

1. Introduction

What is React?



JavaScript library used to create websites



Allows us to easily create **Single Page Apps**
- **SPA's** for short

Single Page Applications



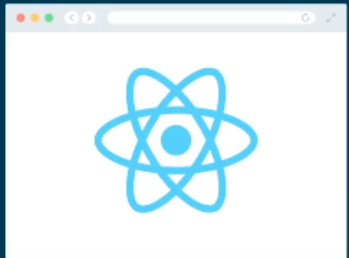
initial request

single HTML page
(index.html)

Server



Single Page Applications



index.html

Server

Single Page Applications

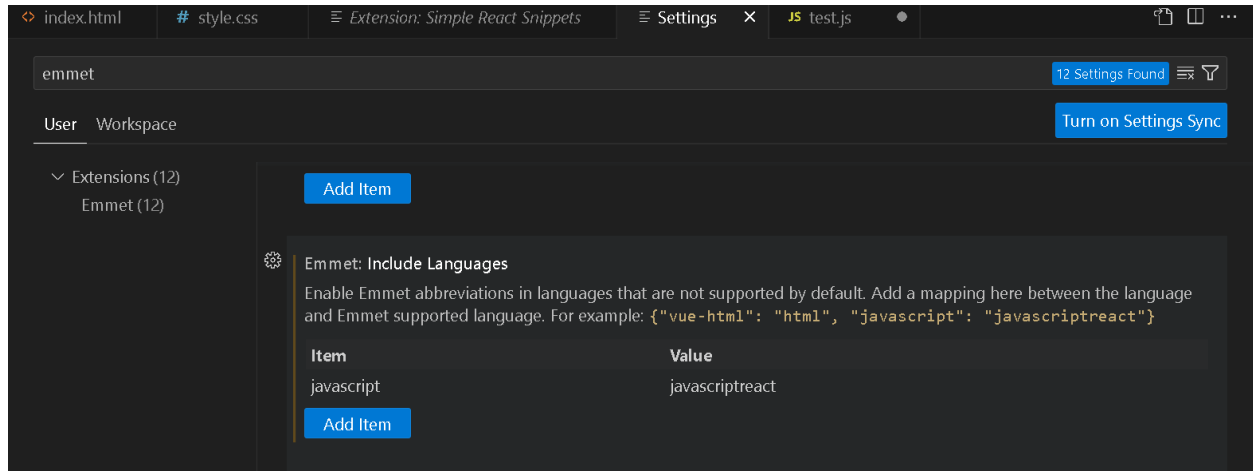


request to /about

about.html

Server

→Go to “Settings” type emmet →”Include Language” add item →**Item: javascript & Value: javascriptreact**



2. Creating a React Application

```
C:\Users\gudiw>node -v  
v20.3.0
```

```
C:\Users\gudiw>cd documents
```

```
C:\Users\gudiw\Documents>cd tuts
```

```
C:\Users\gudiw\Documents\tuts>npx create-react-app dojo-blog
```

Need to install the following packages:

```
create-react-app@5.0.1
```

Ok to proceed? (y) y

npm WARN deprecated tar@2.2.2: This version of tar is no longer supported, and will not receive security updates. Please upgrade asap.

Creating a new React app in C:\Users\gudiw\Documents\tuts\dojo-blog.

Installing packages. This might take a couple of minutes.

Installing react, react-dom, and react-scripts with cra-template...

added 1422 packages in 7m

226 packages are looking for funding

run `npm fund` for details

Initialized a git repository.

Installing template dependencies using npm...

added 74 packages, and changed 1 package in 29s

235 packages are looking for funding

run `npm fund` for details

Removing template package using npm...

removed 1 package, and audited 1496 packages in 10s

235 packages are looking for funding

run `npm fund` for details

74 vulnerabilities (69 moderate, 5 high)

To address issues that do not require attention, run:

npm audit fix

To address all issues (including breaking changes), run:

npm audit fix --force

Run `npm audit` for details.

Created git commit.

Success! Created dojo-blog at C:\Users\gudiw\Documents\tuts\dojo-blog
Inside that directory, you can run several commands:

`npm start`

Starts the development server.

`npm run build`

Bundles the app into static files for production.

`npm test`

Starts the test runner.

`npm run eject`

Removes this tool and copies build dependencies, configuration files
and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

`cd dojo-blog`

`npm start`

Happy hacking!

C:\Users\gudiw\Documents\tuts>**cd dojo-blog**

C:\Users\gudiw\Documents\tuts\dojo-blog>**code .** (open VS code)

C:\Users\gudiw\Documents\tuts\dojo-blog>

VS Code

node_modules --> All Installed Packages available

99% code in the **"src"** folder

--> Suppose if you download project from github **"node_module"** folder is not available then code it won't execute then we need to install the packages --> **"npm install"**

PS C:\Users\gudiw\Documents\tuts\dojo-blog> **npm run start**

Starting the development server...

One of your dependencies, babel-preset-react-app, is importing the "@babel/plugin-proposal-private-property-in-object" package without declaring it in its dependencies. This is currently working because "@babel/plugin-proposal-private-property-in-object" is already in your Compiled successfully!

You can now view dojo-blog in the browser.

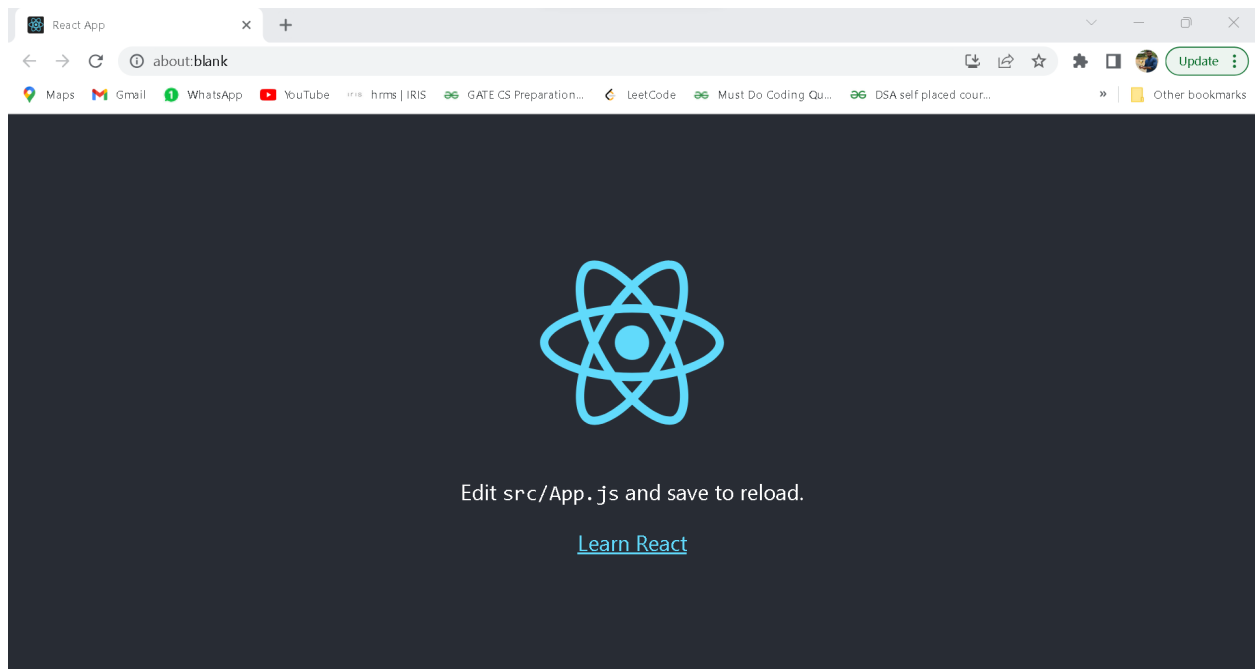
Local: **http://localhost:3000**

On Your Network: **http://192.168.244.65:3000**

Note that the development build is not optimized.
To create a production build, use npm run build.

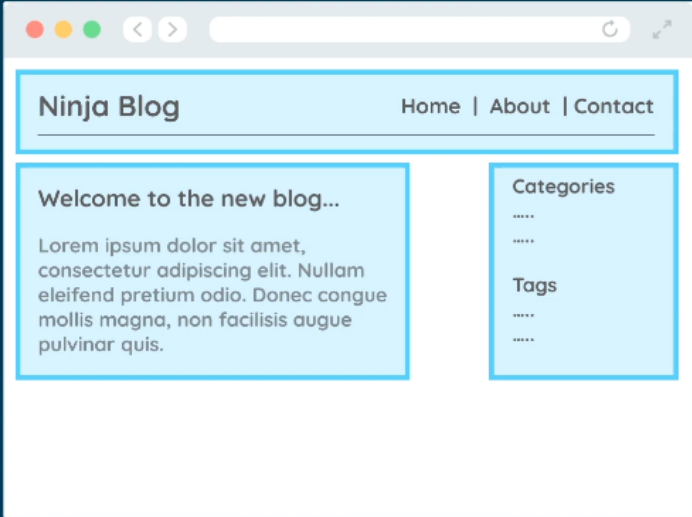
webpack compiled successfully

O/P:



3. Components & Templates

React Components



Components contain:

- Template
- Logic

“className” is jsx in “App.js” In Html that is “class”.

App.js

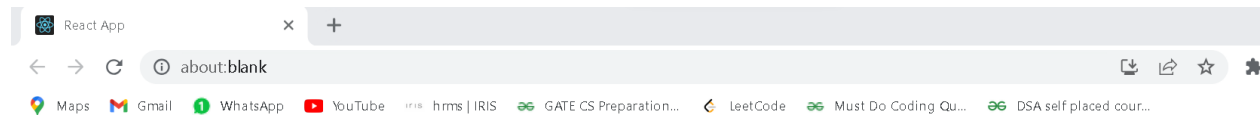
```
import './App.css';

function App() {
  return (
    <div className="App">
      <div className="content">
        <h1>App Component</h1>
      </div>
    </div>
  );
}
```



```
}  
  
export default App;
```

O/P:



App Component

4. Dynamic Values in Templates

App.js

```
import './App.css';  
  
function App() {  
  const title = 'Welcome to the new blog';  
  const likes = 50;  
  const link = "http://www.google.com";  
  
  return (  
    <div className="App">  
      <div className="content">  
        <h1>{ title }</h1>  

```

```

    <p>liked { likes } times</p>

    <p>{ 10 }</p>
    <p>{ "hello, ninjas" }</p>
    <p>{ [1,2,3,4,5] }</p>
    <p>{ Math.random() * 10 }</p>

    <a href={link}>Google Site</a>
  </div>
</div>
);
}

export default App;

```

O/P:



Wlecome to the new blog

liked 50 times

10

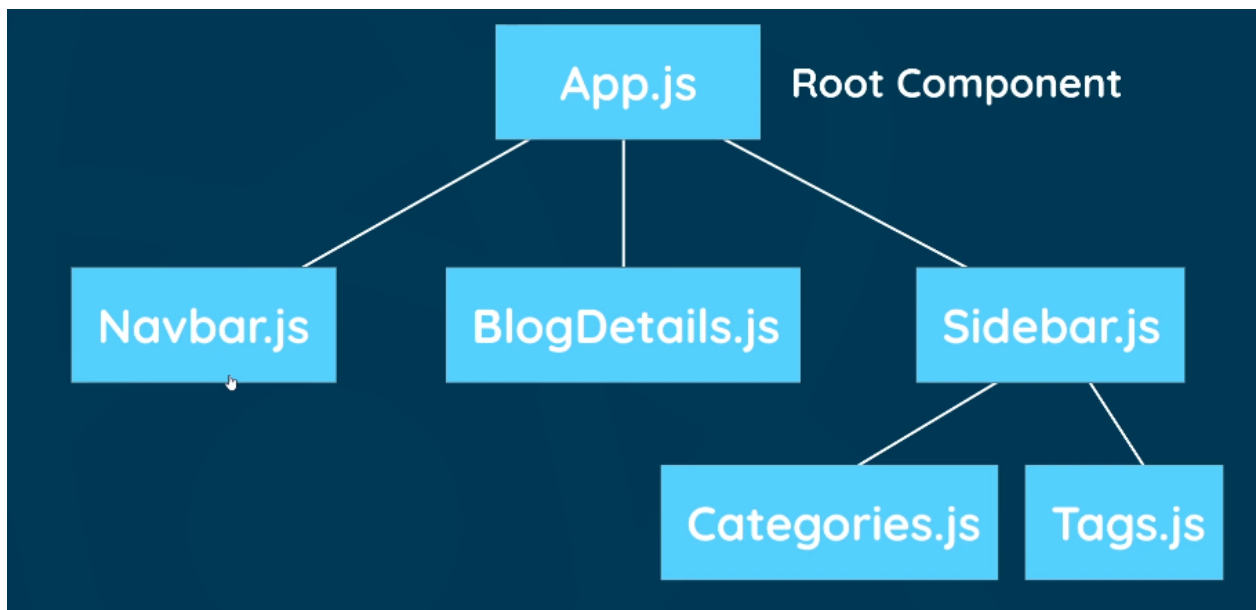
hello, ninjas

12345

5.14137339565556

[Google Site](#)

5. Multiple Components



sfc → **Stateless Function Component**

→ if you type “**sfc**” it will give the below code

```
const = () => {  
  return ( );  
}  
  
export default ;
```

App.js

```
import './App.css';  
import Navbar from './Navbar';  
import Home from './Home';
```

```
function App() {  
  return (  
    <div className="App">  
      <Navbar></Navbar>  
      <div className="content">  
        <Home></Home>  
      </div>  
    </div>  
  );  
}  
  
export default App;
```

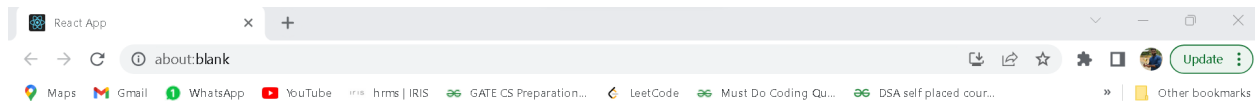
Navbar.js

```
const Navbar = () => {  
  return (  
    <nav className="navbar">  
      <h1>The Dojo Blog</h1>  
      <div className="links">  
        <a href="/">Home</a>  
        <a href="/create">New Blog</a>  
      </div>  
    </nav>  
  );  
}  
  
export default Navbar;
```

Home.js

```
const Home = () => {  
  return (  
    <div className="home">  
      <h2>Homepage</h2>  
    </div>  
  );  
}  
  
export default Home;
```

O/P:



6. Adding Styles

Navbar.js

```
const Navbar = () => {  
  return (  
    <nav className="navbar">  
      <h1>The Dojo Blog</h1>  
      <div className="links">  
        <a href="/">Home</a>  
        <a href="/create" style={{
```

```

        color: "white",
        backgroundColor: '#f1356d',
        borderRadius: '8px'
      }}>New Blog</a>
    </div>
  </nav>
);
}

export default Navbar;

```

index.css

```

@import
url('https://fonts.googleapis.com/css2?family=Quicksand:wght@300;400;500;600;700&display=swap');

/* base styles */
* {
  margin: 0;
  font-family: "Quicksand";
  color: #333;
}

.navbar {
  padding: 20px;
  display: flex;
  align-items: center;
  max-width: 600px;
  margin: 0 auto;
  border-bottom: 1px solid #f2f2f2;
}

.navbar h1 {
  color: #f1356d;
}

```

```

}

.navbar .links {
  margin-left: auto;
}

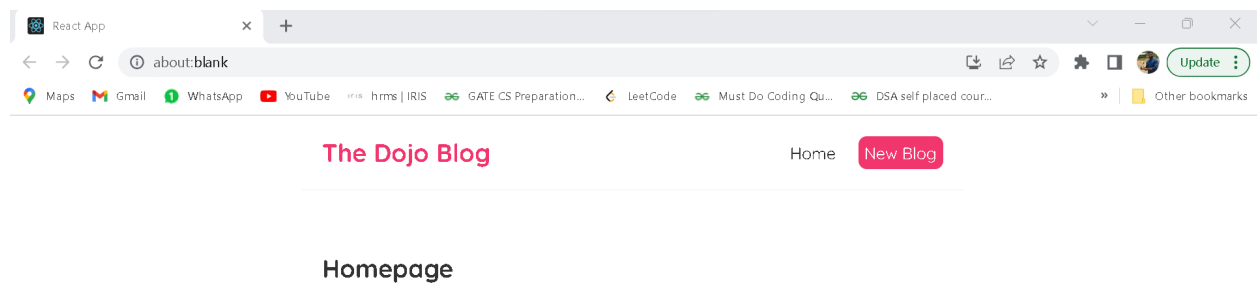
.navbar a {
  margin-left: 16px;
  text-decoration: none;
  padding: 6px;
}

.navbar a:hover {
  color: #f1356d;
}

.content {
  max-width: 600px;
  margin: 40px auto;
  padding: 20px;
}

```

O/P:



7. Click Events

Home.js

```
const Home = () => {

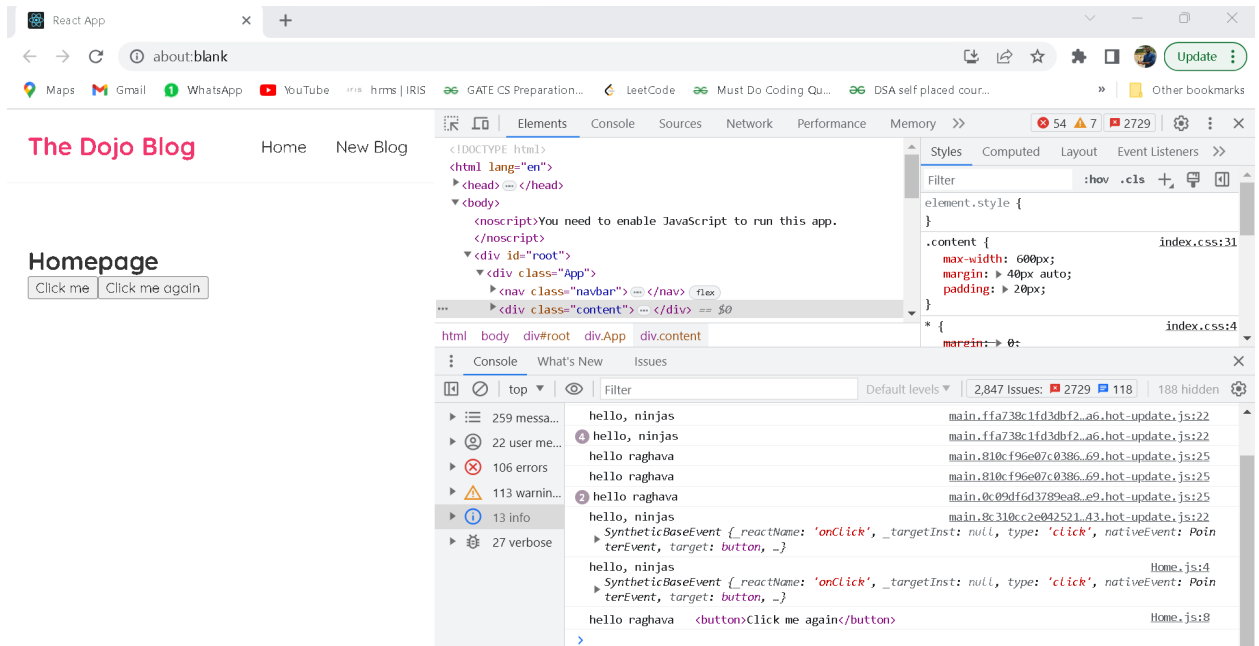
  const handleClick = (e) => {
    console.log('hello, ninjas', e);
  }

  const handleClickAgain = (name, e) => {
    console.log('hello ' + name, e.target);
  }

  return (
    <div className="home">
      <h2>Homepage</h2>
      <button onClick={handleClick}>Click me</button>
      <button onClick={(e) => handleClickAgain('raghava', e)}>Click me
again</button>
    </div>
  );
}

export default Home;
```


O/P:



8. Using State(useState hook)

Home.js

```
import { useState } from "react";

const Home = () => {
  // let name = 'raghava';
  const [name, setName] = useState('raghava');
  const [age, setAge] = useState(25);

  const handleClick = () => {
    //name = 'Gudiwada'
    //console.log(name);
    setName('Priya');
    setAge(30);
  }
}
```

```

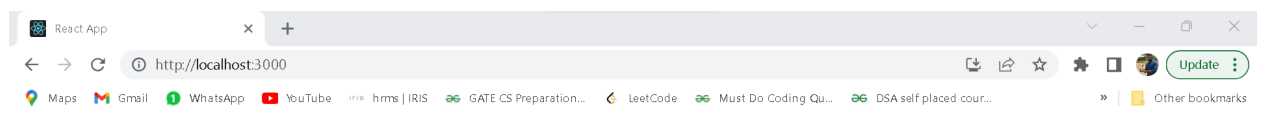
    }

    return (
      <div className="home">
        <h2>Homepage</h2>
        <p>{ name } is { age } years old</p>
        <button onClick={handleClick}>Click
me</button>
      </div>
    );
  }

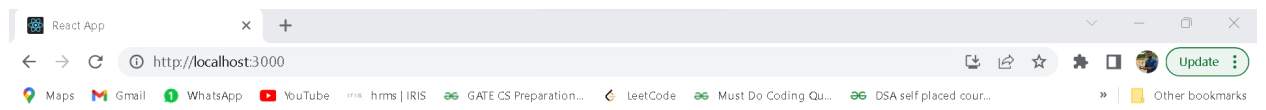
export default Home;

