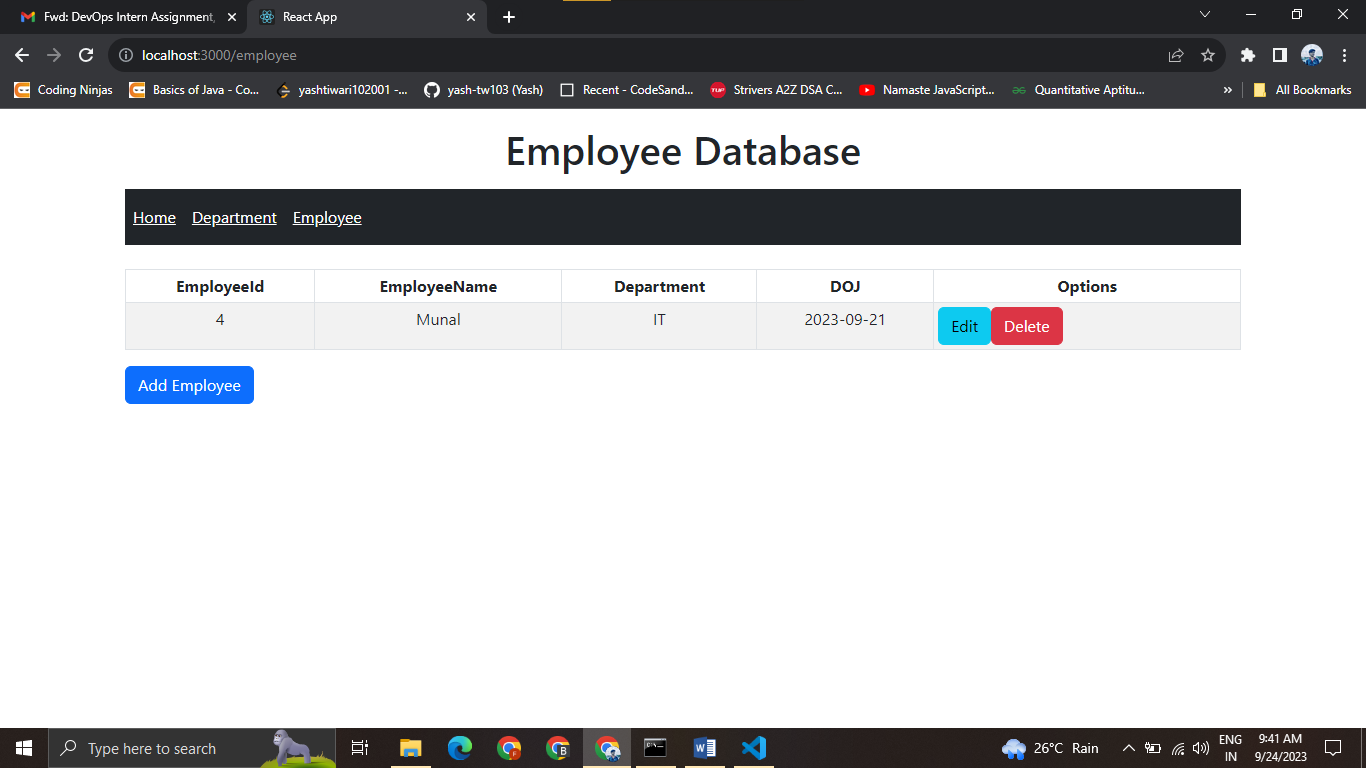
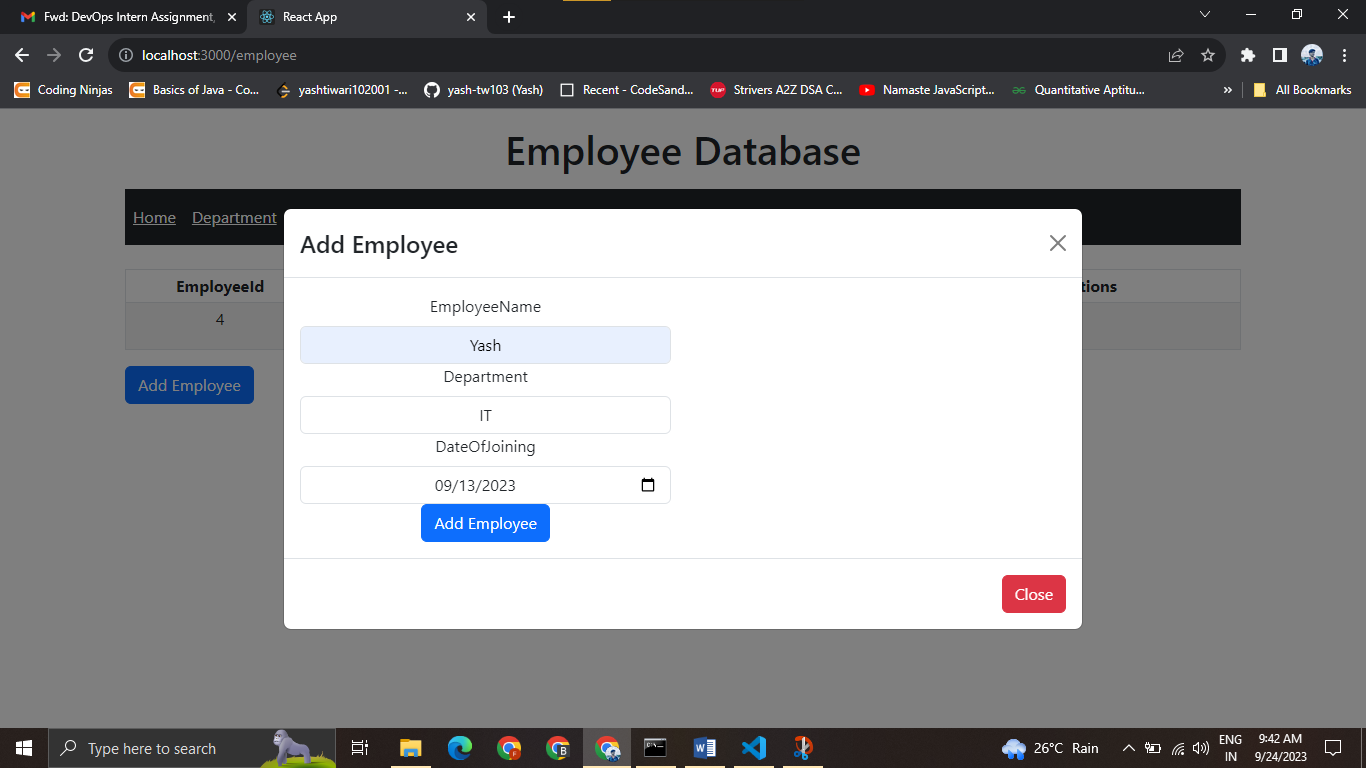
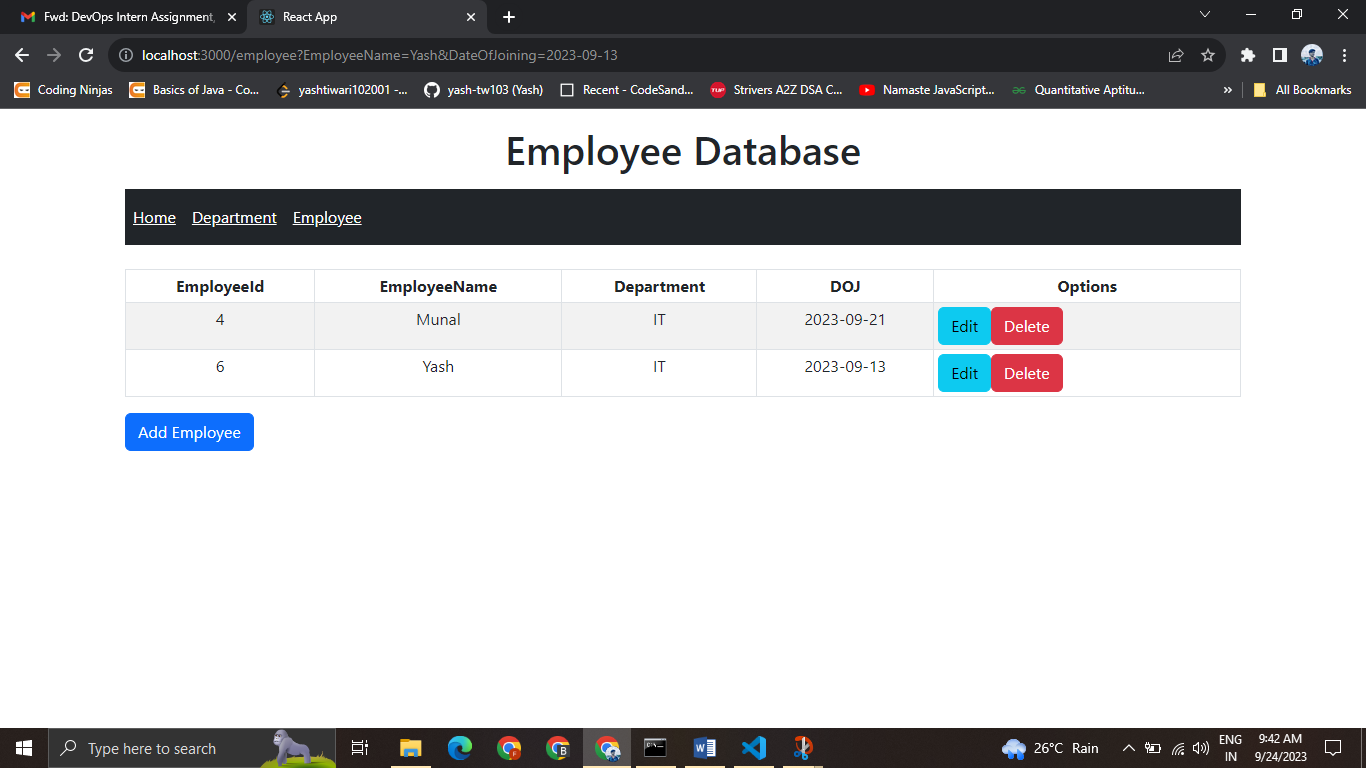


