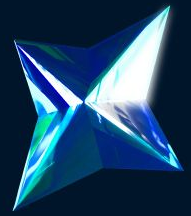
BRO Academy



Decorative Styling

Lesson 9

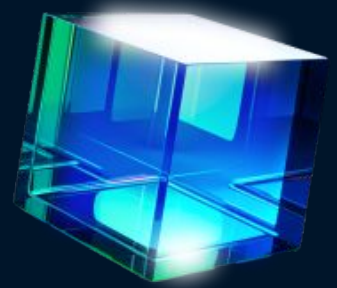
Lesson Plan

1. Learn pseudo-classes and pseudo-elements.
2. Practice using them for flexible styling.

What Are Pseudo-Elements?

allow styling parts of an element, such as the first letter or line

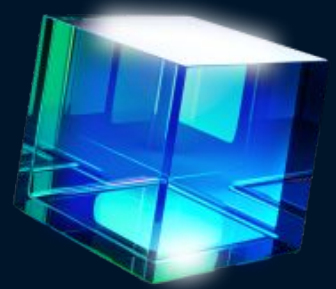
- `::before, ::after` – add content before or after an element.
- `::first-letter, ::first-line` – style specific parts of text.
- `::marker` – style list markers.



Pseudo-classes

Help style specific attributes or states that are not reflected in the DOM.

- **user-action** pseudo-classes
- **lang** pseudo-class
- **negation** pseudo-class
- **structural** pseudo-classes
- **user interface** pseudo-class selectors



Link Pseudo-classes for <a>

`:link` – styles unvisited links

`:visited` – styles visited links

```
a:link { color: blue; }

a:visited { color: purple; }
```

Practice: Link State Pseudo-Classes

Task:
1. Create a list of links.

2. Apply styles:

- Blue color for new links (:link).
- Purple color for visited links (:visited).

User-action pseudo-classes

`:active` – element is being clicked

`:focus` – element is in focus

`:hover` – mouse is hovering over the element

```
a, button, input
```

Practice User-action pseudo-classes

Task:
1. Create a button and a link.
2. Apply styles:
:hover – change color on hover.
:active – change color on click.
:focus – frame around the element.

The negation pseudo-class

Styles are applied to all elements except those matching the selector

`:not(p) { }` – all elements except paragraphs **tags**

`:not(.intro) { }` – all elements except those **with class** .intro

`:not(#news) { }` – all elements except those **with id** #news

`:not(:lang(fr)) { }` – all elements **except** those with the French **language**

`:not([disabled]) { }` – all elements except those **without** the disabled **attribute**

`p:not(.intro) { }` – all **paragraphs** **except** those with the class `.intro`

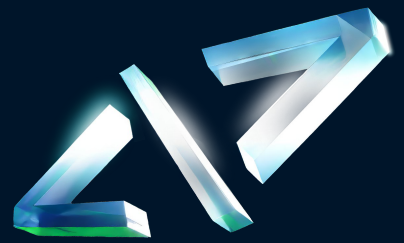
Structural pseudo-classes

Allow you to select elements based on their position in the document structure.

! If the document structure changes, the structural pseudo-class might apply to a different element or potentially to no element at all.

It can sometimes be difficult to determine exactly which element the styles will be applied to.

