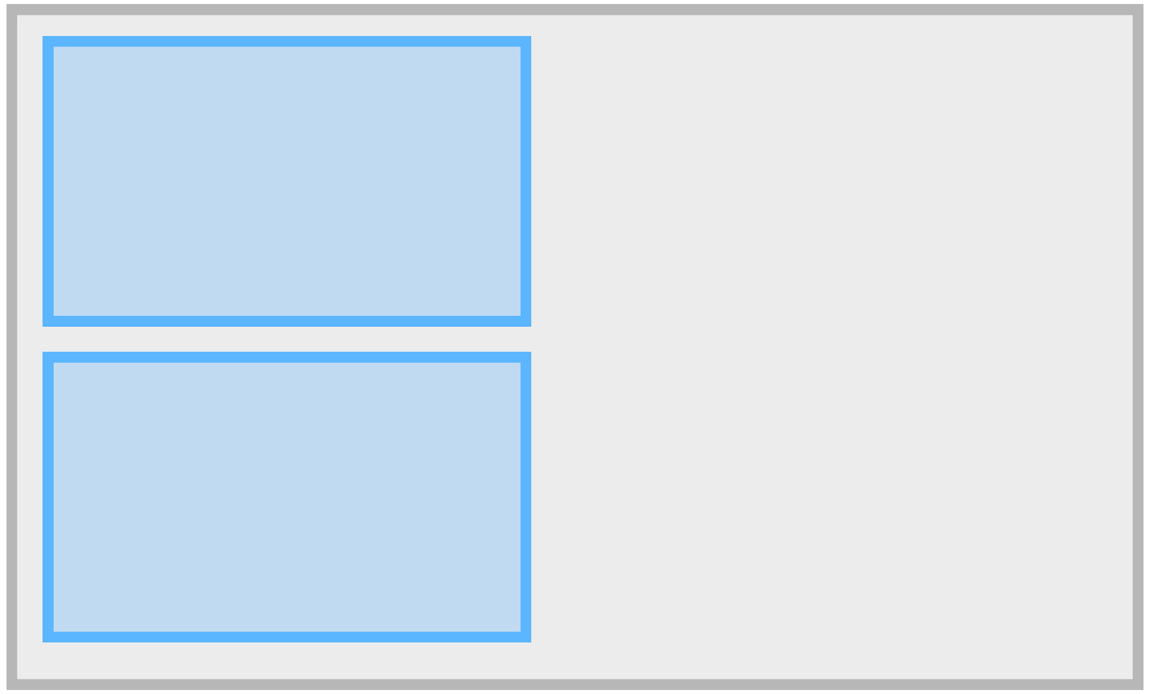
- CSS layouts using floats

CSS Floats

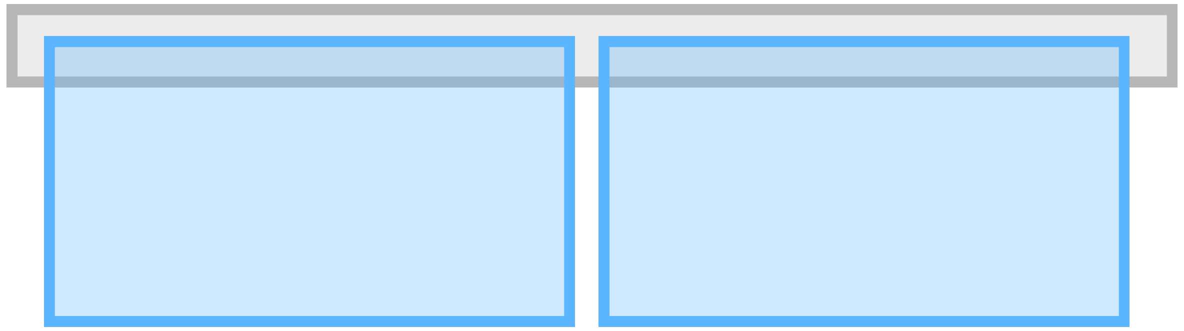


```
.my-div {
  width: 50%;
}
```

CSS Floats

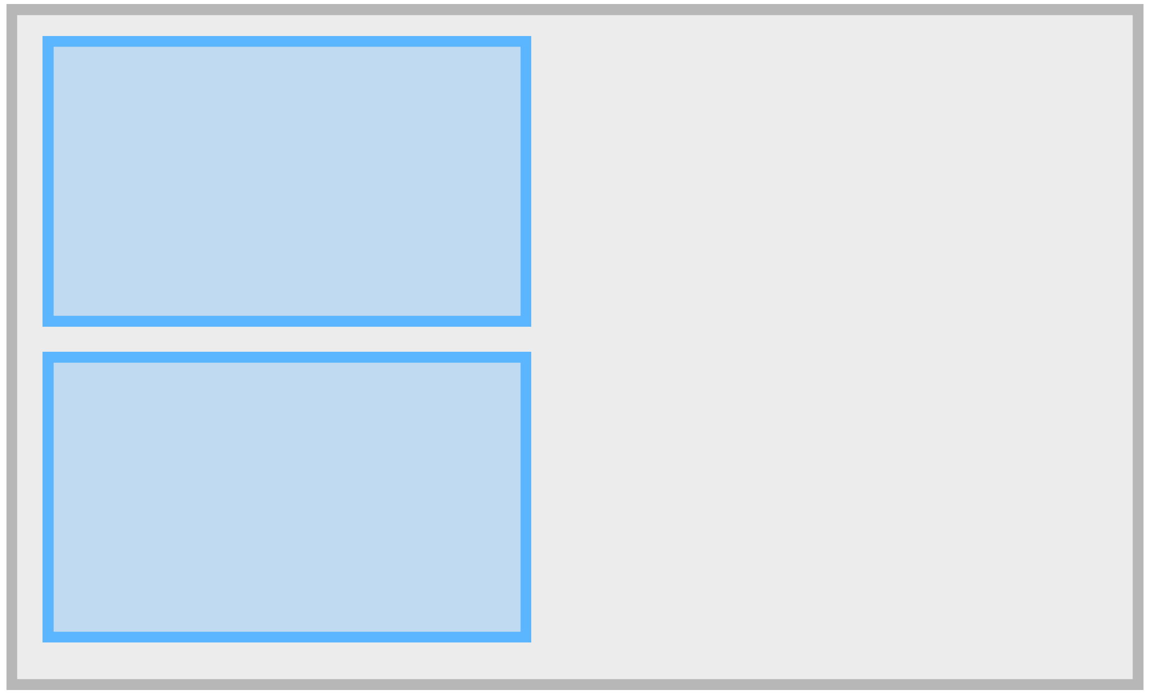


```
.my-div {
  width: 50%;
}
```

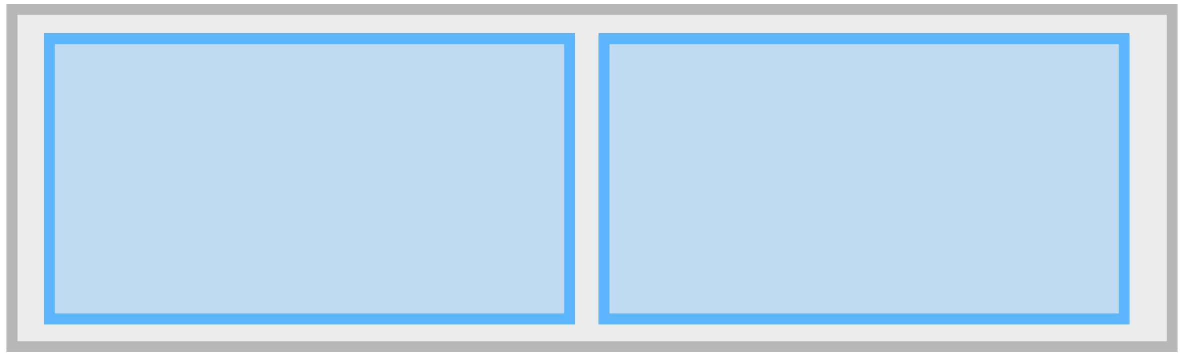


```
.my-div {
  width: 50%;
  float: left;
}
```

CSS Floats



```
.my-div {
  width: 50%;
}
```



```
.my-div {
  width: 50%;
  float: left;
}

.my-parent-div { (clear fixed) }
```

Bonus! CSS Positioning

- Elements may be positioned in CSS using the position property (I know, weird eh)

