

Search elements ...

PINNED ELEMENTS

Container

Section

Block

Heading

Basic Text

LAYOUT

Div

BASIC

Rich Text

Text link

Button

Icon

Image

Video

GENERAL

Nav (Nestable)

Dropdown

Offcanvas

Toggle

Divider

Icon Box

::before and ::after

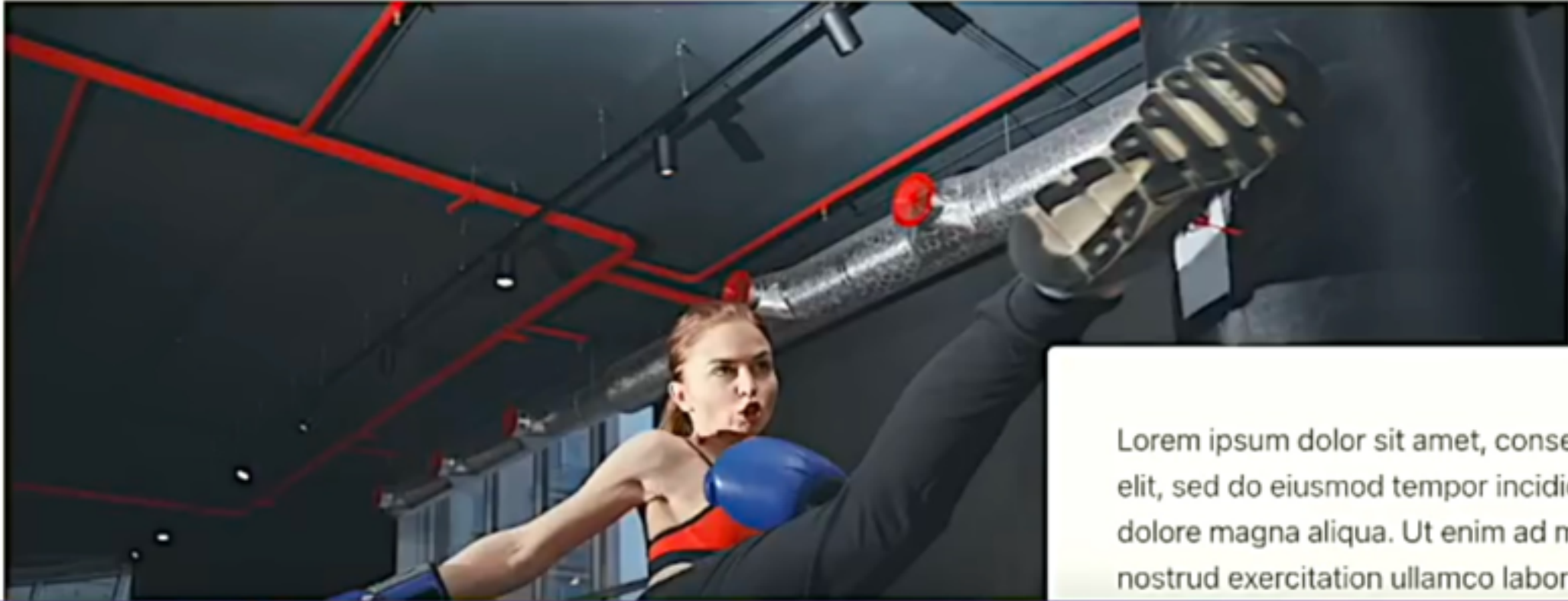
- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

Physical Divider or Pseudo Element?

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



Lorem ipsum dolor sit amet, consectetur adipiscing 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

Structure

Section

Container

Heading

Rich Text

Section

Container

Container

Container

Container

Container

Global

Global

Global

Keep in mind that pseudo elements live inside the box!

03:02

Search elements ...

PINNED ELEMENTS

Container

Section

Block

Heading

Basic Text

LAYOUT

Div

BASIC

Rich Text

Text link

Button

Icon

Image

Video

GENERAL

Nav (Nestable)

Dropdown

Offcanvas

Toggle

Divider

Icon Box

::before and ::after

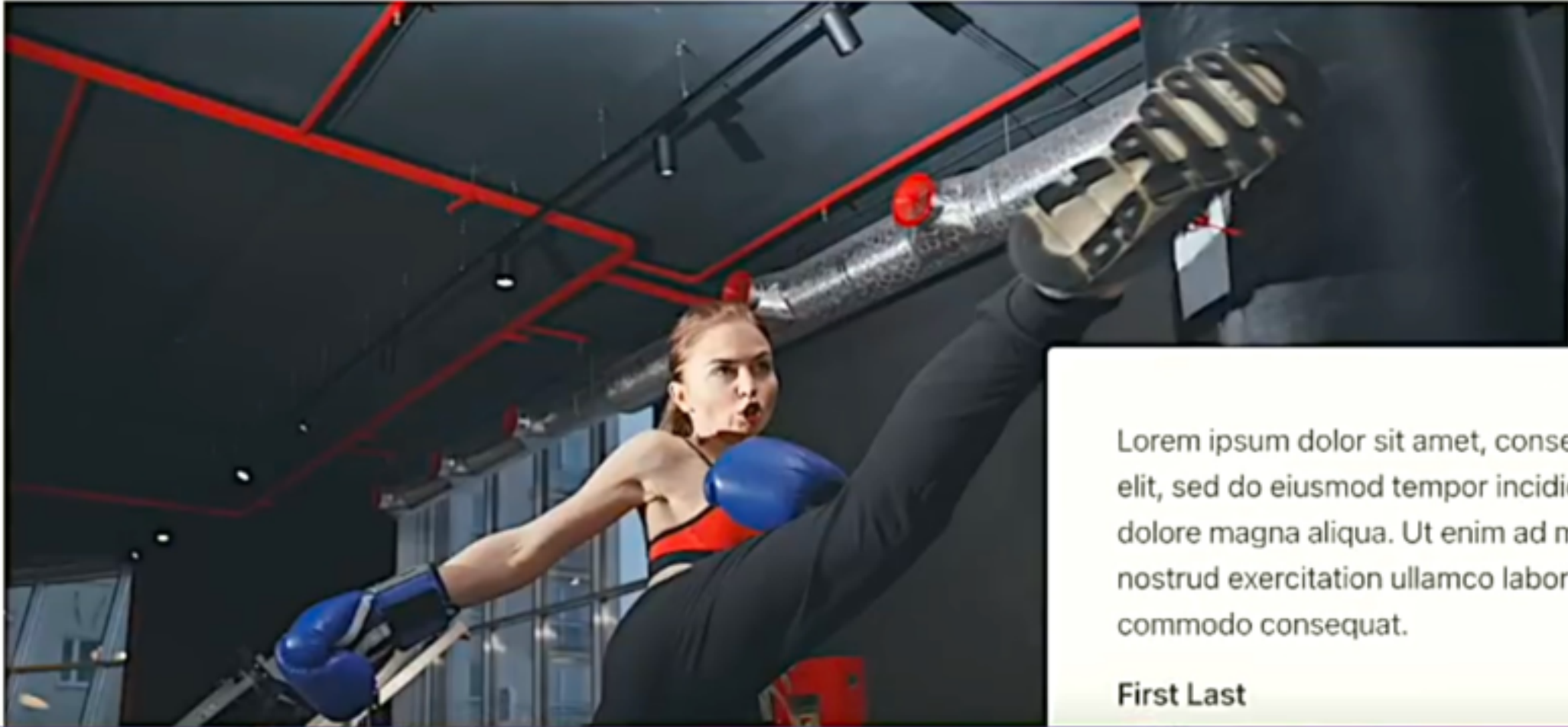
- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

Physical Divider or Pseudo Element?

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



Lorem ipsum dolor sit amet, consectetur adipiscing 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 commodo consequat.

First Last

Structure

Section

Container

Heading

Rich Text

Section

Container

Container

Container

Container

Container

Container

Global

Global

Global

03:16

Search elements ...

PINNED ELEMENTS

Container

Section

Block

Heading

Basic Text

LAYOUT

Div

BASIC

Rich Text

Text link

Button

Icon

Image

Video

GENERAL

Nav (Nestable)

Dropdown

Offcanvas

Toggle

Divider

Icon Box

::before and ::after

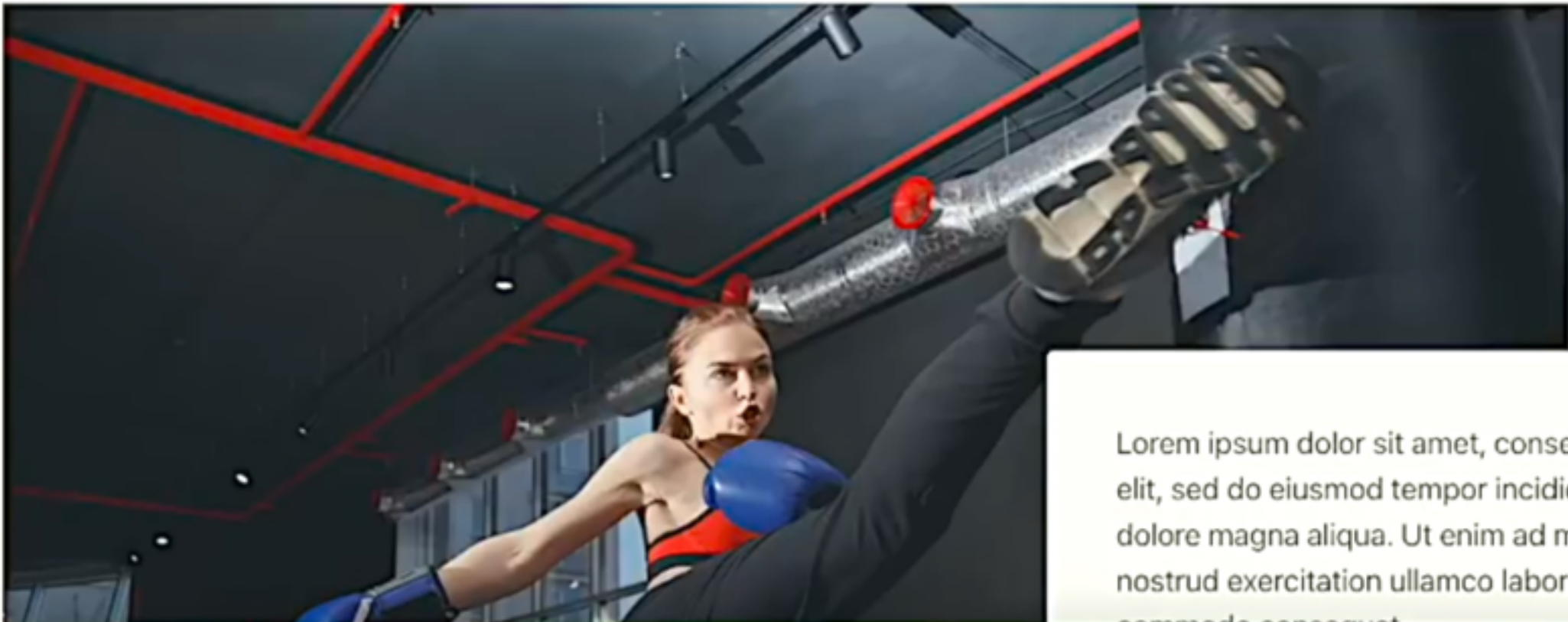
- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

Physical Divider or Pseudo Element?

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



Lorem ipsum dolor sit amet, consectetur adipiscing 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 commodo consequat.

Structure

Section

Container

Heading

Rich Text

Section

Container

Container

Divider


Container

Container

Container

Global

This is the divider element. If you add that across your page you create yourself a maintenance nightmare

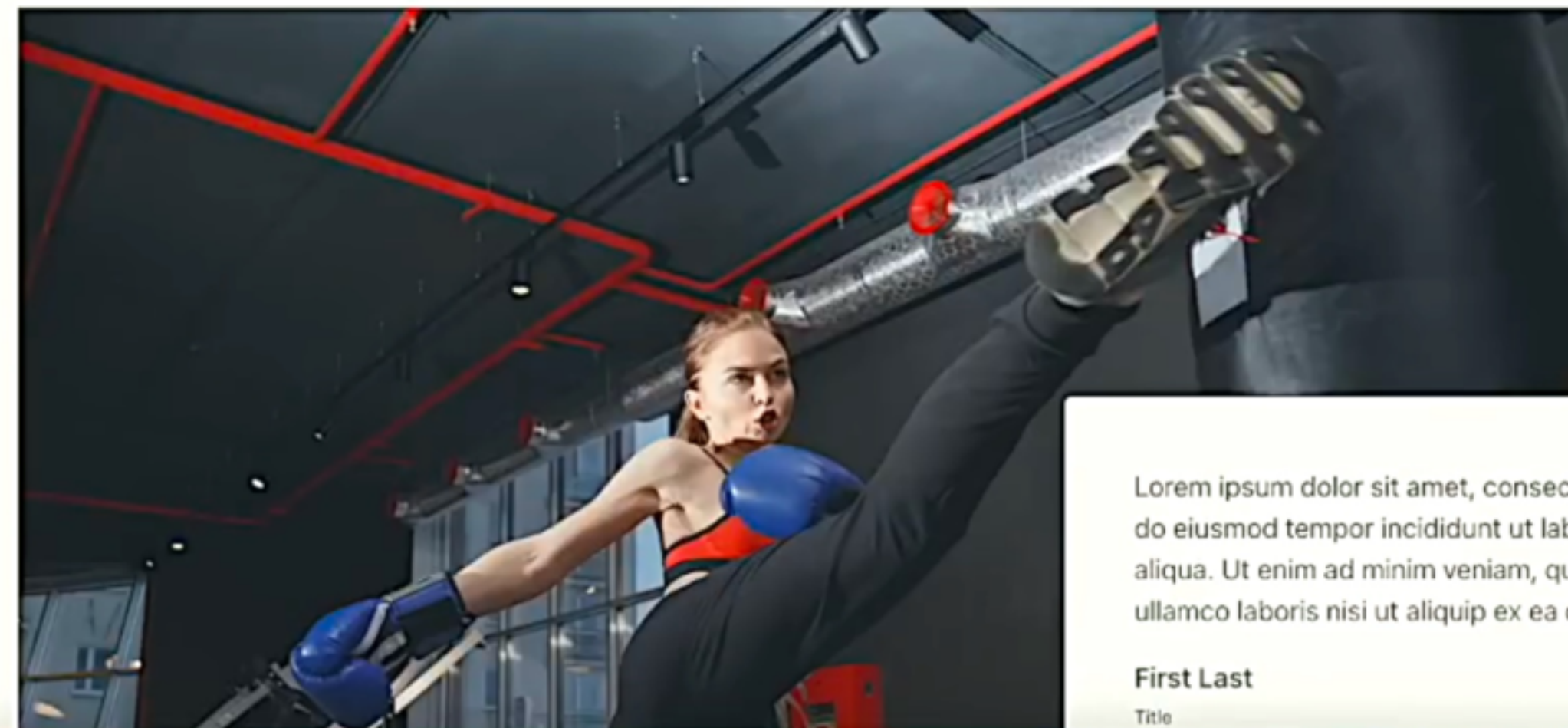


04:01

- Two of the most common and useful psuedo elements are `::before` and `::after`. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" – `::before` and `::after` elements are placed *inside* the box, not outside the box).

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).



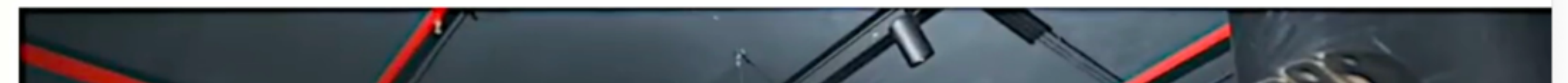
::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).



Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



```
1 root::after {
2   content: "🐱";
3 }
```

More flexible is to write the CSS
but you can do it all inside Bricks

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).