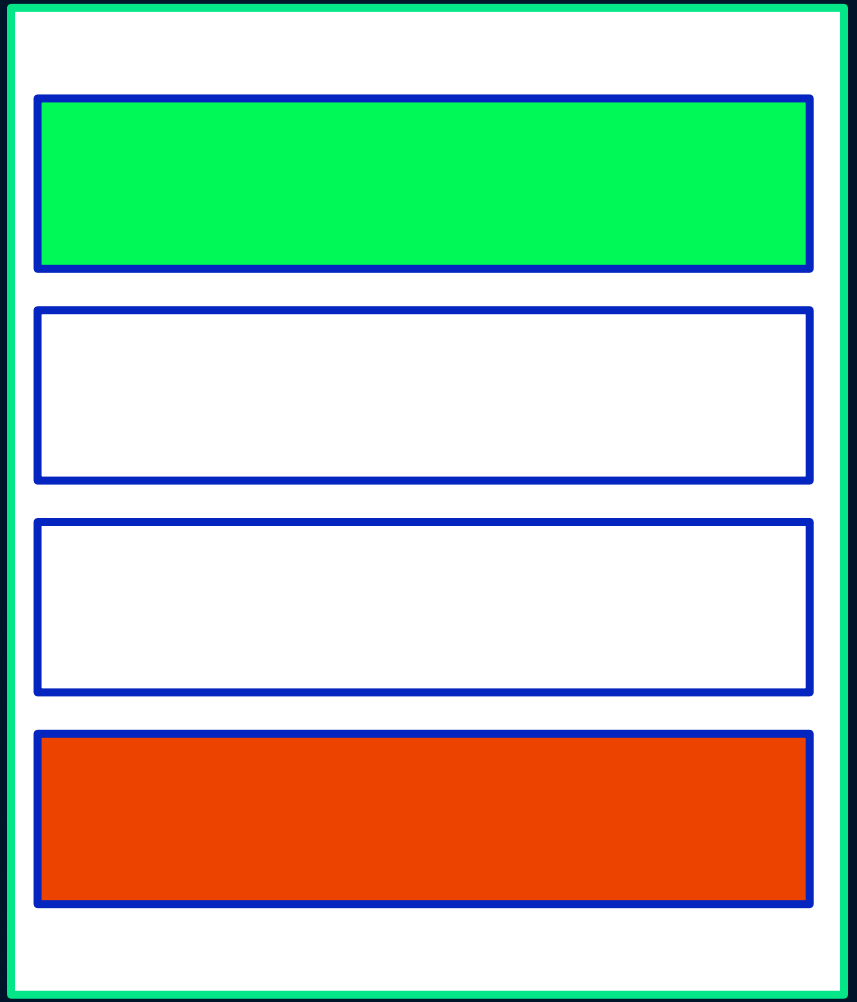
```
:first-child { }
:only-child { }
:nth-child(3n) { }
```

