

So now that we know why we need a JavaScript framework in the first place, let's start by learning about the most popular framework out there. React. So in this lecture we will get a high level overview of what React actually is and how it works. So it's going to be a full packed but super interesting lecture. So let's quickly get started. So, according to the official React documentation, React is a JavaScript library for building user interfaces. Now that's something, but it really isn't that helpful, is it? So let's extend this definition by a bit so that it becomes more helpful or should I say a little bit more crazy like this. So now in this slightly modified definition, React is an extremely popular, declarative, component based, state driven JavaScript library for building user interfaces created by Facebook. And this is actually a lot more helpful because now we can deconstruct this definition and learn about what React actually is step by step. So starting with component based design, React is all about components such as buttons, input fields, search bars and so on. So components are the building blocks of user interfaces in React. And in fact, basically all React does is to take components and draw them on a web page. That's it. And so we build complex UIs in React by building multiple components and then combining these components like Lego pieces, for example, to create a complex application like Airbnb. So here we have exactly the same components that we saw earlier, but all combined into a complex UI. So the nav bar, the search bar, the results panel and the map. Now another thing that we do with components is to reuse them. For example, here in the results panel, there are multiple listings that look quite similar. And so we can create a listing component and then reuse it three times here, simply adjusting the data a little bit. Now since we built complex UIs by combining multiple components, each component must have all the information about what it looks like, right? So in order to describe what each component looks like and how it works, we use a special declarative syntax called JSX. And declarative simply means that we tell React what a component and ultimately the entire UI should look like based on the current state. So React is basically a huge abstraction away from the Dom so that we never have to work with the Dom directly as we would with vanilla JavaScript. So we simply tell react what we want to happen when some data changes, but not how to do it. And again, we do that using JSX. So JSX is basically a syntax that combines parts of HTML, CSS, JavaScript, and it even allows us to reference other React components. And the declarative nature of React is an essential concept that of course we will really dive into later. But now you might wonder if we never touch the Dom, then how does React update the UI? Well, that's where the concept of state comes into play. So remember that the main goal of React is to always keep the UI in sync with data. And from now on, let's actually call that data state. For example, an array of apartments on Airbnb. So based on our initial state, React will render a user interface using the components that we wrote using JSX. Then, based on some event like a button click, the state might change. For example, more data about apartments might be loaded from an API. And so here comes the interesting part. Whenever the state changes, we manually update the state in our app and React will then automatically rerender the user interface to reflect that latest state. Or in other words, react reacts to state changes by re rendering the UI. And that is in fact the whole reason why react is called react in the first place. Now, there's always been some debate over whether React is actually a framework or just a library. Well, the short answer is that React is actually just a library, even though I keep calling it a framework. And the reason for that is that React itself is really only the so called view layer. So if we want to build a complete real world application, we need to choose multiple external libraries to add to our project, for example, for routing or for data fetching. Now to address this, there are actually multiple frameworks that have been built on top of React. So frameworks that include all these functionalities that react is missing out of the box. And the most popular ones are called Next.js and remix. But again, more about all that later in the course. Now moving on. Besides all the great features that we've been talking about, maybe the biggest reason to choose react over all the other frameworks that are similar is because React is extremely popular. So according to the weekly download numbers from NPM, React is by far the most used framework and the difference between React and the other ones only seems to be growing. So react really isn't going anywhere. A big reason for that is that many large companies have adopted React a long time ago and so now more and more smaller companies all over the world are following in their footsteps. This means that all these companies now need react developers for their teams, which of course has created a huge worldwide job market with a high demand for qualified React developers. And these are not just any jobs. They are usually well paying jobs. Now, now even another positive outcome of React huge popularity is that now there is a very large and active React developer community, which means that there are always going to be lots of tutorials, StackOverflow questions and answers and also lots of support given to other React developers. And if you ask me, this is actually one of the strongest benefits of using React right now. Also because this huge community has grown a truly gigantic third party library ecosystem around React. And so now you can integrate countless useful libraries and tools into your own React applications. And now our last topic is that React was created by Facebook. Well, more specifically, React was created in by Jordan Walke, who is an engineer who was working at Facebook at the time. So that's why we say that React is backed by Facebook or now by Meta. So there React was first used on the newsfeed and also the chat app, and from there it spread out to the entire Facebook and also the Instagram applications. Then in , React was open sourced for everyone to use it, and the rest is history. Because React has truly and completely transformed front end web development not only by developers that are using React itself, but because many other modern frameworks were created as a response which really changed the industry forever. Okay. So to summarize, React is really good at two things. First, rendering components on a web page as a user interface based on their current state. And second, React is really good at keeping the user interface in sync with state by re rendering or in other words, by reacting when the state of one of the component changes and as you will see later react does all this by employing something called a virtual Dom, a fiber tree, one way data flow and many other techniques that we will study throughout this course. All right. And so now that you have a rough overview of what React actually is, let's set up our development environment so that we can start writing some serious code finally.