

Agenda

-
- Handwritten notes on a whiteboard titled "DRY" (Don't Repeat Yourself). The notes are organized into a list of five items, with a bracket grouping items 3, 4, and 5. To the right of the list, there are several diagrams and annotations. A "Side bar" is shown with a "Completed" box. A "Video" box contains a "5" and "Video-10". A "repeating yourself" box is marked with a red "X" and "DRY" in a box. At the bottom right, the name "Pranav" is written.
- 1) Com point
 - 2) Props
 - 3) State
 - 4) Map
 - 5) from Library
- Side bar
- Completed
- repeating yourself X
- DRY
- Pranav

- 1) set of helper functions which are
u.e.
- 2) \Rightarrow read
- 3) few differences
- 4) browser ~~master~~ 2 read-only

States

- 1) state is an ^{updateable} object that contains information about a component
- 2) state can be changed
- 3) when changed, component re-renders
- 4) component can change the state
- 5) state is built by that component

CS & SY Chap → prop

- i) state
- boolean
 - Number
 - string
 - object
 - array

- 7) state is preserved on re-sends

React

- 1) Component
- <doctype>
- child
- element
- node
- <html> html element
- doc tag element tag
- <p> ... </p>
- <p> -

2)

-
- The diagram illustrates the difference between static and dynamic class members:
- Static Members:**
 - Associated with the **class**.
 - Shared by all instances.
 - Example: `static int count;`
 - Accessed via `ClassName.memberName` (e.g., `Person.count`).
 - Can be modified directly through the class reference.
 - Dynamic Members:**
 - Associated with individual **instances**.
 - Each instance has its own copy.
 - Example: `int id;` (instance variable).
 - Accessed via `instanceName.memberName` (e.g., `p.id`).
 - Cannot be modified directly through the class reference.
- A note at the bottom states: "dynamic class member".

- 1) import necessary resources
- 2) create a state
- 3) create a method to alter state
- 4) create UI render
- 5) handle event listeners