API

(Application Programming Interface)

API is the acronym **for Application Programming Interface**, which is a software intermediary that allows two applications to talk to each other. Each time you use an app like Facebook, send an instant message, or check the weather on your phone, you're using an API.

It’s methods are:

1. **Get Method API** – Show The Data with API
2. **Post Method API** – Store The Data with API
3. **Put Method API** – Update The Data with API
4. **Delete Method API** – Delete The Data with API

**We are create fake API from ‘Json-Server’ for using Data manipulate**

**https://github.com/typicode/json-server**

**First of all install Json server: npm install -g json-server**

**And then run the server : json-server --watch db.json**

**Okay fine! Then open your** [**http://localhost:3000/posts/**](http://localhost:3000/posts/) **or**

**Brower And its Url Address bar: http://localhost:3000/posts/1**

**Past this link**

**Get Method API**

**App.js :-**

import React,{useEffect, useState} from 'react'

import './App.css'

export default function App(){

    const [user,setUser]=useState([])

   useEffect(() => {

    fetch("http://localhost:3000/posts").then((get)=>{

        get.json().then((data)=>{

            setUser(data)

        })

    })

   }, [])

   console.warn(user)

    return(

        <div className='App'>

            <h1>get method from API</h1>

            <table border="3">

                <tr>

                    <th>ID</th>

                    <th>NAME</th>

                    <th>SURNAME</th>

                    <th>MOBILE</th>

                </tr>

                {

                    user.map((item)=>

                <tr>

                    <td>{item.id}</td>

                    <td>{item.name}</td>

                    <td>{item.surname}</td>

                    <td>{item.mobile}</td>

                </tr>

                    )

                }

            </table>

        </div>

    );

}

**DB.JSON** :- it’s fake API Json-server to fetching for App.js

{

  "posts": [

    {

      "id": 1,

      "name":"sonu",

      "surname":"bandhey",

      "mobile": 783657

    },

    {

      "id": 2,

      "name":"monu",

      "surname":"bandhey",

      "mobile": 536456

    },

    {

      "id": 3,

      "name":"nanu",

      "surname":"bandhey",

      "mobile": 23984759

    },

    {

      "id": 4,

      "name":"buggy",

      "surname":"bandhey",

      "mobile": 895965

    }

  ],

  "comments": [

    {

      "id": 1,

      "body": "some comment",

      "postId": 1

    }

  ],

  "profile": {

    "name": "typicode"

  }

}

**Post Method API**

**App.js :-**

import React,{useState} from "react";

import './App.css'

export default function App(){

    const [name,setName] = useState("");

    const [surname,setSurname] = useState("");

    const [mobile,setMobile] = useState("");

    function Save(){

        console.warn(name,surname,mobile);

        let data = {name,surname,mobile};

        fetch("http://localhost:3000/posts/",{

            method:"POST",

            headers:{

                "Accept":"application/json",

                "content-type":"application/json"

            },

            body:JSON.stringify(data)

        }).then((post)=>{

            console.warn("post",post)

            post.json().then((result)=>{

                console.warn(result)

            })

        })

    }

    return(

        <div className="App">

            <h1>Post API Method</h1>

            <input type="text" value={name} onChange={(e)=>{setName(e.target.value)}} name = "name"/><br/><br/>

            <input type="text" value={surname} onChange={(e)=>{setSurname(e.target.value)}} name = "surname"/><br/><br/>

            <input type="text" value={mobile} onChange={(e)=>{setMobile(e.target.value)}} name = "mobile"/><br/><br/>

            <button type="button" onClick={Save}>Add New Data</button>

        </div>

    )

