Navigation

# HTML

The navigation section is the first section we will work on. Let’s start with the HTML structure. The most appropriate element to use here is **header**. We will give it the class **header**. It adequately describes the header of our web page. Inside, there is a logo which will be a link **a** and will receive the class **logo**. Next to the logo, on the opposite side, there are navigation links. We will create a standard structure based on the **nav** element, followed by an unordered list **ul** (class **header\_nav**). Each **li** will receive the class **nav\_\_element** and link **nav\_\_link**.

We create classes for those elements so that it is easier to get to and style them later in CSS. The class naming convention that we use here is BEM.

However, we know that the logo and navigation have be limited in width to the container. So, between **header** and them, we add a **div** with classes: **container** and **header\_\_container**. The class **container** will set the appropriate size and **header\_\_container** will allow us to style it all in some particular way, e.g. give **display: flex**.

What does our HTML structure look like?

<header class="header">

<div class="header\_\_container container">

<a href="#" class="logo">BestShop</a>

<nav>

<ul class="header\_\_nav">

<li class="nav\_\_element">

<a href="#why-us" class="nav\_\_link">Why us</a>

</li>

<li class="nav\_\_element">

<a href="#benefits" class="nav\_\_link">Benefits</a>

</li>

<li class="nav\_\_element">

<a href="#prices" class="nav\_\_link">Prices</a>

</li>

<li class="nav\_\_element">

<a href="#contact" class="nav\_\_link">Contact</a>

</li>

</ul>

</nav>

</div>

</header>

# CSS

Let’s move on to styling the whole thing.

## Header

Let’s start the class **header.** The design shows that it is supposed to have **70px** height and shadow shift **y: 3px** , blue **6px** and color **rgba(0,0,0,.16)**. We set this shadow right here, because it is supposed to be on 100% of the page width.

.header {

height: 70px;

box-shadow: 0 3px 6px rgba(0, 0, 0, 0.16);

}

## Container

What will **header\_\_container** do? The logo and navigation are to be on opposite sides, both of them centered vertically. The most convenient way to do this is to use **flex** and the corresponding two properties.

.header\_\_container {

display: flex;

justify-content: space-between;

align-items: center;

height: 100%;

}

Why set the **height: 100%**? To make our container the same height as its parent (which is **header**). This allows for vertical centering: **align-items: center**.

# Logo

In this project, the logo is plain text. Its height is **32px** which makes **2rem** (**16px \* 2**). Used the main color and Raleway font with weight 300.

.logo {

font-family: var(--font-raleway);

font-size: 2rem;

font-weight: 300;

color: var(--color-primary);

}

Our variables are already useful!

# Navigation

First of all, the navigation is arranged side by side. Therefore, set **display:flex** for the **ul** element (**header\_\_nav**).

.header\_\_nav {

display: flex;

}

To make the clickable area of links in the navigation as large as possible, we will give the **nav\_\_link** elements appropriate **padding**. How do you get the effect of highlighting a link after hovering over it? The easiest way is to prepare an appropriate **border-bottom** in **transparent** color. When **:hover** is recognized, it will change its color.

.nav\_\_link {

color: var(--color-grey);

font-weight: 700;

font-size: 0.6875rem;

text-transform: uppercase;

padding: 28px 18px 22px;

border-bottom: 4px solid transparent;

letter-spacing: 0.1875rem;

}

.nav\_\_link:hover {

color: var(--color-black);

border-color: var(--color-primary);

}

In order not to make the line emphasizing the active link too long, the rest of **padding** will be given to **nav\_\_element**.

.nav\_\_element {

padding: 0 26px;

}

But now out last element of the navigation, it too far away from the container. Let’s reset its right **padding**.

.nav\_\_element:last-child {

padding-right: 0;

}