Hello, my name is Zsolt Nagy, and this is **CSS architecture** course. We will demystify the topic of how to write solid CSS architectures on all levels. Firstly, let's see why CSS architecture is important.

As you know, basic CSS is one of the easiest languages you can learn in software development. This is not only an advantage, but also a disadvantage because it is very easy to create bad CSS. You have most likely seen a lot of cases when you saw unmaintainable CSS containing a lot of overrides, containing a lot of erroneous declarations that are completely different in two separate browsers.

Given that it is very easy to make maintainability aerated mistakes in CSS, we have to establish some guidelines that make sure that we keep writing CSS at the highest standard. But wait a minute. What makes our CSS maintainable? Is it the tools we use, the preprocessors, the frameworks?

Yes, they all contribute to our comfort, but using the right tools, using the right preprocessors and frameworks are not enough. Even if a lot of people use these tools, preprocessors, and frameworks, only a small fraction of software developers are capable of making good CSS: well structured, maintainable, and scalable.

By the end of this course, you should have a good idea of how to design, create, and contribute to stylesheets that can be found in large applications or component libraries. Our goal is to learn how to create structured, scalable, and maintainable CSS. First, you will learn the goals of maintainable CSS architecture.

You will find out how to apply software engineering principles to write maintainable CSS. We will put a lot of effort into understanding how to write specific, efficient, and descriptive CSS selectors. Once we know how to write maintainable code on selector level and how to apply the principles, it's time to establish a solid CSS hierarchy.

Let me briefly introduce myself. My name is Zsolt Nagy. I'm a technical development lead in Berlin, Germany. I have a front end and full stack development background, which means that I have a perspective on applications from multiple angles. For instance, my last course on SitePoint had a front end development topic, namely the introduction of TypeScript.

In order to benefit the most from this course, I highly recommend that you learn the principles of writing basic HTML and CSS. The knowledge of how to preprocess CSS using, for instance, Sass or Less will be required to understand the practical examples belonging to this course. We will go through six lessons:

- During the lesson 1 we will define the goals of CSS architecture
- Lesson 2 is about writing maintainable CSS code, mainly on selector and rule level
- Lesson 3 introduces some principles of maintainable CSS architecture
- In lesson 4 you will learn how to establish a style hierarchy from the ground up
- In lesson 5 we will put theory into practice
- Lesson 6 is the end of the course where we will draw some conclusions.