::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



```
1 root::after {
2   content: "";
3   width: 100%;
4   height: 2px;
5 }
```

Now we gave the pseudo element dimension but we still can't see it because it's transparent by default

Use "root" to target the element wrapper: root { background: blue }

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

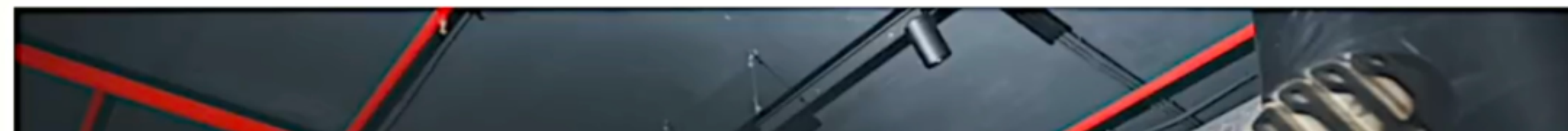


::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



.pseudo-divider

x 1

.pseudo-divider

CONTENT

STYLE

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

width 0s ease 0s

[Learn more about CSS transitions](#)

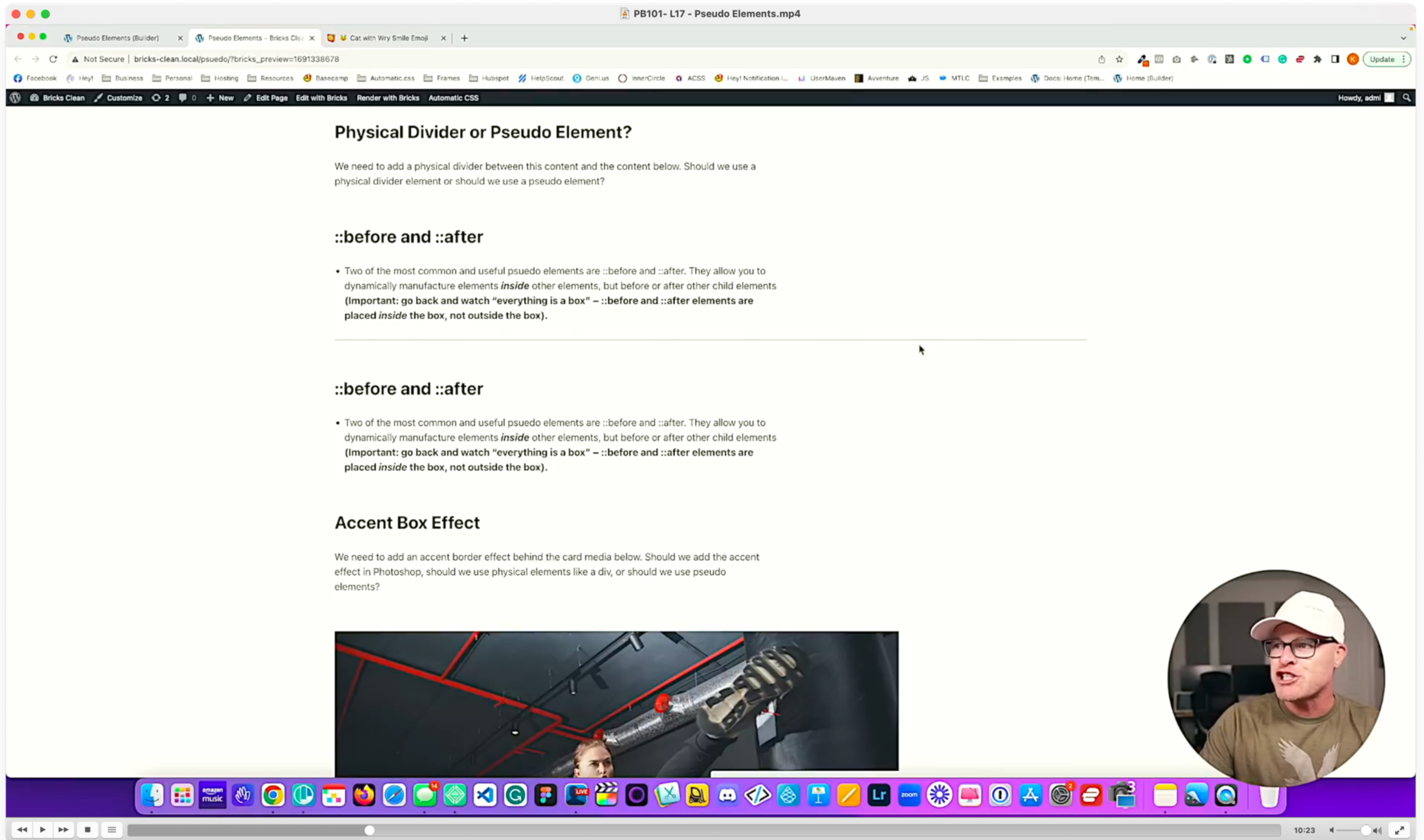
• Custom CSS

```
1 root::after {  
2   content: "";  
3   width: 100%;  
4   height: 2px;  
5   background-color: red;  
6 }
```

Tip: Use a bright, ugly color to begin with because then you can see the element on screen

Use "root" to target the element wrapper: root { background: blue }





PB101- L17 - Pseudo Elements.mp4

Pseudo Elements (Builder) x Pseudo Elements - Bricks Clean x Cat with Wry Smile Emoji x +

Not Secure | bricks-clean.local/psuedo/?bricks=run

Facebook Hey! Business Personal Hosting Resources Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS Hey! Notification UserMaven Avventure JS MTLC Examples Docs: Home (Tem... Home (Builder)

Divider

.pseudo-divder x 1

.pseudo-divder

CONTENT STYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

CSS

CSS Filters

Learn more about CSS filters

Transition

width 0s ease 0s

Learn more about CSS transitions

Custom CSS

```
1 root::after {
2   content: "";
3   width: 100%;
4   height: 2px;
5   background-color: var(--shade-light);
6   margin-top: var(--content-gap);
7 }
```

Physical Divider or Pseudo Element?

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

Accent Box Effect

Now both dividers are the same because they share the same class. With a modifier we can manipulate the element

Structure

- Section
 - Container
 - Heading
 - Rich Text
- Section
 - Container
 - Heading
 - Basic Text
- Divider
- Divider
- Container
- Container
- Container
- Container
- Container
- Global
- Global
- Global

11:21

Divder

.pseudo-divder--top

.pseudo-divder

.pseudo-divder--top

CONTENT

STYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

CSS

CSS Filters

Learn more about CSS filters

Transition

width 0s ease 0s

Learn more about CSS transitions

Custom CSS

1

Physical Divder or Pseudo Element?

Modifier class

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to

Physical Divider or Pseudo Element?

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

Accent Box Effect

Because of the flex container we can easily move the pseudo element to the top

```
1 root::after {
2   order: -1;
3 }
```


We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" – ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

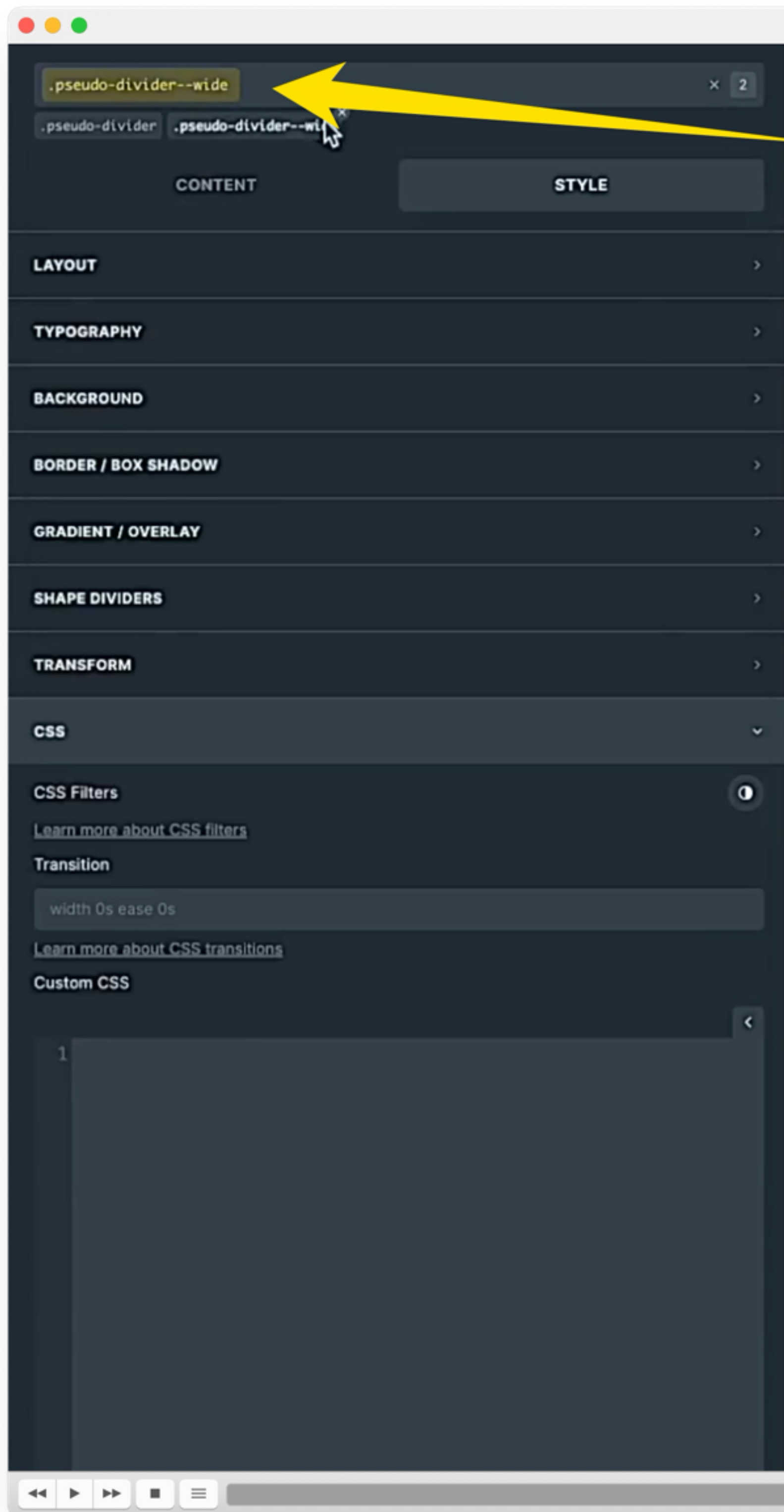
- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" – ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" – ::before and ::after elements are placed *inside* the box, not outside the box).

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



Physical Divider or Pseudo Element?

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

GLOBAL CLASS CREATED

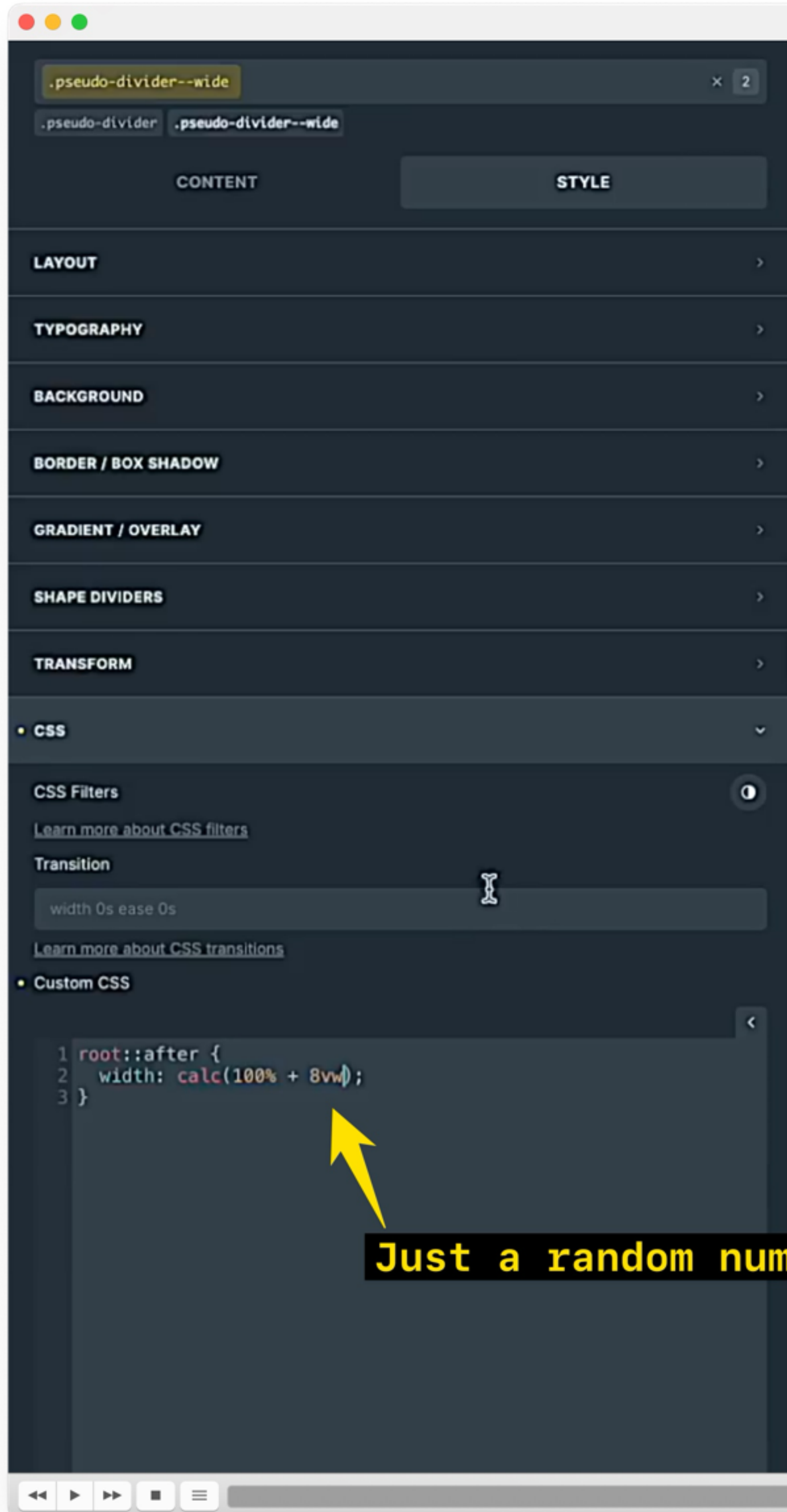
- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

+ [] + ÷

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

Accent Box Effect



Physical Divider or Pseudo Element?