OUR EMPLOYEE DATA LOOKS LIKE THIS







Now to create this project we will take the following Steps

* We create our environment for python In our project folder , for that we do
* Python –**m venv myenv**
* **myenv\Scripts\activate**
* We’ll get (**myenv) C:\Users**\..... our path
* Then we do **pip install django**
* Then we install **pip install djangorestframework**
* **Now we make our project with -> Django startproject DjangoAPI**
* We do **cd DjangoAPI**
* Then we run our server with -> **python manage.py runserver**
* Open **SQLLite** studio and add our sqllite3 database to it
* Then we installed cors headers by ->**pip install** **django-cors-headers**
* Add cors headers in **installedApps and Middleware**

INSTALLED\_APPS = [

    'django.contrib.admin',

    'django.contrib.auth',

    'django.contrib.contenttypes',

    'django.contrib.sessions',

    'django.contrib.messages',

    'django.contrib.staticfiles',

'corsheaders'

]

CORS\_ORIGIN\_ALLOW\_ALL = True

MIDDLEWARE = [

    'corsheader.middleware.CorsMiddleware',

* Then we start our App
* Python mange.py startapp EmployeeApp and add these below
* 'corsheaders ',
* 'EmployeeApp.apps.EmployeeappConfig',
* 'rest\_framework'
* Now In our models.py we create our Database
* # Create your models here.
* class Departments(models.Model):
* DepartmentId = models.AutoField(primary\_key=True)
* DepartmentName = models.CharField(max\_length=100)
* class Employees(models.Model):
* EmployeeId = models.AutoField(primary\_key=True)
* EmployeeName = models.CharField(max\_length=100)
* Department = models.CharField(max\_length=100)
* DateOfJoining = models.DateField()
* PhotoFileName = models.CharField(max\_length=100)
* Then we make migrations with the command
* **Python manage.py makemigrations Employee**
* **Python manage.py migrate EmployeeApp(this will migrate our models to the sql tables)**

we create a file Nmae **serializers.py**  and then

from rest\_framework import serializers

from EmployeeApp.models import Departments , Employees

class DepartmentSerializer(serializers.ModelSerializer):

    class Meta:

        model = Departments

        fields = ('DepartmentId',

                  'DepartmentName')

class EmployeeSerializer(serializers.ModelSerializer):

    class Meta:

        model = Employees

        fields = ('EmployeeId',

                  'EmployeeName',

                  'Department',

                  'DateOfJoining',

                  'PhotoFileName')