Bonus! CSS Positioning

- Elements may be positioned in CSS using the position property (I know, weird eh)
- Valid values are `static` (default), `relative`, `absolute` and `fixed`.

Bonus! CSS Positioning

- Elements may be positioned in CSS using the position property (I know, weird eh)
- Valid values are `static` (default), `relative`, `absolute` and `fixed`.

```
.my-div {
  position: relative;
}
```

Static positioning

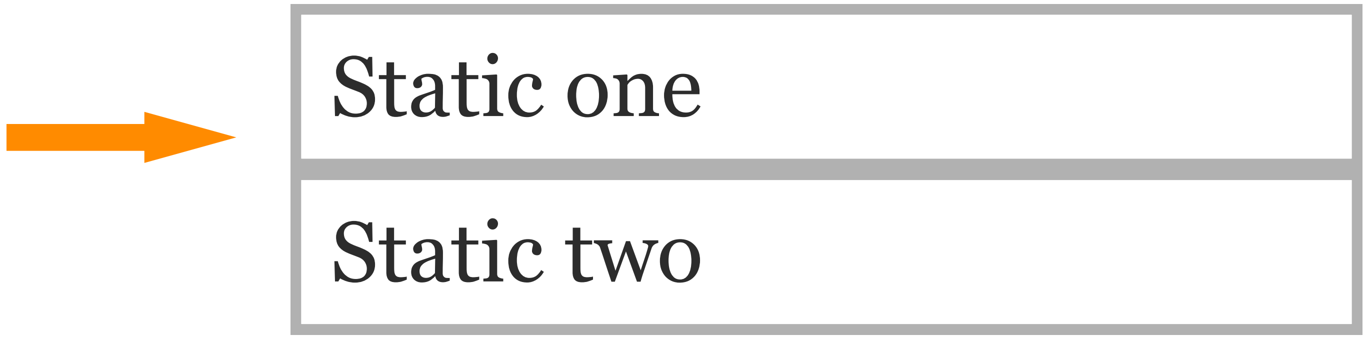
Static positioning

- Statically positioned elements are positioned according to their document order