```

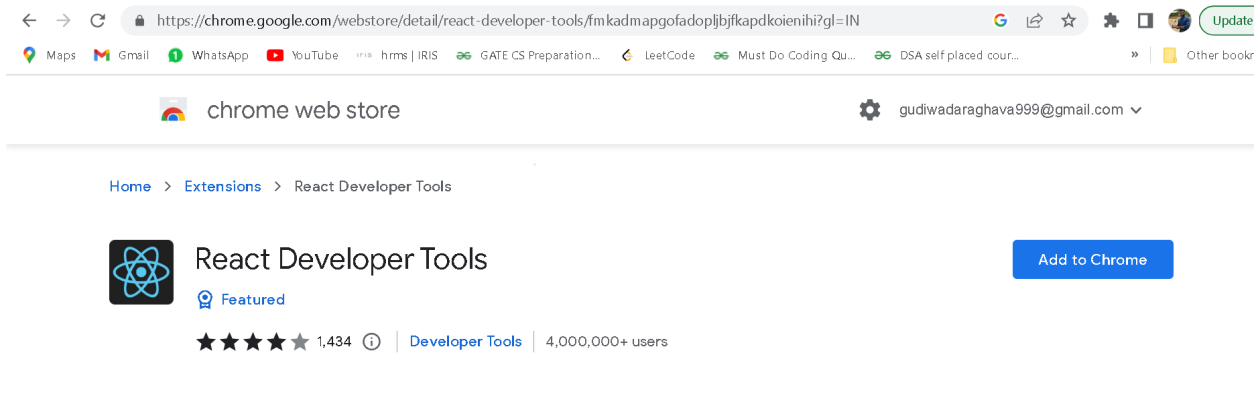
O/P:



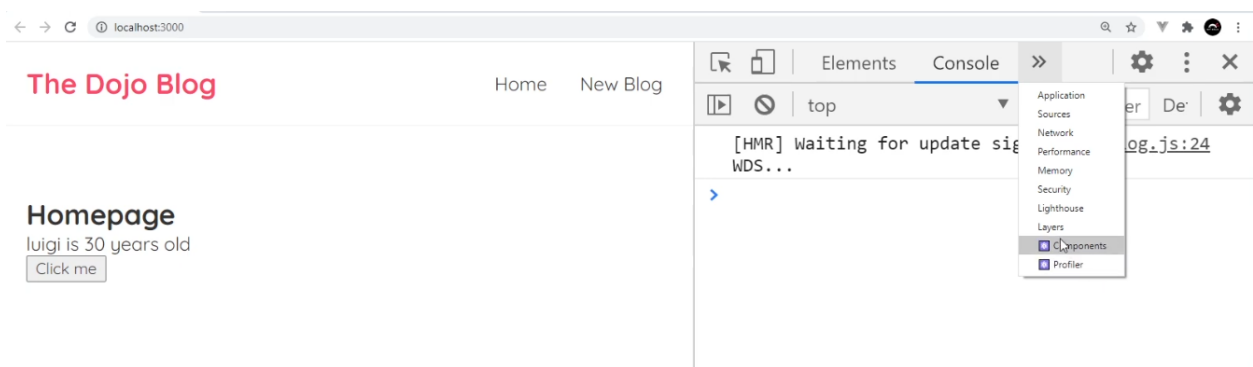
→After Click





9. Intro to React Dev Tools





→ React Dev Tool we are getting extra tabs








Elements


 Components


















Search (text or /regex/)









▼ App

Navbar

Home

App



props

new entry: ""


rendered by

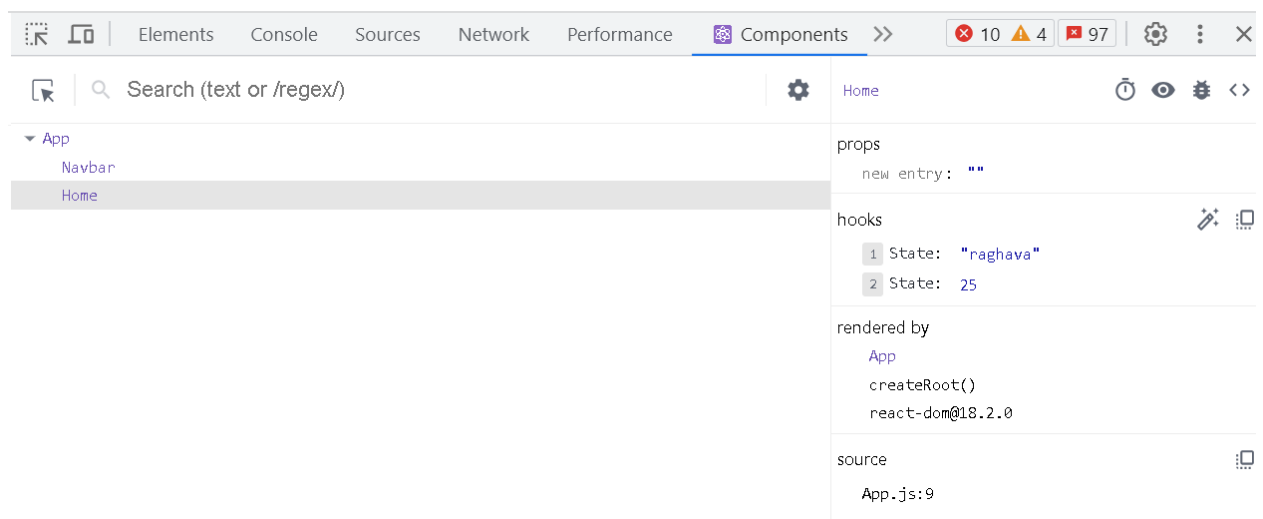
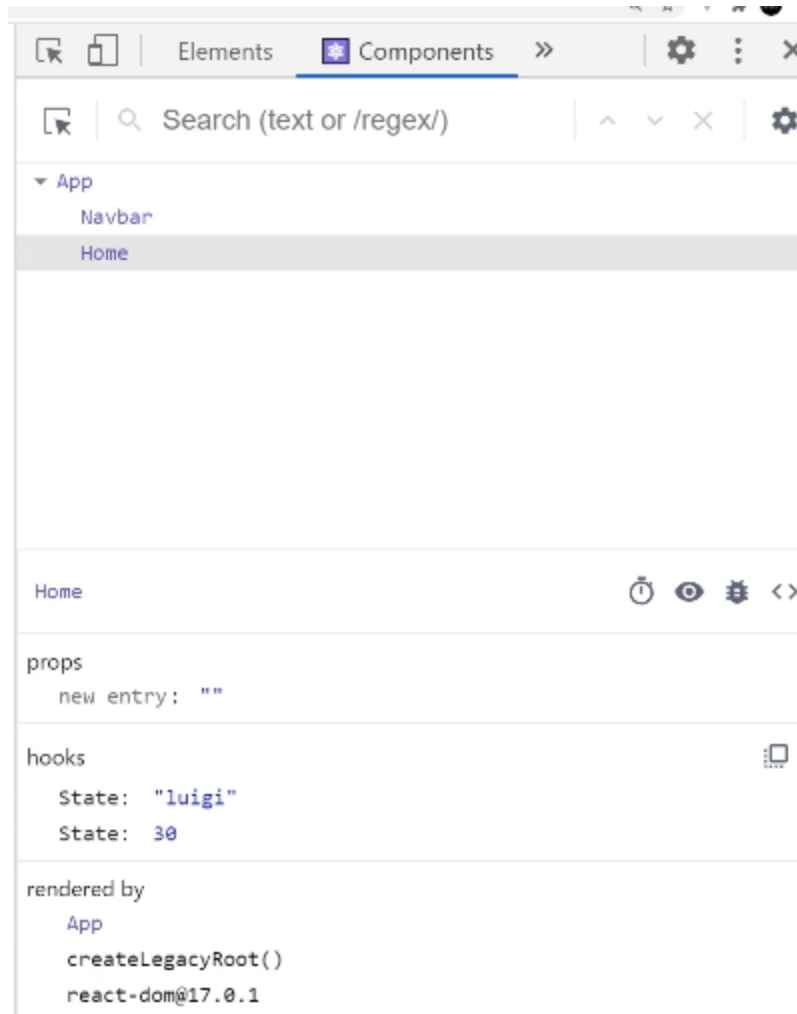
createLegacyRoot()

react-dom@17.0.1

source

src/index.js:8





→After Click

The screenshot shows a web browser at `http://localhost:3000` displaying 'The Dojo Blog' homepage. The page has a header with 'Home' and 'New Blog' links. The main content area shows 'Homepage' with the text 'Priya is 30 years old' and a 'Click me' button. The React DevTools Components panel is open on the right, showing the component tree with 'App' at the root, containing 'Navbar' and 'Home'. The 'Home' component's props are shown as `new entry: ''`. The hooks section shows two hooks: `State: 'Priya'` and `State: 30`. The rendered by section shows `App` and `createRoot()` from `react-dom@18.2.0`. The source section shows `App.js:9`.

The screenshot shows the React DevTools Console panel. The left sidebar shows 41 messages, 10 user messages, 18 errors, 8 warnings, 10 info, and 5 verbose. The main console area shows a log for `[Click to expand] <Home />` from `react_devtools_backend_compact.js:7937`. The log shows the props as `{}` and the hooks as `{id: 1, isStateEditable: true, name: 'State', value: 30, subHooks: Array(0), ...}`. The log also shows the nodes as `[div.home]` from `react_devtools_backend_compact.js:7956`. The location is `{fileName: 'C:\\Users\\gudiw\\Documents\\tuts\\dojo-blog\\src\\App.js', lineNumber: 9, columnNumber: 9}` from `react_devtools_backend_compact.js:7960`. A message at the bottom says 'Right-click any value to save it as a global variable for further inspection.' from `react_devtools_backend_compact.js:7964`.

