



[Course](#) > [React R...](#) > [Presen...](#) > [Presen...](#)

Audit Access Expires Jun 20, 2020

You lose all access to this course, including your progress, on Jun 20, 2020.

Presentational vs Container Components



Video

[Download video file](#)

Transcripts

[Download SubRip \(.srt\) file](#)

[Download Text \(.txt\) file](#)

Presentational vs Container Components

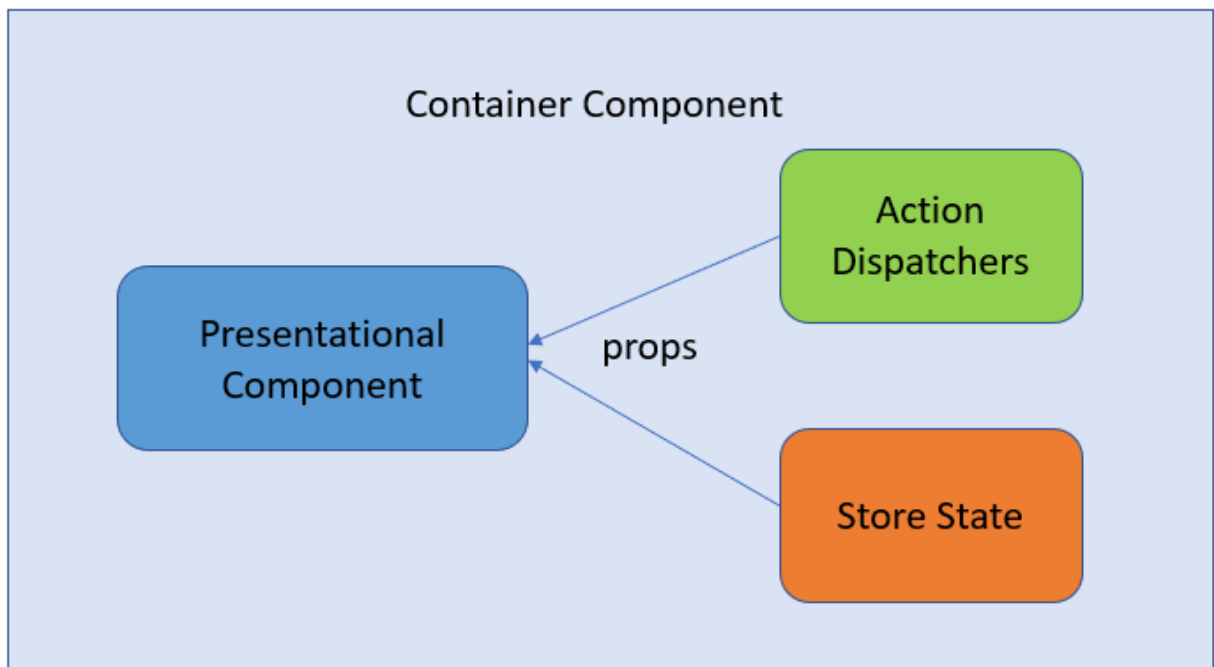
Presentational vs Container Components

React Redux embraces the idea of separating your components in two types: `presentational` components and `container` components.

In a nut shell, presentational components deal with the way things look. Presentational components should not have to know anything about Redux at all. They behave just like normal React components, except they read all their data from props and dont use state. All important data and callbacks are passed down to the presentional components as props.

On the other hand, container components are essentially a wrapper around presentional components that deal with anything Redux related. Container components specify what part of the store state should be passed down to the presentational components and which actions the presentional component should be able to dispatch. React Redux has a few special methods that make it easy to pass down Redux actions dispatchers and store state to React components.

We recommend organizing presentational components and container components into separate folders.



Presentational component example:

```
//returns a list of items given an array
const ItemsList = (props) => {
  return (
    <div>
      {
        props.items.map( (item) => {
          return (
            <div> Name: {item.name} | Price: {item.price}</div>
          )
        })
      }
    </div>
  )
}
```

Container component example:

```
//takes array from store state and passes it as a props to ItemsList

const mapStateToProps = (state) => {
  return {
    items: state
  }
}

const ItemsListContainer = connect(mapStateToProps)(ItemsList)
```