# Todo List

- Common examples in webdev
- Lets do it in React
- Write as a component passed the data
    - State changes are OUTSIDE component
- How will this change with services?
    - Foreshadowing!

# The Todo Data Model Plan

- list of items
    - have unique id
        - store as object, not array
- each item has:
    - task (the text)
    - done (boolean)
    - id (unique identifier)

# The Todo UI Plan

- Each item
    - shows name
    - shows done status (how?)
    - allows a toggle interaction (done)
    - allows a delete option
- Can add new items
    - input for task
        - assume not done
        - id is auto-assigned

# Todo in React

- pass in functions to manage state changes
- a component that is is charge of display
    - Can use child components
    - Needs to include form to add

# An Example breakdown

`<TodoList/>`

- `todos` - object of todos
- `onToggle` - called when a todo changes done
- `onDelete` - called when a todo is deleted
- `onAdd` - called when a todo is created

# Further Breakdown

- many `<TodoListItem/>`
    - `key` (id) - we are an array
    - `todo` individual todo item
    - `onToggle`
    - `onDelete`
- `<TodoAddForm/>`
    - manages temporary state
    - `onAdd`