10. Outputting Lists

Home.js

```
import { useState } from "react";

const Home = () => {
  const [blogs, setBlogs] = useState([
    { title: 'My new website', body: 'lorem ipsum...',
author: 'raghava', id: 1 },
    { title: 'Welcome party!', body: 'lorem ipsum...',
author: 'joshi', id: 2 },
    { title: 'Web dev top tips', body: 'lorem ipsum...',
author: 'Harini', id: 3 }
  ]);

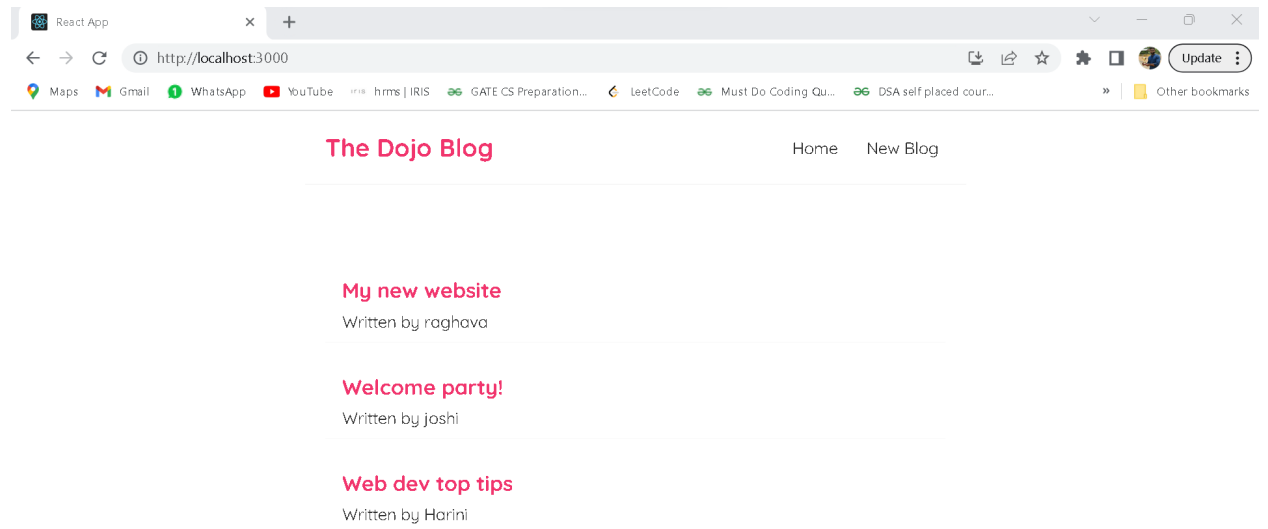
  return (
    <div className="home">
      {blogs.map((blog) => (
        <div className="blog-preview" key={blog.id}>
          <h2>{ blog.title }</h2>
          <p>Written by { blog.author }</p>
        </div>
      ))}
    </div>
  );
}

export default Home;
```

index.css

```
/* blog previews / list */
.blog-preview {
  padding: 10px 16px;
  margin: 20px 0;
  border-bottom: 1px solid #fafafa;
}
.blog-preview:hover {
  box-shadow: 1px 3px 5px rgba(0,0,0,0.1);
}
.blog-preview h2 {
  font-size: 20px;
  color: #f1356d;
  margin-bottom: 8px;
}
```

O/P:



11. Props

In React, "props" is a short form of "properties" and refers to a mechanism for passing data from a parent component to a child component.

Home.js

```
import { useState } from "react";
import BlogList from "../BlogList";

const Home = () => {
  const [blogs, setBlogs] = useState([
    { title: 'My new website', body: 'lorem ipsum...',
author: 'raghava', id: 1 },
    { title: 'Welcome party!', body: 'lorem ipsum...',
author: 'joshi', id: 2 },
    { title: 'Web dev top tips', body: 'lorem ipsum...',
author: 'Harini', id: 3 }
  ]);

  return (
    <div className="home">
      <BlogList blogs= {blogs} title="All
Blogs!"></BlogList>
    </div>
  );
}

export default Home;
```

BlogList.js

```
const BlogList = ({ blogs, title }) => {  
  //const BlogList = (props) => {  
    // const blogs = props.blogs;  
    // const title = props.title;  
  
    return (  
      <div className="blog-list">  
        <h2>{ title }</h2>  
        {blogs.map((blog) => (  
          <div className="blog-preview" key={blog.id}>  
            <h2>{ blog.title }</h2>  
            <p>Written by { blog.author }</p>  
          </div>  
        ))}  
      </div>  
    );  
  }  
}  
  
export default BlogList;
```

O/P:

The screenshot shows a web browser at `http://localhost:3000` displaying a blog titled "The Dojo Blog". The blog has three posts: "My new website" by raghava, "Welcome party!" by joshi, and "Web dev top tips" by Harini. The Redux DevTools console is open, showing the state of the application. The state is an object with a `blogs` array containing three blog entries. The first entry is for "My new website" by raghava, the second is for "Welcome party!" by joshi, and the third is for "Web dev top tips" by Harini. The console also shows the `BlogList.js:4` and `react_devtools_backend_compact.js:2421` logs.

The screenshot shows a web browser at `http://localhost:3000` displaying a blog titled "The Dojo Blog". The blog has a navigation bar with "Home" and "New Blog" links. Below the navigation bar, there is a section titled "All Blogs!". Under this section, there are three blog entries: "My new website" by raghava, "Welcome party!" by joshi, and "Web dev top tips" by Harini.

12. Reusing Components

Home.js

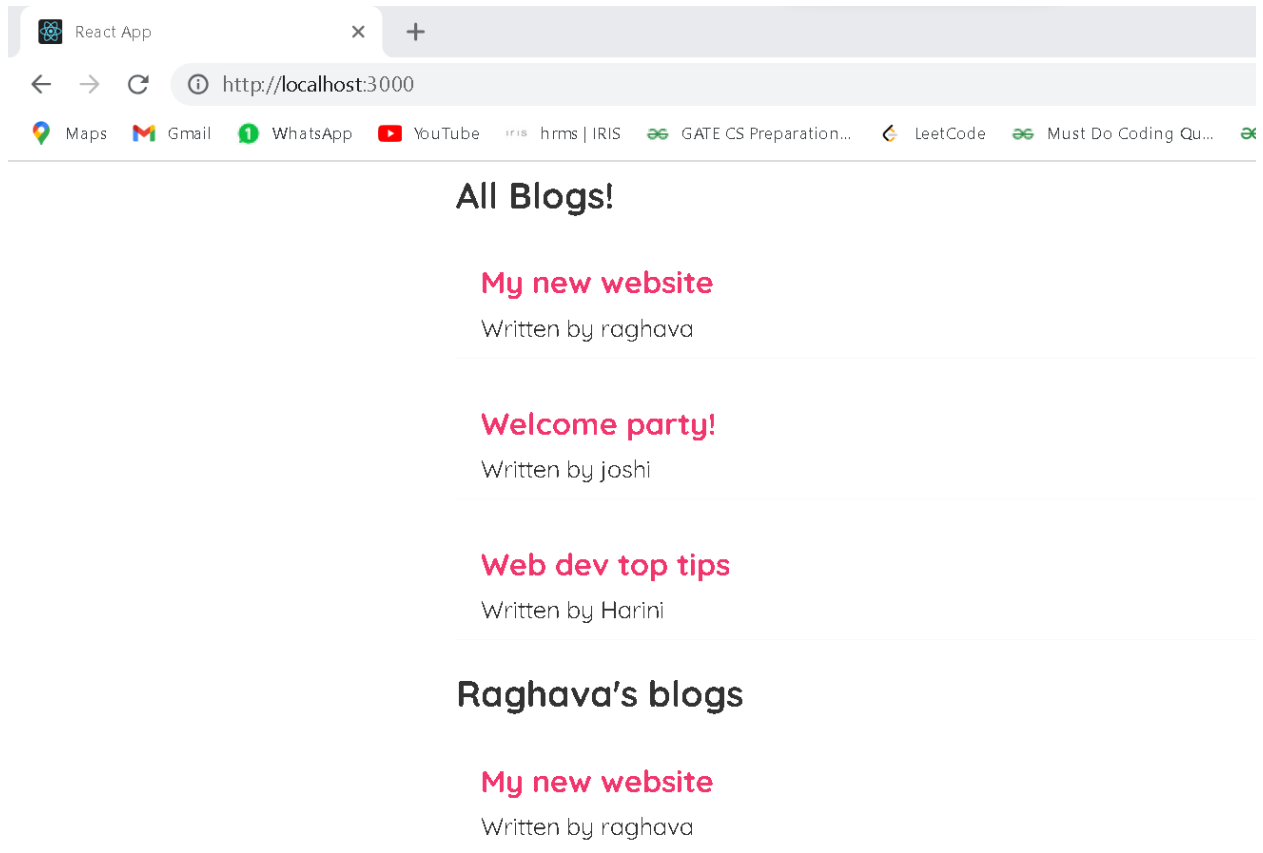
```
import { useState } from "react";
import BlogList from "./BlogList";

const Home = () => {
  const [blogs, setBlogs] = useState([
    { title: 'My new website', body: 'lorem ipsum...', author: 'raghava', id: 1 },
    { title: 'Welcome party!', body: 'lorem ipsum...', author: 'joshi', id: 2 },
    { title: 'Web dev top tips', body: 'lorem ipsum...', author: 'Harini', id: 3 }
  ]);

  return (
    <div className="home">
      <BlogList blogs= {blogs} title="All Blogs!"></BlogList>
      <BlogList blogs= {blogs.filter((blog) => blog.author === 'raghava')} title="Raghava's
blogs"></BlogList>
    </div>
  );
}

export default Home;
```

O/P:



13. Functions as Props

Home.js

```
import { useState } from "react";
import BlogList from "../BlogList";

const Home = () => {
  const [blogs, setBlogs] = useState([
    { title: 'My new website', body: 'lorem ipsum...',
author: 'raghava', id: 1 },
    { title: 'Welcome party!', body: 'lorem ipsum...',
author: 'joshi', id: 2 },
    { title: 'Web dev top tips', body: 'lorem ipsum...',
author: 'Harini', id: 3 }
  ]);

  const handleDelete = (id) => {
    const newBlogs = blogs.filter(blog => blog.id !==
id);
    setBlogs(newBlogs);
  }

  return (
    <div className="home">
      <BlogList blogs= {blogs} title="All Blogs!"
handleDelete={handleDelete}></BlogList>
    </div>
  );
}

export default Home;
```