Then in our views.py we create all of our requests

from django.shortcuts import render

from django.views.decorators.csrf import csrf\_exempt

from rest\_framework.parsers import JSONParser

from django.http.response import JsonResponse

from EmployeeApp.models import Departments, Employees

from EmployeeApp.serializers import DepartmentSerializer, EmployeeSerializer

# Create your views here.

@csrf\_exempt

def departmentApi(request,id=0):

    if request.method== 'GET':

        departments = Departments.objects.all()

        departments\_serializer= DepartmentSerializer(departments, many=True)

        return JsonResponse(departments\_serializer.data, safe=False)

    elif request.method=='POST':

        department\_data=JSONParser().parse(request)

        department\_serializer = DepartmentSerializer (data=department\_data)

        if department\_serializer.is\_valid():

            department\_serializer.save()

            return JsonResponse("Added Successfully!!", safe=False)

        return JsonResponse("Failed to Add.", safe=False)

    elif request.method== 'PUT':

        department\_data = JSONParser().parse(request)

        department=Departments.objects.get(DepartmentId=department\_data['DepartmentId'])

        department\_serializer=DepartmentSerializer(department, data=department\_data)

        if department\_serializer.is\_valid():

            department\_serializer.save()

            return JsonResponse("Updated Successfully!!", safe=False)

        return JsonResponse("Failed to Update.", safe=False)

    elif request.method=="DELETE":

        department=Departments.objects.get(DepartmentId=id)

        department.delete()

        return JsonResponse("Deleted Successffully!!", safe=False)

* Now we set our Urls for the main
* We create a file name urls.py in EmployeeApp
* from django.urls import re\_path
* from EmployeeApp import views
* urlpatterns=[
* re\_path(r'^department/$',views.departmentApi),
* re\_path(r'^department/([0-9]+)$',views.departmentApi)
* ]

Then we import and include these urls in our DjangoApi

from django.contrib import admin

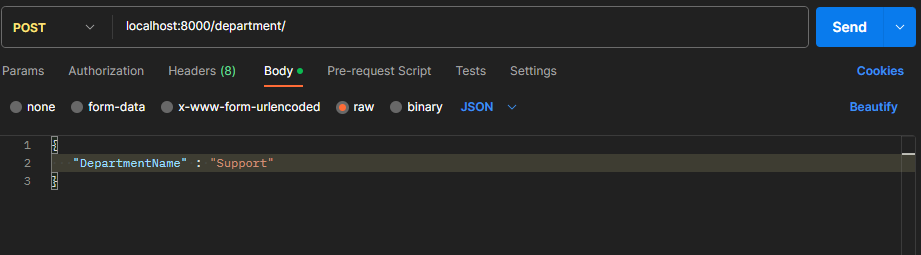
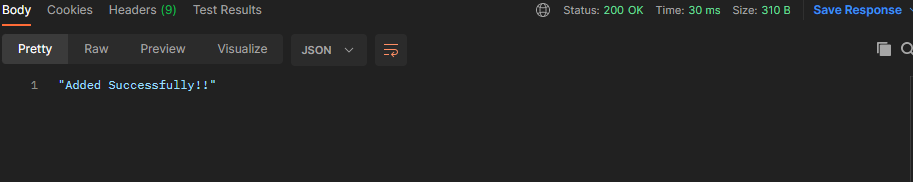
from django.urls import path , include , re\_path

urlpatterns = [

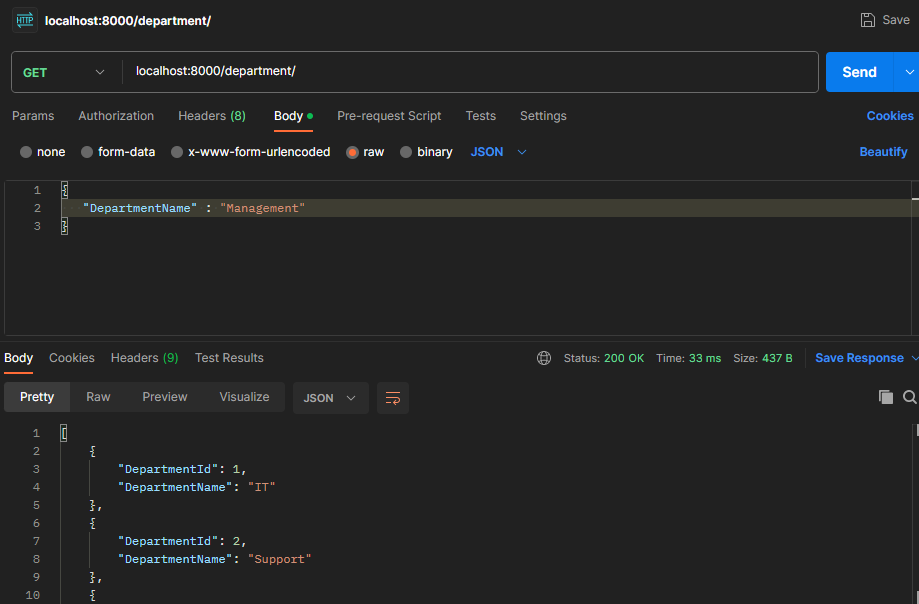
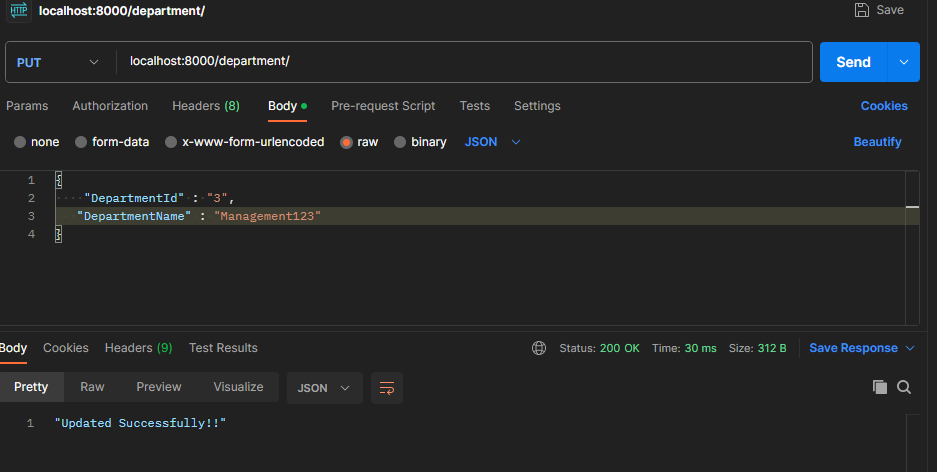
    path('admin/', admin.site.urls),

    re\_path(r'^',include('EmployeeApp.urls'))