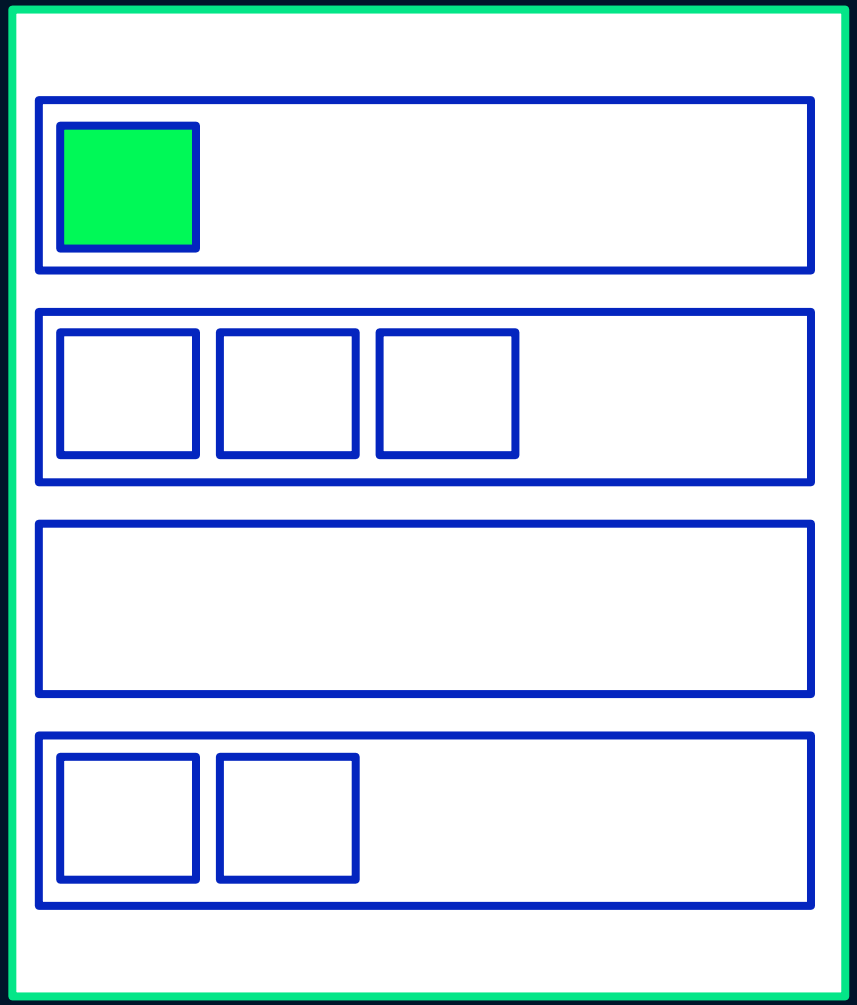


:first-child
:last-child

– Selects the element that is the **first/last child** of another element.

```
article:first-child { }
article:last-child { }

<section>
   <article> 1 </article>
   <article> 2 </article>
   <article> 3 </article>
   <article> 4 </article>
</section>
```



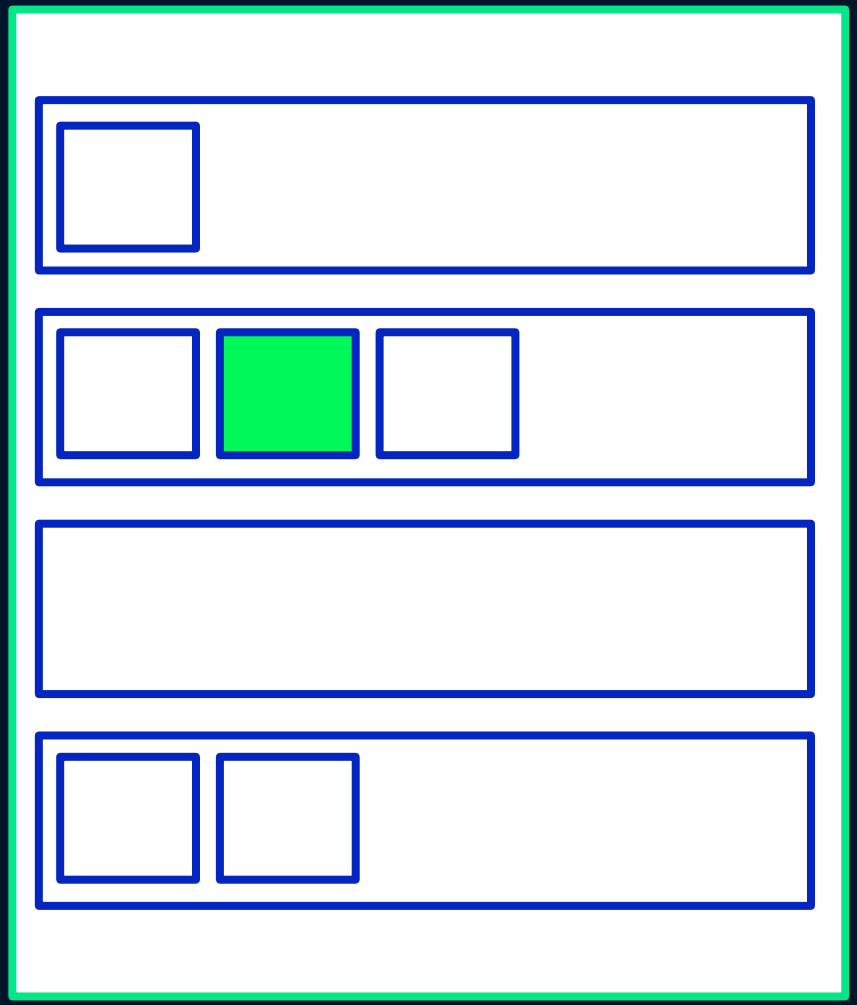


pseudo-class

:only-child

– Selects an element that is the only child of another element.

```
div:only-child { }

<article>
   <div> 1 </div>
</article>
```



