Another option to Separate Concerns

ToDo Sample is very top-down MVC logic

- App.jsx defines all the action functions
- Other components all "dumb"
    - Ignorant of how to update state

This works

- but pushes most of the logic into App.jsx

What is another option?

Components by Responsibility

Each Component aware of their "Responsibility"

- Coupled to the state
    - But that's not bad, we HAVE the state
- App doesn't define the actions
- Each Component manages their responsibility

Manages their Responsibility?

- Login handles "logging in"
  - UI
  - Temporary state
  - Service calls
  - Updating state
- TodoList handles displaying Todos
  - Including service calls for state changes
  - Updating state as a result
- Etc.

Which is better? It depends!

You are deciding *where* complexity is, not *if*

- Separation by Responsibility
    - More common in medium-large applications
    - Component is coupled with changing state
    - Logic is decoupled from outer application
- Top-down Logic
    - Helpful in small applications
    - Components are Presentation only
    - Logic in one place

"Logic in one place" can be "good" or "bad"