BlogList.js

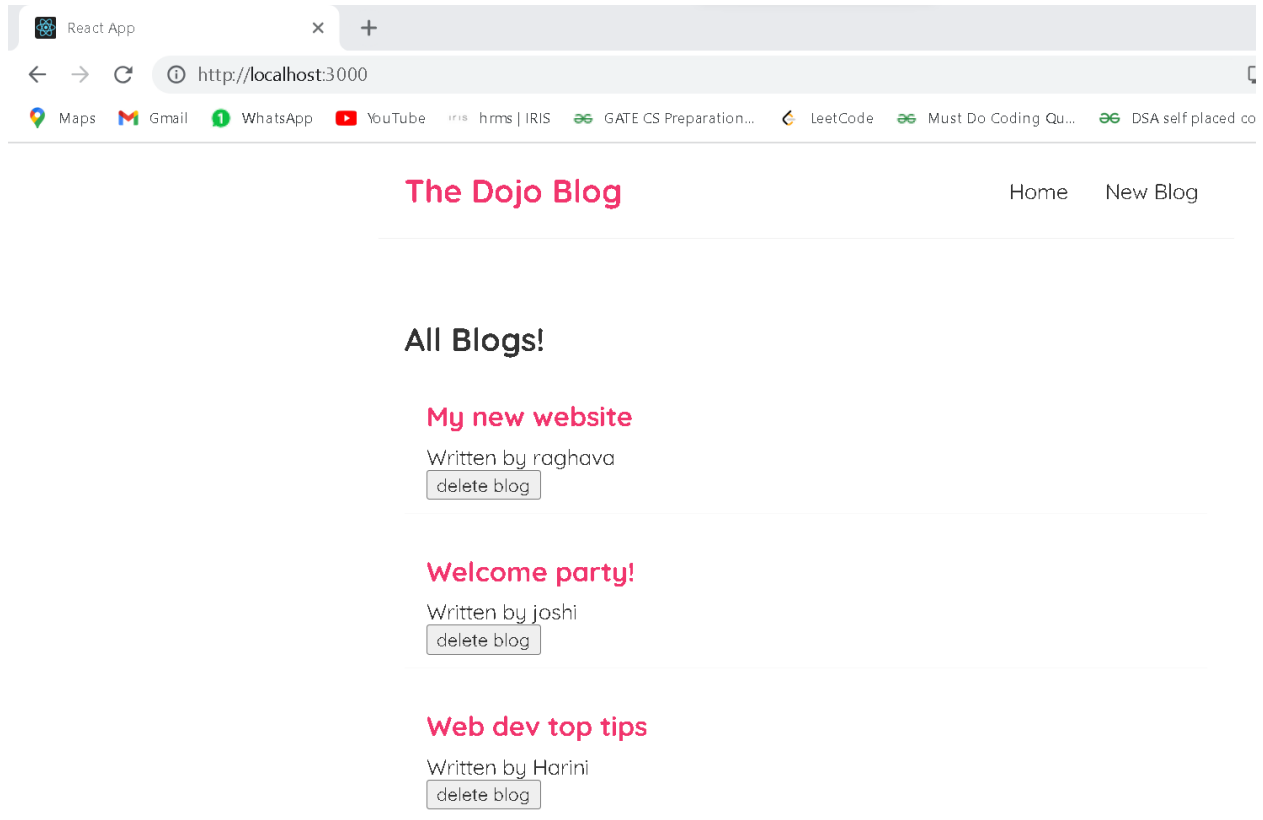
```
const BlogList = ({ blogs, title, handleDelete }) => {

  return (
    <div className="blog-list">
      <h2>{ title }</h2>
      {blogs.map((blog) => (
        <div className="blog-preview" key={blog.id}>
          <h2>{ blog.title }</h2>
          <p>Written by { blog.author }</p>
          <button onClick={() =>
handleDelete(blog.id)}>delete blog</button>
        </div>
      ))}
    </div>
  );
}

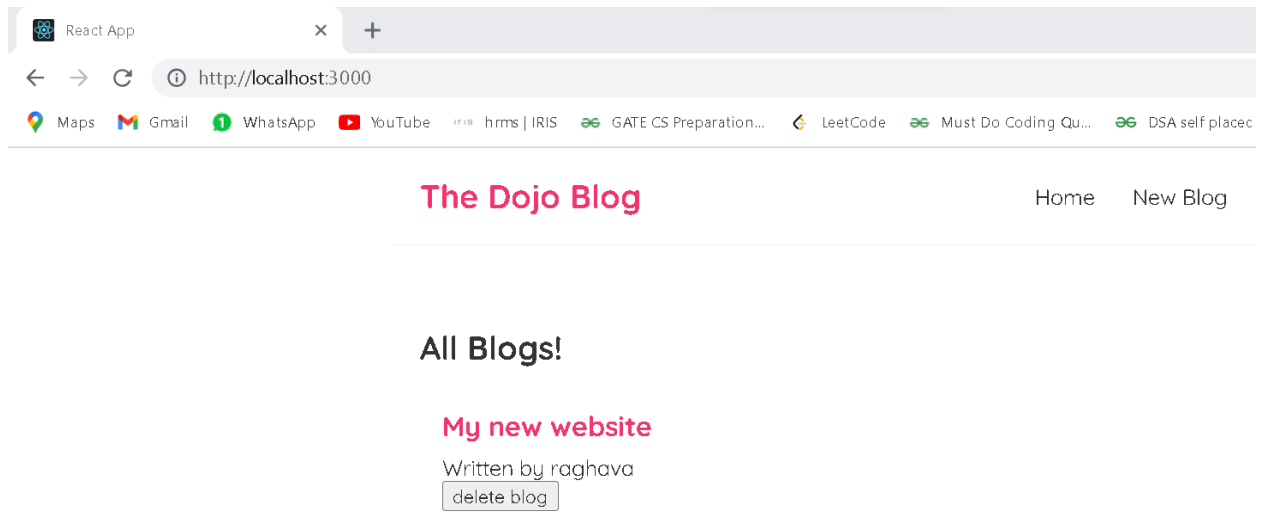
export default BlogList;
```

O/P:

→Initially



→After Deletion



14. useEffect Hook(the basics)

Home.js

```
import { useState, useEffect } from "react";
import BlogList from "../BlogList";

const Home = () => {
  const [blogs, setBlogs] = useState([
    { title: 'My new website', body: 'lorem ipsum...',
author: 'raghava', id: 1 },
    { title: 'Welcome party!', body: 'lorem ipsum...',
author: 'joshi', id: 2 },
    { title: 'Web dev top tips', body: 'lorem ipsum...',
author: 'Harini', id: 3 }
  ]);

  const handleDelete = (id) => {
```

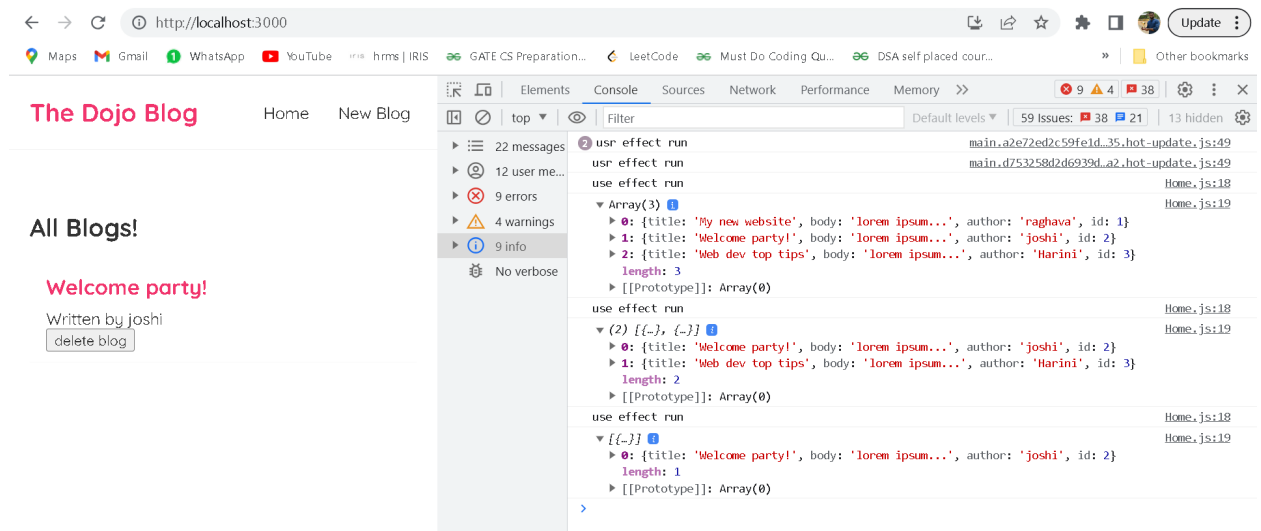
```
    const newBlogs = blogs.filter(blog => blog.id !==
id);
    setBlogs(newBlogs);
  }

  //This function fires for every vendor
  useEffect(() =>{
    console.log('use effect run');
    console.log(blogs);
  });

  return (
    <div className="home">
      <BlogList blogs= {blogs} title="All Blogs!"
handleDelete={handleDelete}></BlogList>
    </div>
  );
}

export default Home;
```

O/P:



15. useEffect Dependencies

Home.js

```
import { useState, useEffect } from "react";
import BlogList from "./BlogList";

const Home = () => {
  const [blogs, setBlogs] = useState([
    { title: 'My new website', body: 'lorem ipsum...',
author: 'raghava', id: 1 },
    { title: 'Welcome party!', body: 'lorem ipsum...',
author: 'joshi', id: 2 },
    { title: 'Web dev top tips', body: 'lorem ipsum...',
author: 'Harini', id: 3 }
  ]);

  const [name, setName] = useState('raghava');
```

```

    const handleDelete = (id) => {
      const newBlogs = blogs.filter(blog => blog.id !==
id);
      setBlogs(newBlogs);
    }

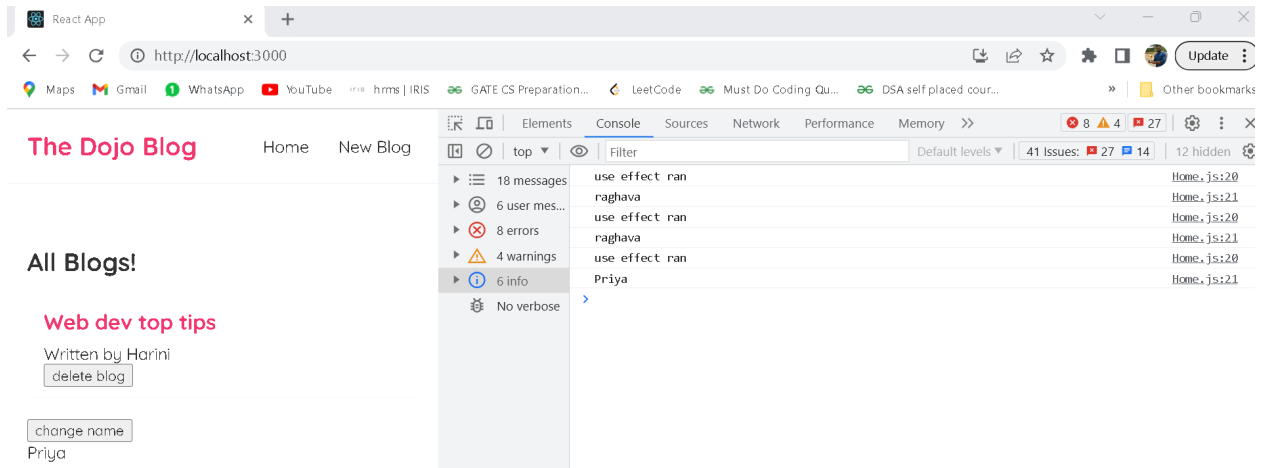
    useEffect(() =>{
      console.log('use effect ran');
      console.log(name);
    }, [name]);

    return (
      <div className="home">
        <BlogList blogs= {blogs} title="All Blogs!"
handleDelete={handleDelete}></BlogList>
        <button onClick={ () =>
setName('Priya')}>change name</button>
        <p>{ name }</p>
      </div>
    );
  }

export default Home;

```

O/P:



16. Using JSON Server

→Data Folder →db.json

db.json

```
{
  "blogs": [
    {
      "title": "My First Blog",
      "body": "Why do we use it?\nIt is a long established
fact that a reader will be distracted by the readable content
of a page when looking at its layout. The point of using Lorem
Ipsum is that it has a more-or-less normal distribution of
letters, as opposed to using 'Content here, content here',
making it look like readable English. Many desktop publishing
packages and web page editors now use Lorem Ipsum as their
default model text, and a search for 'lorem ipsum' will
uncover many web sites still in their infancy. Various
```

versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).

Where does it come from?

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, *consectetur*, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "*de Finibus Bonorum et Malorum*" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, "Lorem ipsum dolor sit amet..", comes from a line in section 1.10.32.

The standard chunk of Lorem Ipsum used since the 1500s is reproduced below for those interested. Sections 1.10.32 and 1.10.33 from "*de Finibus Bonorum et Malorum*" by Cicero are also reproduced in their exact original form, accompanied by English versions from the 1914 translation by H. Rackham.