pseudo-class

:only-of-type

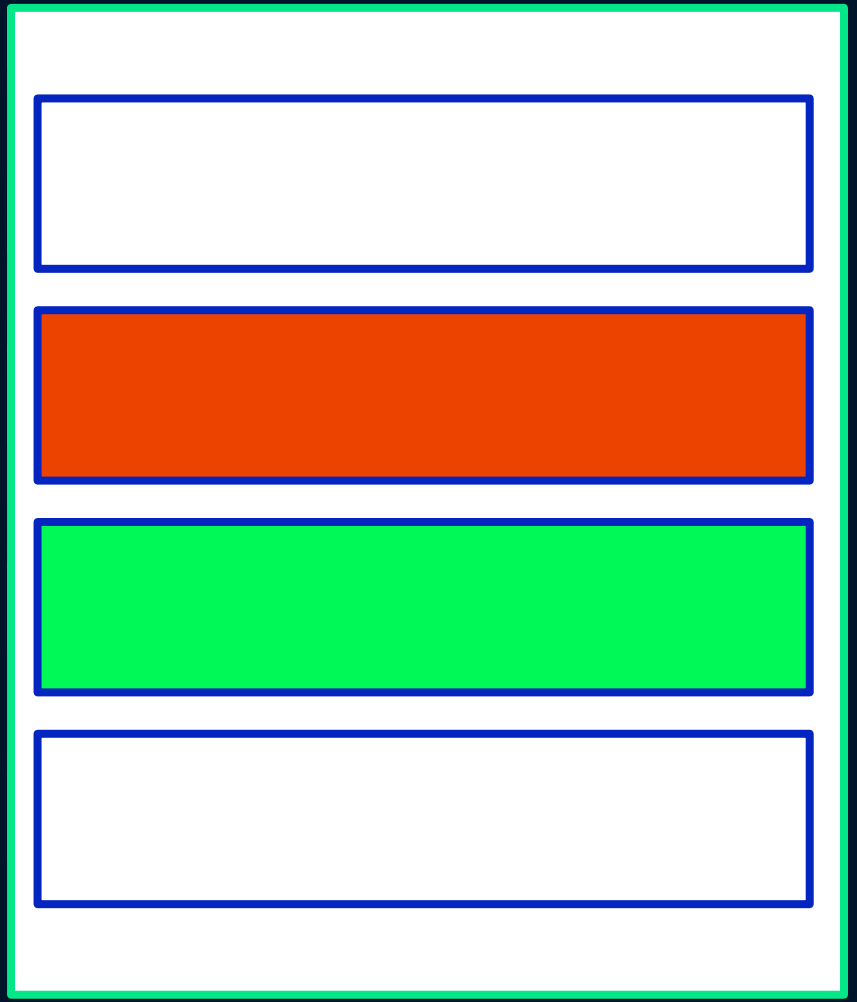
– Selects an element that is the **only** element of its **type** within its parent.

```
p:only-of-type { }

<article>
   <div> 1 </div>
   <p> 1 </p>
   <div> 1 </div>
</article>
```

:first-of-type
:last-of-type

– Selects an element that is the **first/last** of its **type** within its parent element.

```
p:first-of-type { }
p:last-of-type { }

<section>
   <article> 1 </article>
   <p> 2 </p>
   <p> 3 </p>
   <article> 4 </article>
</section>
```