}

**Method :-** post data for there location address.

**Headers :-** they status for location address. What content-type to accept and what content-type to use.

**Body :-** it’s brower body to show and we are looking it.

**JSON.stringify :- it is json format file to show the contents is string.**

**Db-json :-**

{

  "posts": [

    {

      "id": 1,

      "name": "sonu",

      "surname": "bandhey",

      "mobile": 783657

    },

    {

      "id": 2,

      "name": "monu",

      "surname": "bandhey",

      "mobile": 536456

    },

    {

      "id": 3,

      "name": "nanu",

      "surname": "bandhey",

      "mobile": 23984759

    },

    {

      "id": 4,

      "name": "buggy",

      "surname": "bandhey",

      "mobile": 895965

    },

    {

      "name": "iron-man",

      "surname": "iron@man8787",

      "mobile": 8347569387,

      "id": 5

    },

    {

      "name": "dfgdfg",

      "surname": "dfgdfgfg",

      "mobile": "d67567",

      "id": 6

    },

    {

      "name": "dfgdfg",

      "surname": "dfgdfgfg",

      "mobile": "d67567",

      "id": 7

    },

    {

      "name": "dfgdfg",

      "surname": "dfgdfgfg",

      "mobile": "d67567",

      "id": 8

    },

    {

      "name": "dfgdfg",

      "surname": "dfgdfgfg",

      "mobile": "d67567",

      "id": 9

    },

    {

      "name": "buggy",

      "surname": "bandhey",

      "mobile": "5646984964",

      "id": 10

    }

  ],

  "comments": [

    {

      "id": 1,

      "body": "some comment",

      "postId": 1

    }

  ],

  "profile": {

    "name": "typicode"

  }

}

**Delete Method API**

**App.js :-**

import React,{useEffect, useState} from 'react'

import './App.css'

export default function App(){

    const [user,setUser]=useState([])

   useEffect(() => {

    getData();

   }, [])

   function getData(){

    fetch("http://localhost:3000/posts").then((get)=>{

        get.json().then((data)=>{

            setUser(data)

        })

    })

   }

   function deleteData(id){

       alert(id)

       fetch(`http://localhost:3000/posts/${id}`,{

           method:"DELETE"

       }).then((del)=>{

           del.json().then((result)=>{

               console.warn(result)

           })

       })

       getData();

   }

   console.warn(user)

    return(

        <div className='App'>

            <h1>get method from API</h1>

            <table border="3">

                <tr>

                    <th>ID</th>

                    <th>NAME</th>

                    <th>SURNAME</th>

                    <th>MOBILE</th>

                    <th>OPERATION</th>

                </tr>

                {

                    user.map((item,i)=>

                <tr key={i}>

                    <td>{item.id}</td>

                    <td>{item.name}</td>

                    <td>{item.surname}</td>

                    <td>{item.mobile}</td>

                    <td><button type="button" onClick={()=>deleteData(item.id)}>Delete</button></td>

                </tr>

                    )

                }

            </table>

        </div>

    );

}

**Db.json :-**

{

  "posts": [

  ],

  "comments": [

    {

      "id": 1,

      "body": "some comment",

      "postId": 1

    }

  ],

  "profile": {

    "name": "typicode"

  }

}

**Pre-fill form method**

**App.js:-**

import React,{useState, useEffect} from 'react'

import './App.css'

export default function App(){

  const [user,setUser] = useState([])

  const [name,setName] = useState("")

  const [surname,setSurname] = useState("")

  const [mobile,setMobile] = useState("")

  useEffect(()=>{

    getData();

  },[])

  function getData(){

    fetch("http://localhost:3000/posts").then((get)=>{

      get.json().then((result)=>{

        setUser(result)

        setName(result[0].name)

        setSurname(result[0].surname)

        setMobile(result[0].mobile)

      })

    })

  }

  function deleteData(id){

    fetch(`http://localhost:3000/posts/${id}`,

    {method: "DELETE"}).then((del)=>{

      del.json().then((result)=>{

        console.warn(result)

      })

    })

    getData();

  }

  function addUser(id){

        console.warn(user[id-1])

        let add = user[id-1]

        setName(add.name)

        setSurname(add.surname)

        setMobile(add.mobile)

  }

    return(

      <div className='App'>

        <h1>DATA PRE-FILL METHOD</h1>

          <table border="2" style={{float:"left"}}>

            <tbody>

              <tr>

                <th>ID</th>

                <th>NAME</th>

                <th>SURNAME</th>

                <th>MOBILE</th>

                <th colSpan={2}>OPERATION</th>

              </tr>

              {

                user.map((item,i)=>

                  <tr key={i}>

                    <td>{item.id}</td>

                    <td>{item.name}</td>

                    <td>{item.surname}</td>

                    <td>{item.mobile}</td>

                    <td><button onClick={()=>deleteData(item.id)}>DELETE</button></td>

                    <td><button onClick={()=>addUser(item.id)}>UPDATE</button></td>

                  </tr>

                )

              }

            </tbody>

          </table>

          <form >

             <input type="text" value={name}/><br/><br/>

             <input type="text" value={surname}/><br/><br/>

             <input type="text" value={mobile}/><br/><br/>

             <input type="submit"/>

          </form>

      </div>

    );