Static positioning

- Statically positioned elements are positioned according to their document order

```
<div>Static one</div>
<div>Static two</div>
```



Relative positioning

Relative positioning

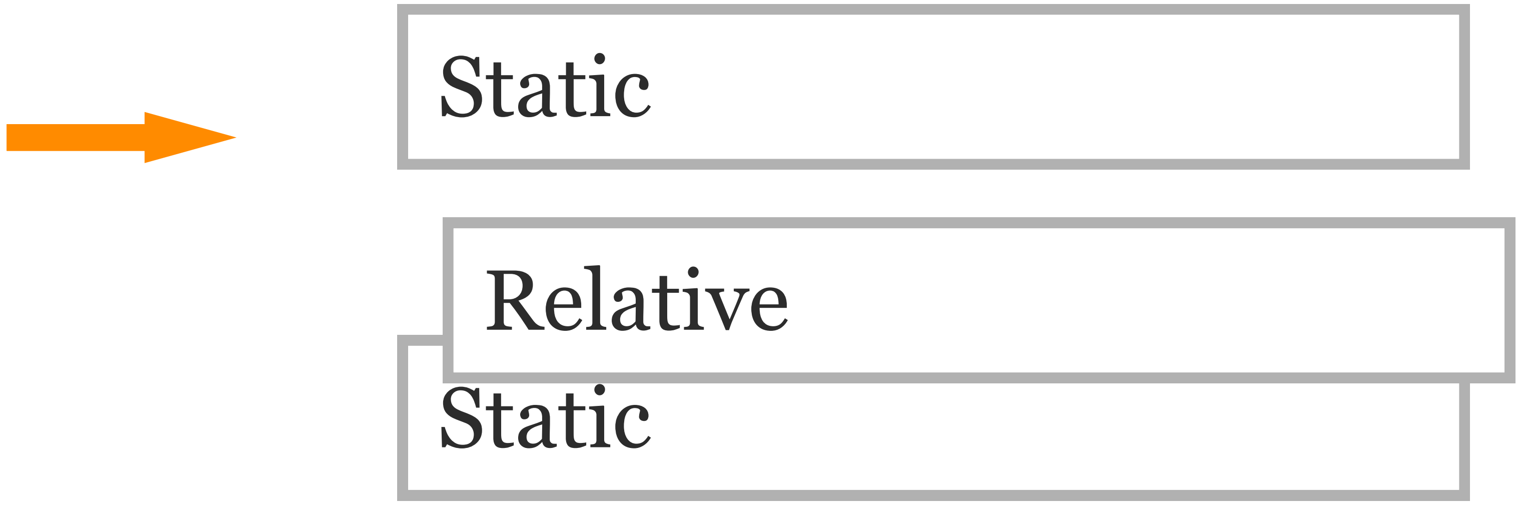
- Relatively positioned elements leave a footprint however can be shifted according to positioning

Relative positioning

- Relatively positioned elements leave a footprint however can be shifted according to positioning

```
<div>Static</div>
<div class=“offset”>
  Relative</div>
<div>Static</div>

   .offset {
     position: relative;
     top: 10px;
     left: 10px;
   }
```



Absolute positioning

Absolute positioning

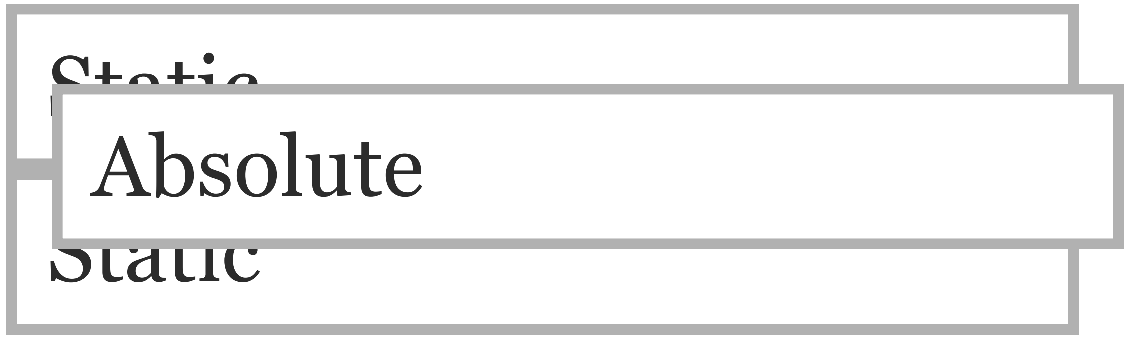
- Absolutely positioned elements are taken out of flow (no footprint) and are positioned accordingly

Absolute positioning

- Absolutely positioned elements are taken out of flow (no footprint) and are positioned accordingly

```
<div>Static</div>
<div class=“absolute-div”>
  Absolute</div>
<div>Static</div>
```

```
   .absolute-div {
     position: absolute;
     top: 10px;
     left: 10px;
   }
```



