1.

Props are data passed to a component from another component. This prop cannot be modified, it is read only.

e.g



Components can manage and store their data. This data can change dynamically depending on the user input. These changes can also be rendered dynamically.

e.g:

here a the listnote state is being updated by creating a new list but with the added note. This happens when the form is submitted. Finally the state of the note is set to null to empty the input box of the form.

const handleSubmit = (event) => {

        event.preventDefault();

        setlistNote([...listNote, note]);

        setNote('');

    }

2.

A functor is an object that implements the map function. The map function takes in another function as an argument to modify the values inside the object. The return is a new functor the same size.

e.g

Here it returns a functor where each element is modified by incrementing it by 11.

const nums = [1,2,3,4];

console.log(nums.map(n => n + 11));

3.

|  |  |  |
| --- | --- | --- |
|  | Advantage | Disadvantage |
| Callbacks | Easy to use, and simple to understand in simple cases | Nested callbacks lead to difficulty in debugging. As the request of the second async request inside the first, depends on the response of the first request. |
| Promise | Promises can be chained but are simpler to reason about and debug with the use of .then() and .catch() methods. | Promises will only return a success or failure in operations. Initiated promise cannot be stopped. |
| Streams | Streams allow for more control and increase in performance with memory limited. | Difficult to learn |

4.

A diagram of a website

Description automatically generated with medium confidence

Content is the actual object in the box

Padding is the space around it and the border

Border surrounds padding

Margin is the space between other elements

5.

1. user goes to url in browser. Browser goes to an address book where the location of the website is, querying the DNS.

2. Browser makes a TCP/IP connection to the server’s IP address based on the port associated with the URL. Protocols tell where the files on the server it is located in.

3. Application server or HTTP proxy, listens on a specific port and accepts the the connection request from the browser.

4.Connection is now established, and browser sends a HTTP request to the server.

5. Server responds to request by parsing it on the same TCP/IP connection rendering the files of the website.