Frameworks

As you may have noticed in your own projects, lots of applications rely on common functionality that we'd like to avoid re-writing for every new project. This functionality could include things like re-rendering components when underlying data changes, or fancier features like server-side rendering. These days, almost all web applications you see and use are built using a web framework – some of the most common frameworks include React, Angular, and Vue.

> ℹ️ **What's the difference between a programming language and a framework?**
>
> *A programming language tells the computer what it should do. Every programming language features a syntax and a particular set of rules, which need to be followed every time the code is written. A software framework is built on top of a programming language.*

Using a web framework enables you to start from a higher level of abstraction than raw HTML and CSS, reducing the amount of boilerplate and repetitive coding that isn't very interesting. It lets you focus on the core functionality of the app you're building, rather than minutia like rerendering components on the page at precisely the right times. Frameworks like React are built and maintained by people who know what they're doing, and trusted by large companies.

Frameworks have several key advantages over rolling your own app from scratch, that make development more productive and efficient:

- Frameworks are written with security in mind and consistently updated with the latest patches to security vulnerabilities
- Frameworks let you avoid repetitive code by sharing components
- Standardized structure lets multiple developers collaborate on the same project, and allows new developers to contribute quickly because they are familiar with the framework's libraries and systems

Frameworks exist for all sorts of different application types: websites, data science, databases, mobile apps, you name it. Some common JavaScript frameworks you'll often hear about are listed below:

Name	Language	Use Case
React	JavaScript / TSX	Web
Angular	JavaScript / TSX	Web
Vue	JavaScript / TSX	Web
Node.js	JavaScript	Backend
React Native	JavaScript / TSX	Mobile

Cordova	JavaScript / HTML	Mobile
Flutter	Dart	Mobile

React

In this book, we'll be focusing on React as our primary UI framework. React was created and is maintained by Facebook, and is used throughout Instagram, Facebook, and WhatsApp. React is one of the fastest growing and most popular web frameworks, and it's used by lots of companies all over the world.

React apps are built with a combination of JavaScript and HTML that can be embedded into JavaScript code. This combination is called JSX, and you'll often see files within React apps with a .jsx extension.

One of the defining features of React is its ability to keep track of **state**, and act on changes to state. Each React component can keep track of some set of data, called its state. This could include information about the app's current situation, like if a user is logged in or not, or what text was entered into a text box. Whenever your state changes, React automatically updates all components that rely on that state, saving you the headache of manually re-rendering components on data change.

The best way to learn React is by working with it.

- This Codecademy course provides an excellent introduction to React through hands-on exercises
- The React Tutorial is the classic way to learn React concepts through building representative applications

Further Reading:
https://hackr.io/blog/what-is-frameworks
https://reactjs.org
https://stackshare.io/react