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

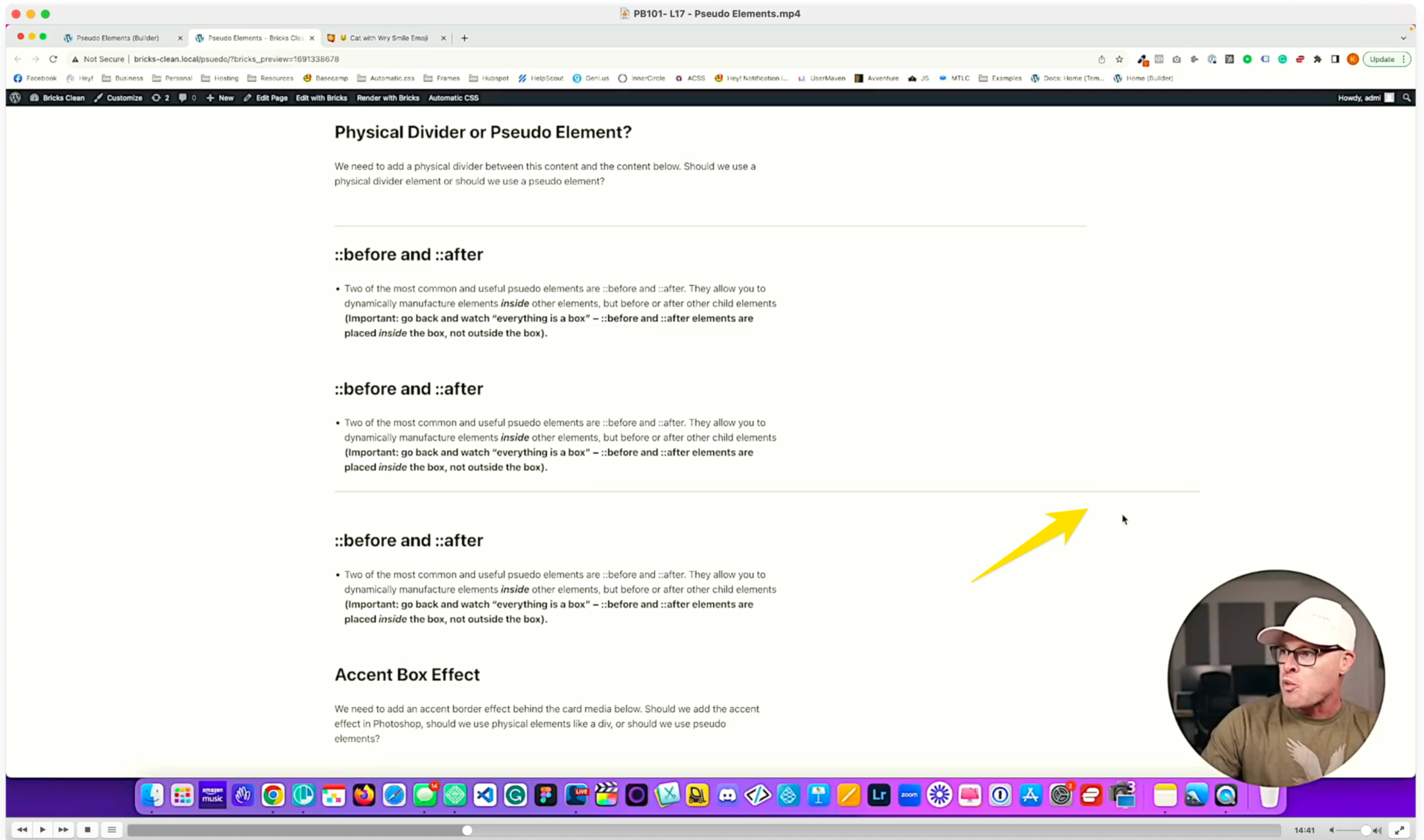
GRAND

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

Accent Box Effect



Physical Divider or Pseudo Element?

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

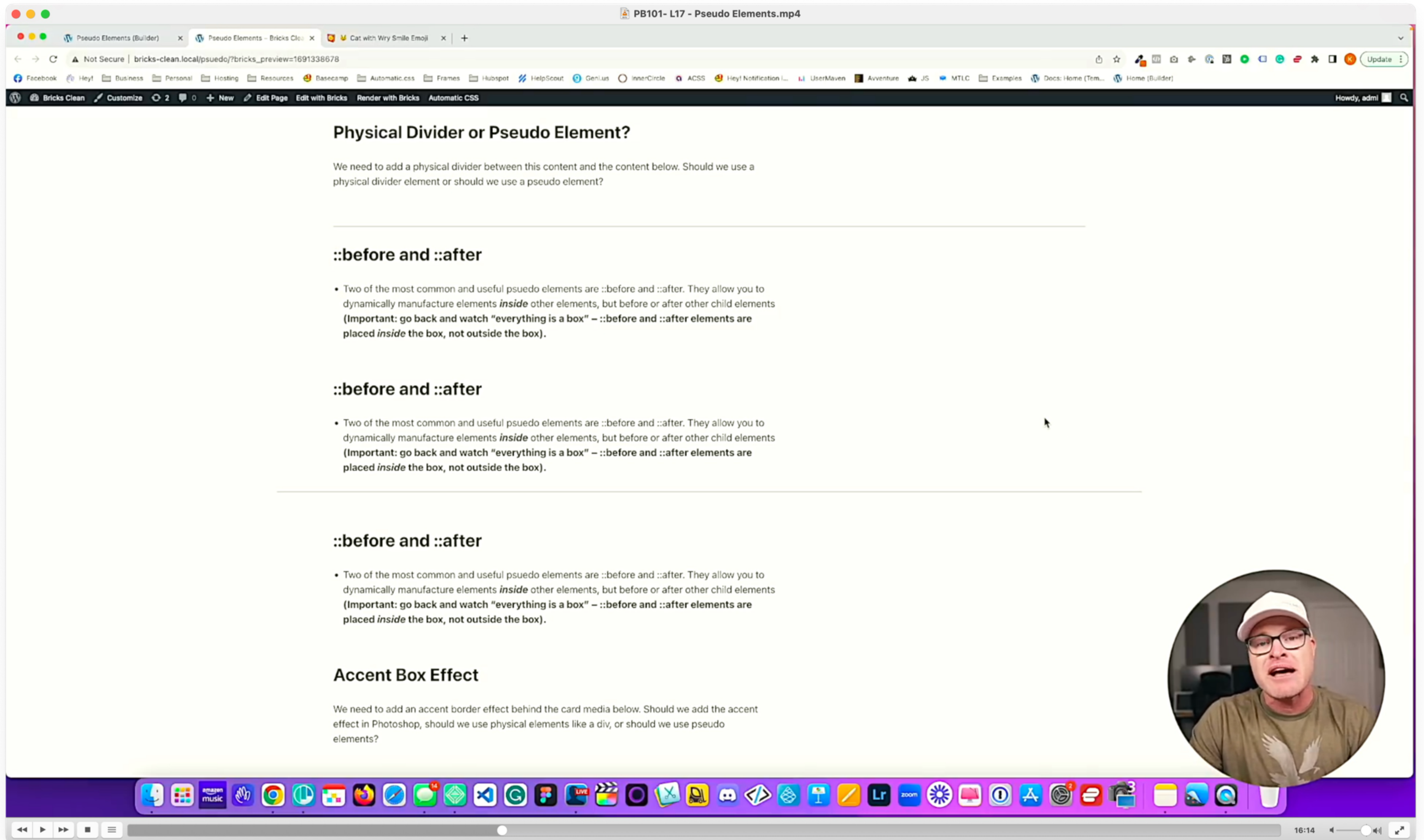
::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

This CSS technique positioned the pseudo element in the middle of the viewport

```
1 root::after {
2   width: calc(100% + 8vw);
3   transform: translateX(-50%);
4   margin-left: 50%;
5 }
```





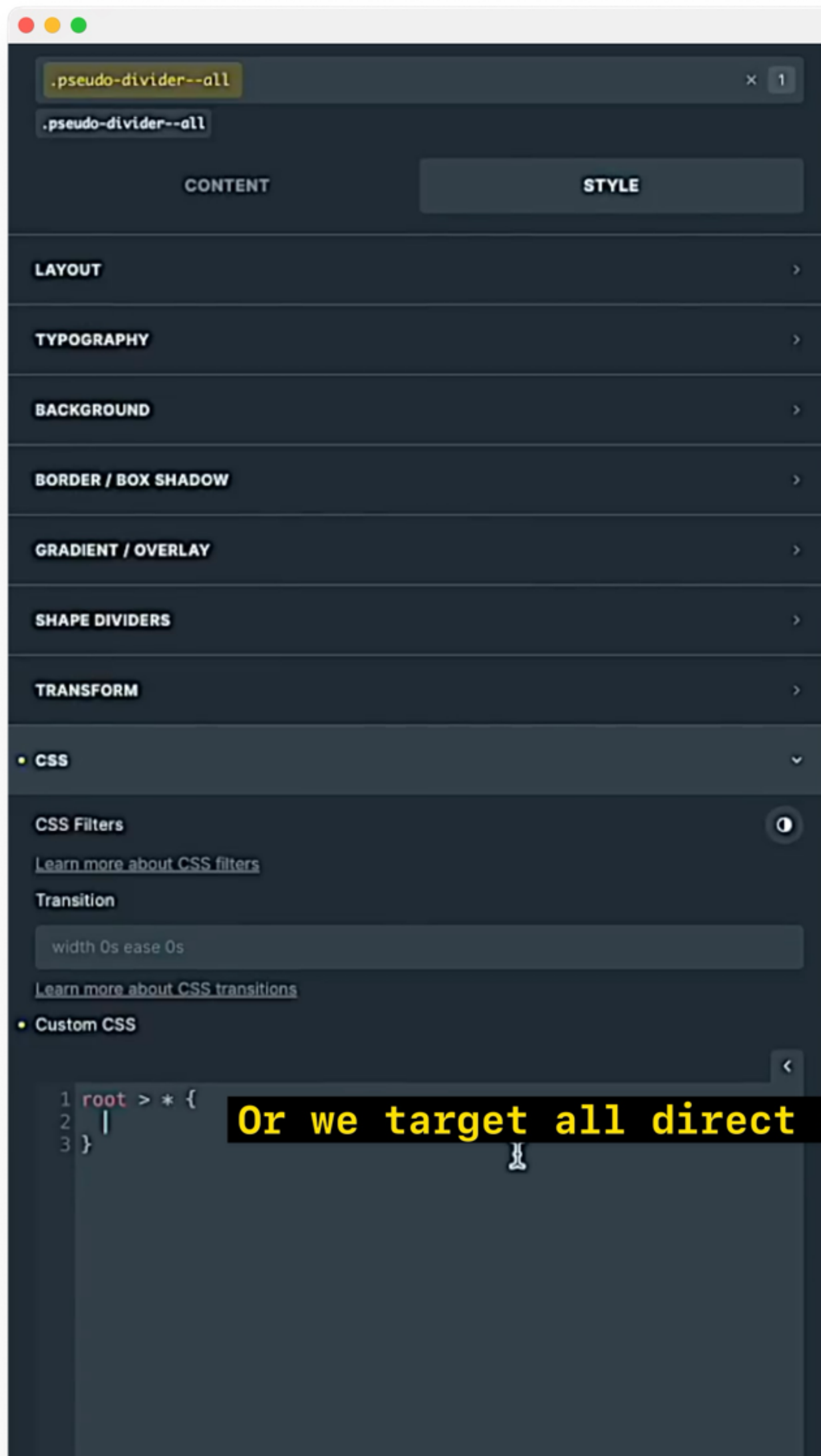


- ## Physical Divider or Pseudo Element?

::before and ::after

- The class is on the parent element (in this case the section) and we could target all Bricks containers or...

- Two of the most common and useful pseudo elements are `::before` and `::after`. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - `::before` and `::after` elements are placed *inside* the box, not outside the box).



- They're very difficult to imagine and conceptualize, so it's best if we take a look at some real-world examples.

Physical Divider or Pseudo Element?

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

::before and ::after

- Two of the most common and useful psuedo elements are ::before and ::after. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - ::before and ::after elements are placed *inside* the box, not outside the box).

× 1

.pseudo-divider--all

CONTENT

STYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

- **CSS**

CSS Filters

[Learn more about CSS filters](#)

Transition

[Learn more about CSS transitions](#)

- Custom CSS

```
1 root > *::after {
2   content: '';
3   width: 100%;
4   height: 2px;
5   background-color: red;
6   margin-top: calc(var(--container-gap) - var(--content-
7     gap));
```

Now your variables...

Physical Divider or Pseudo Element?

We need to add a physical divider between this content and the content below. Should we use a physical divider element or should we use a pseudo element?

::before and ::after

SAVED

- Two of the most common and useful psuedo elements are `::before` and `::after`. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - `::before` and `::after` elements are placed *inside* the box, not outside the box).

::before and ::after

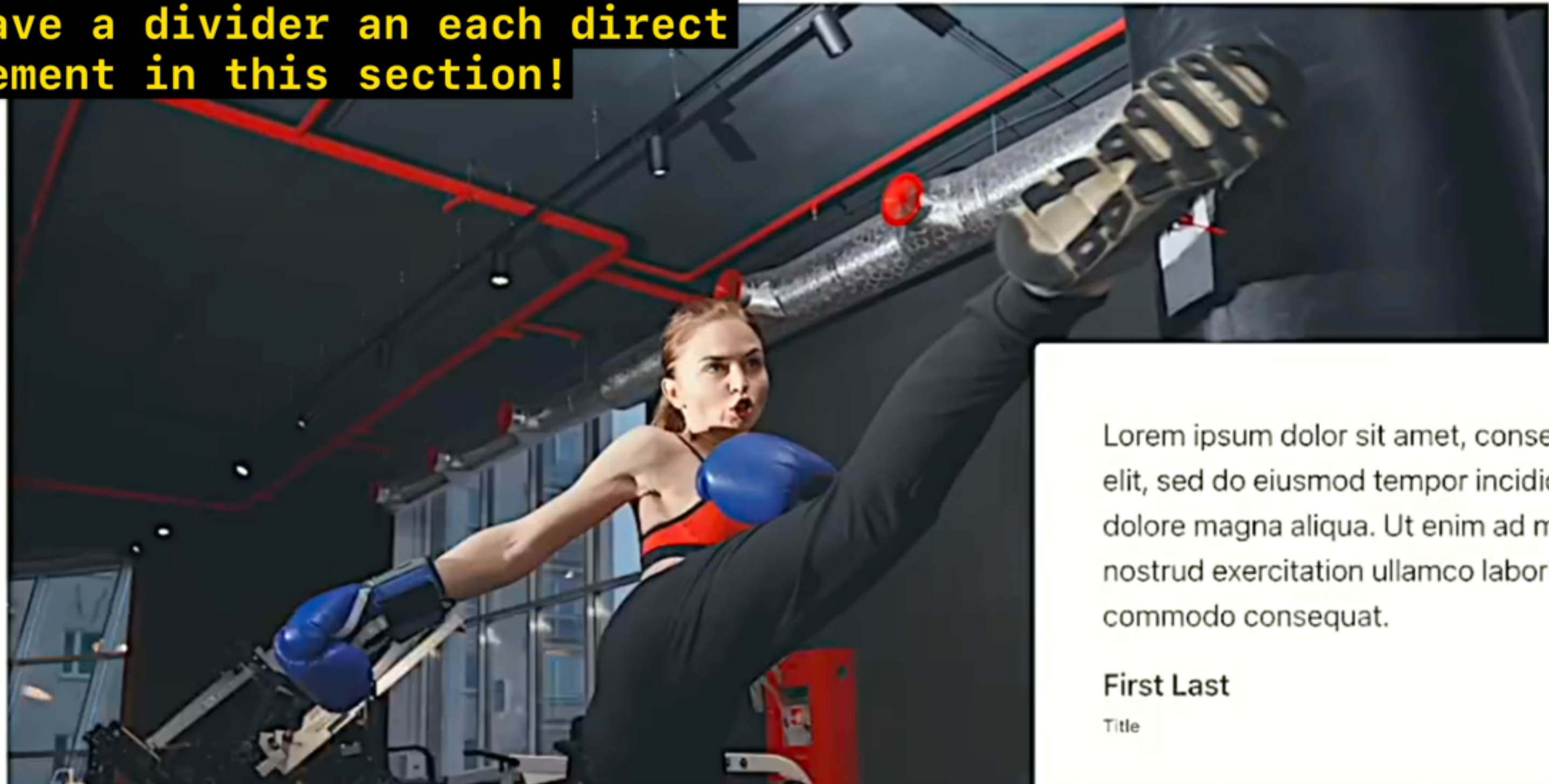
- Two of the most common and useful psuedo elements are `::before` and `::after`. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - `::before` and `::after` elements are placed *inside* the box, not outside the box).

- Two of the most common and useful pseudo elements are `::before` and `::after`. They allow you to dynamically manufacture elements *inside* other elements, but before or after other child elements (Important: go back and watch "everything is a box" - `::before` and `::after` elements are placed *inside* the box, not outside the box).

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a `div`, or should we use pseudo elements?

Now we have a divider on each direct child element in this section!



Lorem ipsum dolor sit amet, consectetur adipiscing 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 commodo consequat.

First Last

Title

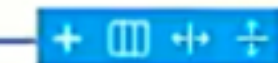
Mastering ::before & ::after will immediately level up your game.