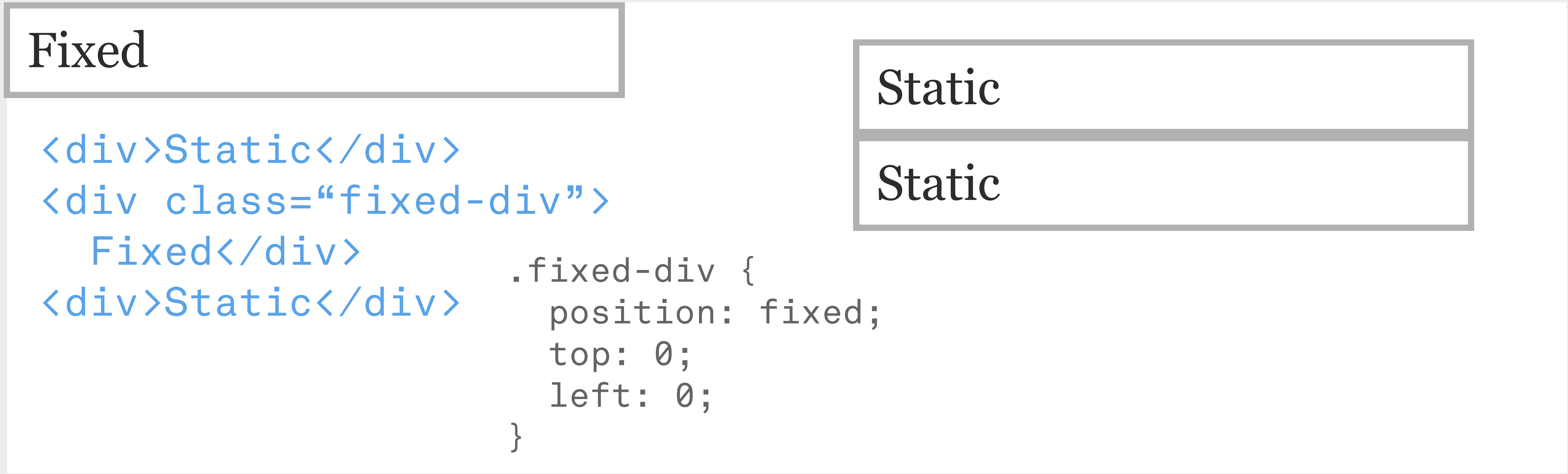
Fixed positioning

Fixed positioning

- Fixed positioned elements are taken out of flow and are positioned according to the viewport

Fixed positioning

- Fixed positioned elements are taken out of flow and are positioned according to the viewport



```
<div>Static</div>
<div class="fixed-div">
  Fixed</div>
<div>Static</div>
```

```
.fixed-div {
  position: fixed;
  top: 0;
  left: 0;
}
```

Positioning

Demo time

PART 3

Intro + Bonus

CSS Typography

UI Elements

A brief history of type on the web

A brief history of type on the web

- Originally all text was limited to pre-installed system fonts (i.e. Arial, Times New Roman)

A brief history of type on the web

- Originally all text was limited to pre-installed system fonts (i.e. Arial, Times New Roman)
- Which isn’t such a bad thing for usability but very limiting for creating branded experiences

A brief history of type on the web

- Originally all text was limited to pre-installed system fonts (i.e. Arial, Times New Roman)
- Which isn’t such a bad thing for usability but very limiting for creating branded experiences
- Workarounds including images-as-text and Flash as a replacement for headlines

A brief history of type on the web

Using images / Flash as text is bad because:

A brief history of type on the web

Using images / Flash as text is bad because:

- Search engines can’t read content

A brief history of type on the web

Using images / Flash as text is bad because:

- Search engines can’t read content
- Screen readers (for vision impaired) also can’t

A brief history of type on the web

Using images / Flash as text is bad because:

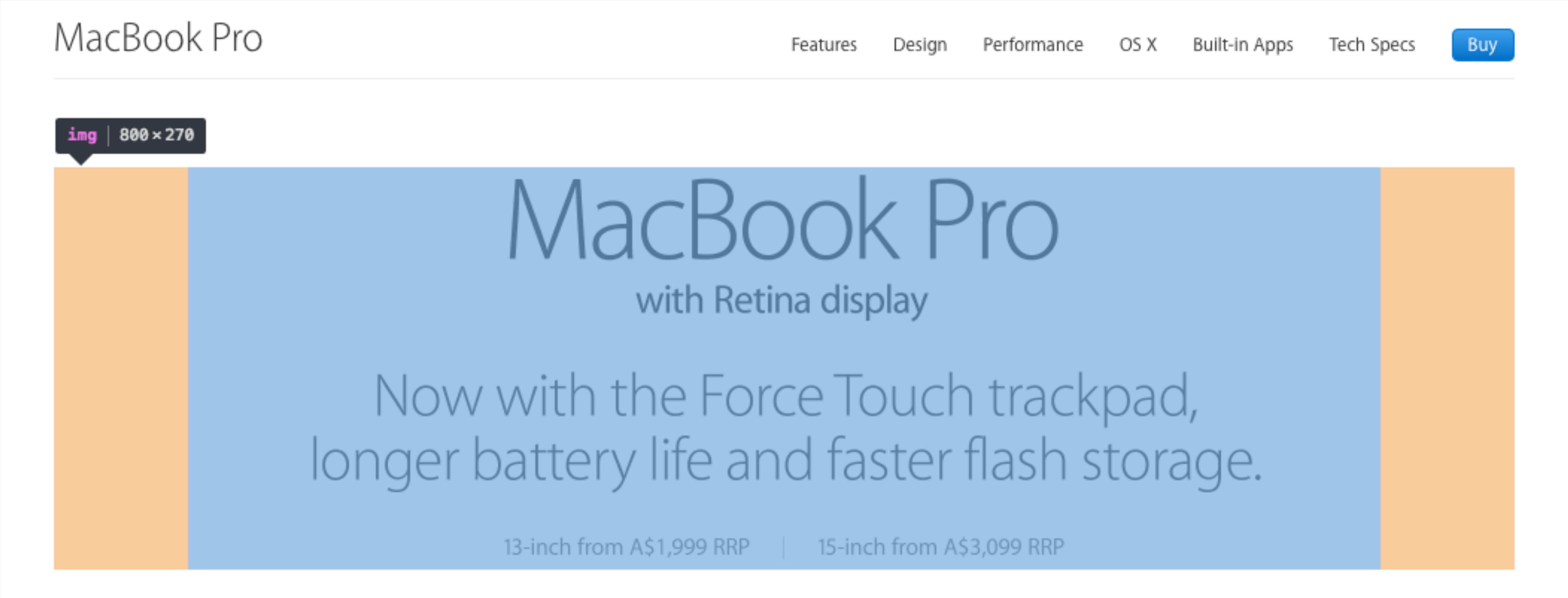
- Search engines can’t read content
- Screen readers (for vision impaired) also can’t
- Won’t scale to display

A brief history of type on the web

Using images / Flash as text is bad because:

- Search engines can’t read content
- Screen readers (for vision impaired) also can’t
- Won’t scale to display
- Can’t copy / paste

Images-as-text



MacBook Pro

MacBook Pro

with Retina display

Now with the Force Touch trackpad, longer battery life and faster flash storage.

13-inch from A$1,999 RRP | 15-inch from A$3,099 RRP

The present: Webfonts

The best practise today is using webfonts - font files that can be loaded on the browser allowing any fonts to be embedded in a webpage.

The present: Webfonts

The best current practise is using webfonts.

The present: Webfonts

The best current practise is using webfonts.

- Webfonts refer to embeddable font files

The present: Webfonts

The best current practise is using webfonts.

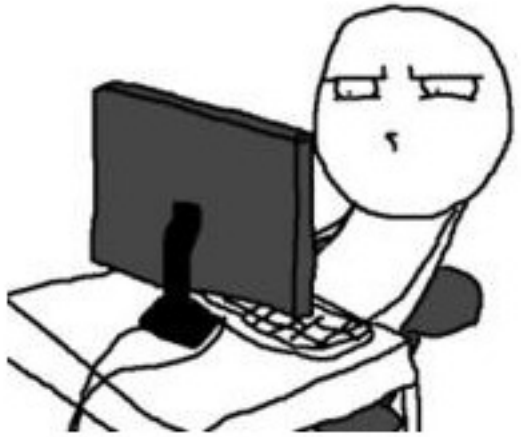
- Webfonts refer to embeddable font files
- Multiple formats for multiple vendors

The present: Webfonts

The best current practise is using webfonts.

- Webfonts refer to embeddable font files
- Multiple formats for multiple vendors
- Can be loaded directly though some licenses require fonts to be hosted remotely

Webfonts syntax



```
@font-face {
  font-family: 'Journal';
    src: url('http://your-own.site/fonts/journal.woff') format('woff'),
    url('http://your-own.site/fonts/journal.svg#Journal') format('svg'),
    url('http://your-own.site/fonts/journal.ttf') format('truetype'),
    url('http://your-own.site/fonts/journal.eot'),
    url('http://your-own.site/fonts/journal.eot?#iefix') format('embedded-opentype');
  font-weight: normal;
  font-style: normal;
}
```

Webfonts

Generators such as `fontsquirrel.com` can create all the required webfont files.

Demo time

Limitations

- Reduced selection of fonts

Limitations

- Reduced selection of fonts
- No hyphenation for justfified text

Limitations

- Reduced selection of fonts
- No hyphenation for justfified text
- Lack of kerning support

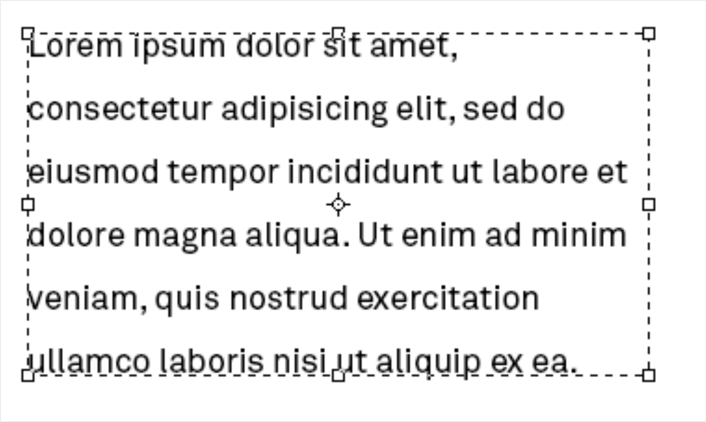
Limitations

- Reduced selection of fonts
- No hyphenation for justfified text
- Lack of kerning support
- No control over size of text elements (orphans)

