**ReactJs**

* **Standard code Structure ::** [**https://www.bezkoder.com/react-node-express-mongodb-mern-stack/**](https://www.bezkoder.com/react-node-express-mongodb-mern-stack/)
* **CRUD WITHOUT COMPONENT :: https://www.youtube.com/watch?v=eJyZ7k6lWfE**
* **Random image ::** [**https://picsum.photos/**](https://picsum.photos/)
* **Random data :: https://jsonplaceholder.typicode.com/**
* **Installation in ubuntu ::** [**https://www.tecmint.com/install-reactjs-on-ubuntu/**](https://www.tecmint.com/install-reactjs-on-ubuntu/)
* **Installation** in windows ::
  + Check version:: create -react -app -v
* **History** :: **Jordan Walke a software facebook developer**
  + OpenSource **Javascript library** for build UI(user interact kar sakte hai easily)
  + It’s handling the view layer for web and mobile apps ,
  + **Create and maintained by facebook** and large company developers
  + It’s early called **faxJs**
* **Terminal Command for create react project ::**
  + cd /home/st22122101/Projects/MEAR/MERN/FrontEnd
  + npm create react-app <app-name>
* **Run project in terminal ::**
  + cd /home/st22122101/Projects/MEAR/MERN/FrontEnd
  + npm start
* **In terminal if server is on type this ::** npx kill-port 3000
* **Once project ready for deployment then create production build of react application .its automatic build backend** 
  + **Yarn build**
* **React approach ::** 
  + **Component ::** it’s core building of react **(piece of react) and it is easier to build react**
    - **(It’s like ek obj ne man time,many place ae use karva )()make by piece of code**
  + **Declarative Approach ::** client doesn’t know which functionality use to create a system but they require a specific project
  + **DOM updates are handled gracefully**
  + **Reusable code**
* **ECMAScript6 :: Javascript compiler ,currently called ECMAScript**
  + **Our browser didn’t understand js code so ECMAScript work as (pehle browser understand json lekin ab hamara code modern ho gaya hai to browser didn’t understand our code so we use babel is a compiler jo hamara code compile karke convert karta hai browser language me )**
* **JSX :: JavaScript extension/xml** 
  + **When we use jsx must import react**
* **Hooks :: It’s a function that will help us to connect/use to react features.**
* **UIschema :: can control how to render (kese data catch karna hai)**
* **Json :: can control what to data render (konsa data send karna hai ui me )**
* **Solid EXAMPLE ::** [**https://codesandbox.io/examples/package/@rjsf/material-ui**](https://codesandbox.io/examples/package/@rjsf/material-ui)
* **PROPS EXAMPLE :** [**https://developer.here.com/blog/street-address-validation-with-reactjs-and-here-geocoder-autocomplete**](https://developer.here.com/blog/street-address-validation-with-reactjs-and-here-geocoder-autocomplete)
* **Between validation ::** [**https://formvalidation.io/guide/validators/between/**](https://formvalidation.io/guide/validators/between/)
* **LatitudeAnsLongitude ::**
* **Latitude: ^([+-])?(?:90(?:\\.0{1,6})?|((?:|[1-8])[0-9])(?:\\.[0-9]{1,6})?)$**
* **Longitude: ^([+-])?(?:180(?:\\.0{1,6})?|((?:|[1-9]|1[0-7])[0-9])(?:\\.[0-9]{1,6})?)$**

**ReactJs Redux**

* **React is unidirectional (parent-child)**
* **Prop drilling :: we transfer data from root ->new component->new component->required component(passsing data) problem is we have transfer data to each component till require component**
* **Props drilling solution :: useContext**
* **Another solution :: component can refer data from redux store**
* **Redux :: pattern and library that can manage or update application state (jo application state ko manage kart hai help of centralize store we catch data from centaralized file and kis component ne kis file ko call kiya hai jo previous dat aor now ka data hai wo predictable fashion me rahega )**
* **Redux mai topic :: (incremet-decrement eaxmple / juicer)**
  + **Action(What to do?)(two action perform hogi 1.incre and 2.decre)**
  + **Reducer(How to do?)(1.incre per kitna 2.decre per kitna)**
  + **Store :: data store is a centralized**
  + **Functions :: functio store in action index file is actionCreators**
  + **Action(What to do?)(round handle)**
  + **Reducer(How to do?)(fruid dal ke kese round ghumana hai )**
  + **Store :: data store is a centralized(initial fruit and updated fruit milega)**
  + **Functions :: functio store**
* **Dispatcher :: event or function call by dispatcher**
* **Selecter :: use show some initial state** 
  + **Const myselector = useSelecor(()=>{state.changeNumer})**
  + **(Here changeNumber is in reducer)**
  + **Const dispatch = useDispatcher();**
  + **onclick(()=>(dispatch(incNumber())))**
  + **Import (reduxStore)store in app.js**
  + **Import Provider from “react-redux”**
  + **store.subscriber(()=>console.log(store.geState()))**
  + **<Provider store={store}><App/></Provider>**

**Convert file to Base64**

* <https://codesandbox.io/s/convert-file-to-base64-in-react-lqi1e?file=/src/App.js>

**Redux**

* https://blog.bitsrc.io/simplifying-redux-with-redux-toolkit-6236c28cdfcb