So many creative and scalability challenges are overcome with ::before and ::after pseudo elements, especially in combination with absolute positioning, z-index, pseudo-classes, and other techniques. Work to master the use of ::before and ::after pseudo elements!

Pseudo elements can be assigned to classes, IDs, or HTML elements, making them very flexible, scalable, and maintainable where adding additional HTML is probably not.

Doesn't work on images - they need a div wrapper or <figure> tag.

Pseudo elements aren't selectable and aren't accessible to many screen readers, so only use them for decorative content.



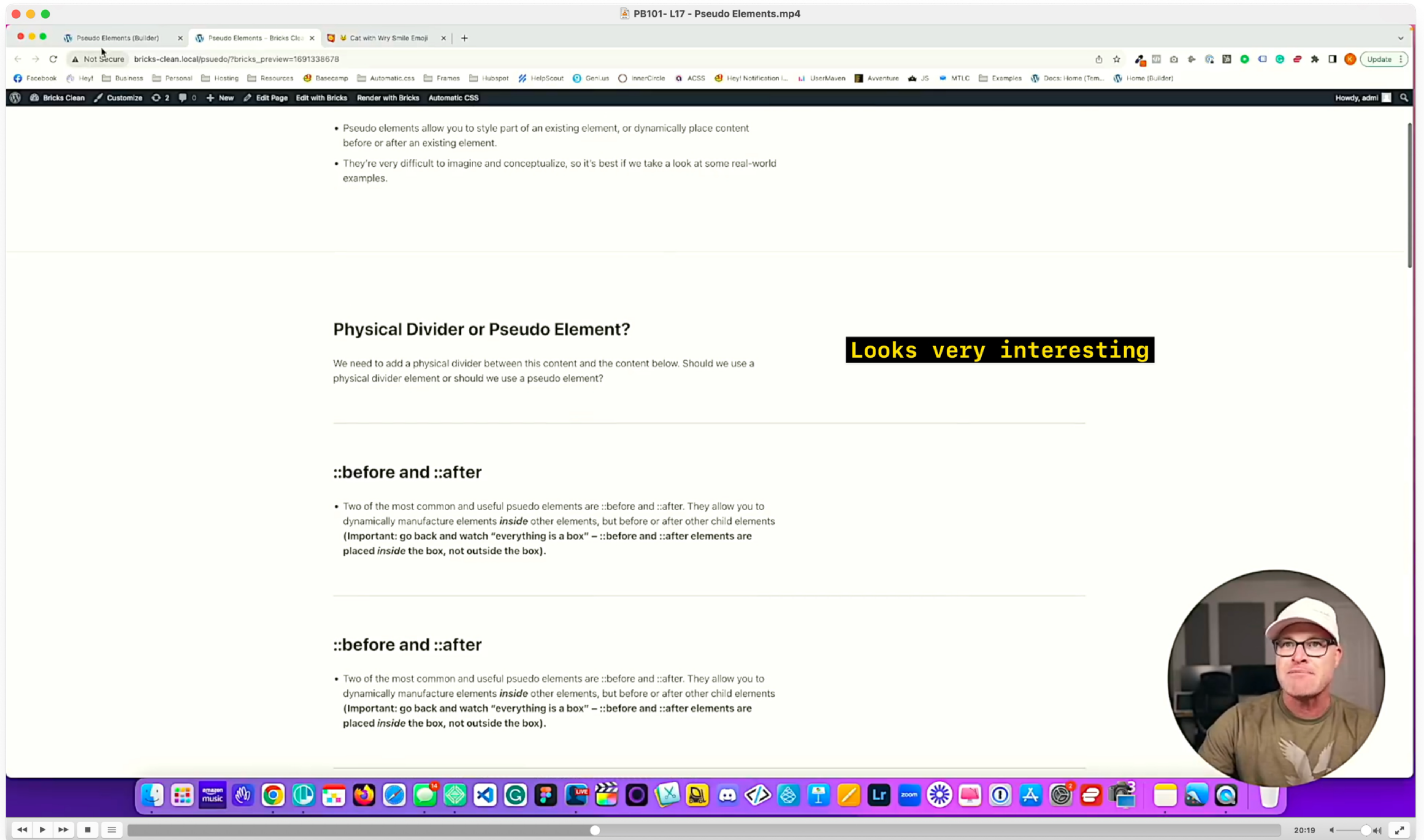
::marker

The icons in front of list items are called markers. They can be styled with the ::marker pseudo element.

- List item one
- List item two
- List item three

Now the element does not display on the last child in the section

```
1 root > *:not(:last-child)::after {
2   content: '';
3   width: 100%;
4   height: 2px;
5   background-color: red;
6   margin-top: calc(var(--container-gap) - var(--content-gap));
7 }
```

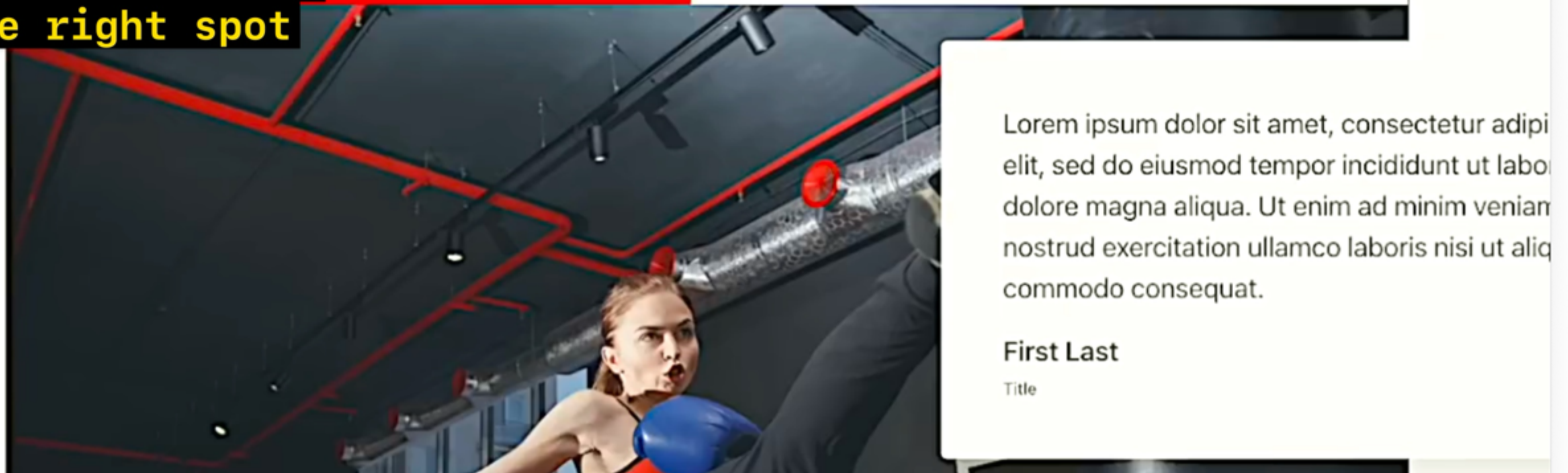



Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



Now that we can see the pseudo element we can easily position it to the right spot



.media-accent x 2

.fr-testimonial-card-foxtrot__media-wrapper .media-accent

CONTENT

STYLE

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

width 0s ease 0s

[Learn more about CSS transitions](#)

• Custom CSS

```
1 root::before {  
2   content: '';  
3   width: 500px;  
4   height: 500px;  
5   background-color: red;  
6   |  
7 }
```

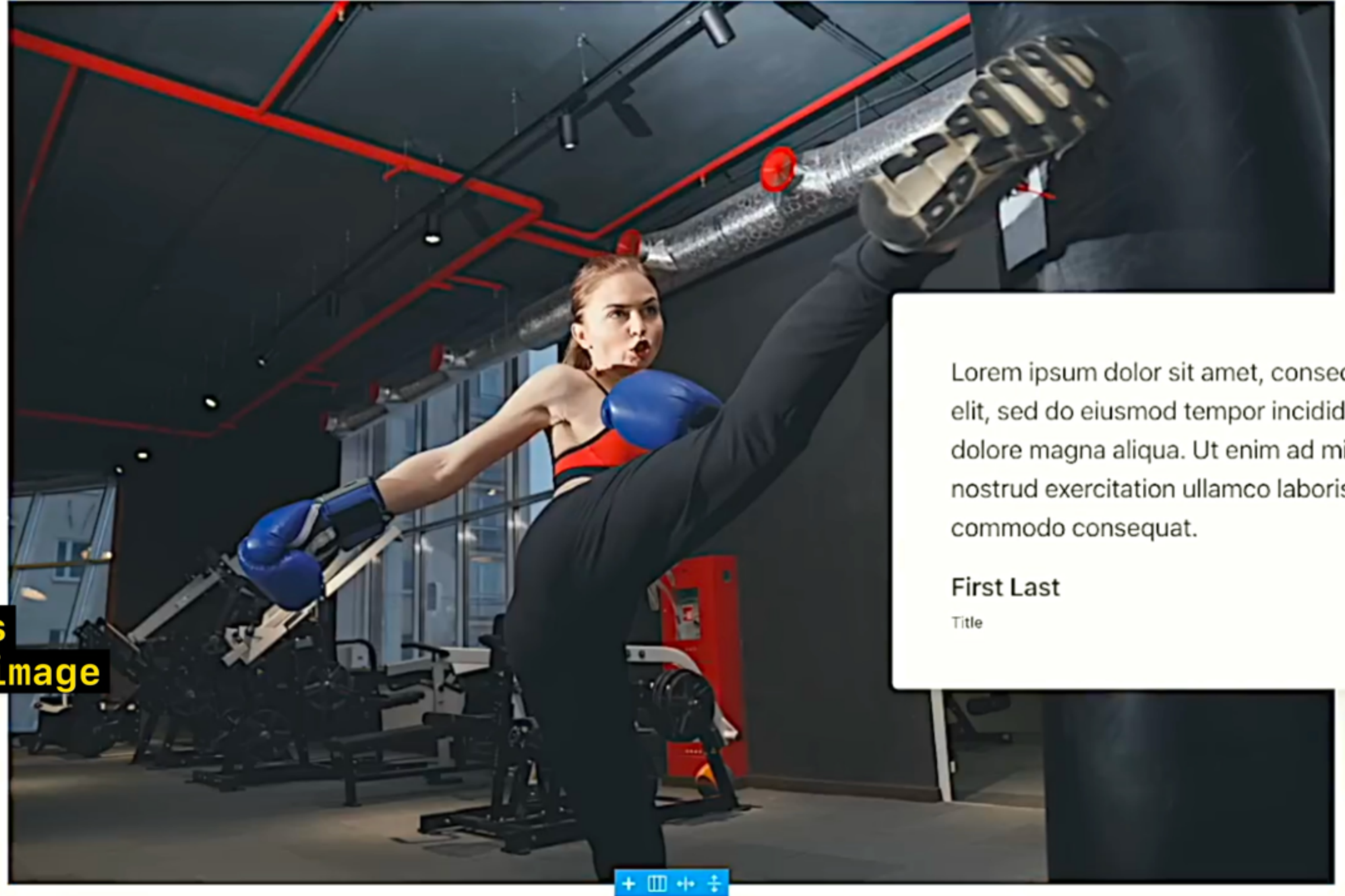
Use "root" to target the element wrapper: root { background: blue }

CSS classes

Separated by space. Without class dot.

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



Now the element is right behind the image

Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?

.media-accent

.fr-testimonial-card-foxtrot__media-wrapper .media-accent

CONTENT

STYLE

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

width 0s ease 0s

[Learn more about CSS transitions](#)

• Custom CSS

```
1 root::before {
2   content: '';
3   width: 500px;
4   height: 500px;
5   background-color: red;
6   position: absolute;
7 }
```

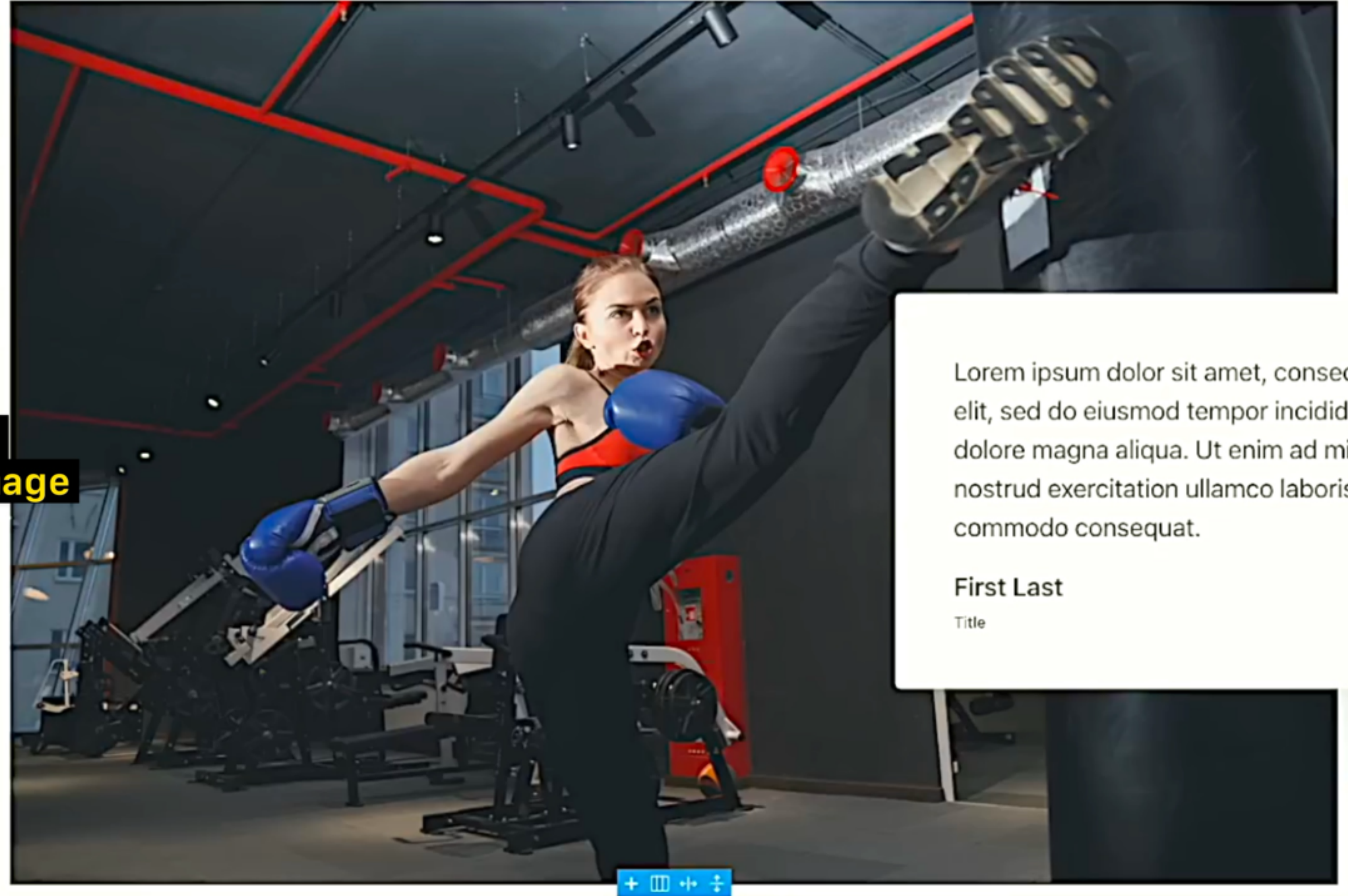
Use "root" to target the element wrapper: root { background: blue }

CSS classes

Separated by space. Without class dot.