Limitations

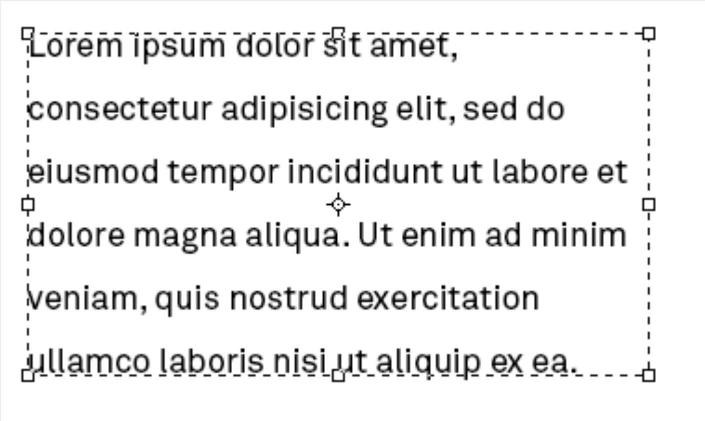
- Reduced selection of fonts
- No hyphenation for justfified text
- Lack of kerning support
- No control over size of text elements (orphans)
- Browser text vertical alignment

Vertical alignment


Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea.

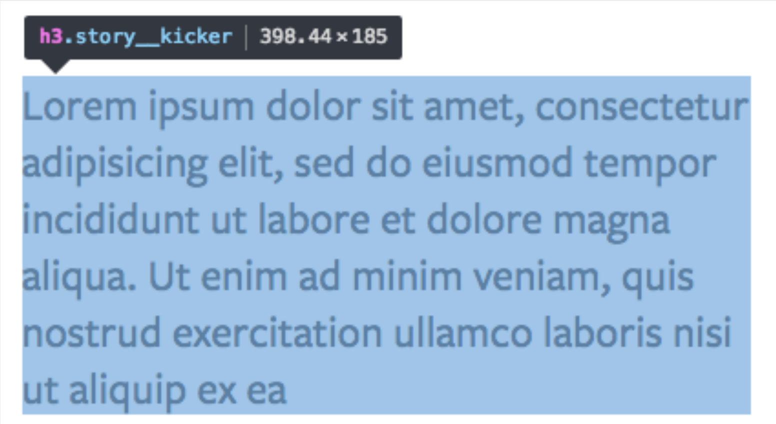
Photoshop

Vertical alignment



Lorem ipsum dolor sit amet,
consectetur adipisicing elit, sed do
eiusmod tempor incididunt ut labore et
dolore magna aliqua. Ut enim ad minim
veniam, quis nostrud exercitation
ullamco laboris nisi ut aliquip ex ea.

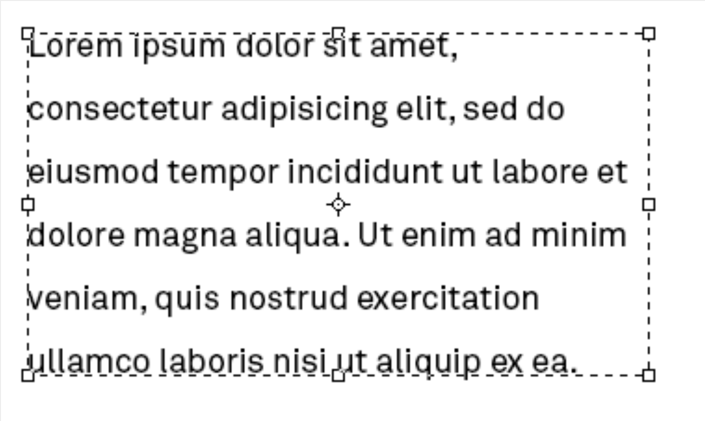
Photoshop



Lorem ipsum dolor sit amet, consectetur
adipisicing elit, sed do eiusmod tempor
incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis
nostrud exercitation ullamco laboris nisi
ut aliquip ex ea