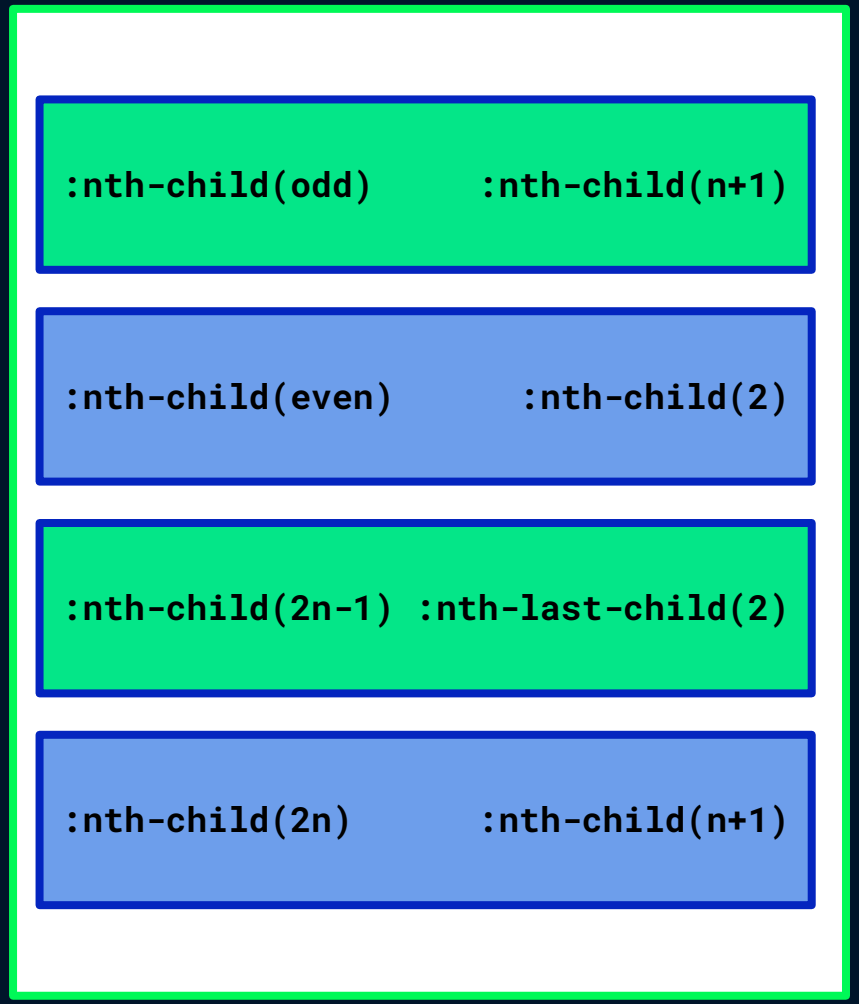


:nth-child(n)
:nth-last-child(n)

– Selects **specific child elements** in a parent element starting from the beginning or the end.

n:

- **number**
- **number + n** (selects every **n-th** element)
- expression with **+/-** (allows starting from an element other than the first)
- **even** (all even elements)
- **odd** (all odd elements)

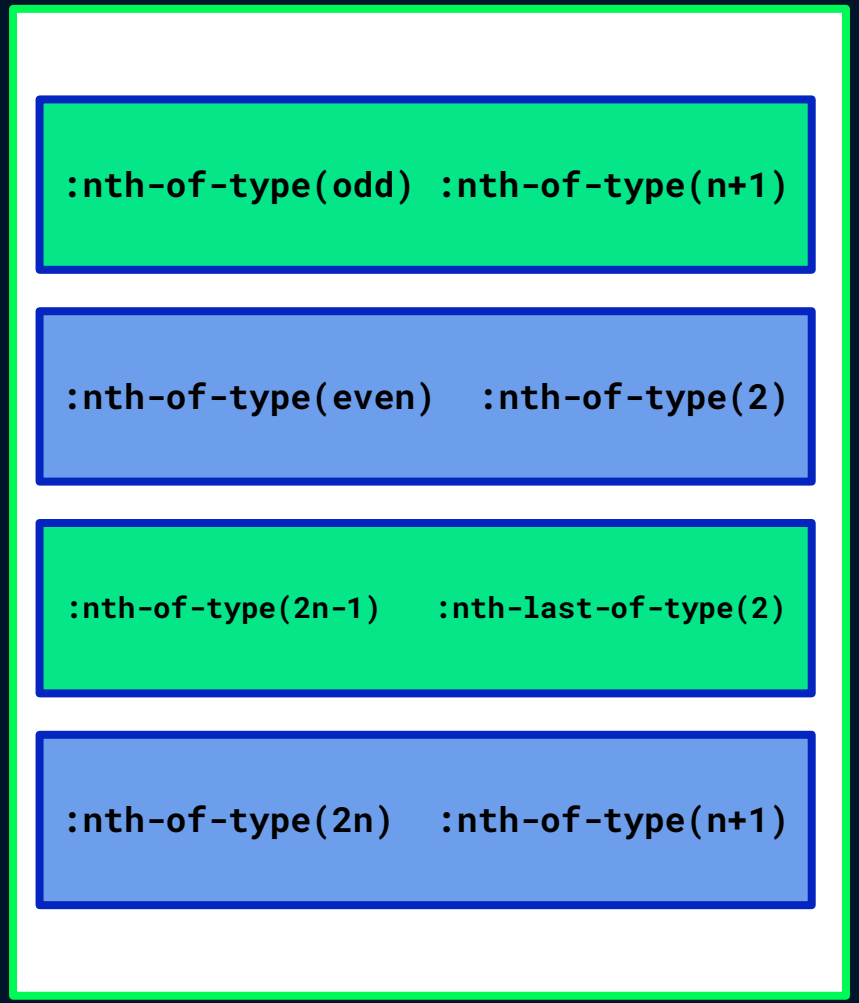


:nth-of-type(n)
:nth-last-of-type(n)

