

BlogController.java

```
package com.blog.blogspringboot;
import java.util.*;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.CrossOrigin;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.RestController;

@RestController
@CrossOrigin(origins="*")
public class BlogController {

    @Autowired
    ArrayList<Blogs> l;

    @GetMapping("/blogs")
    public ResponseEntity<ArrayList<Blogs>> getblogs()
    {
        return new ResponseEntity<ArrayList<Blogs>>(l,HttpStatus.OK);
    }

    @PostMapping("/add")
    public void addblogs(@RequestBody Blogs blog)
    {
        l.add(blog);
    }
}
```

Blogs.java with @Component and @Bean

```
package com.blog.blogspringboot;
import java.util.ArrayList;

import org.springframework.context.annotation.Bean;
import org.springframework.stereotype.Component;

@Component
public class Blogs {
    String name;
    String content;

    public Blogs() {
    }

    public Blogs(String name, String content) {
        this.name = name;
        this.content = content;
    }

    public String getName() {
        return this.name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getContent() {
        return this.content;
    }

    public void setContent(String content) {
        this.content = content;
    }

    @Bean
    public ArrayList<Blogs> l()
    {
        return new ArrayList<Blogs>();
    }
}
```

CorsConfig.java

```
package com.blog.blogspringboot;

import org.springframework.context.annotation.Configuration;
import org.springframework.web.servlet.config.annotation.CorsRegistry;
import org.springframework.web.servlet.config.annotation.WebMvcConfigurer;
@Configuration
public class CorsConfig implements WebMvcConfigurer{

    @Override
    public void addCorsMappings(CorsRegistry registry)
    {
        registry.addMapping("/**")
            .allowedOrigins("*")
            .allowedMethods("GET", "POST", "PUT", "DELETE");
    }

}
```

App.js with routing

```
import logo from './logo.svg';
import './App.css';
import {BrowserRouter as Router, Routes, Route, Link} from 'react-router-dom'
import Home from './Home'
import Blogs from './Blogs'

function App() {
  return (
    <div className="App">

      <Router>
      <nav>
        <li><Link to="/">Home</Link></li>

        <li><Link to="/blogs">Insert</Link></li>

      </nav>

      <Routes>
        <Route path="/" element={<Home/>}/>
        <Route path="/blogs" element={<Blogs/>}/>
      </Routes>

    </Router>
    </div>
  );
}

export default App;
```

Home.js with fetching

```
import React, { useEffect,useState } from 'react'
import './Home.css'

export default function Home() {

  const [blogs,setblogs] = useState([])

  useEffect(() => {
    fetch('http://localhost:8080/blogs')
      .then((res)=>res.json())
      .then((data) => {
        setblogs(data)
      })
      .catch((err) => {
        console.log(err.message)
      })
  })
  return (
    <div>

      <p id='Posts'>
        Posts
      </p>
      {
        blogs.map((blog) => {
          return(
            <div id='container' key={blog.id}>
              <p id='title'>{blog.name}</p>
              <p id='content'> {blog.content}</p>
            </div>
          );
        })
      }

    </div>
  )
}
```

Blogs.js with axios.post

```
import axios from 'axios'
import React, { useState } from 'react'
import './Blogs.css'
export default function Blogs() {

  const [name,setname] = useState('')
  const [content,setcontent] = useState('')

  const handlesubmit = (e) => {
    axios.post('http://localhost:8080/add',{name:name,content
:content})
    .catch((err)=>{console.log(err.message)})
  }
  return (
    <div>

      <p id='add'>
        Add New Content
      </p>
      <form id='form'>
        <input type='text' placeholder='Title' id='title1'
onChange={(event)=>{setname(event.target.value)}}/>
        <textarea placeholder='Content' id='content1'
onChange={(event) =>{setcontent(event.target.value)}}/>
        <input type='submit' id='submit'
onClick={handlesubmit}/>
      </form>
    </div>
  )
}
```