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



Now the element is anchored to the image

```
1 root {  
2   position: relative;  
3 }  
4  
5 root::before {  
6   content: '';  
7   width: 500px;  
8   height: 500px;  
9   background-color: red;  
10  position: absolute;  
11 }
```

Use "root" to target the element wrapper: root { background: blue }

CSS classes

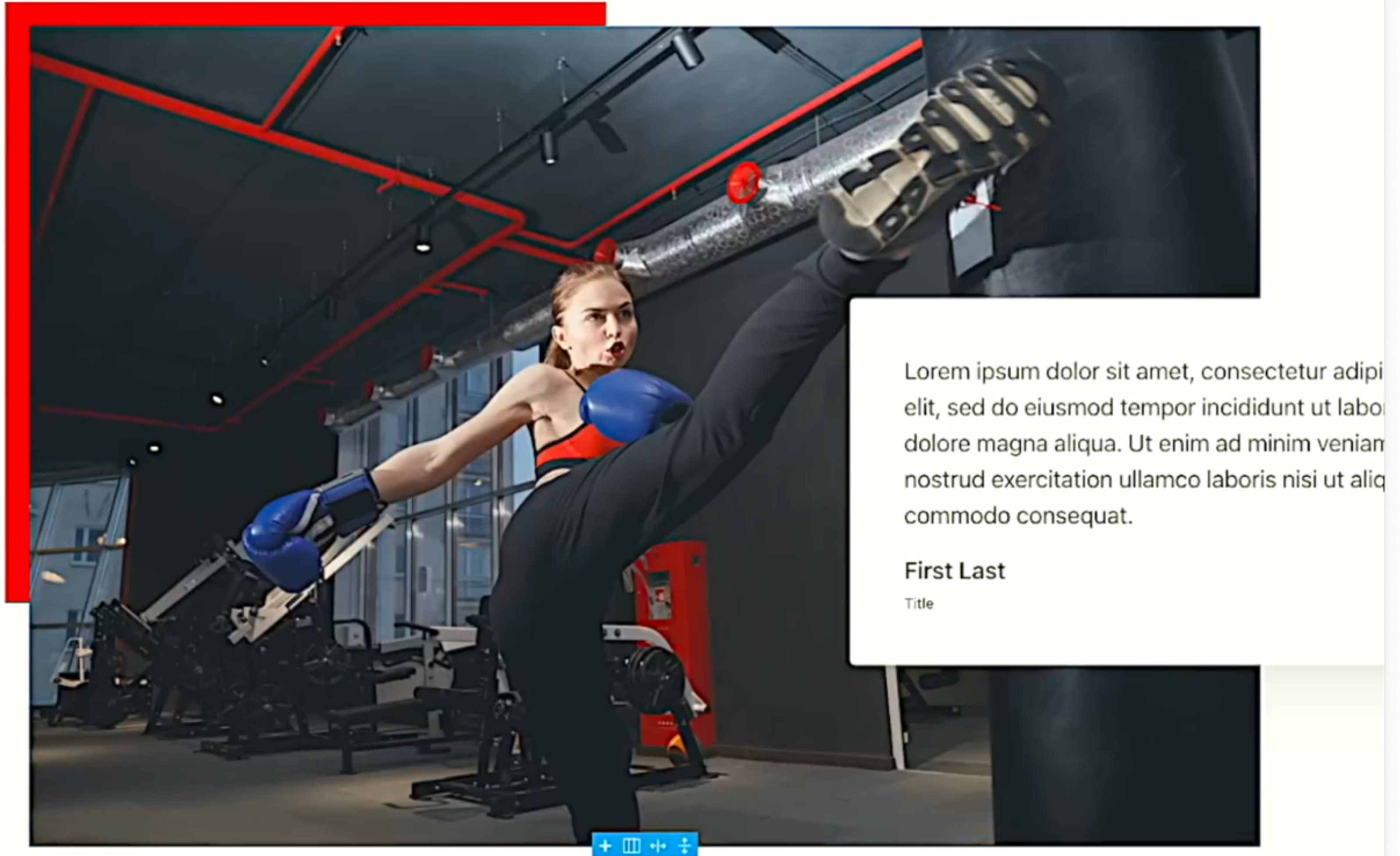
Separated by space. Without class dot.

Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore dolore magna aliqua. Ut enim ad minim veniam nostrud exercitation ullamco laboris nisi ut aliquammodo consequat.

First Last

Title

Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?

.media-accent

× 2

.fr-testimonial-card-foxtrot__media-wrapper .media-accent

CONTENT

STYLE

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

width 0s ease 0s

[Learn more about CSS transitions](#)

• Custom CSS

```
1 root {
2   position: relative;
3 }
4
5 root::before {
6   content: '';
7   width: 500px;
8   height: 500px;
9   background-color: red;
10  position: absolute;
11  top: -1em;
12  left: -1em;
13 }
```

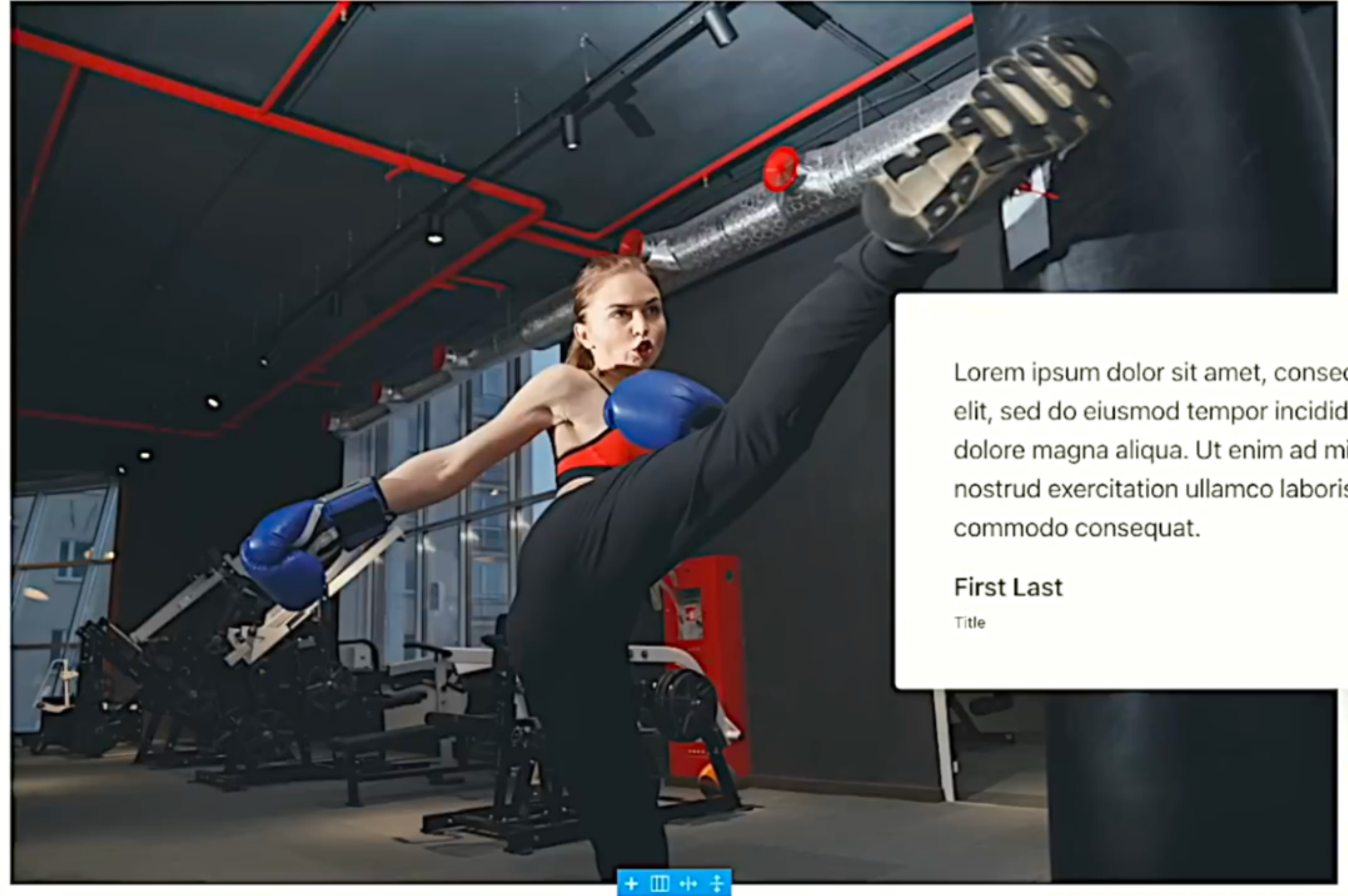
Use "root" to target the element wrapper: root { background: blue }

CSS classes

Separated by space. Without class dot.

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore dolore magna aliqua. Ut enim ad minim veniam nostrud exercitation ullamco laboris nisi ut aliquammodo consequat.

First Last

Title

Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?

.media-accent x 2

.fr-testimonial-card-foxtrot__media-wrapper .media-accent

CONTENT

STYLE

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

width 0s ease 0s

[Learn more about CSS transitions](#)

• Custom CSS

```
1 root {  
2   position: relative;  
3 }  
4  
5 root::before {  
6   content: '';  
7   width: 500px;  
8   height: 500px;  
9   background-color: transparent;  
10  border-left: 2px solid red;  
11  border-top: 2px solid red;  
12  position: absolute;  
13  top: -1em;  
14  left: -1em;  
15 }
```

Use "root" to target the element wrapper: root { background: blue }

CSS classes

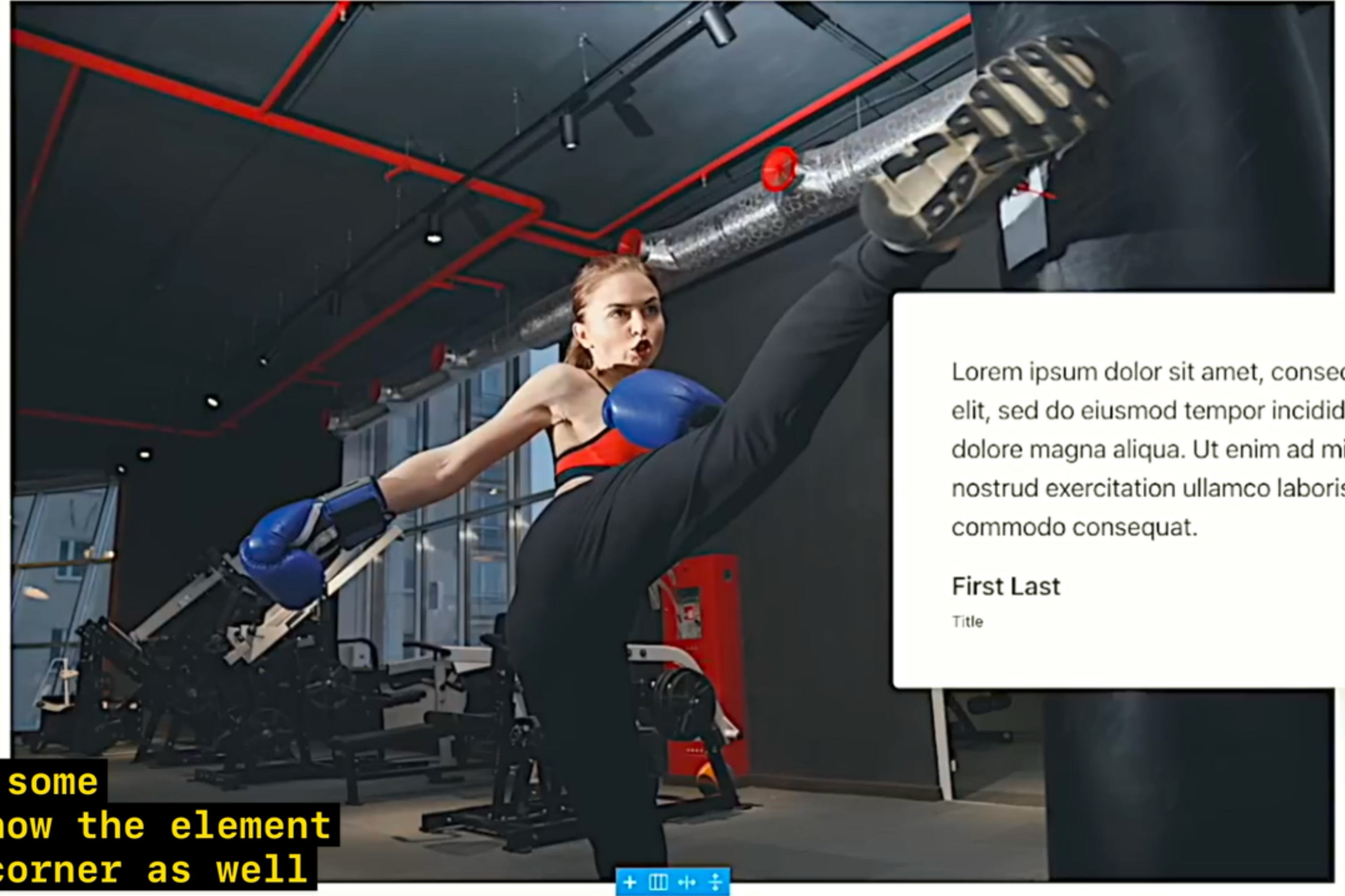
Separated by space. Without class dot.



24:15

Accent Box Effect

We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



With a duplicate and some adjustments we can show the element on the bottom right corner as well

Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?

.media-accent x 2

.fr-testimonial-card-foxtrot__media-wrapper .media-accent

CONTENT

STYLE

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

width 0s ease 0s

[Learn more about CSS transitions](#)

• Custom CSS

```
4
5 root::before {
6   content: '';
7   width: 300px;
8   height: 300px;
9   background-color: transparent;
10  border-left: 2px solid red;
11  border-top: 2px solid red;
12  position: absolute;
13  top: -1em;
14  left: -1em;
15 }
16
17 root::after {
18   content: '';
19   width: 300px;
20   height: 300px;
21   background-color: transparent;
22   border-right: 2px solid red;
23   border-bottom: 2px solid red;
24   position: absolute;
25   bottom: -1em;
26   right: -1em;
27 }
```

Use "root" to target the element wrapper: root { background: blue }

CSS classes

Separated by space. Without class dot.

A woman in a gym is performing a high kick, wearing blue boxing gloves and a red sports bra. She is in a dynamic pose, with one leg raised high and the other extended forward. The background shows gym equipment and a red structure. A white text box is overlaid on the right side of the image, containing placeholder text and a title.

We need the background image below to have an overlay. Can we use a pseudo element?