– Selects elements of a **specific type in the parent** element starting from the beginning or the end.

n:

- **number**
- **number + n** (selects every **n-th** element)
- expression with **+/-** (allows starting from an element other than the first)
- **even** (all even elements)
- **odd** (all odd elements)



:root

– Selects the root element of the document (tag **<html>**).

:empty

– Selects an element that has no content or child elements (an empty element).

A space is already a character, so the tag is no longer considered empty.
It also applies to input elements where no value has been entered.

```
:root { }

<html>
   <head> 1 </head>
   <body> 1 </body>
</html>

p:empty { }

<article>
   <p> 1 </p>
   <p> </p>
   <p></p>
   <p><span></span></p>
</article>
```

Pseudo-elements

– **(fake elements)** Allow styling elements that are not in the document tree.

`::-webkit-scrollbar` – styles the scrollbar

+ Other pseudo-elements of the form `::-webkit-scrollbar-*,` are used only with prefixes and only in `webkit` browsers

```
.invisible-scrollbar::-webkit-scrollbar {
  display: none;
}
```

Pseudo-elements for text

`::first-line` – styles the first line of text

`::first-letter` – styles the first letter of text

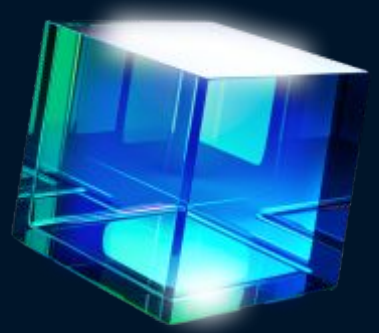
```
p::first-line { }
p:first-letter { }
<p>
   This is the first line
   of a paragraph of text
</p>
```

Pseudo-elements for Lists

- Usage of counters in lists
- Styling list markers

`::marker` – Styling list markers.

```
ol {
  counter-reset: section;
}
li::before {
  counter-increment: section;
  content: counter(section);
}
li::marker { }
```



Summary

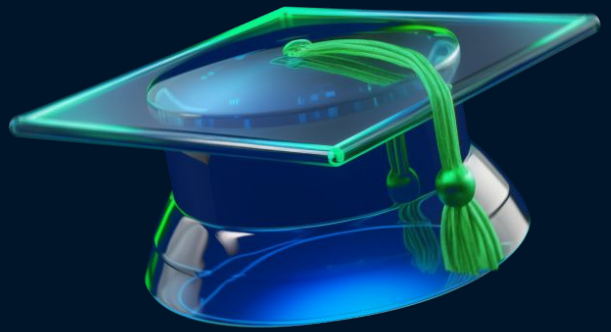
What we covered:

Link state pseudo-classes (:link, :visited).

Structural pseudo-classes (:nth-child, :only-child).

Special pseudo-classes (:root, :empty).

Pseudo-elements (::before, ::after, ::first-line, ::marker).

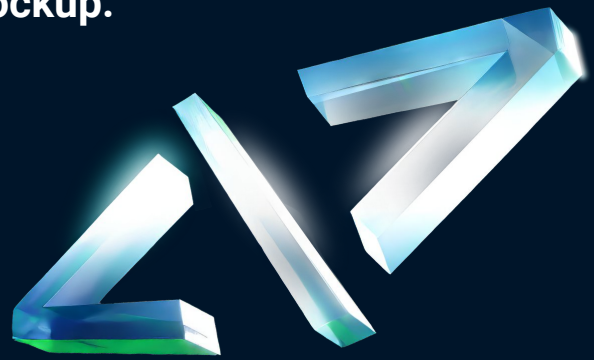


Homework

1. Complete one of the following courses to reinforce your understanding of the theory
2. Achieve the highest level of accuracy with the design mockup:
   - Apply all states for links and buttons:

     hover, active, focus, according to the UI kit
   - Apply visual effects such as

     shadows, shapes, filters and etc
   - Use pseudo-elements where necessary
   - Set all internal and external margins and padding