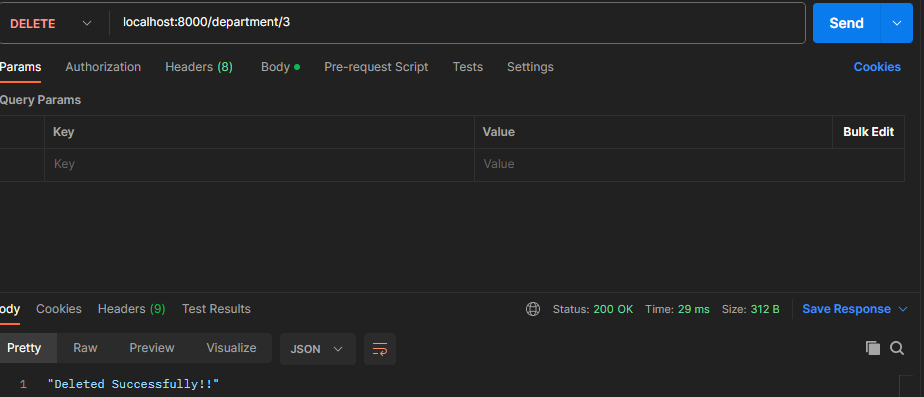
]

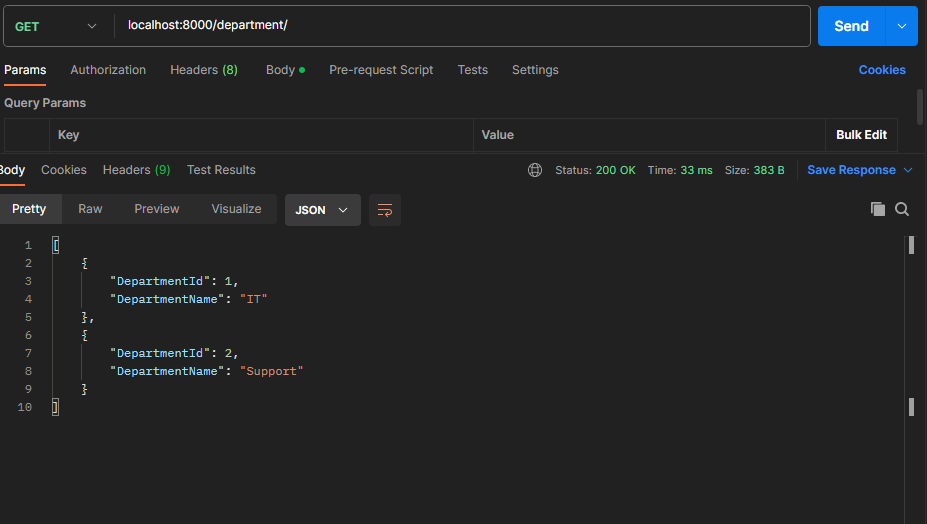
* Now we can check our request through Postman Api
* Initially there was no data
* So a get request will give an empty array
* Now we add data inside our app through post request
* 
* And we add “DepartmentName ” :”IT”, and send the request we get
* 

Now when we send a get request we get our data we get

* 
* Our pust request work like this
* 

Our delete request





* Now we create employee api
* @csrf\_exempt
* def employeeApi(request,id=0):
* if request.method== 'GET':
* employees = Employees.objects.all()
* employees\_serializer= EmployeeSerializer(employees, many=True)
* return JsonResponse(employees\_serializer.data, safe=False)
* elif request.method=='POST':
* employee\_data=JSONParser().parse(request)
* employee\_serializer = EmployeeSerializer (data=employee\_data)
* if employee\_serializer.is\_valid():
* employee\_serializer.save()
* return JsonResponse("Added Successfully!!", safe=False)
* return JsonResponse("Failed to Add.", safe=False)

* elif request.method== 'PUT':
* employee\_data = JSONParser().parse(request)
* employee=Employees.objects.get(EmployeeId=employee\_data['EmployeeId'])
* employee\_serializer=EmployeeSerializer(employee, data=employee\_data)
* if employee\_serializer.is\_valid():
* employee\_serializer.save()
* return JsonResponse("Updated Successfully!!", safe=False)
* return JsonResponse("Failed to Update.", safe=False)
* elif request.method=="DELETE":
* employee=Departments.objects.get(EmployeeId=id)
* employee.delete()
* return JsonResponse("Deleted Successfully!!", safe=False)
* Now to upload photo
* In **views.py**

from django.core.files.storage import default\_storage

def SaveFile(request):

    file = request.FILES['myFile']

    file\_name = default\_storage.save(file.name , file)

    return JsonResponse(file\_name , safe=False)

in our urls.py in EmployeeApp

from django.conf.urls.static import static

from django.conf import settings

urlpatterns=[

re\_path(r'^department/$',views.departmentApi),

re\_path(r'^department/([0-9]+)$',views.departmentApi),

#employeeapi

re\_path(r'^employee/$',views.employeeApi),

re\_path(r'^employee/([0-9]+)$',views.employeeApi),

re\_path(r'^Employee/SaveFile$',views.SaveFile)

] + static(settings.MEDIA\_URL, document\_root = settings.MEDIA\_ROOT)