Block

.media-accent

.media-accent

CONTENTSTYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

CSS

CSS Filters

Learn more about CSS filters

Transition

width 0s ease 0s

Learn more about CSS transitions

Custom CSS

```
1 root {
2   position: relative;
3 }
4
5 root::before {
6   content: '';
7   width: 300px;
8   height: 300px;
9   background-color: transparent;
10  border-left: 2px solid red;
11  border-top: 2px solid red;
12  position: absolute;
13  top: -1em;
14  left: -1em;
15 }
16
17 root::after {
18   content: '';
19   width: 300px;
20   height: 300px;
21   background-color: transparent;
22   border-right: 2px solid red;
23   border-bottom: 2px solid red;
24   position: absolute;
```

Structure

Section

Section

Container

Divider

Divider

Container

Container

Container

Testimonial Card Foxtrot (C

Content Wrapper

Media Wrapper

Image

Container

Heading

Basic Text

Block


Container

Global

Global

Global

W 1655H 100%



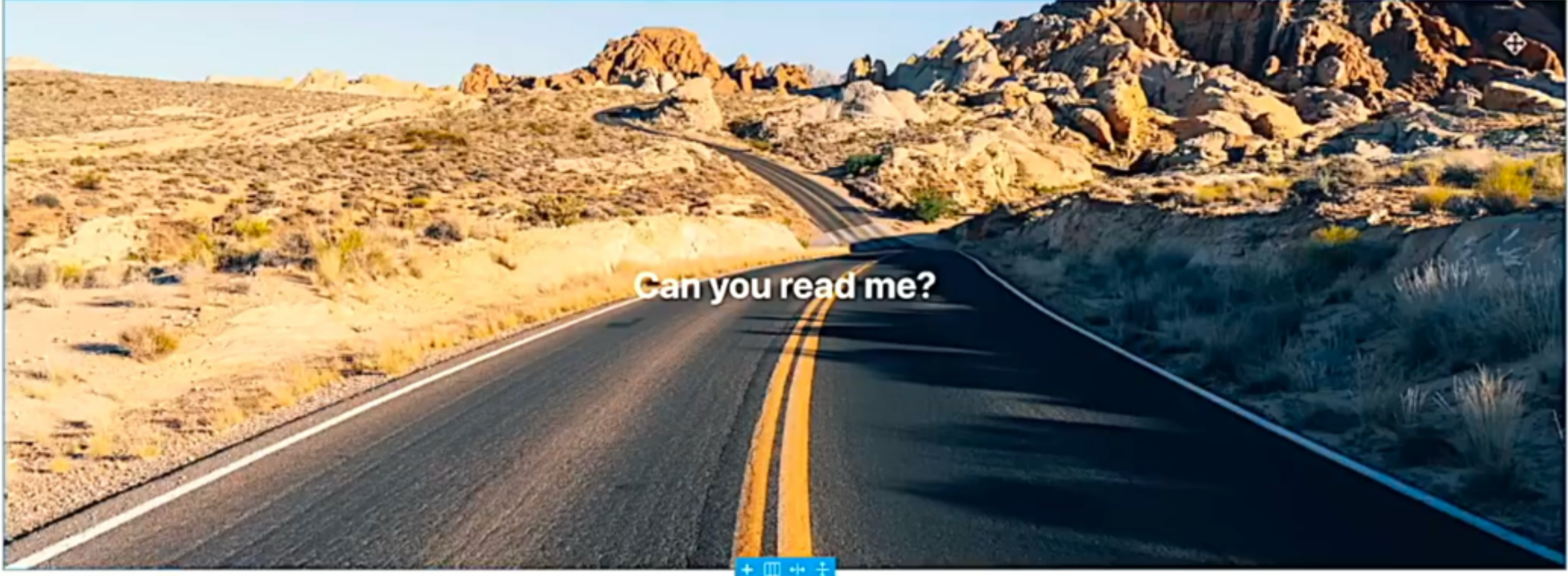
dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

First Last

Title

Background image overlay.

We need the background image below to have an overlay. Can



Can you read me?

Mastering ::before & ::after will immediately level up your game.

25:39

We need the background image below to have an overlay. Can we use a pseudo element?

A photograph of a winding asphalt road in a desert landscape. The road has double yellow lines in the center and white lines on the edges. The surrounding terrain is arid with sparse vegetation and rocky hills in the background. A yellow rectangular box with the word "variable" in black text is positioned on the left side of the road. The text "Can you read me?" is written in white across the middle of the road. At the bottom center, there is a small blue icon with a white crosshair.

Herestering ::before & ::after will immediately level up your game.

Locally scoped variable

Used here

Used here

stering ::before & ::after will immediately level up

Media Wrapper

`.media-accent`

x 2

`.fr-testimonial-card-foxtrot__media-wrapper``.media-accent`

CONTENT

STYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

`width 0s ease 0s`[Learn more about CSS transitions](#)

• Custom CSS

```
1 root {  
2   position: relative;  
3   --border-color: var(--action);  
4 }  
5  
6 root::before {  
7   content: '';  
8   width: 300px;  
9   height: 300px;  
10  background-color: transparent;  
11  border-left: 2px solid var(--border-color);  
12  border-top: 2px solid var(--border-color);  
13  position: absolute;  
14  top: -1em;  
15  left: -1em;  
16 }  
17  
18 root::after {  
19   content: '';  
20   width: 300px;
```

Now we can change
the color in 1 place

Lorem ipsum dolor sit amet, consectetur adipiscing 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 commodo consequat.

First Last

Title

Background image overlay.

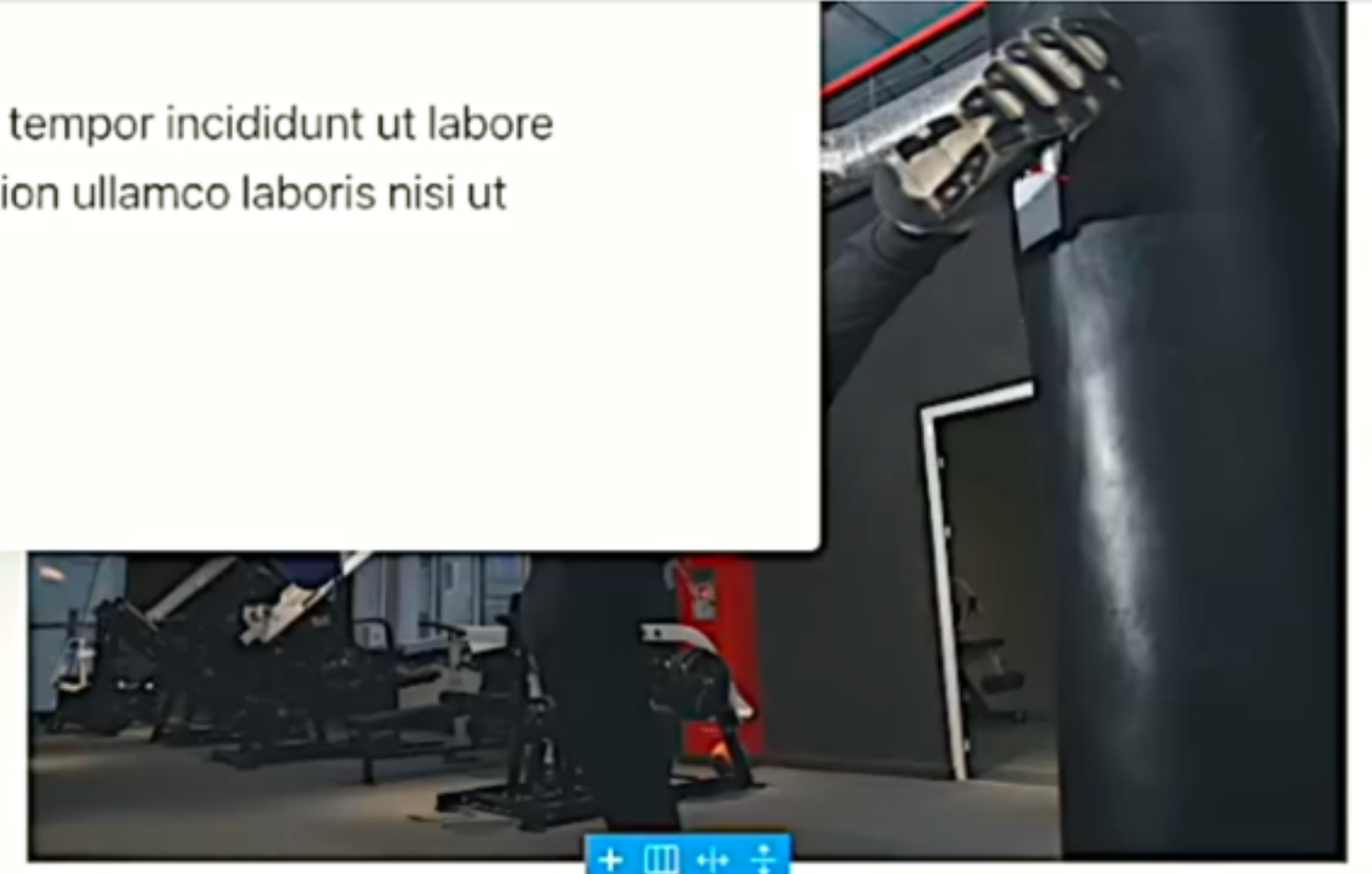
We need the background image below to have an overlay. Can we use a pseudo element?



Lorem ipsum dolor sit amet, consectetur adipiscing 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 commodo consequat.

First Last

Title



Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?



Now we can change the border width in one place as well

We're consolidating the rules both pseudo elements have in common

.media-accent

fr-testimonial-card-foxtrot__media-wrapper .media-accent

CONTENT

STYLE

BORDER / BOX SHADOW

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

CSS

CSS Filters

Learn more about CSS filters

Transition

width 0s ease 0s

Learn more about CSS transitions

Custom CSS

```
1 root {
2   position: relative;
3   --border-color: var(--action);
4   --border-width: 2px;
5 }
6
7 root::before, root::after {
8   content: '';
9   width: 300px;
10  height: 300px;
11  background-color: transparent;
12  position: absolute;
13 }
14
15 root::before {
16   border-left: var(--border-width) solid var(--border-color);
17   border-top: 2px solid var(--border-color);
18   top: -1em;
19   left: -1em;
20 }
21
22 root::after {
23   border-right: 2px solid var(--border-color);
24   border-bottom: 2px solid var(--border-color);
25 }
```

Use "root" to target the element wrapper: root { background: blue }

CSS classes

Separated by space. Without class dot.

.media-accent

.fr-testimonial-card-foxtrot__media-wrapper .media-accent

CONTENTSTYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

width 0s ease 0s

[Learn more about CSS transitions](#)

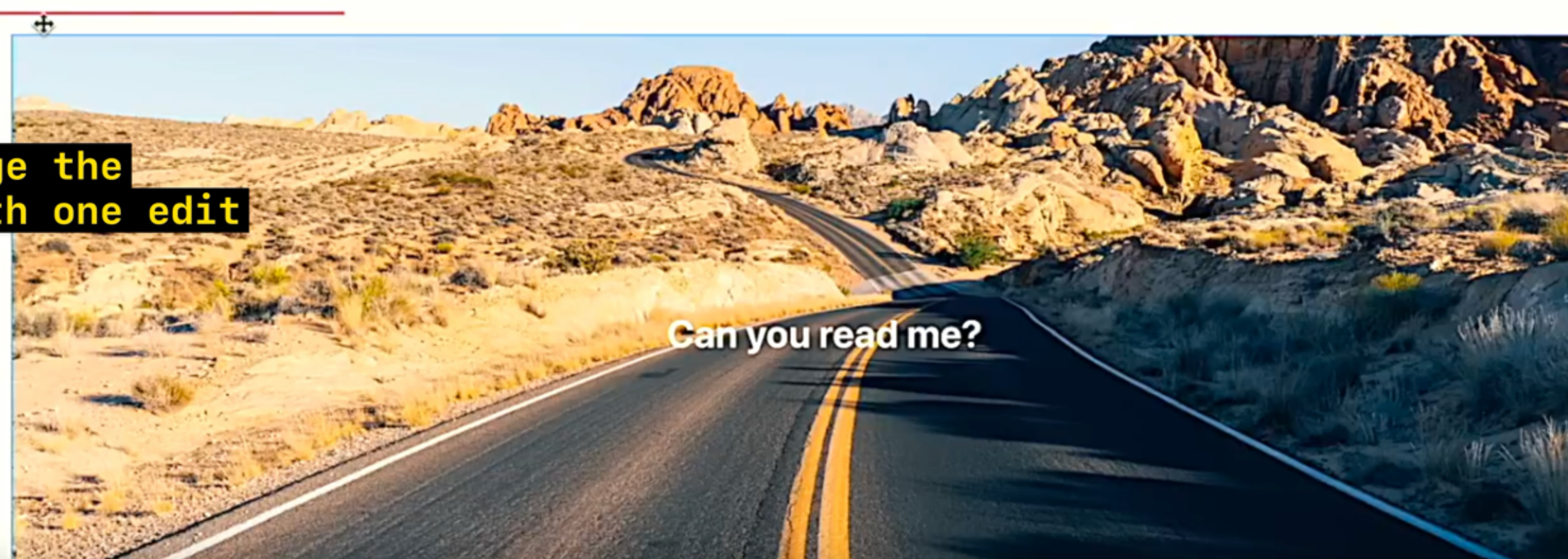
• Custom CSS

```
1 root {
2   position: relative;
3   --border-color: var(--action);
4   --border-width: 3px;
5   --border-style: solid;
6   --border-offset: -1em;
7 }
8
9
10 root::before, root::after {
11   content: '';
12   width: 300px;
13   height: 300px;
14   background-color: transparent;
15   position: absolute;
16 }
17
18 root::before {
19   border-left: var(--border-width) var(--border-style) var(--border-color);
20   border-top: var(--border-width) var(--border-style) var(--border-color);
21   top: var(--border-offset);
22   left: var(--border-offset);
```

Now we can change the border width with one edit

Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?



Lorem ipsum dolor sit amet, consectetur adipiscing 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 commodo consequat.

First Last

Title

Can you read me?

We need the background image below to have an overlay. Can we use a pseudo element?

We need the background image below to have an overlay. Can we use a pseudo element?

Can you read me?

Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?



Mastering ::before & ::after will immediately level up your game.

So many creative and scalability challenges are overcome with ::before and ::after pseudo elements, especially in combination with absolute positioning, z-index, pseudo-classes, and other techniques. Work to master the use of ::before and ::after pseudo elements!

Pseudo elements can be assigned to classes, IDs, or HTML elements, making them very flexible, scalable, and maintainable where adding additional HTML is probably not.

Doesn't work on images - they need a div wrapper or <figure> tag.

Lorem ipsum dolor sit amet, consectetur adipiscing
 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
 commodo consequat.

First Last

Title

We need the background image below to have an overlay. Can we use a pseudo element?

We need the background image below to have an overlay. Can we use a pseudo element?

With this setting we force the elements behind the image **Can you read me?**

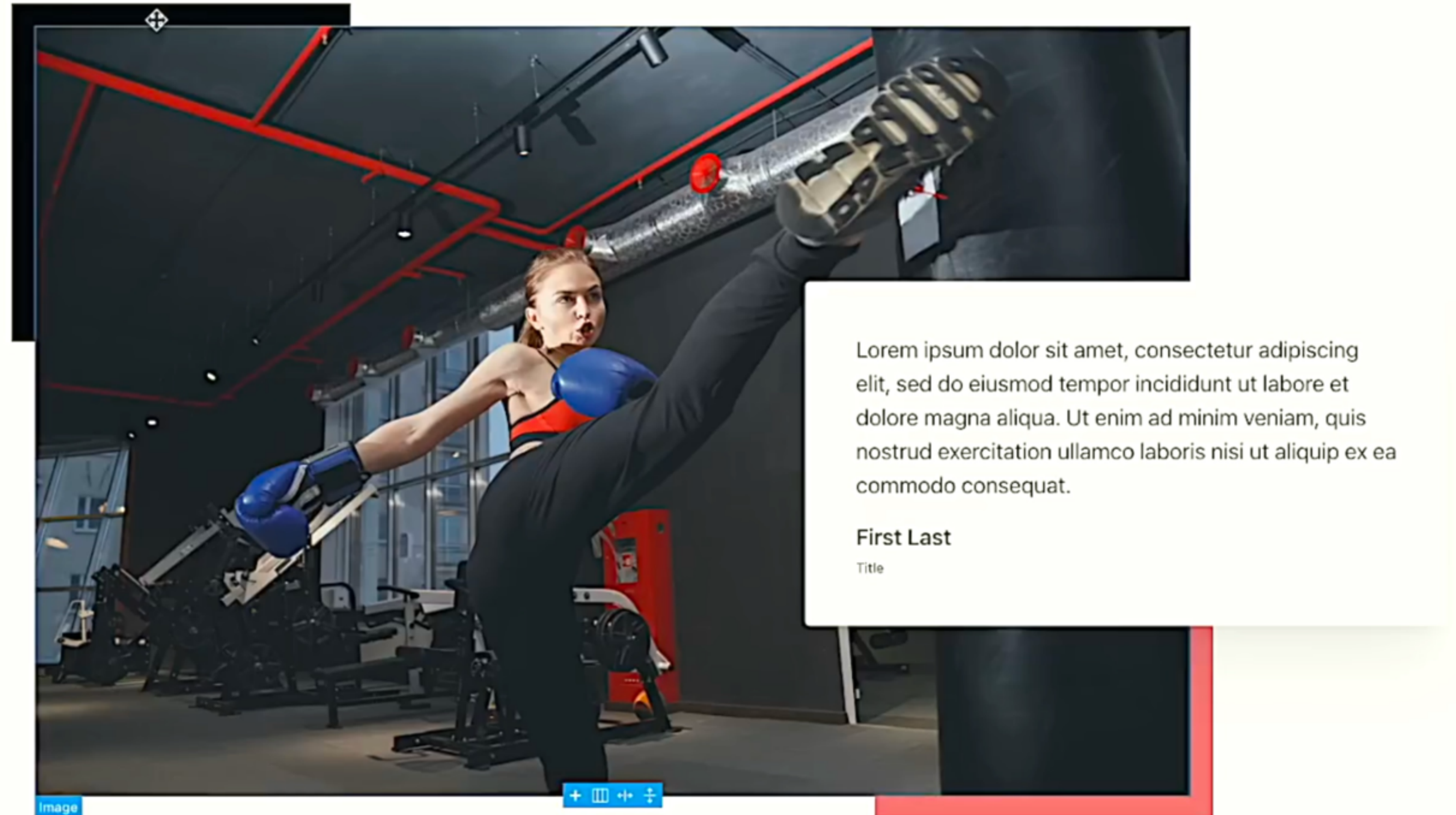
Can you read me?

```

8
9
10 root::before, root::after {
11   content: '';
12   width: 300px;
13   height: 300px;
14   background-color: transparent;
15   position: absolute;
16   z-index: -1;
17 }
18
19 root::before {
20   border-left: var(--border-width) var(--border-style) var(--border-color);
21   border-top: var(--border-width) var(--border-style) var(--border-color);
22   top: var(--border-offset);
23   left: var(--border-offset);
24 }
25
26 root::after {
27   border-right: var(--border-width) var(--border-style) var(--border-color);
28   border-bottom: var(--border-width) var(--border-style)

```


We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?