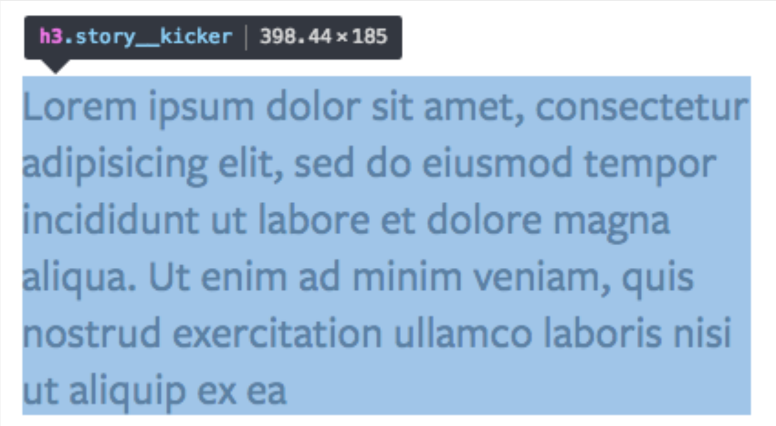
Browser

Vertical alignment



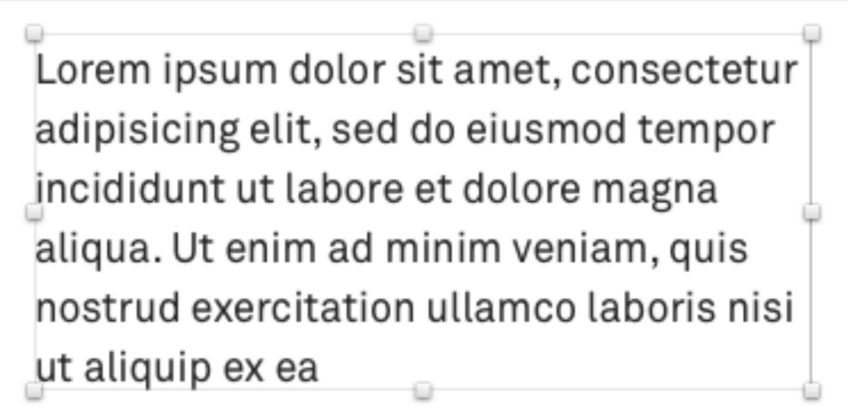
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea.

Photoshop



Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea

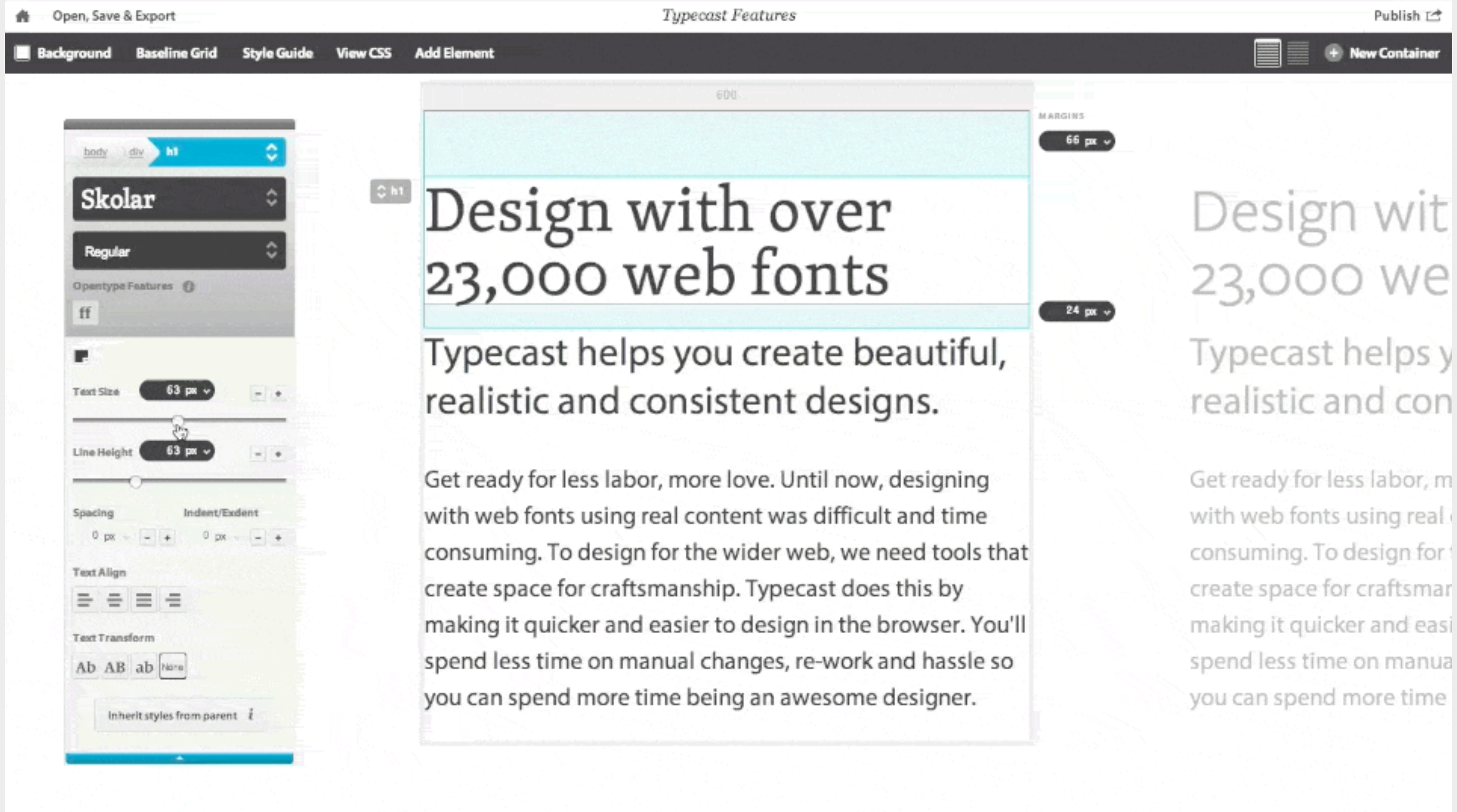
Browser



Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea

Sketch

Typecast for prototyping



Skolar

Regular

Text Size 63 px

Line Height 63 px

Spacing 0 px

Indent/Exdent 0 px

66 px

24 px

Design with over 23,000 web fonts

Typecast helps you create beautiful, realistic and consistent designs.