**OUR BACKEND IS COMPLETE LET’S CREATE OUR FRONTEND**

**We create our react application with npx create-react-app**

* **Now we will create our Home Page**
* **We will create routes for Adding department and Adding Employee**
* **We create Navigation For setting Routes**
* **In navigation.js**
* import React from 'react'
* import { NavLink } from 'react-router-dom';
* import { Navbar, Nav } from 'react-bootstrap';
* function Navigation() {
* return (
* <div>
* <center>
* <Navbar bg="dark" expand="lg">
* <Navbar.Toggle aria-controls="basic-navbar-nav" />
* <Navbar.Collapse id="basic-navbar-nav">
* <Nav>
* <NavLink className="d-inline p-2 bg-dark text-white" to="/">
* Home
* </NavLink>
* <NavLink className="d-inline p-2 bg-dark text-white" to="/department">
* Department
* </NavLink>
* <NavLink className="d-inline p-2 bg-dark text-white" to="/employee">
* Employee
* </NavLink>
* </Nav>
* </Navbar.Collapse>
* </Navbar>
* </center>
* </div >
* )
* }
* export default Navigation

**Now we create add department and employee , with basic functionalities with edit and delete**

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

import { Table, Button, ButtonToolbar } from 'react-bootstrap';

import  AddDepModal  from './AddDepModal';

import  EditDepModal  from './EditDepModal';

function Department() {

  const [deps, setDeps] = useState([]);

  const [addModalShow, setAddModalShow] = useState(false);

  const [editModalShow, setEditModalShow] = useState(false);

  const [depid, setDepId] = useState(null);

  const [depname, setDepName] = useState('');

  const url = 'http://127.0.0.1:8000/'

  const refreshList = () => {

    fetch(`${url}department/`)

      .then((response) => response.json())

      .then((data) => {

        setDeps(data);

      });

  };

  useEffect(() => {

    refreshList();

  }, []);

  const deleteDep = (depid) => {

    if (window.confirm('Are you sure?')) {

      fetch(`${url}department/${depid}`, {

        method: 'DELETE',

        headers: {

          Accept: 'application/json',

          'Content-Type': 'application/json',

        },

      });

    }

  };

  const addModalClose = () => setAddModalShow(false);

  const editModalClose = () => setEditModalShow(false);

  return (

    <div>

      <Table className="mt-4" striped bordered hover size="sm">

        <thead>

          <tr>

            <th>DepartmentId</th>

            <th>DepartmentName</th>

            <th>Options</th>

          </tr>

        </thead>

        <tbody>

          {deps.map((dep) => (

            <tr key={dep.DepartmentId}>

              <td>{dep.DepartmentId}</td>

              <td>{dep.DepartmentName}</td>

              <td>

                <ButtonToolbar>

                  <Button

                    className="mr-2"

                    variant="info"

                    onClick={() => {

                      setEditModalShow(true);

                      setDepId(dep.DepartmentId);

                      setDepName(dep.DepartmentName);

                    }}

                  >

                    Edit

                  </Button>

                  <Button

                    className="mr-2"

                    variant="danger"

                    onClick={() => deleteDep(dep.DepartmentId)}

                  >

                    Delete

                  </Button>

                  <EditDepModal

                    show={editModalShow}

                    onHide={editModalClose}

                    depid={depid}

                    depname={depname}

                  />

                </ButtonToolbar>

              </td>

            </tr>

          ))}

        </tbody>

      </Table>

      <ButtonToolbar>

        <Button

          variant="primary"

          onClick={() => setAddModalShow(true)}

        >

          Add Department

        </Button>

        <AddDepModal

          show={addModalShow}

          onHide={addModalClose}

        />

      </ButtonToolbar>

    </div>

  );

}

export default Department;

import React from 'react';

import { Modal, Button, Row, Col, Form } from 'react-bootstrap';

function EditDepModal(props) {

    const url = 'http://127.0.0.1:8000/'

  const handleSubmit = (event) => {

    // event.preventDefault();

    fetch(`${url}department/`, {

      method: 'PUT',

      headers: {

        Accept: 'application/json',

        'Content-Type': 'application/json',

      },

      body: JSON.stringify({

        DepartmentId: event.target.DepartmentId.value,

        DepartmentName: event.target.DepartmentName.value,

      }),

    })

      .then((res) => res.json())

      .then(

        (result) => {

          alert(result);

        },

        (error) => {

          alert('Failed');

        }

      );

  };

  return (

    <div className="container">

      <Modal

        {...props}

        size="lg"

        aria-labelledby="contained-modal-title-vcenter"

        centered

      >

        <Modal.Header closeButton>

          <Modal.Title id="contained-modal-title-vcenter">

            Edit Department

          </Modal.Title>

        </Modal.Header>

        <Modal.Body>

          <Row>

            <Col sm={6}>

              <Form onSubmit={handleSubmit}>

                <Form.Group controlId="DepartmentId">

                  <Form.Label>DepartmentId</Form.Label>

                  <Form.Control

                    type="text"

                    name="DepartmentId"

                    required

                    disabled

                    defaultValue={props.depid}

                    placeholder="DepartmentId"

                  />

                </Form.Group>

                <Form.Group controlId="DepartmentName">

                  <Form.Label>DepartmentName</Form.Label>

                  <Form.Control

                    type="text"

                    name="DepartmentName"

                    required

                    defaultValue={props.depname}

                    placeholder="DepartmentName"

                  />

                </Form.Group>

                <Form.Group>

                  <Button variant="primary" type="submit">

                    Update Department

                  </Button>

                </Form.Group>

              </Form>

            </Col>

          </Row>

        </Modal.Body>

        <Modal.Footer>

          <Button variant="danger" onClick={props.onHide}>

            Close

          </Button>

        </Modal.Footer>

      </Modal>

    </div>

  );

}

export default EditDepModal;

import React from 'react';

import { Modal, Button, Row, Col, Form } from 'react-bootstrap';

function AddDepModal(props) {

    const url = 'http://127.0.0.1:8000/'

  const handleSubmit = (event) => {

    // event.preventDefault();

    fetch(`${url}department/`, {

      method: 'POST',

      headers: {

        Accept: 'application/json',

        'Content-Type': 'application/json',

      },

      body: JSON.stringify({

        DepartmentId: null,

        DepartmentName: event.target.DepartmentName.value,

      }),

    })

      .then((res) => res.json())

      .then(

        (result) => {

          alert(result);

        },

        (error) => {

          alert('Failed');