}

**Db.json:-**

{

  "posts": [

    {

      "id": 1,

      "name": "sonu",

      "surname": "bandhey",

      "mobile": 783657

    },

    {

      "id": 2,

      "name": "monu",

      "surname": "bandhey",

      "mobile": 536456

    },

    {

      "id": 3,

      "name": "nanu",

      "surname": "bandhey",

      "mobile": 23984759

    },

    {

      "id": 4,

      "name": "aashu",

      "surname": "bandhey",

      "mobile": 895965

    },

    {

      "name": "iron-man",

      "surname": "iron@man8787",

      "mobile": 8347569387,

      "id": 5

    }

  ],

  "comments": [

    {

      "id": 1,

      "body": "some comment",

      "postId": 1

    }

  ],

  "profile": {

    "name": "typicode"

  }

}

**Put Method API**

**App.js:-**

import React, { useState, useEffect } from 'react'

import './App.css'

export default function App() {

  const [user, setUser] = useState([])

  const [name, setName] = useState("")

  const [surname, setSurname] = useState("")

  const [mobile, setMobile] = useState("")

  const [userID, setUserID] = useState(null)

  useEffect(() => {

    getData();

  }, [])

  function getData() {

    fetch("http://localhost:3000/posts").then((get) => {

      get.json().then((result) => {

        setUser(result)

        setName(result[0].name)

        setSurname(result[0].surname)

        setMobile(result[0].mobile)

        setUserID(result[0].id)

      })

    })

  }

  function deleteData(id) {

    fetch(`http://localhost:3000/posts/${id}`,

      { method: "DELETE" }).then((del) => {

        del.json().then((result) => {

          console.warn(result)

          getData()

        })

      })

  }

  function addUser(id) {

    console.warn(user[id - 1])

    let add = user[id - 1]

    setUserID(add.id)

    setName(add.name)

    setSurname(add.surname)

    setMobile(add.mobile)

  }

  function selectUser() {

    let item = { name, surname, mobile, userID }

    fetch(`http://localhost:3000/posts/${userID}`,

      {

        method: "PUT",

        headers: {

          'Accept': 'application/json',

          'Content-Type': 'application/json'

        },

        body: JSON.stringify(item)

      }).then((del) => {

        del.json().then((result) => {

          console.warn(result)

          getData()

        })

      })

  }

  return (

    <div className='App'>

      <h1>DATA PRE-FILL METHOD</h1>

      <table border="2" style={{ float: "left" }}>

        <tbody>

          <tr>

            <th>ID</th>

            <th>NAME</th>

            <th>SURNAME</th>

            <th>MOBILE</th>

            <th colSpan={2}>OPERATION</th>

          </tr>

          {

            user.map((item, i) =>

              <tr key={i}>

                <td>{item.id}</td>

                <td>{item.name}</td>

                <td>{item.surname}</td>

                <td>{item.mobile}</td>

                <td><button onClick={() => deleteData(item.id)}>DELETE</button></td>

                <td><button onClick={() => addUser(item.id)}>UPDATE</button></td>

              </tr>

            )

          }

        </tbody>

      </table>

      <div >

        <input type="text" value={name} onChange={(e) => setName(e.target.value)} /><br /><br />

        <input type="text" value={surname} onChange={(e) => setSurname(e.target.value)} /><br /><br />

        <input type="text" value={mobile} onChange={(e) => setMobile(e.target.value)} /><br /><br />

        <button onClick={selectUser}>Modify</button>

      </div>

    </div>

  );

}

**Db.json:-**

{

  "posts": [

    {

      "name": "sonu",

      "surname": "bandhey",

      "mobile": 783657,

      "userID": 1,

      "id": 1

    },

    {

      "id": 2,

      "name": "monu",

      "surname": "bandhey",

      "mobile": 536456

    },

    {

      "id": 3,

      "name": "nanu",

      "surname": "bandhey",

      "mobile": 23984759

    },

    {

      "name": "aashu",

      "surname": "bandhey",

      "mobile": "875234",

      "userID": 4,

      "id": 4

    },

    {

      "name": "iron-man",

      "surname": "iron@man8787",

      "mobile": 8347569387,

      "id": 5

    }

  ],

  "comments": [

    {

      "id": 1,

      "body": "some comment",

      "postId": 1

    }

  ],

  "profile": {

    "name": "typicode"

  }

}