|  |
| --- |
| {"success":true,"data":[{"\_id":"56918…. |

|  |
| --- |
| index.js  import React from 'react';  import ReactDOM from 'react-dom';  ReactDOM.render( <App />,document.getElementById('root') );  create App.jsx    App.jsx:  import React, {Component} from 'react';  class App extends Component{  render(){  return (  <h1> This is the Home Page </h1>  );  }  }  export default App;  index.js  import React from 'react';  import ReactDOM from 'react-dom';  **import App from './App';**  ReactDOM.render( <App />,document.getElementById('root') );  import React, {Component} from 'react';  class App extends Component{  render(){  return (  **<div> // more than one obj we need div**  **<Header />**  **<Content />**  **</div>**  );  }  }  class Header extends Component{  render(){  return (  <h1> This is the Header </h1>  );  }  }  class Content extends Component{  render(){  return (  <h1> This is the Content </h1>  );  }  }  export default App; |

| **Passing variables and Constructors**  **send to our component** |
| --- |
| import React, {Component} from 'react';  class App extends Component{  constructor(props){ **// it HAS TO be props**  super(props);  this.state={  myvar: "this is just some text",  data: [ // aray with 3 elements  {  "id" :1,  "name": "Dave",  "age" : "20"  },  {  "id" :2,  "name": "Joe",  "age" : "30"  },  {  "id" :3,  "name": "Bob",  "age" : "35"  }  ]  }  }  render(){  return (  <div>  <h1>**{this.props.headerProp**}</h1>  <h2>**{this.props.contentProp}**</h2>  </div>  );  }  }  index.js  import React from 'react';  import ReactDOM from 'react-dom';  import App from './App';  ReactDOM.render( **// any name we can pass here**  <App headerProp="Header from prop" contentProp="CONTENT from prop" />,  document.getElementById('root')  ); |

| **Default props** |
| --- |
| App.defaultProps={  headerProp:"Header from default prop",  contentProp:"CONTENT from default prop"  }    ReactDOM.render(  <App />, // GET RID OFF previous  document.getElementById('root')  ); |

| **send from component to sub component** |
| --- |
| class App extends Component{  constructor(props){  super(props);  this.state={  myvar: "this is just some text",  data: [  {  "id" :1,  "name": "Dave",  "age" : "20"  },  {  "id" :2,  "name": "Joe",  "age" : "30"  },  {  "id" :3,  "name": "Bob",  "age" : "35"  }  ]  }  }  render(){  return (  <div>  <h1>{this.props.headerProp}</h1>  <h2>{this.props.contentProp}</h2>  </div>  );  }  render(){  return(  **<table>**  **<tbody>**  **//MAP IS ITERATOR FOR ARRAY IT IS JAVA SCRIPT**  **{this.state.data.map((person,i) => <TableRow key={i} tableData={person} />)}**  **</tbody>**  **</table>**  );  }  }  //sub component  //tableDatais arbitary name  class TableRow extends Component{  render(){  return(  <tr>  <td><h1> { this.props.tableData.id }</h1> </td>  <td> { this.props.tableData.name }</td>  <td> { this.props.tableData.age }</td>  </tr>    )  }  } |

**today’s code**

part 1

const express = require('express');

const server = express();

const bodyParser =require('body-parser');

const mongoose =require('mongoose');

// CREATE A CONNECTION TO THE MONGO DB

//At home localhost:27017/movie

mongoose.connect("mongodb://192.168.75.128:27017/movie",{useNewUrlParser:true})

server.use(bodyParser.json());

// DEFINE SCHEMA OBJECT CORRESPONDING TO DB COLLECTOIN

var movieSchema= new mongoose.Schema({

title:{type: String, required:false},

year:{type: Number, required:false},

imdb:{type: String, required:false},

type:{type: String, required:false}

});

// CREATE MODEL OBJECT BASED ON THE ABOVE SCHEMA OBJECT

var Movie= mongoose.model("Movie" , movieSchema);

//CREATING ROUTE

//ROUTE FOR OUR BACKEND

server.get('/', function(req,res){

res.redirect("/movie");

})

//GET LIST ALL ROUTE

server.get('/movie', function(req,res){

Movie.find({} , function( err , movies ){

if(err){

return res

.status(400)

.json( { success : false , error : err } );

}

if( !movies.length){

return res

.status(404)

.json({ success : false , error : "No movies found"});

}

return res

.status(200)

.json({ success : true , data : movies});

});

});

/////////////////////////////////////////////////2022\_03\_15

/\*\* show one get /data/:id

\* Dar inja mikhaim baray 1 movie route benvism ke id ra gereftim (/movie/:id)

\* We can cosider id from user: req.params.id we saw in session1

\*

\* ---.get('/movie/:id', function(req,res){

Movie.find(RECORDS , function( err , movies ){

if(err){

}

if( !movies.length){

}

SUCCESS

FOR ALL WE HAVE return res.status(.........).json(....)

error code sucess:true/false , message

\*/npx create-react-app

server.get('/movie/:id', function(req,res){

Movie.findById(req.params.id , function( err , movie ){

if(err){

return res

.status(400)

.json( { success : false , error : err } );

}

if(!movie){

return res

.status(404)

.json({ success : false , error : "The movie wasn\'t found"});

}

return res

.status(200)

.json({ success : true , data : movie});

});

});

//set up server to listen for connection

server.listen(3010,function(){

console.log("Server started....")

});

second part react

LOOK AT ABOVE MORE

import { render } from '@testing-library/react';

import React, {Component} from 'react';

class App extends Component{

constructor(props){

super(props);

this.state={

myvar: "this is just some text",

data: [

{

"id" :1,

"name": "Dave",

"age" : "20"

},

{

"id" :2,

"name": "Joe",

"age" : "30"

},

{

"id" :3,

"name": "Bob",

"age" : "35"

}

]

}

}

render(){

return (

<div>

<h1>{this.props.headerProp}</h1>

<h2>{this.props.contentProp}</h2>

</div>

);

}

render(){

return(

<table>

<tbody>

{this.state.data.map((person,i) => <TableRow key={i} tableData={person} />)}

</tbody>

</table>

);

}

}

//sub component

//tableDatais arbitary name

class TableRow extends Component{

render(){

return(

<tr>

<td><h1> { this.props.tableData.id }</h1> </td>

<td> { this.props.tableData.name }</td>

<td> { this.props.tableData.age }</td>

</tr>

)

}

}

// class Header extends Component{

// render(){

// return (

// <h1> This is the Header </h1>

// );

// }

// }

// class Content extends Component{

// render(){

// return (

// <h1> This is the Content </h1>

// );

// }

// }

App.defaultProps={

headerProp:"Header from default prop",

contentProp:"CONTENT from default prop"

}

export default App;