Where can I get some?

There are many variations of passages of Lorem Ipsum available, but the majority have suffered alteration in some form, by injected humour, or randomised words which don't look even slightly believable. If you are going to use a passage of Lorem Ipsum, you need to be sure there isn't anything embarrassing hidden in the middle of text. All the Lorem Ipsum generators on the Internet tend to repeat predefined chunks as necessary, making this the first true generator on the Internet. It uses a dictionary of over 200 Latin words, combined with a handful of model sentence structures, to generate Lorem Ipsum which looks reasonable. The generated

Lorem Ipsum is therefore always free from repetition, injected humour, or non-characteristic words etc.",

```
"author": "raghava",
```

```
"id": 1
```

```
},
```

```
{
```

```
"title": "Opening Party!",
```

```
"body": "Why do we use it?\nIt is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).\n\nWhere does it come from?\nContrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of \"de Finibus Bonorum et Malorum\" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, \"Lorem ipsum dolor sit amet..\", comes from a line in section 1.10.32.\n\nThe standard chunk of Lorem Ipsum used since the 1500s is reproduced below for those
```

```
interested. Sections 1.10.32 and 1.10.33 from \"de Finibus
Bonorum et Malorum\" by Cicero are also reproduced in their
exact original form, accompanied by English versions from the
1914 translation by H. Rackham.\n\nWhere can I get
some?\nThere are many variations of passages of Lorem Ipsum
available, but the majority have suffered alteration in some
form, by injected humour, or randomised words which don't look
even slightly believable. If you are going to use a passage of
Lorem Ipsum, you need to be sure there isn't anything
embarrassing hidden in the middle of text. All the Lorem Ipsum
generators on the Internet tend to repeat predefined chunks as
necessary, making this the first true generator on the
Internet. It uses a dictionary of over 200 Latin words,
combined with a handful of model sentence structures, to
generate Lorem Ipsum which looks reasonable. The generated
Lorem Ipsum is therefore always free from repetition, injected
humour, or non-characteristic words etc.",
  "author": "joshi",
  "id": 2
}
1
}
```

O/P:

```
PS C:\Users\gudiw\Documents\tuts\dojo-blog> npx json-server --watch
data/db.json --port 8000
```

Need to install the following packages:

json-server@0.17.3

Ok to proceed? (y) y

```
\{^_^}/ hi!
```


Loading data/db.json

Done

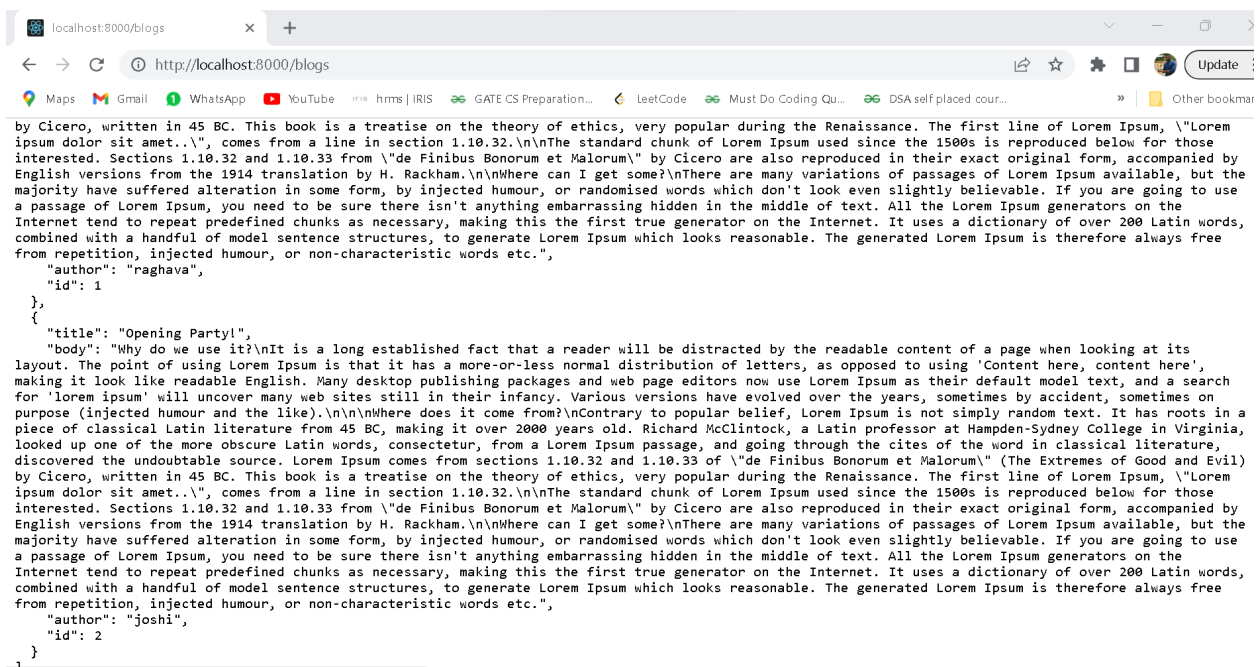
Resources

<http://localhost:8000/blogs>

Home

<http://localhost:8000>

→the link of “Resources” is enter in the localhost: 3000 we will see the json data



```
{
  "title": "Opening Party!",
  "body": "Why do we use it?\nIt is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).\n\nWhere does it come from?\nContrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of \"de Finibus Bonorum et Malorum\" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, \"Lorem ipsum dolor sit amet..\", comes from a line in section 1.10.32.\n\nThe standard chunk of Lorem Ipsum used since the 1500s is reproduced below for those interested. Sections 1.10.32 and 1.10.33 from \"de Finibus Bonorum et Malorum\" by Cicero are also reproduced in their exact original form, accompanied by English versions from the 1914 translation by H. Rackham.\n\nWhere can I get some?\nThere are many variations of passages of Lorem Ipsum available, but the majority have suffered alteration in some form, by injected humour, or randomised words which don't look even slightly believable. If you are going to use a passage of Lorem Ipsum, you need to be sure there isn't anything embarrassing hidden in the middle of text. All the Lorem Ipsum generators on the Internet tend to repeat predefined chunks as necessary, making this the first true generator on the Internet. It uses a dictionary of over 200 Latin words, combined with a handful of model sentence structures, to generate Lorem Ipsum which looks reasonable. The generated Lorem Ipsum is therefore always free from repetition, injected humour, or non-characteristic words etc.\".",
  "author": "raghava",
  "id": 1
},
{
  "title": "Why do we use it?",
  "body": "Why do we use it?\nIt is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).\n\nWhere does it come from?\nContrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of \"de Finibus Bonorum et Malorum\" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, \"Lorem ipsum dolor sit amet..\", comes from a line in section 1.10.32.\n\nThe standard chunk of Lorem Ipsum used since the 1500s is reproduced below for those interested. Sections 1.10.32 and 1.10.33 from \"de Finibus Bonorum et Malorum\" by Cicero are also reproduced in their exact original form, accompanied by English versions from the 1914 translation by H. Rackham.\n\nWhere can I get some?\nThere are many variations of passages of Lorem Ipsum available, but the majority have suffered alteration in some form, by injected humour, or randomised words which don't look even slightly believable. If you are going to use a passage of Lorem Ipsum, you need to be sure there isn't anything embarrassing hidden in the middle of text. All the Lorem Ipsum generators on the Internet tend to repeat predefined chunks as necessary, making this the first true generator on the Internet. It uses a dictionary of over 200 Latin words, combined with a handful of model sentence structures, to generate Lorem Ipsum which looks reasonable. The generated Lorem Ipsum is therefore always free from repetition, injected humour, or non-characteristic words etc.\".",
  "author": "joshi",
  "id": 2
}
}
```

Endpoints

/blogs	GET	Fetch all blogs
/blogs/{ id }	GET	Fetch a single blog
/blogs	POST	Add a new blog
/blogs/{ id }	DELETE	Delete a blog

17. Fetching Data with useEffect

Home.js

```
import { useState, useEffect } from "react";
import BlogList from "../BlogList";

const Home = () => {
  const [blogs, setBlogs] = useState(null);

  useEffect(() => {
    fetch('http://localhost:8000/blogs')
      .then(res => {
        return res.json()
      })
      .then(data => {
        setBlogs(data);
      })
  }, []);
```

```

    return (
      <div className="home">
        {blogs && <BlogList blogs= {blogs} title="All
Blogs!"></BlogList> }
      </div>
    );
  }

export default Home;

```

BlogList.js

```

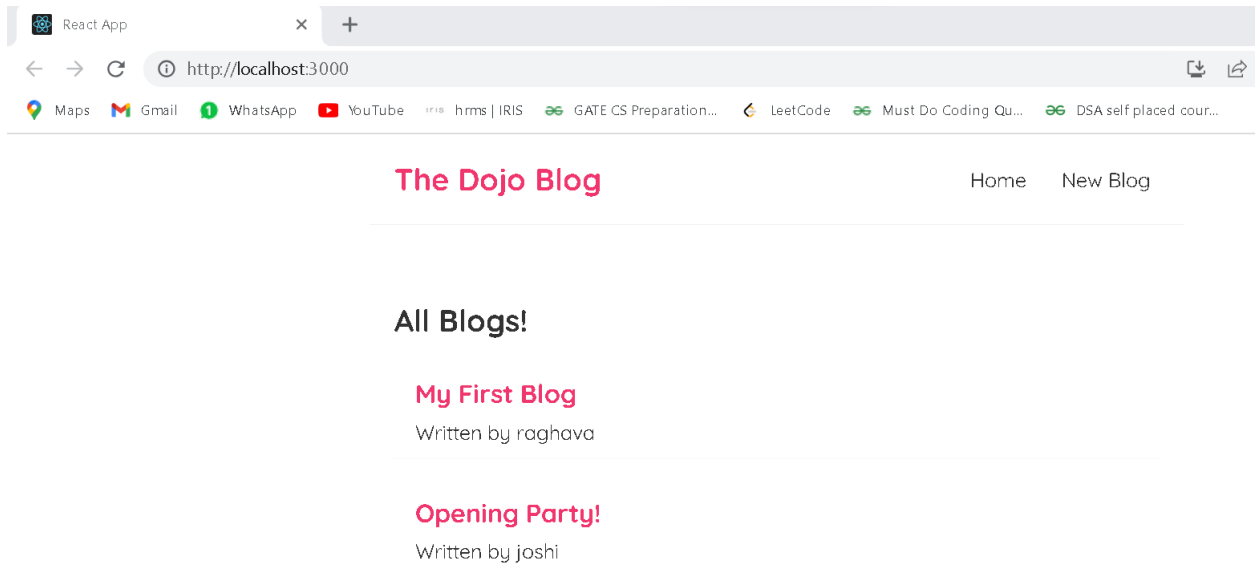
const BlogList = ({ blogs, title }) => {

  return (
    <div className="blog-list">
      <h2>{ title }</h2>
      {blogs.map((blog) => (
        <div className="blog-preview" key={blog.id}>
          <h2>{ blog.title }</h2>
          <p>Written by { blog.author }</p>
        </div>
      ))}
    </div>
  );
}

export default BlogList;

