**INCEPTION EPISODE - 1**

1. VS Code use Emmet is generates some code for developers.
2. React Uses two CDN links so if we links both two cdn links in our project that we got the superpower of react so we need to put two cdn first is react.development it is the core react and second is react-dom.development and this is the bridge between the react and browser
3. React.createElement(para1, para2, para3) the first one is take a tag of html second is take the attributes like class name, id extras. and the third one is going to be a content or next nested html tag or it would be React.Element like nested stuff
4. But we creating a root and rendring inside the dom it is the job of React.Dom ReactDOM.createRoot(document.getElementById(“root”));
5. Most Constly or Expensive Oprations inside Browser page is when web page intractive each other or chages then the dom need to be manipulated suppose you hit or click the button then model appears so this is the intractivity.
6. When we write the React.createElement() it always create the object it’s not create the html

**Igniting Our App EPISODE - 2**

1. When we create our react app using **npx-create-react-app** this basically create the production ready app that have lot’s of configuration file
2. Npm is everything but not node package manager. Npm is manages the packages but it’s not standards for node package manager. Npm is the biggest packages managers.
3. Package.json is the configuration for npm
4. **Bundler**: when we have normal html, css and js file our whole code needs to be bundle together, compressed, clean , minified cached bundler helps us to do these things **webpacl**, **parcel** , **vite** bundlers basically bundle our app or packages our app so we can shift our application to the production.
5. There are two types of dependencis first one is dev and normal dependencis
6. "parcel": "^2.15.4" here the ^ is known as caret
7. The caret basically automatically fetches the current version of the package. Caret upgrade the minor ve (an attribute not part of the primary rsion of the package and **~ tilde** is upgraded the major version
8. Package-lock.json keep the excet version of the package that installed in the project
9. Node modules is like a database that contains the dependency that also contains it’s internal packages dependency that node\_modules like our project dependency have parcel now parcel as a project have it’s own dependency. And those dependency have it’s own dependency and those have it’s own this is known as transitive dependency
10. If we have package and package-lock.json we can re-create node\_modules
11. If we want to install something we need to use npm but if we want to execute something then we should run **npx** like **npx parcel index.html**
12. **Parcel** is doing lot’s of things like **Dev Build, Local Server, HMR-Hot Module Replacement, File watching algorithm** and it is written in C++, Caching – Faster Builds, Image Optimization, Minification, Bundling, Compress the file like removed the white space, Code Splitting, Differential Bundling-> means to support older browsers, parcel is giving us Diagnostic it also Error Handling, HTTPS, Tree Shaking -> remove unused code, Different bundle dev and Prod bundles

**Laying The Foundation EPISODE - 3**

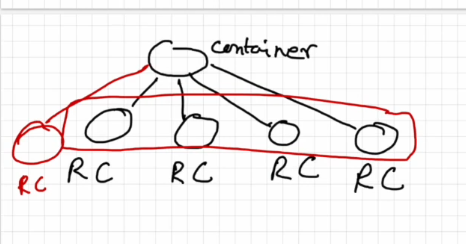
1. The react element `React.createElement()` is not html element it is an object so react element is end of the day is an object and when we render this object into the dom then it become the html element
2. React.createElement('h1', { id: 'name' }, "Hello World")

Writing like this in core react is very clumsy so react developer realized that this is not going to work for developers this is not a good way to create a browser elements it’s not developer friendly it’s not redable also to get off this things facebook developers create something known as **JSX** is a Javascript syntax which is easier to create react elements.

1. JSX is different lots of people said the jsx is the part of the react but it’s not. Lot of developer claim that react can not be written without JSX. But we can write react without JSX also we can build all the big application without using the JSX also but JSX makes our developers life easy
2. JSX is not html inside JavaScript. Jsx is not html jsx is a html like syntax or more closer to xml
3. Javascript does not comes jsx built inside it. Our js engine does not understand JSX . Javascript is code that JS engine understands. Js engine understand ecmascript.
4. const jsxheading = <h1>Hello World using JSX</h1>
5. if you paste this code in browser console it will throw error like this ` VM43:1 Uncaught SyntaxError: Unexpected token '<'`
6. because browser does not understand jsx it will give us the syntax error if browser does not understand the jsx then how react application work on the browser then parcel is doing the job behind the seen.
7. When we write jsx code then first the jsx code is transpiled before it reaches the javascript engine
8. Transpiled means this jsx code is converted to the code that browsers can understands that react can understands and who is transpiling the code the transpiling is done by parcel the webpack or bundler and parcel is doing itself absolutylt no it is done by babel that parcel uses inside it babel job is convert the jsx into code that browsers undnerstands
9. Now react element is the only things that react renders or work with then how jsx is written by us and it works correctly so when we write jsx it first transpiled into React element then react element is create the js element or object and then this js object is renderd as an html element so the jsx is end of the day is react.createElement and babel is doing all these things like jsx converts into react element babel is a javascript compiler.
10. In React Component two type class based component and second is functional component first one is old way and second is modern classes based component uses javascript classes to create it same functional component is uses javascript functions to create components.
11. React functional component is normal javascript function which return some peace of JSX code.
12. A function which returning a react element it will become a functional component
13. **Component Composition** -> when we put one component in another component this is called component composition