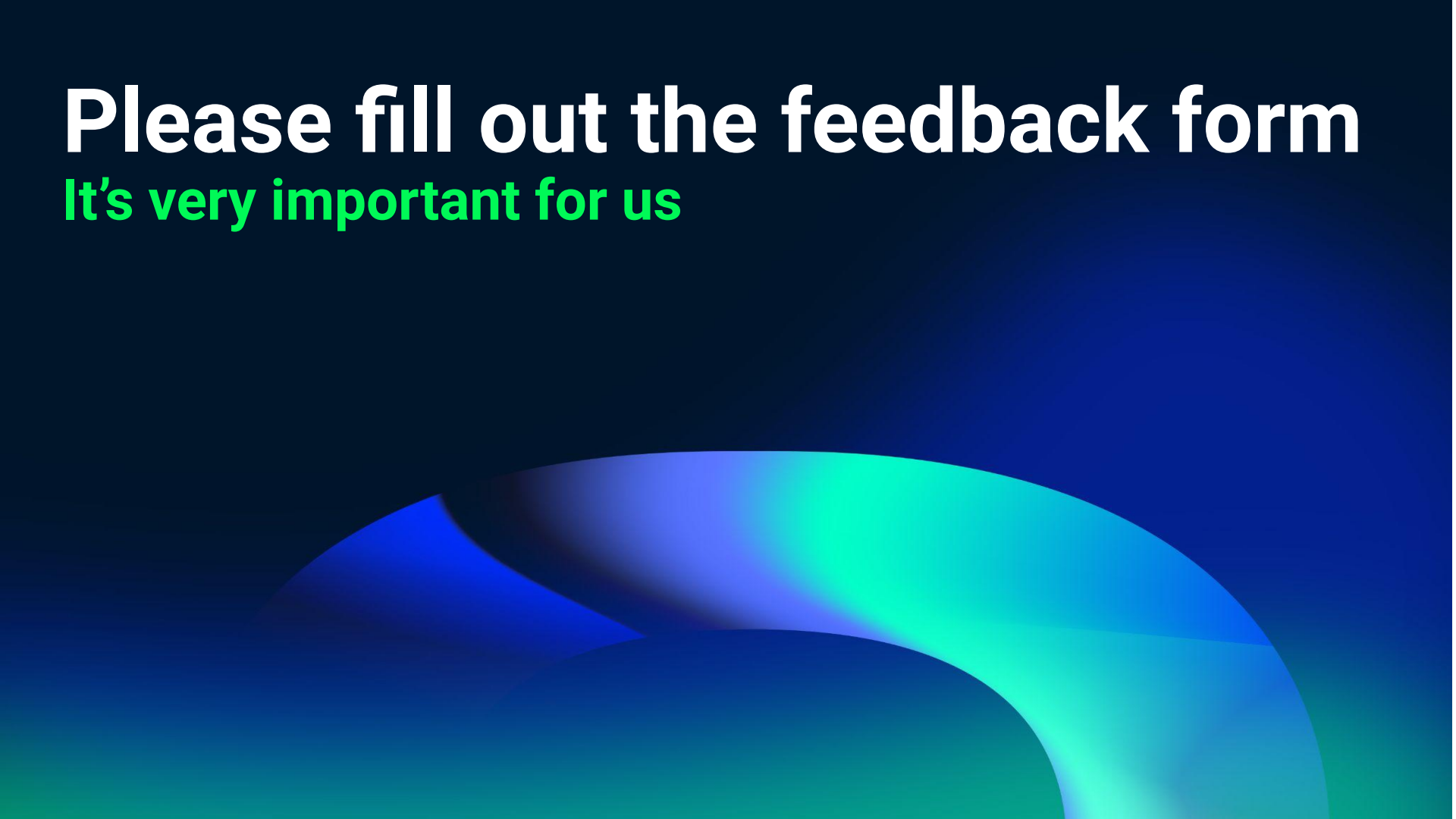
Your website should look exactly like the design mockup.
This is the final stage of work on the website. Next, we will only be adding animations.

BRO Academy

QUESTIONS?

Please fill out the feedback form

It’s very important for us




THANK YOU!
Have a good evening!