```

O/P:



18. Conditional Loading Message

Home.js

```
import { useState, useEffect } from "react";
import BlogList from "./BlogList";

const Home = () => {
  const [blogs, setBlogs] = useState(null);
  const [isPending, setIsPending] = useState(true);
  useEffect(() => {
    setTimeout(() => {
      fetch('http://localhost:8000/blogs')
        .then(res => {
          return res.json();
        })
        .then(data => {
```

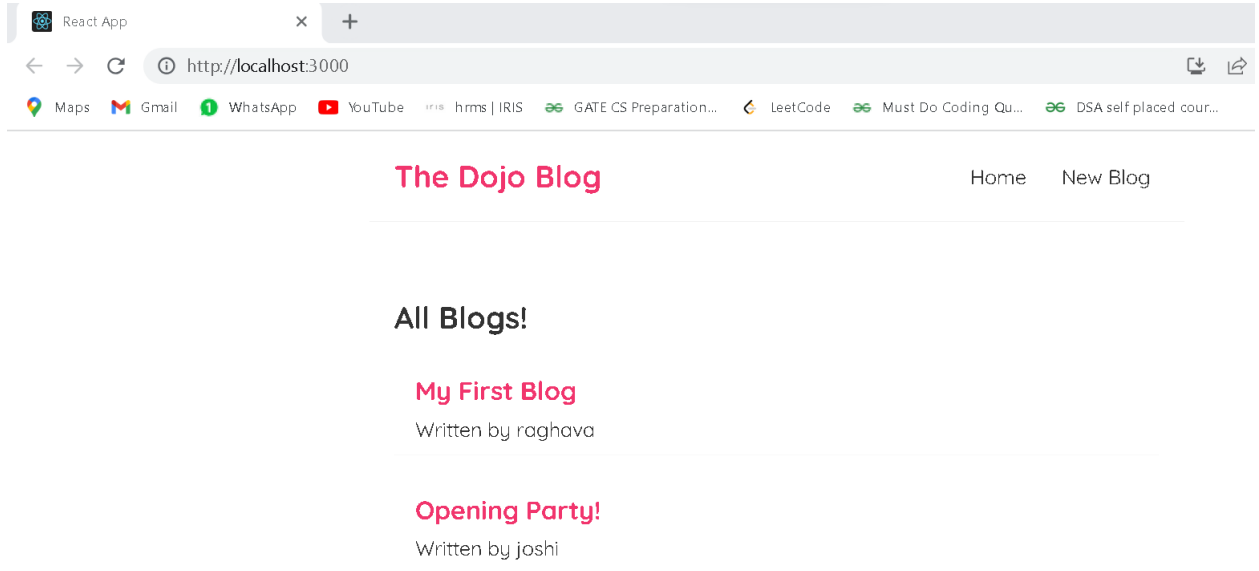
```
        setBlogs(data);
        setIsPending(false);
    });
    }, 1000);
}, []);

return (
    <div className="home">
        { isPending && <div>Loading...</div> }
        {blogs && <BlogList blogs= {blogs} title="All
Blogs!"></BlogList> }
    </div>
);
}

export default Home;
```

O/P:

→ **One sec is Loading.. message**



19. Handling Fetch Errors

Home.js

```
import { useState, useEffect } from "react";
import BlogList from "./BlogList";

const Home = () => {
  const [blogs, setBlogs] = useState(null);
  const [isPending, setIsPending] = useState(true);
  const [error, setError] = useState(null);

  useEffect(() => {
    setTimeout(() => {
      fetch('http://localhost:8000/blogs')
```

```

        .then(res => {
            if(!res.ok) {
                throw Error('could not fetch the data for
that resource');
            }
            return res.json();
        })
        .then(data => {
            setBlogs(data);
            setIsPending(false);
            setError(null);
        })
        .catch(err => {
            setIsPending(false);
            setError(err.message);
        })
    }, 1000);
}, []);

return (
    <div className="home">
        { error && <div>{ error }</div>}
        { isPending && <div>Loading...</div> }
        {blogs && <BlogList blogs= {blogs} title="All
Blogs!"></BlogList> }
    </div>
);
}

export default Home;

```

O/P:

The screenshot displays a web browser window with the address bar at `http://localhost:3000`. The page title is "The Dojo Blog" in pink, with navigation links for "Home" and "New Blog". The main content area, titled "All Blogs!", lists two blog entries: "My First Blog" (written by raghava) and "Opening Party!" (written by joshi). Below the browser window, the Chrome DevTools console is open, showing two network requests from `Home.js:11`. Both requests are CORS responses with status 200. The first request is a simple response. The second request includes a detailed body: `{body: {...}, bodyUsed: true, headers: {}, ok: true, redirected: false, status: 200, statusText: "OK", type: "cors", url: "http://localhost:8000/blogs"}`. The console also shows a summary of 286 issues: 259 errors, 27 warnings, and 27 hidden messages.

React App x +

← → ↻ ⓘ http://localhost:3000

📍 Maps 📧 Gmail 📞 WhatsApp 📺 YouTube 📺 hms | IRIS 📺 GATE CS Preparation... 📺 LeetCode 📺 Must Do Coding Qu... 📺 DSA self placed cour..

The Dojo Blog

Home New Blog

All Blogs!

My First Blog

Written by raghava

Opening Party!

Written by joshi

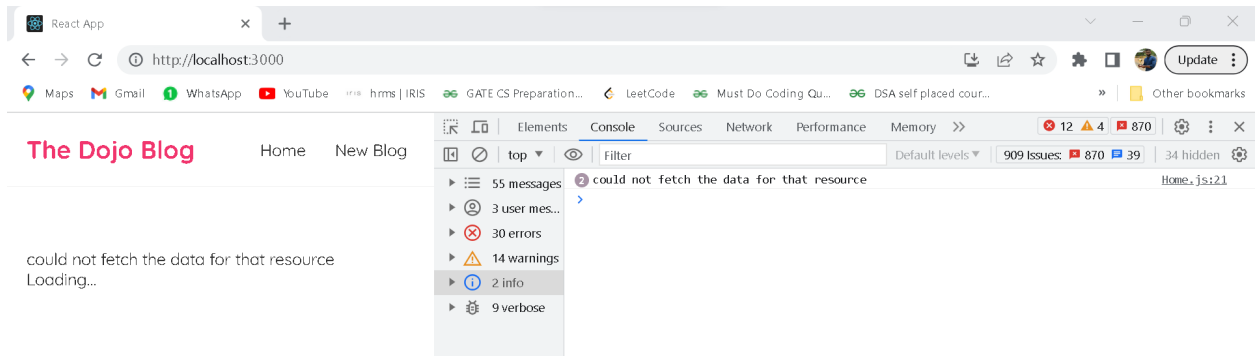
29 messages
4 user mes...
15 errors
7 warnings
2 info
5 verbose

Home.js:11
Response {type: 'cors', url: 'http://localhost:8000/blogs', redirected: false, status: 200, ok: true, ...}

Home.js:11
Response {type: 'cors', url: 'http://localhost:8000/blogs', redirected: false, status: 200, ok: true, ...}
body: {...}
bodyUsed: true
headers: Headers {}
ok: true
redirected: false
status: 200
statusText: "OK"
type: "cors"
url: "http://localhost:8000/blogs"
[[Prototype]]: Response

286 Issues: 259 27 27 hidden

→ When the error will be found it will catch the error



20. Making a Custom Hook

Home.js

```
import { useState, useEffect } from "react";
import BlogList from "./BlogList";
import useFetch from "./useFetch";

const Home = () => {
  const { data: blogs, isPending, error } =
    useFetch('http://localhost:8000/blogs');

  return (
    <div className="home">
      { error && <div>{ error }</div> }
      { isPending && <div>Loading...</div> }
      { blogs && <BlogList blogs={blogs} title="All
Blogs!"></BlogList> }
    </div>
  );
}
```

```
export default Home;
```

useFetch.js

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

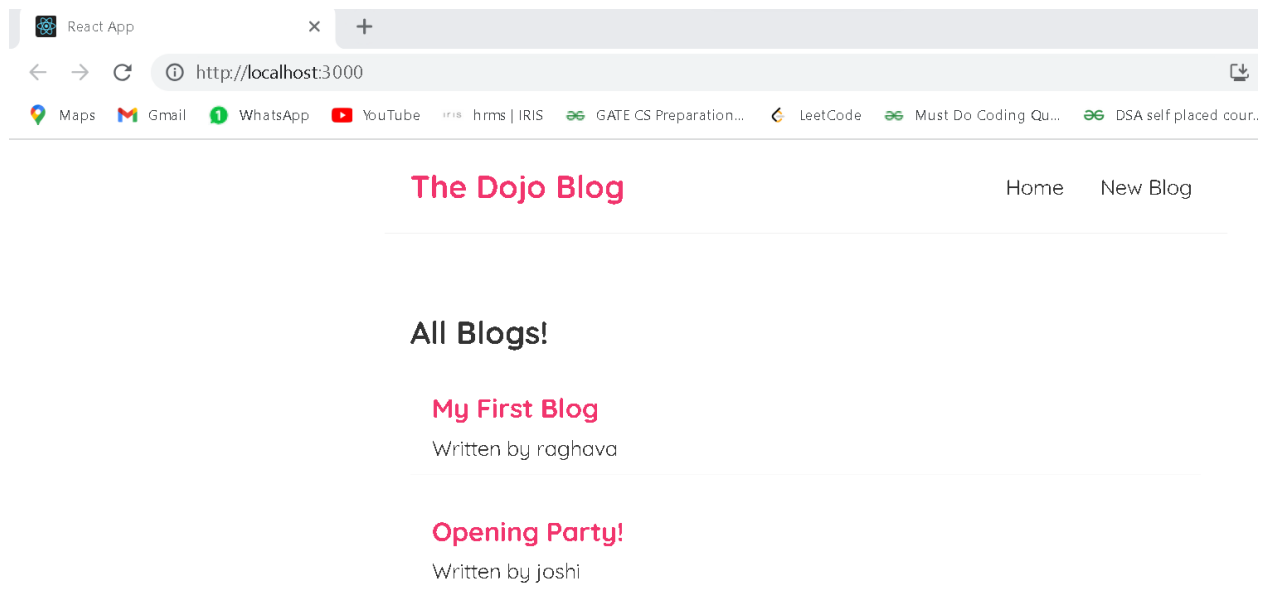
const useFetch = (url) => {
  const [data, setData] = useState(null);
  const [isPending, setIsPending] = useState(true);
  const [error, setError] = useState(null);

