|  |  |  |
| --- | --- | --- |
| **CURD Operation in react** | | |
|  | CreateURD |  |
| Steps   1. Make a from 2. Get from data & make it an object. 3. Make a POST API in backend/Server 4. Send Data To the server and get a confirmation sms | Step 1 => Make a from  <form onSubmit={handleSubmit}>    <input type="text" name="personName" placeholder="Your Name" required />    <input type="email" name="personEmail" placeholder="Your Email" required />    <input type="submit" value={"Add User"} />  </form>  Step 2 => Get from data  const handleSubmit = (e) => {    e.preventDefault();    const personName = e.target.personName.value;    const personEmail = e.target.personEmail.value;    const userInfo = {personName, personEmail};    console.log(userInfo);  Step 4 => Send data to the server  fetch(    "https://url/addNewUser",    {      method: "POST",      headers: {        "Content-Type": "application/json",      },      body: JSON.stringify(userInfo),    }  )    .then((res) => res.json())    .then((data) => {      if (data.insertedId) {  do something      }      form.reset();    });  }; | Step 3 => Make a API in server  // post data to database  app.post("/addNewUser", async (req, res) => {    const userInfo = req.body;    const result = await dbCollectionName.insertOne(userInfo);    res.send(result);  });   * url url same hobe (<https://url/addNewUser> === /addNewUser) * mthod: “POST” === app.post * user theke data body te jay and server er req.body te giye joma hoy ( userInfo obj === body: JSON.stringify(userInfo) === req.body) * server theke res send kore j taka call korsilo ekta confirmation sms day (res.send(result) === ((res) => res.json()) |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  | **All Data Get /** CUReadD |  |
|  | {  path: "/allTouristsSpot",  element: <AllTouristsSpot />,  loader: () => fetch('https://b9a10-server-side-sakhawat-code-info.vercel.app/allData’),  },  // all data sent  app.get('/ allData’, async (req, res) => {  const cursor = touristSpotData.find();  const result = await cursor.toArray();  res.send(result);  }) | Jai page e data chai  const touristDataFromDB = useLoaderData();  const [allData, setAllData] = useState(touristDataFromDB);  // console.log(allData)  const sortBy = (sortData) => {  if (sortData === "averageCost") {  const sortByNumber = [].concat(allData).sort((a, b) => b.average\_cost - a.average\_cost);  setAllData(sortByNumber);  }  };  Show data  <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3">  {/\* single component \*/}  {  allData?.map(SingleTouristSpotData => <SingleTouristSpot  key={SingleTouristSpotData.\_id}  SingleTouristSpotData={SingleTouristSpotData}  ></SingleTouristSpot>)  } |
|  | CURDelete |  |
|  | <button onClick={() => handleDataDelete(singleData.\_id)} className="flex items-center p-2 transition ease-in duration-200 uppercase rounded-full hover:bg-gray-800 hover:text-white border-2 border-gray-900 focus:outline-none">  <MdOutlineDeleteForever size={20} />  </button>  const handleDataDelete = (id) => {  fetch(`https://b9a10-server-side-sakhawat-code-info.vercel.app/singleDataDelete/${id}`, {  method: "DELETE"  })  .then(res => res.json())  .then(data => {  if (data.deletedCount === 1) {  give him a sms  } | app.delete('/singleDataDelete/:deleteId', async (req, res) => {  const deleteId = req.params.deleteId;  const query = { \_id: new ObjectId(deleteId) };  const result = await touristSpotData.deleteOne(query);  res.send(result);  }) |
|  | CUpdateRD |  |
|  | {  path: "/updateTouristSpotDataPage/:needUpdateThisId",  element: <ProtectedRoutes><UpdateTouristSpotDataPage /></ProtectedRoutes>,  loader: ({ params }) => fetch(`https://b9a10-server-side-sakhawat-code-info.vercel.app/findSingleDataForUpdate/${params.needUpdateThisId}`)  }, | app.put('/updateSingleData/:updateNowThisId', async (req, res) => {  const updateData = req.body;  const updateNowThisId = req.params.updateNowThisId;  const filter = { \_id: new ObjectId(updateNowThisId) };  const options = { upsert: true };  const updateDoc = {  $set: {  country\_Name: updateData.country\_Name,  location: updateData.location,  tourists\_spot\_name: updateData.tourists\_spot\_name,  locationImgLink: updateData.locationImgLink,  shortDescription: updateData.shortDescription,  seasonality: updateData.seasonality,  travel\_time: updateData.travel\_time,  totalVisitorsPerYear: updateData.totalVisitorsPerYear,  average\_cost: updateData.average\_cost,  userName: updateData.userName,  userEmail: updateData.userEmail,  userCreatePostDateAndTime: updateData.userCreatePostDateAndTim  },  }; |
|  |  |  |

**56.1**