Props & State
Sección 3, Clase 39

`props` and `state` are **CORE concepts** of React. Actually, only changes in `props` and/ or `state` trigger React to re-render your components and potentially update the DOM in the browser (a detailed look at how React checks whether to really touch the real DOM is provided in section 6).

Props

`props` allow you to pass data from a parent (wrapping) component to a child (embedded) component.

Example:

AllPosts Component:

```
1  const posts = () => {
2      return (
3          <div>
4              <Post title="My first Post" />
5          </div>
6      );
7  }
```

Here, `title` is the custom property (`prop`) set up on the custom `Post` component. We basically replicate the default HTML attribute behavior we already know (e.g. `<input type="text">` informs the browser about how to handle that input).

Post Component:

```
1  const post = (props) => {
2      return (
3          <div>
4              <h1>{props.title}</h1>
5          </div>
6      );
7  }
```

The `Post` component receives the `props` argument. You can of course name this argument whatever you want - it's your function definition.