        }

      );

  };

  return (

    <div className="container">

      <Modal

        {...props}

        size="lg"

        aria-labelledby="contained-modal-title-vcenter"

        centered

      >

        <Modal.Header closeButton>

          <Modal.Title id="contained-modal-title-vcenter">

            Add Department

          </Modal.Title>

        </Modal.Header>

        <Modal.Body>

          <Row>

            <Col sm={6}>

              <Form onSubmit={handleSubmit}>

                <Form.Group controlId="DepartmentName">

                  <Form.Label>DepartmentName</Form.Label>

                  <Form.Control

                    type="text"

                    name="DepartmentName"

                    required

                    placeholder="DepartmentName"

                  />

                </Form.Group>

                <Form.Group>

                  <Button variant="primary" type="submit">

                    Add Department

                  </Button>

                </Form.Group>

              </Form>

            </Col>

          </Row>

        </Modal.Body>

        <Modal.Footer>

          <Button variant="danger" onClick={props.onHide}>

            Close

          </Button>

        </Modal.Footer>

      </Modal>

    </div>

  );

}

export default AddDepModal;

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

import { Table, Button, ButtonToolbar } from 'react-bootstrap';

import  AddEmpModal  from './AddEmpModal';

import  EditEmpModal  from './EditEmpModal';

function Employee() {

  const [emps, setEmps] = useState([]);

  const [addModalShow, setAddModalShow] = useState(false);

  const [editModalShow, setEditModalShow] = useState(false);

  const [empDetails, setEmpDetails] = useState({

    empid: null,

    empname: '',

    depmt: '',

    photofilename: '',

    doj: '',

  });

  const url = 'http://127.0.0.1:8000/'

  const refreshList = () => {

    fetch(url+'employee')

      .then((response) => response.json())

      .then((data) => {

        setEmps(data);

      });

  };

  useEffect(() => {

    refreshList();

  }, []);

  const deleteEmp = (empid) => {

    if (window.confirm('Are you sure?')) {

      fetch(`${url}employee/${empid}`, {

        method: 'DELETE',

        headers: {

          Accept: 'application/json',

          'Content-Type': 'application/json',

        },

      });

    }

  };

  const editModalClose = () => {

    setEditModalShow(false);

  };

  return (

    <div>

      <Table className="mt-4" striped bordered hover size="sm">

        <thead>

          <tr>

            <th>EmployeeId</th>

            <th>EmployeeName</th>

            <th>Department</th>

            <th>DOJ</th>

            <th>Options</th>

          </tr>

        </thead>

        <tbody>

          {emps.map((emp) => (

            <tr key={emp.EmployeeId}>

              <td>{emp.EmployeeId}</td>

              <td>{emp.EmployeeName}</td>

              <td>{emp.Department}</td>

              <td>{emp.DateOfJoining}</td>

              <td>

                <ButtonToolbar>

                  <Button

                    className="mr-2"

                    variant="info"

                    onClick={() =>

                      setEmpDetails({

                        empid: emp.EmployeeId,

                        empname: emp.EmployeeName,

                        depmt: emp.Department,

                        photofilename: emp.PhotoFileName,

                        doj: emp.DateOfJoining,

                      })

                    }

                  >

                    Edit

                  </Button>

                  <Button

                    className="mr-2"

                    variant="danger"

                    onClick={() => deleteEmp(emp.EmployeeId)}

                  >

                    Delete

                  </Button>

                  <EditEmpModal

                    show={editModalShow}

                    onHide={editModalClose}

                    empid={empDetails.empid}

                    empname={empDetails.empname}

                    depmt={empDetails.depmt}

                    photofilename={empDetails.photofilename}

                    doj={empDetails.doj}

                  />

                </ButtonToolbar>

              </td>

            </tr>

          ))}

        </tbody>

      </Table>

      <ButtonToolbar>

        <Button variant="primary" onClick={() => setAddModalShow(true)}>

          Add Employee

        </Button>

        <AddEmpModal show={addModalShow} onHide={() => setAddModalShow(false)} />

      </ButtonToolbar>

    </div>

  );

}

export default Employee;

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

import { Modal, Button, Row, Col, Form, Image } from 'react-bootstrap';

function EditEmpModal(props) {

  const [deps, setDeps] = useState([]);

//   const [photofilename, setPhotoFilename] = useState('anonymous.png');

//   const [imagesrc, setImageSrc] = useState(

//     process.env.REACT\_APP\_PHOTOPATH + photofilename

//   );

const url = 'http://127.0.0.1:8000/'

  useEffect(() => {

    fetch(url + 'department')

      .then((response) => response.json())

      .then((data) => {

        setDeps(data);

      });

  }, []);

  const handleSubmit = (event) => {

    event.preventDefault();

    fetch(process.env.REACT\_APP\_API + 'employee', {

      method: 'PUT',

      headers: {

        Accept: 'application/json',

        'Content-Type': 'application/json',

      },

      body: JSON.stringify({

        EmployeeId: event.target.EmployeeId.value,

        EmployeeName: event.target.EmployeeName.value,

        Department: event.target.Department.value,

        DateOfJoining: event.target.DateOfJoining.value,

        // PhotoFileName: photofilename,

      }),

    })

      .then((res) => res.json())

      .then(

        (result) => {

          alert(result);

        },

        (error) => {

          alert('Failed');

        }

      );

  };

//   const handleFileSelected = (event) => {

//     event.preventDefault();

//     setPhotoFilename(event.target.files[0].name);

//     const formData = new FormData();

//     formData.append('myFile', event.target.files[0], event.target.files[0].name);

//     fetch(process.env.REACT\_APP\_API + 'Employee/SaveFile', {

//       method: 'POST',

//       body: formData,

//     })

//       .then((res) => res.json())

//       .then(

//         (result) => {

//           setImageSrc(process.env.REACT\_APP\_PHOTOPATH + result);

//         },

//         (error) => {

//           alert('Failed');

//         }

//       );

//   };

  return (

    <div className="container">

      <Modal {...props} size="lg" aria-labelledby="contained-modal-title-vcenter" centered>