**Talk in cheap show me the code - EPISODE - 4**

1. When we are going to build a application first we need to make it first low wireframe so we can have an idea that what we are going to create.
2. In React we have something known as props, props mean properties which we can simply pass to the components, suppose I want to dynamically pass data to some component, how we can do that we can do it using props. For example the component is end of the day is normal JavaScript functions similarly the props are end of the day are just normal arguments to the functions.
3. Config-driven UI (CDUI) - A design pattern where the user interface is dynamically generated based on a configuration file, rather than being hardcoded into the application for example if some food delivering app the landing page carousel is different for different city or state because their carousel or UI changed automatically via the Configuration of the specific city, all the UI is driven by the data. That is what config-driven UI. Like we have written the UI once and now according the to data which is coming from the backend my UI is getting changed this is known as Config-driven UI (CDUI). The UI is powered by Data this is how built config driven UI. In any react application there is a UI layer and data layer both layers are made an application.



1. When we loop over a lot of things so suppose this was a card container and these restaurant card (RC), is all its Children, so react optimizes its render cycles, for example these are the components are the same level that we looped in so these components need to have unique ID. Like so they need to be uniquely represented, suppose if the new restaurant came in at the first place these is a new red colored restaurant (RC) came in at the first place, so if we will not pass the unique id to the cards then react will re renders all the cards inside the container because react does not know which card have to be renders because we didn’t pass the unique key to the restaurant cards

Some developers uses the index as their unique key and this is ok but should not pass the index to the cards or mapping over the elements because index Is good and these logically looks right because every time there is a unique key for each and every cards or elements but react itself says we should not use keys as form in indexes we can use indexes as an id but it’s not recommended. We always need to pass the unique keys and this is the best practice

**Let’s get Hooked EPISODE – 5**

1. Everything we can do in react we can simply do using html and JavaScript then why we need react and react is end of the day is JavaScript so whatever the framework or library we use to makes our developer experience easy this the job of every framework.
2. Always to the industry practice we should always put our code in src folders src means source code
3. Never ever keep any hard coded data in the component files.
4. React something uses **Reconciliation Algorithm** and this also knowns as **React Fiber** Virtual Dom is the Represent of the Actual DOM virtual DOM is object or react element so this react virtual DOM is basically kind of nothing but a object. It’s basically a nested object or representation of actual DOM.
5. **Diff** **Algorithm** basically finds out difference between two virtual DOM. It finds out the differences between the previous virtual dom and updated virtual dom. It will calculate the difference nodes and update the dom. In react16 a new algorithm to update the dom came out that algorithm is know as **Reconciliation** wheneversomething changes in UI this is known as **Reconciliation** after react 16 these algorithm knows as **React Fiber** now we work with React18
6. React is doing efficient dom manipulation this is why react fast because it’s have virtual dom virtual dom is become a popular concept when react has came in virtual dom concept existed long back virtual dom is not react thing what is virtual dom virtual dom is basically that dom you see that tags and elemetns that you see it’s kind of like a javascript representation of it that html code and object representation of it the concept existed long time ago in software engineering world. react took that and built it’s code algoritham over that virtual dom. React can effecncitly find out the difference the between virtual dom and update the UI this is the core of algoritham.
7. When we use state and it’s setState so the setState function basically trigger the diff algorithm and It will automatically re render the component state in react is a superpowerfyll variable that react keep eye on it

**Exploring the World EPISODE – 6**

1. Monolith and Micro service arcetecture
2. When our whole component is render out useeffect hook works