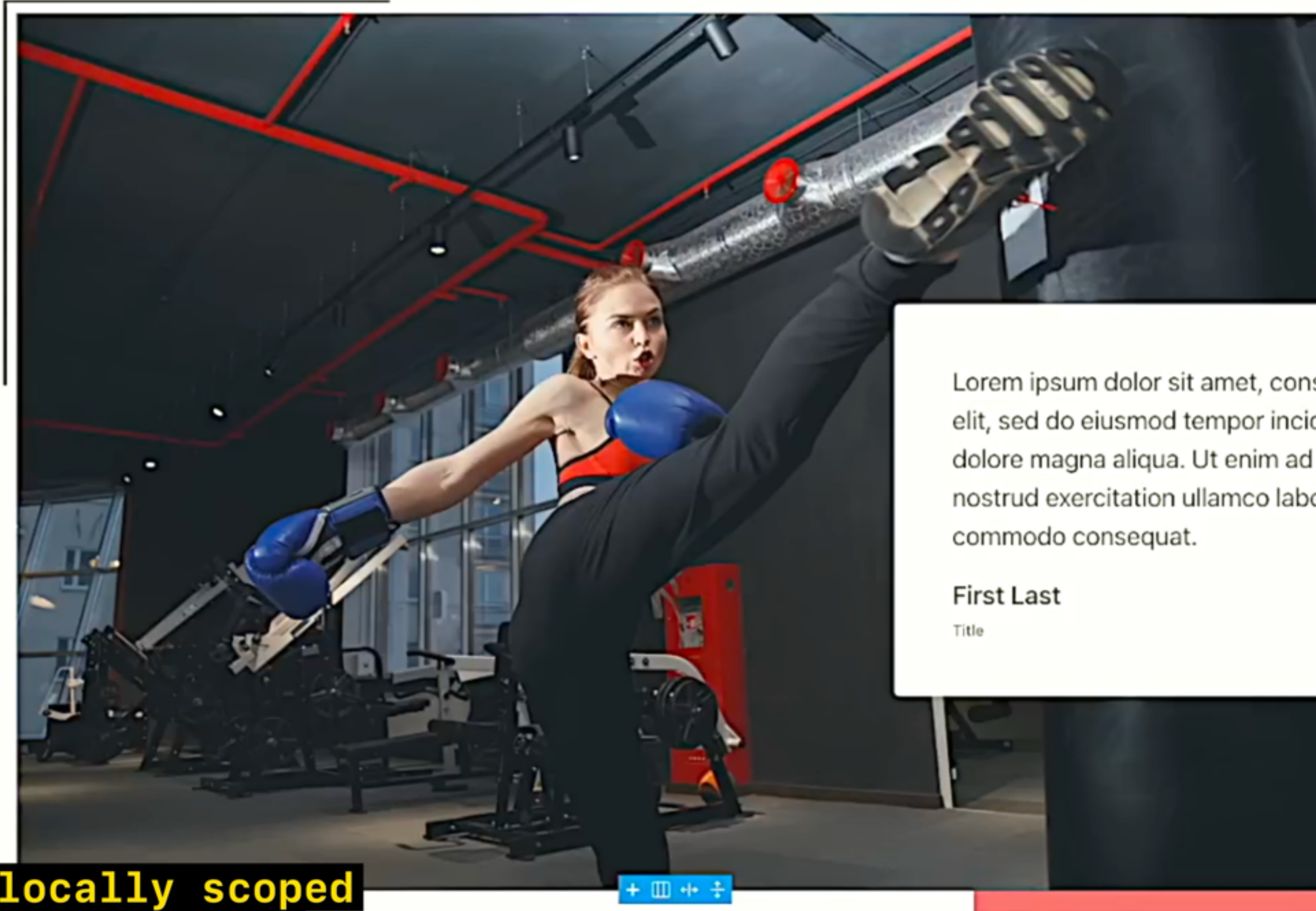


Now we can manipulate single elements on the ID level to change values for image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?



We need to add an accent border effect behind the card media below. Should we add the accent effect in Photoshop, should we use physical elements like a div, or should we use pseudo elements?



Lorem ipsum dolor sit amet, consectetur adipiscing 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 commodo consequat.

First Last

Title

We just redefine the locally scoped variables, not the values itself!

Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?

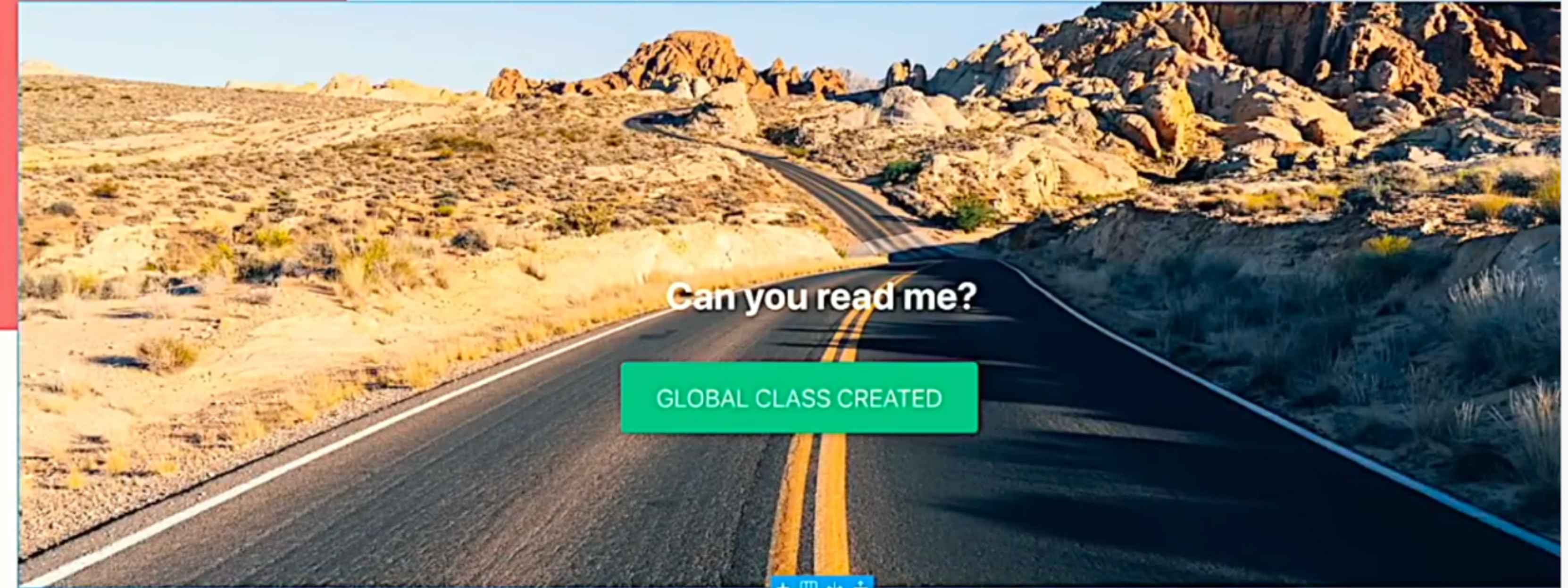


```
1 root::before {  
2   --border-color: black;  
3   --border-width: 2px;  
4   --border-offset: -.5em;  
5   |  
6 }
```




my-custom-overlay

We need the background image below to have an overlay. Can we use a pseudo element?



Mastering ::before & ::after will immediately level up your game.

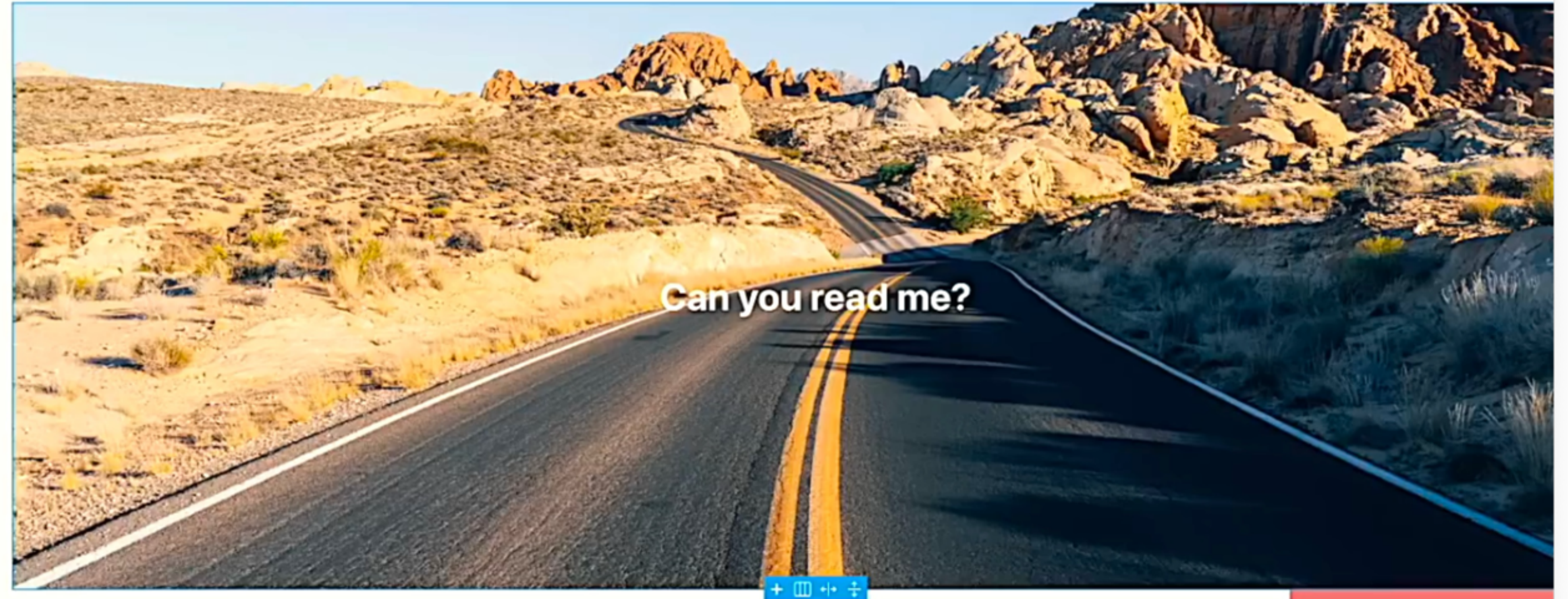
So many creative and scalability challenges are overcome with `::before` and `::after` pseudo elements, especially in combination with absolute positioning, z-index, pseudo-classes, and other techniques. Work to master the use of `::before` and `::after` pseudo elements!

Pseudo elements can be assigned to classes, IDs, or HTML elements, making them very flexible, scalable, and maintainable where adding additional HTML is probably not.

Doesn't work on images - they need a div wrapper or `<figure>` tag.

Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?



Mastering ::before & ::after will immediately level up your game.

Inset stretches the overlay over the whole image and scalability challenges are overcome with ::before and ::after pseudo elements, especially in combination with absolute positioning, z-index, pseudo-classes, and other techniques.

Work to master the use of ::before and ::after pseudo elements!

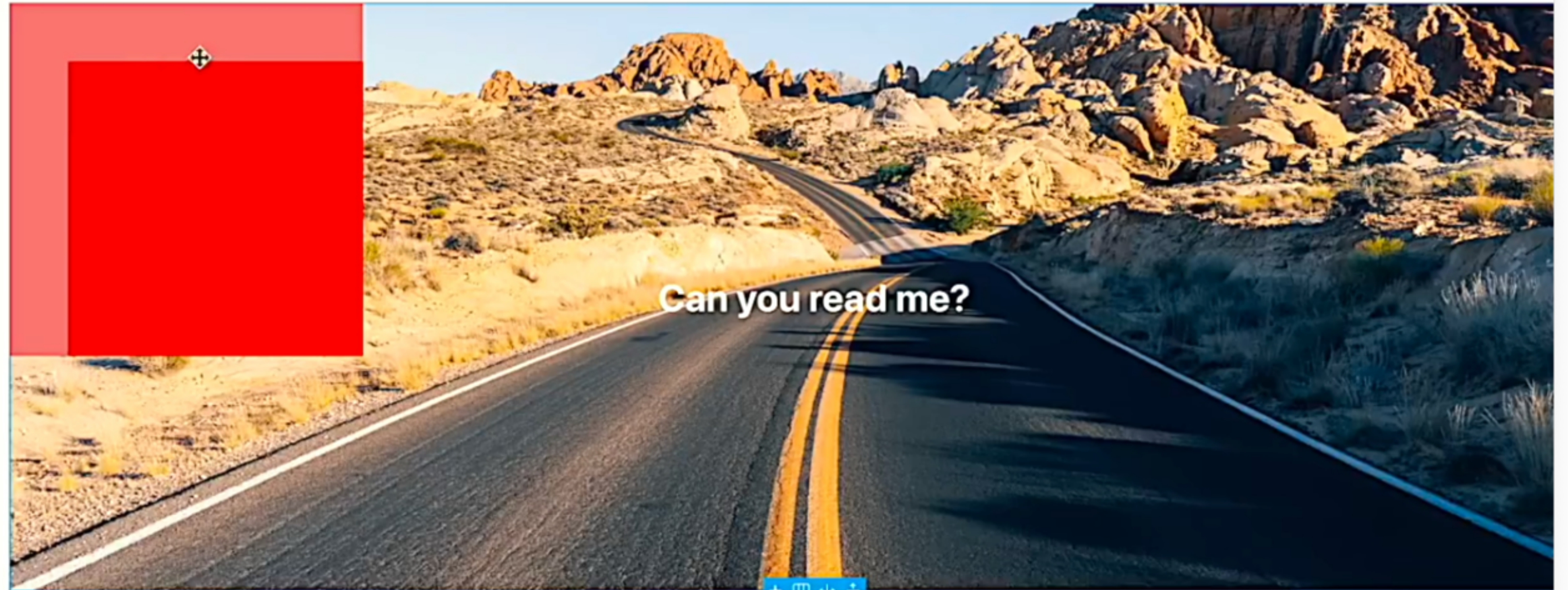
Pseudo elements can be assigned to classes, IDs, or HTML elements, making them very flexible, scalable, and maintainable where adding additional HTML is probably not.