        <Modal.Header closeButton>

          <Modal.Title id="contained-modal-title-vcenter">Edit Employee</Modal.Title>

        </Modal.Header>

        <Modal.Body>

          <Row>

            <Col sm={6}>

              <Form onSubmit={handleSubmit}>

                <Form.Group controlId="EmployeeId">

                  <Form.Label>EmployeeId</Form.Label>

                  <Form.Control

                    type="text"

                    name="EmployeeId"

                    required

                    placeholder="EmployeeId"

                    disabled

                    defaultValue={props.empid}

                  />

                </Form.Group>

                <Form.Group controlId="EmployeeName">

                  <Form.Label>EmployeeName</Form.Label>

                  <Form.Control

                    type="text"

                    name="EmployeeName"

                    required

                    defaultValue={props.empname}

                    placeholder="EmployeeName"

                  />

                </Form.Group>

                <Form.Group controlId="Department">

                  <Form.Label>Department</Form.Label>

                  <Form.Control as="select" defaultValue={props.depmt}>

                    {deps.map((dep) => (

                      <option key={dep.DepartmentId}>{dep.DepartmentName}</option>

                    ))}

                  </Form.Control>

                </Form.Group>

                <Form.Group controlId="DateOfJoining">

                  <Form.Label>DateOfJoining</Form.Label>

                  <Form.Control

                    type="date"

                    name="DateOfJoining"

                    required

                    defaultValue={props.doj}

                  />

                </Form.Group>

                <Form.Group>

                  <Button variant="primary" type="submit">

                    Update Employee

                  </Button>

                </Form.Group>

              </Form>

            </Col>

            {/\* <Col sm={6}>

              <Image width="200px" height="200px" src={imagesrc} />

              <input onChange={handleFileSelected} type="File" />

            </Col> \*/}

          </Row>

        </Modal.Body>

        <Modal.Footer>

          <Button variant="danger" onClick={props.onHide}>

            Close

          </Button>

        </Modal.Footer>

      </Modal>

    </div>

  );

}

export default EditEmpModal;

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

import { Modal, Button, Row, Col, Form, Image } from 'react-bootstrap';

function AddEmpModal(props) {

  const [deps, setDeps] = useState([]);

  const [photofilename, setPhotoFilename] = useState('anonymous.png');

//   const [imagesrc, setImageSrc] = useState(

//     process.env.REACT\_APP\_PHOTOPATH + photofilename

//   );

  const url = 'http://127.0.0.1:8000/'

  useEffect(() => {

    fetch(url+ 'department/')

      .then((response) => response.json())

      .then((data) => {

        setDeps(data);

      });

  }, []);

  const handleSubmit = (event) => {

    // event.preventDefault();

    fetch(url+ 'employee/', {

      method: 'POST',

      headers: {

        Accept: 'application/json',

        'Content-Type': 'application/json',

      },

      body: JSON.stringify({

        EmployeeId: null,

        EmployeeName: event.target.EmployeeName.value,

        Department: event.target.Department.value,

        DateOfJoining: event.target.DateOfJoining.value,

        PhotoFileName: photofilename,

      }),

    })

      .then((res) => res.json())

      .then(

        (result) => {

          alert(result);

        },

        (error) => {

          alert('Failed');

        }

      );

  };

//   const handleFileSelected = (event) => {

//     event.preventDefault();

//     const selectedFile = event.target.files[0];

//     setPhotoFilename(selectedFile.name);

//     const formData = new FormData();

//     formData.append('myFile', selectedFile, selectedFile.name);

//     fetch(process.env.REACT\_APP\_API + 'Employee/SaveFile', {

//       method: 'POST',

//       body: formData,

//     })

//       .then((res) => res.json())

//       .then(

//         (result) => {

//           setImageSrc(process.env.REACT\_APP\_PHOTOPATH + result);

//         },

//         (error) => {

//           alert('Failed');

//         }

//       );

//   };

  return (

    <div className="container">

      <Modal

        {...props}

        size="lg"

        aria-labelledby="contained-modal-title-vcenter"

        centered

      >

        <Modal.Header closeButton>

          <Modal.Title id="contained-modal-title-vcenter">

            Add Employee

          </Modal.Title>

        </Modal.Header>

        <Modal.Body>

          <Row>

            <Col sm={6}>

              <Form onSubmit={handleSubmit}>

                <Form.Group controlId="EmployeeName">

                  <Form.Label>EmployeeName</Form.Label>

                  <Form.Control

                    type="text"

                    name="EmployeeName"

                    required

                    placeholder="EmployeeName"

                  />

                </Form.Group>

                <Form.Group controlId="Department">

                  <Form.Label>Department</Form.Label>

                  <Form.Control as="select">

                    {deps.map((dep) => (

                      <option key={dep.DepartmentId}>{dep.DepartmentName}</option>

                    ))}

                  </Form.Control>

                </Form.Group>

                <Form.Group controlId="DateOfJoining">

                  <Form.Label>DateOfJoining</Form.Label>

                  <Form.Control

                    type="date"

                    name="DateOfJoining"

                    required

                    placeholder="DateOfJoining"

