**React** is all about “components” because all user interface in the end are made up of components. With React, you **define the target UI state(s)** – not the steps to get there! Instead, React will figure out & perform necessary steps. (Declarative programming)

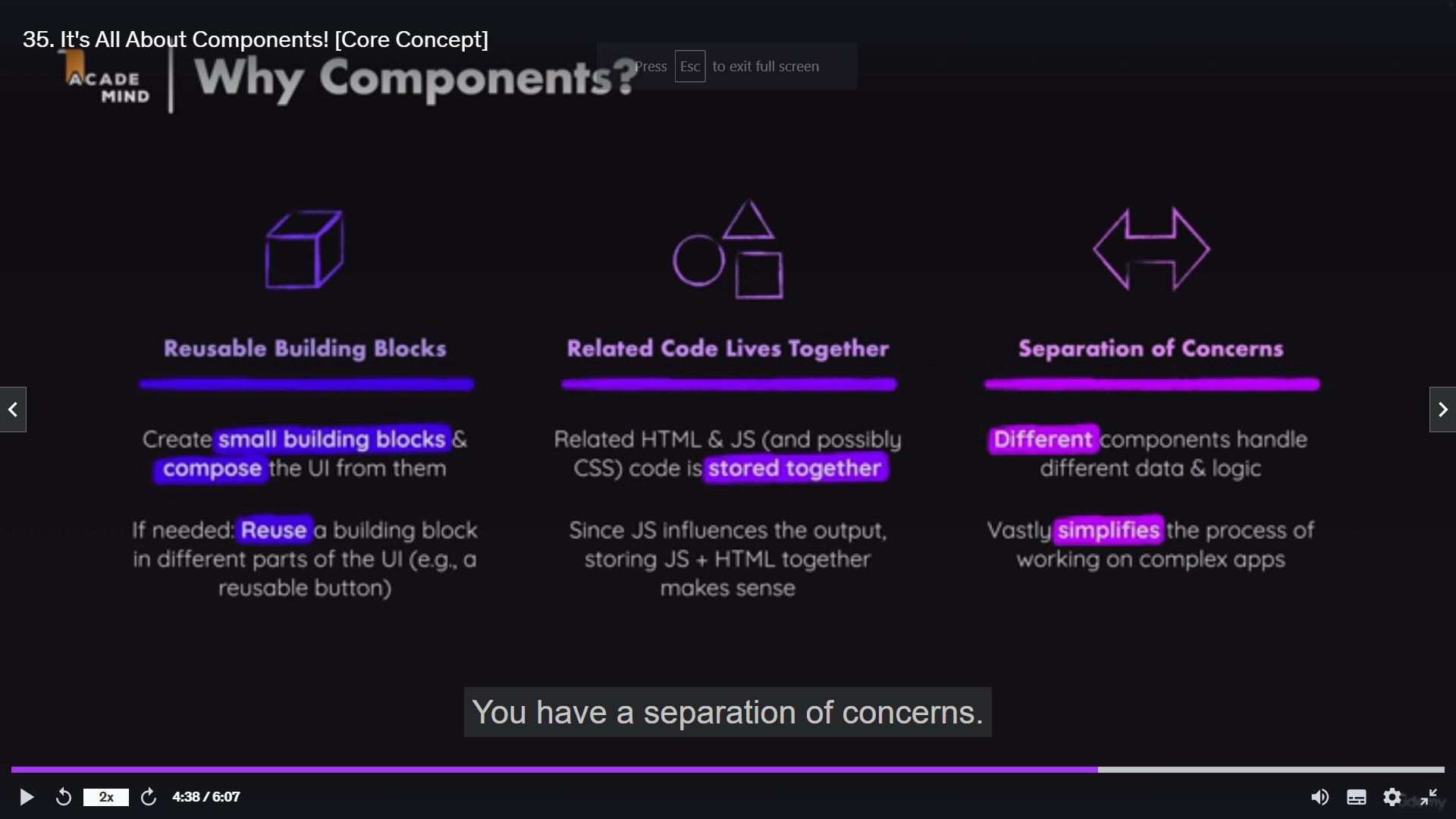
React project uses a build process:

* Raw, unprocessed React code **won’t execute** in the browser because React uses JSX feature.
* In addition, the code would **not be optimized for production** (e.g. not minified)

**Components** are reusable building blocks in your user interface (just a combination of HTML code, CSS styling, JS logic).

Ex:

Reason why we use components:



Graphical user interface, application

Description automatically generatedHow components are built:

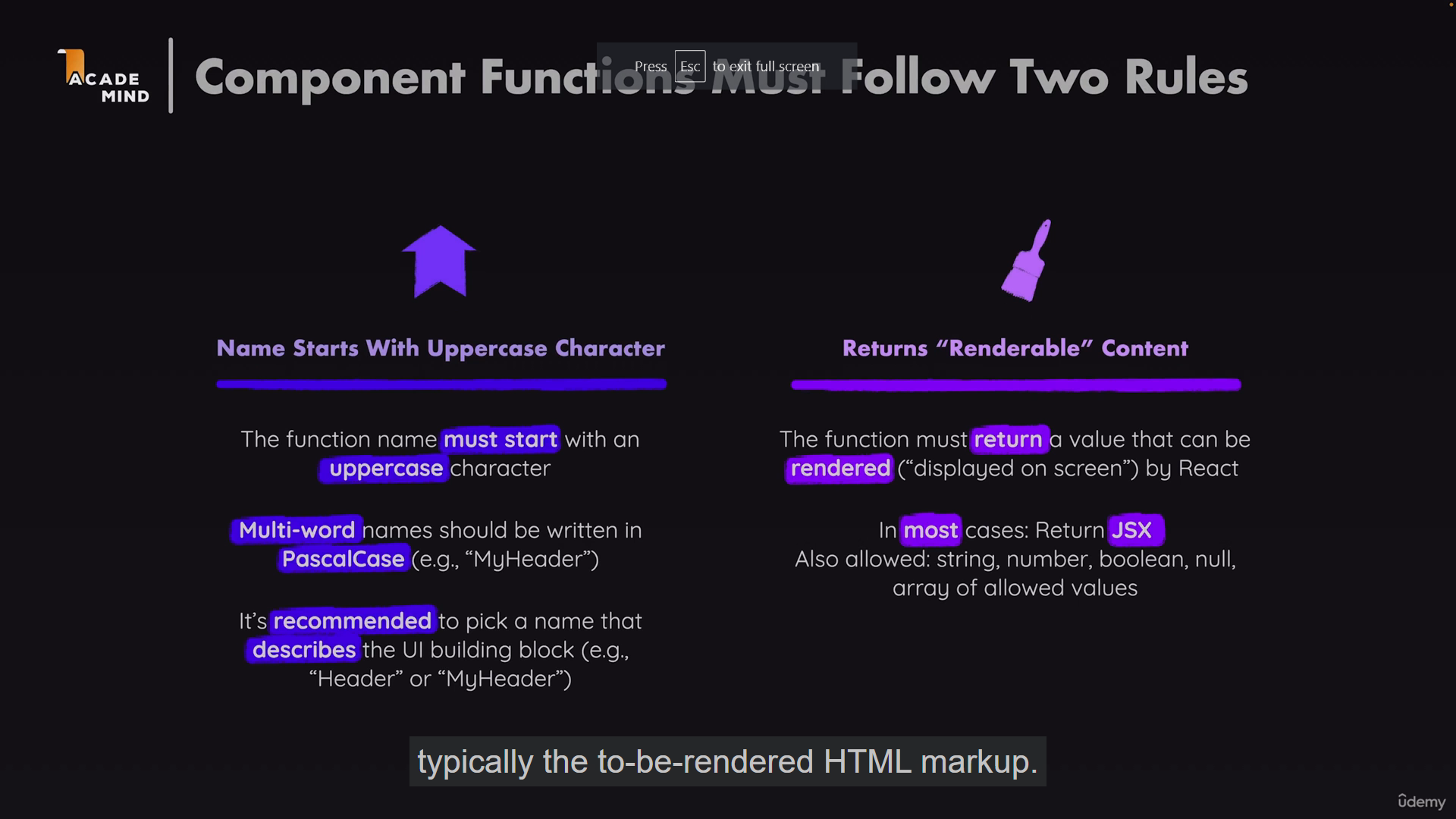
**React code** is written in a “declarative way”. With React, we will not tell that a certain HTML element should be created and inserted into a specific place. We will always define the desired end state, and React’s job to figure out which elements on actual webpage might be added or removed or updated.

Graphical user interface, website

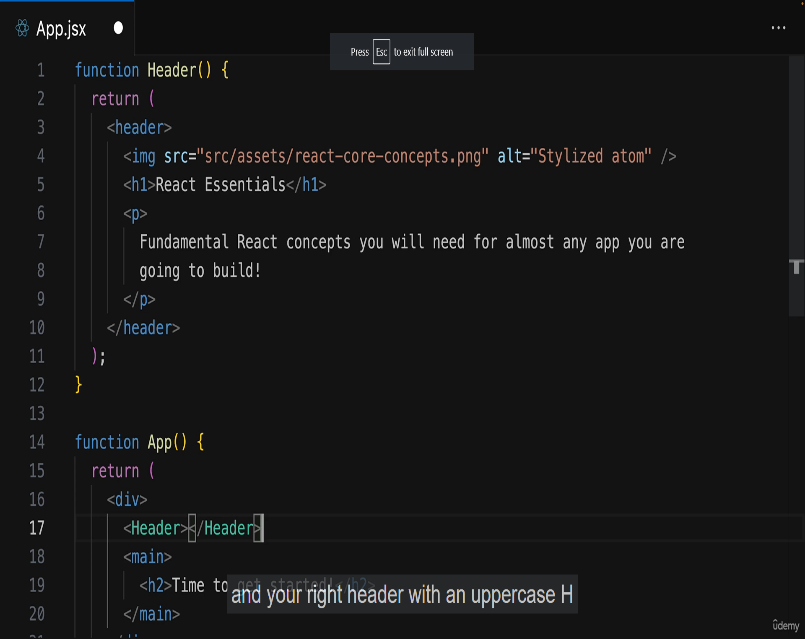
Description automatically generated

**JSX (JavaScript XML)** is basically HTML code inside of JavaScript.

Component function rules:



With React, we basically create component by building up the component tree

Graphical user interface, diagram

Description automatically generated

**Props in React:** let’s say we have custom component which should output a list of items where the goalItem should be output dynamically. We can pass data to custom component by adding a attribute. And inside that component, we can then get access to all these attributes.

Diagram

Description automatically generated