Doesn't work on images - they need a div wrapper or <figure> tag.

```
1 root {  
2   position: relative;  
3 }  
4  
5 root::before {  
6   content: '';  
7   position: absolute;  
8   inset: 0;  
9 }
```


Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?



Mastering ::before & ::after will immediately level up your game.

Now we can see the element

So many creative and scalability challenges are overcome with ::before and ::after pseudo elements, especially in combination with absolute positioning, z-index, pseudo-classes, and other techniques. Work to master the use of ::before and ::after pseudo elements!

Pseudo elements can be assigned to classes, IDs, or HTML elements, making them very flexible, scalable, and maintainable where adding additional HTML is probably not.

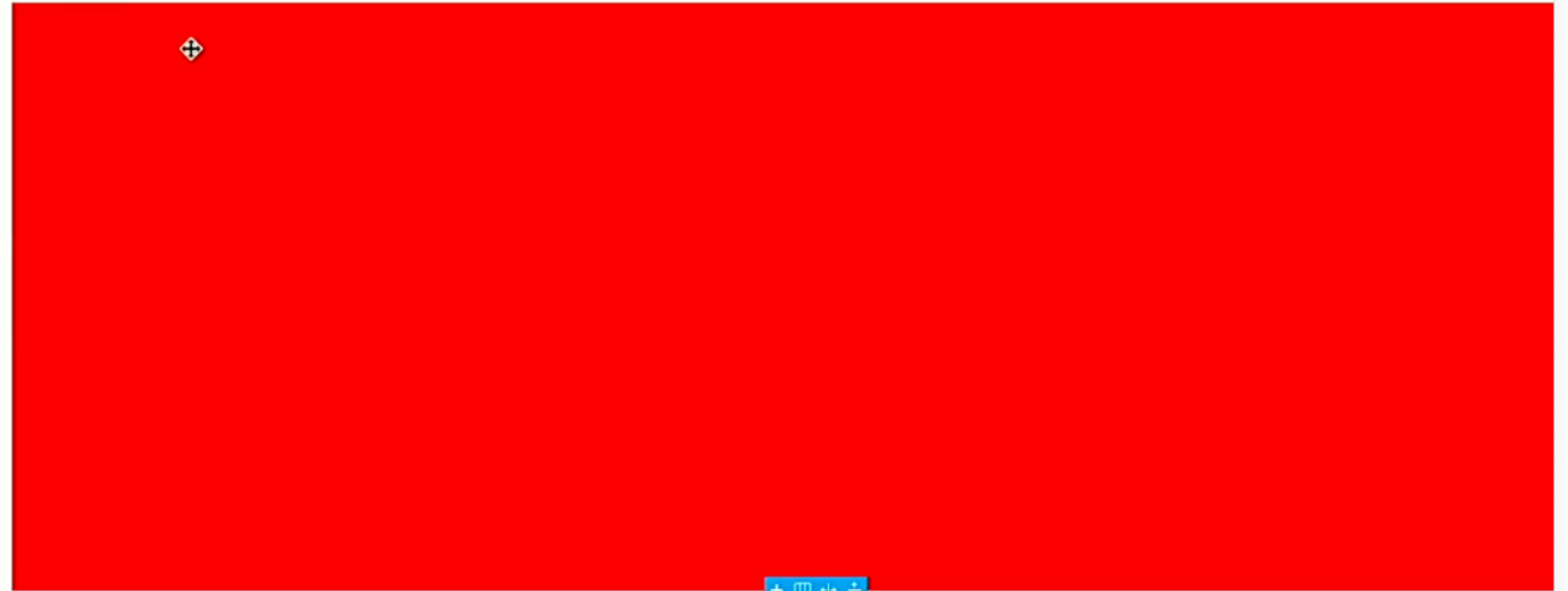
Doesn't work on images - they need a div wrapper or <figure> tag.

```

1 root {
2   position: relative;
3 }
4
5 root::before {
6   content: '';
7   position: absolute;
8   inset: 0;
9   background-color: red;
10  z-index: 1;
11 }
  
```


Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?



Mastering ::before & ::after will immediately level up your game.

You can add an extra wrapper because with every wrapper you get 2 new pseudo elements!

come with ::before and ::after pseudo elements, especially in combination with absolute positioning, z-index, pseudo-classes, and other techniques. Work to master the use of ::before and ::after pseudo elements!

Pseudo elements can be assigned to classes, IDs, or HTML elements, making them very flexible, scalable, and maintainable where adding additional HTML is probably not.

Doesn't work on images - they need a div wrapper or <figure> tag.

.my-custom-overlay x 1

CONTENT STYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

width 0s ease 0s

[Learn more about CSS transitions](#)

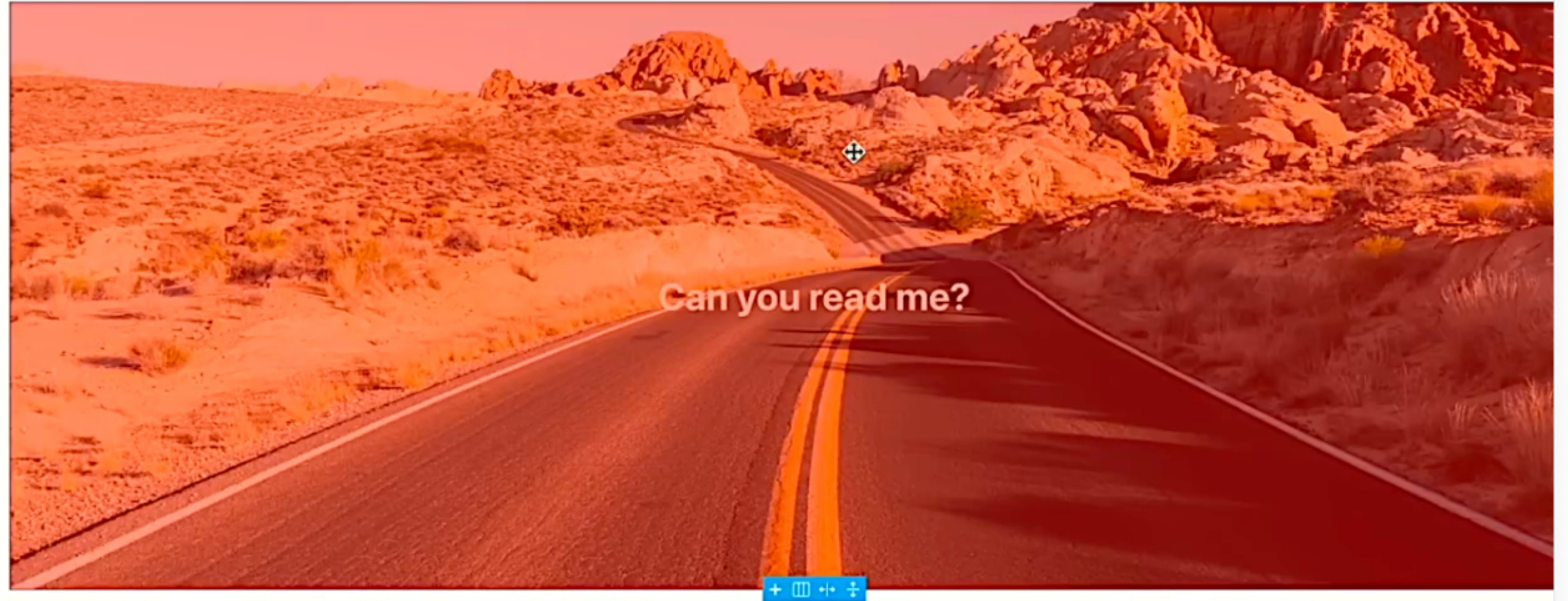
• Custom CSS

```

1 root {
2   position: relative;
3 }
4
5 root::before {
6   content: '';
7   position: absolute;
8   inset: 0;
9   width: 100%;
10  height: 100%;
11  background-color: red;
12  z-index: 1;
13 }
```


Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?



Mastering ::before & ::after will immediately level up your game.

So many creative and scalability challenges are overcome with ::before and ::after pseudo elements, especially in combination with absolute positioning, z-index, pseudo-classes, and other techniques. Work to master the use of ::before and ::after pseudo elements!

Pseudo elements can be assigned to classes, IDs, or HTML elements, making them very flexible, scalable, and maintainable where adding additional HTML is probably not.

Doesn't work on images - they need a div wrapper or <figure> tag.

.my-custom-overlay

x 1

.my-custom-overlay

CONTENT

STYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

SHAPE DIVIDERS

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

width 0s ease 0s

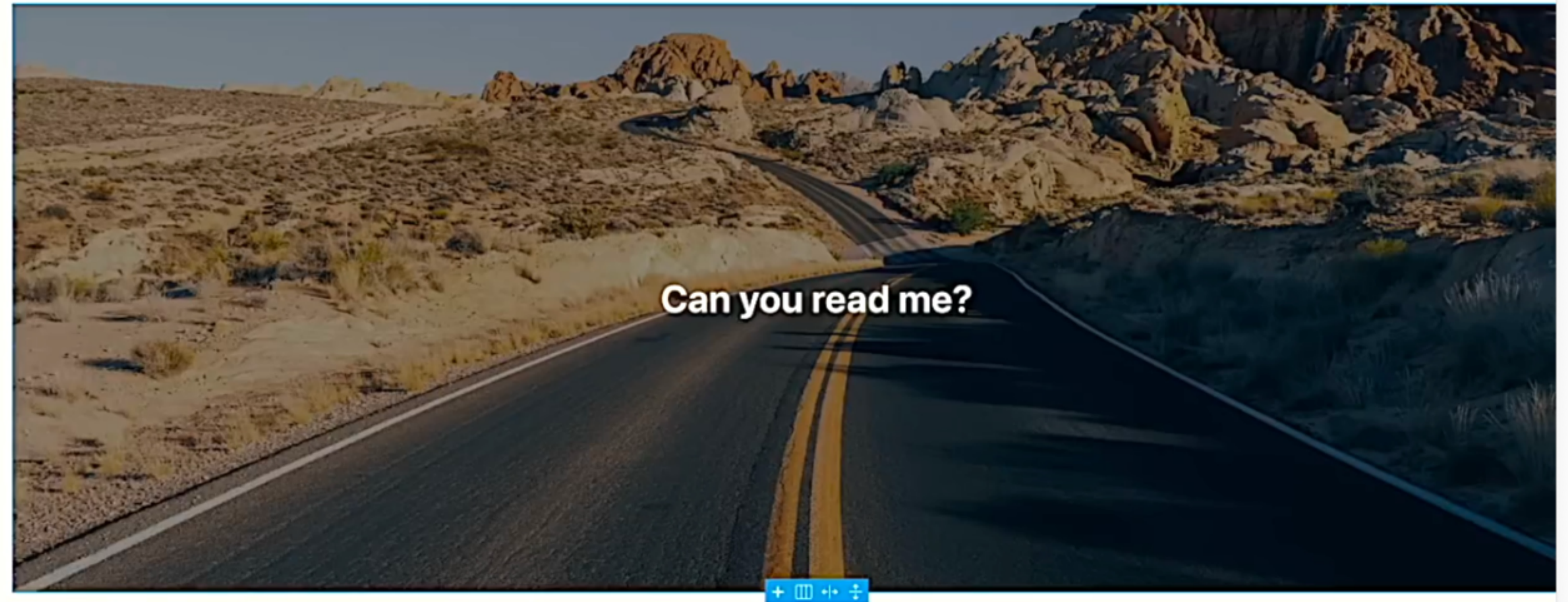
[Learn more about CSS transitions](#)

• Custom CSS

```
1 root {
2   position: relative;
3 }
4
5 root::before {
6   content: '';
7   position: absolute;
8   inset: 0;
9   width: 100%;
10  height: 100%;
11  background-color: red;
12  opacity: .5;
13  z-index: 1;
14 }
```


Background image overlay.

We need the background image below to have an overlay. Can we use a pseudo element?



This resets the stacking **for this element** **your game.** **:before & ::after will immediately level up**

So many creative and scalability challenges are overcome with `::before` and `::after` pseudo elements, especially in combination with absolute positioning, z-index, pseudo-classes, and other techniques. Work to master the use of `::before` and `::after` pseudo elements!

Pseudo elements can be assigned to classes, IDs, or HTML elements, making them very flexible, scalable, and maintainable where adding additional HTML is probably not.

Doesn't work on images - they need a div wrapper or `<figure>` tag.

```
1 root {
2   position: relative;
3   isolation: isolate;
4 }
5
6 root::before {
7   content: '';
8   position: absolute;
9   inset: 0;
10  width: 100%;
11  height: 100%;
12  background-color: black;
13  opacity: .5;
14  z-index: -1;
15 }
```


Great job, we're almost done!





::marker

The icons in front of list items are called markers.
They can be styled with the ::marker pseudo element.

- List item one
- List item two
- List item three

::first-letter & ::first-line

We can style the first letter of this paragraph.

We can style the entire first line of this paragraph, even if the break changes.

Section

```
1 root li::marker {  
2   color: red;  
3 }
```


::marker

The icons in front of list items are called markers.
They can be styled with the ::marker pseudo element.

- List item one
- List item two
- List item three

Section

::first-letter & ::first-line

We can style the first letter of this paragraph.

We can style the entire first line of this paragraph, even if the break changes.

Now the list item and the
marker have different colors

```
1 root li {  
2   color: black;  
3 }  
4  
5 root li::marker {  
6   color: red;  
7 }
```


::marker

The icons in front of list items are called markers.
They can be styled with the ::marker pseudo element.

- ♥ List item one
- ♥ List item two
- ♥ List item three

::first-letter & ::first-line

We can style the first letter of this paragraph.

We can style the entire first line of this paragraph, even if the break changes.

.marker--custom x 1

.marker--custom

CONTENT

STYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

all 0s ease 0s

[Learn more about CSS transitions](#)

• Custom CSS

```
1 root li {  
2   color: black;  
3   list-style: none;  
4 }  
5  
6 root li::marker {  
7   content: '♥';  
8   color: red;  
9 }
```


::marker

The icons in front of list items are called markers.
They can be styled with the ::marker pseudo element.

- ♥ List item one
- ♥ List item two
- ♥ List item three

::first-letter & ::first-line

We can style the first letter of this paragraph.

We can style the entire first line of this paragraph, even if the break changes.

::marker

The icons in front of list items are called markers.
They can be styled with the ::marker pseudo element.

- ♥ List item one
- ♥ List item two
- ♥ List item three

::first-letter & ::first-line

We can style the first letter of this paragraph.

We can style the entire first line of this paragraph, even if the break changes.

.marker--custom x 1

.marker--custom

CONTENT

STYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

all 0s ease 0s

[Learn more about CSS transitions](#)

• Custom CSS

```
1 root li {  
2   color: black;  
3   list-style: none;  
4   padding-left: .5em;  
5 }  
6  
7 root li::marker {  
8   content: '♥';  
9   color: red;  
10 }
```


::marker

The icons in front of list items are called markers.
They can be styled with the ::marker pseudo element.

- ♥ List item one
- ♥ List item two
- ♥ List item three

::first-letter & ::first-line

We can style the first letter of this paragraph.

We can style the entire first line of this paragraph, even if the break changes.

```
1 root::first-letter {  
2   color: red;  
3   font-weight: 700;  
4   float: left;  
5   font-size: 150%;  
6 }
```


`.highlight-first`

1

x 1

`.highlight-first`

CONTENT

STYLE

LAYOUT

TYPOGRAPHY

BACKGROUND

BORDER / BOX SHADOW

GRADIENT / OVERLAY

TRANSFORM

• CSS

CSS Filters

[Learn more about CSS filters](#)

Transition

`all 0s ease 0s`[Learn more about CSS transitions](#)

• Custom CSS

```
1 root::first-line {  
2   font-weight: 700;  
3   color: var(--action);  
4 }
```

2

::marker

The icons in front of list items are called markers.
They can be styled with the ::marker pseudo element.

♥ List item one

♥ List item two

♥ List item three

::first-letter & ::first-line

We can style the first letter of this paragraph.

We can style the entire first line of this paragraph, even if the break changes.

Section