Get ready for less labor, more love. Until now, designing with web fonts using real content was difficult and time consuming. To design for the wider web, we need tools that create space for craftsmanship. Typecast does this by making it quicker and easier to design in the browser. You'll spend less time on manual changes, re-work and hassle so you can spend more time being an awesome designer.

Design wit
23,000 we

Typecast helps y
realistic and con

Get ready for less labor, m
with web fonts using real
consuming. To design for
create space for craftsmar
making it quicker and easi
spend less time on manua
you can spend more time

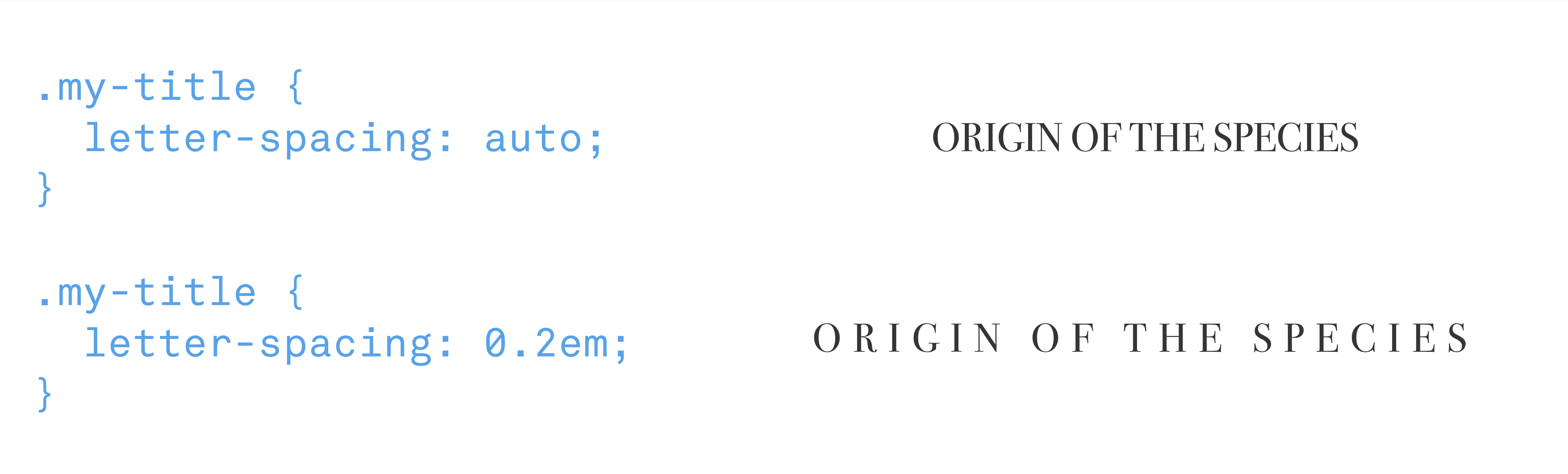
Kerning

Kerning

Kerning type is generally restrictive however we can specify `letter-spacing` for whole titles.

Kerning

Kerning type is generally restrictive however we can specify `letter-spacing` for whole titles.



```
.my-title {
  letter-spacing: auto;
}
```

ORIGIN OF THE SPECIES

```
.my-title {
  letter-spacing: 0.2em;
}
```

ORIGIN OF THE SPECIES

Units

Units

- Many possibilities: `px`, `em`, `rem`, `%`, `pt`

Units

- Many possibilities: `px`, `em`, `rem`, `%`, `pt`
- For `font-size`, `line-height`, `letter-spacing`

Units

- Many possibilities: `px`, `em`, `rem`, `%`, `pt`
- For `font-size`, `line-height`, `letter-spacing`
- Pixels (`px`) are simplest

Units

- Many possibilities: `px`, `em`, `rem`, `%`, `pt`
- For `font-size`, `line-height`, `letter-spacing`
- Pixels (px) are simplest
- Ems (em) are most difficult yet powerful for responsive design

Demo time

PART 3

Intro + Bonus

CSS Typography

UI Elements

HTML / CSS user interface

HTML / CSS user interface

- For dynamic websites we often need to accept user input for actions or data

HTML / CSS user interface

- For dynamic websites we often need to accept user input for actions or data
- Form elements are HTML elements that are usually used to accomplish this

HTML / CSS user interface

- For dynamic websites we often need to accept user input for actions or data
- Form elements are HTML elements that are usually used to accomplish this
- Examples include text-fields, dropdown menus and text areas

Demo time

Beyond forms

Beyond forms

- For interactive experiences we often need to apply CSS beyond the static approach

Beyond forms

- For interactive experiences we often need to apply CSS beyond the static approach
- For many components absolute and fixed positioning is used

Beyond forms

- For interactive experiences we often need to apply CSS beyond the static approach
- For many components absolute and fixed positioning is used
- Let’s see how CSS is used for a common UI approach; an image carousel

Demo time

Thank you.