                  />

                </Form.Group>

                <Form.Group>

                  <Button variant="primary" type="submit">

                    Add Employee

                  </Button>

                </Form.Group>

              </Form>

            </Col>

            {/\* <Col sm={6}>

              <Image width="200px" height="200px" src={imagesrc} />

              <input onChange={handleFileSelected} type="File" />

            </Col> \*/}

          </Row>

        </Modal.Body>

        <Modal.Footer>

          <Button variant="danger" onClick={props.onHide}>

            Close

          </Button>

        </Modal.Footer>

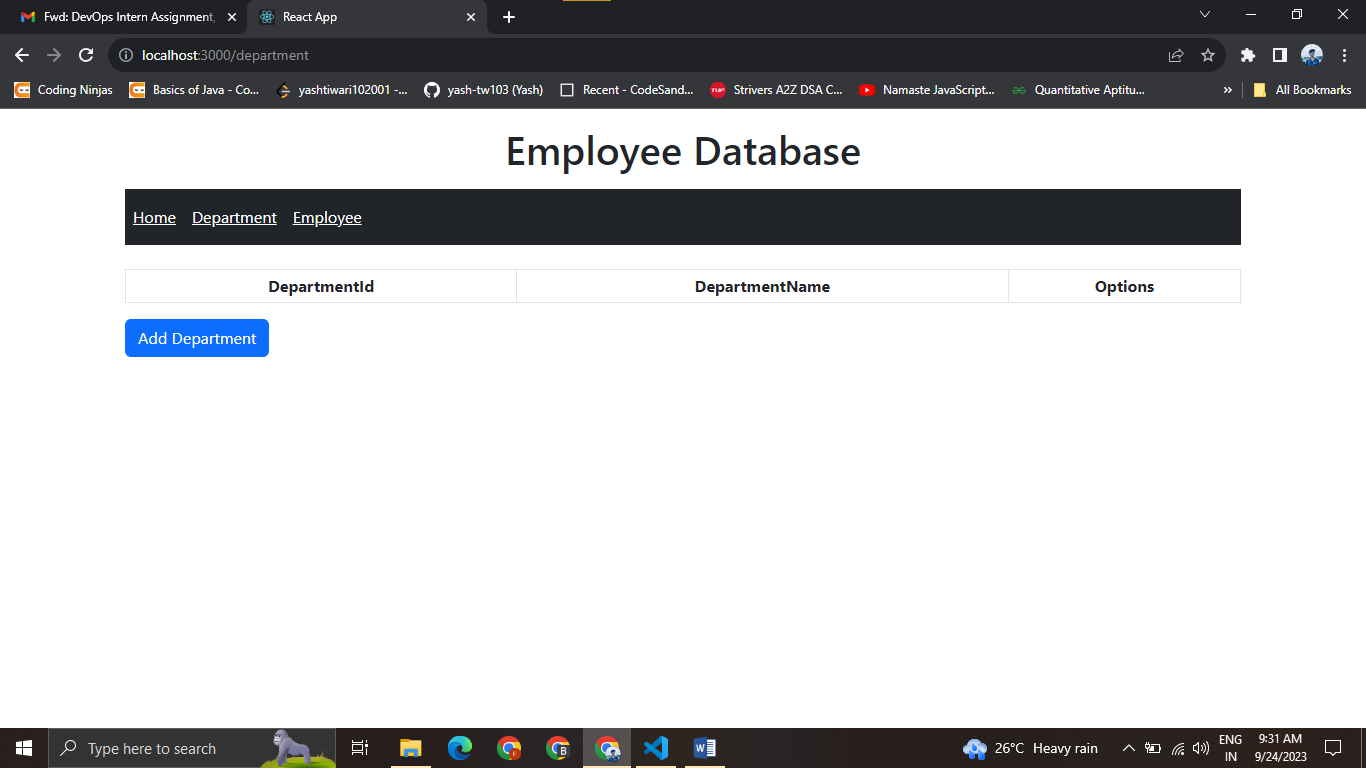
      </Modal>

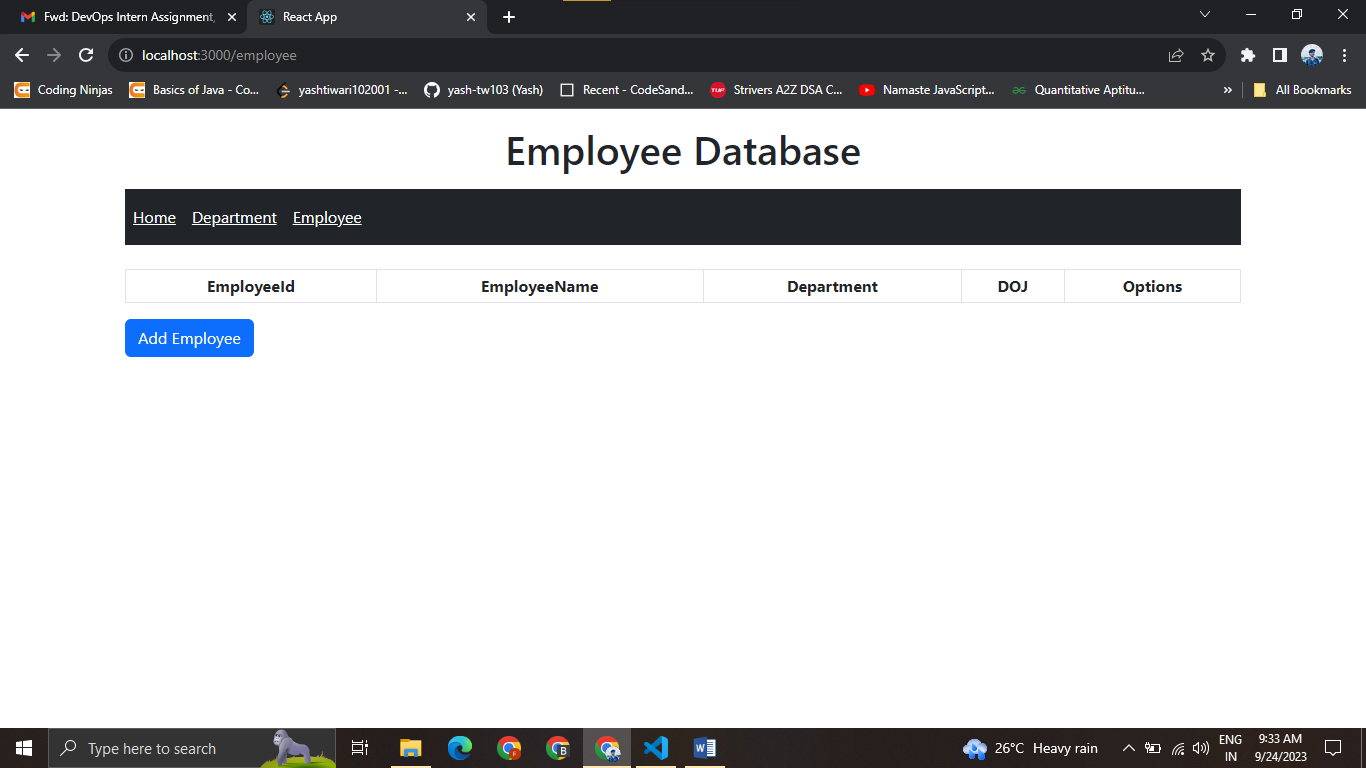
    </div>

  );

}

export default AddEmpModal;

****

****

**When we fetch the data we get**