  useEffect(() =>{
    setTimeout(() => {
      fetch(url)
        .then(res => {
          if(!res.ok){
            throw Error('could not fetch the data
for that resource');
          }
          return res.json();
        })
        .then(data => {
          setData(data);
          setIsPending(false);
          setError(null);
        })
        .catch(err => {
          setIsPending(false);
          setError(err.message);
        })
      }, 1000);
    }, [url]);
  }
```

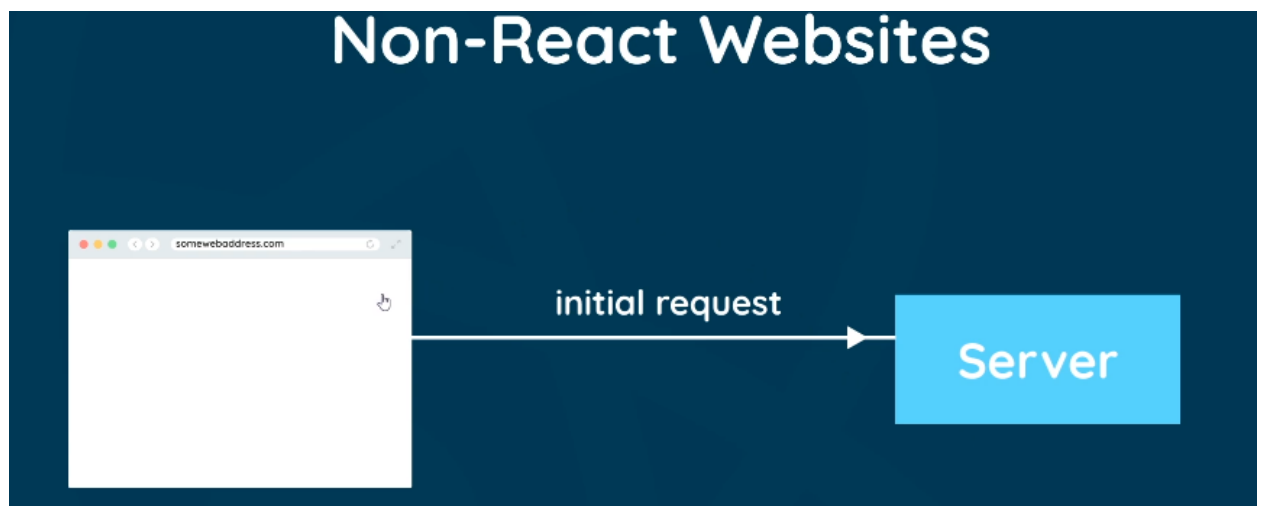
```
    return { data, isPending, error}
  }

export default useFetch;
```

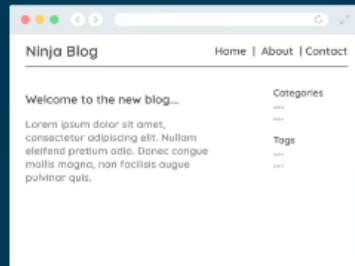
O/P:



21. The React Router



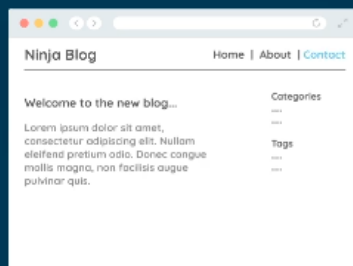
Non-React Websites



index.html

Server

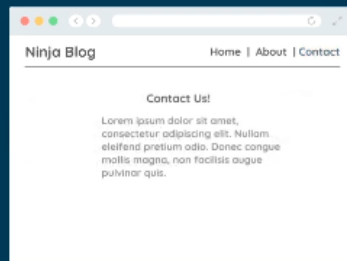
Non-React Websites



request to /contact

Server

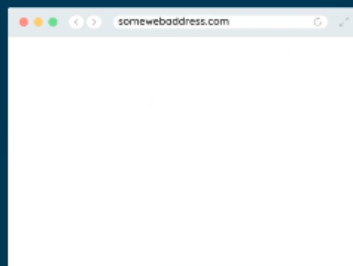
Non-React Websites



contact.html

Server

React Websites (SPA's)



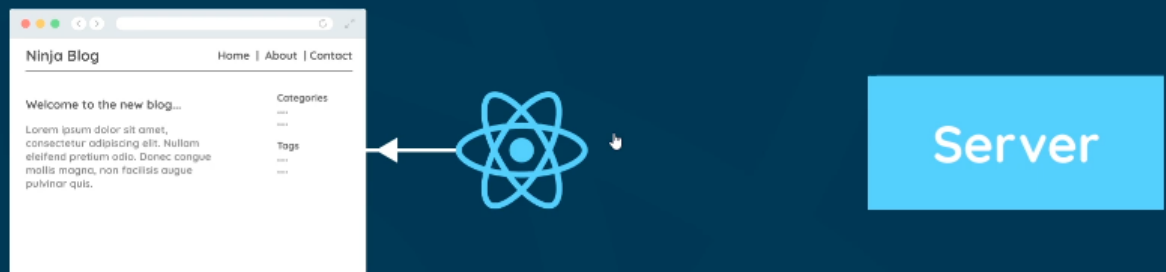
initial request

Server

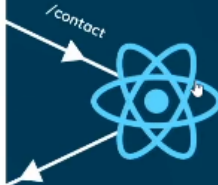
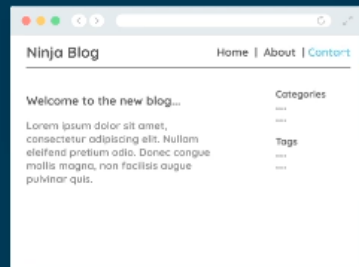
React Websites (SPA's)



React Websites (SPA's)

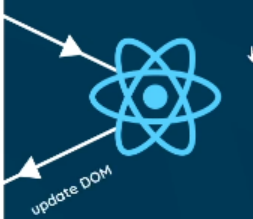
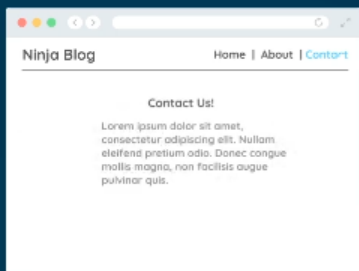


React Websites (SPA's)



Server

React Websites (SPA's)



Server

→By using React less contact with server because it is faster



```
PS C:\Users\gudiw\Documents\tuts\dojo-blog> npm install  
react-router-dom@5
```

added 12 packages, and audited 1508 packages in 26s

235 packages are looking for funding
run `npm fund` for details

74 vulnerabilities (69 moderate, 5 high)

```
npm audit fix
```

To address all issues (including breaking changes), run:
`npm audit fix --force`

Run `npm audit` for details.

```
PS C:\Users\gudiw\Documents\tuts\dojo-blog>
```

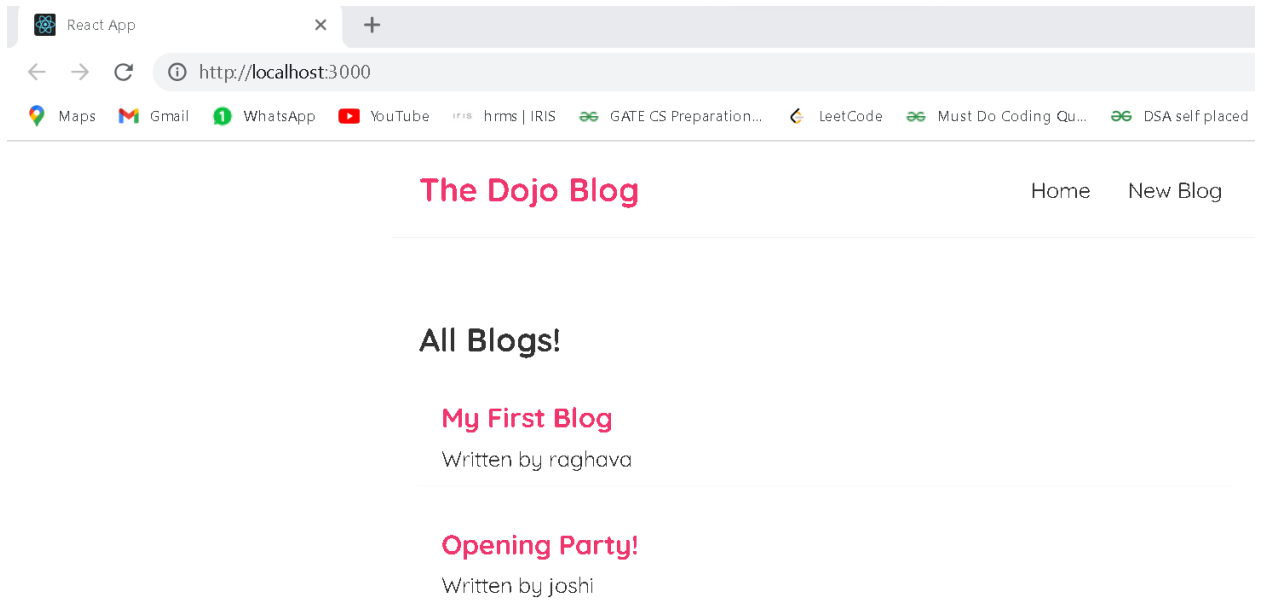

App.js

```
import Navbar from './Navbar';
import Home from './Home';
import { BrowserRouter as Router, Route, Switch } from
'react-router-dom';

function App() {
  return (
    <Router>
      <div className="App">
        <Navbar></Navbar>
        <div className="content">
          <Switch>
            <Route path="/">
              <Home></Home>
            </Route>
          </Switch>
        </div>
      </div>
    </Router>
  );
}

export default App;
```

O/P:



22. Exact Match Routes

App.js

```
import Navbar from './Navbar';
import Home from './Home';
import { BrowserRouter as Router, Route, Switch } from
'react-router-dom';
import Create from './Create';

function App() {
  return (
    <Router>
      <div className="App">
        <Navbar></Navbar>
        <div className="content">
          <Switch>
            <Route exact path="/">
```

```

        <Home></Home>
      </Route>
      <Route path="/create">
        <Create></Create>
      </Route>
    </Switch>
  </div>
</div>
</Router>
);
}

export default App;

```

Create.js

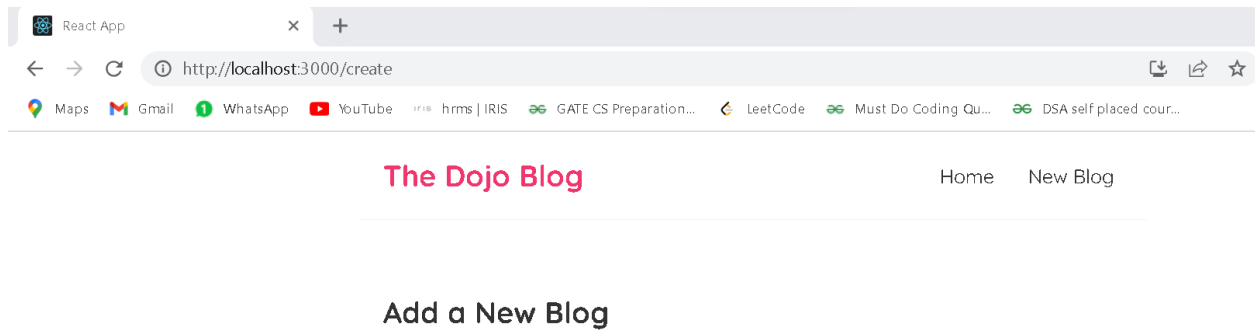
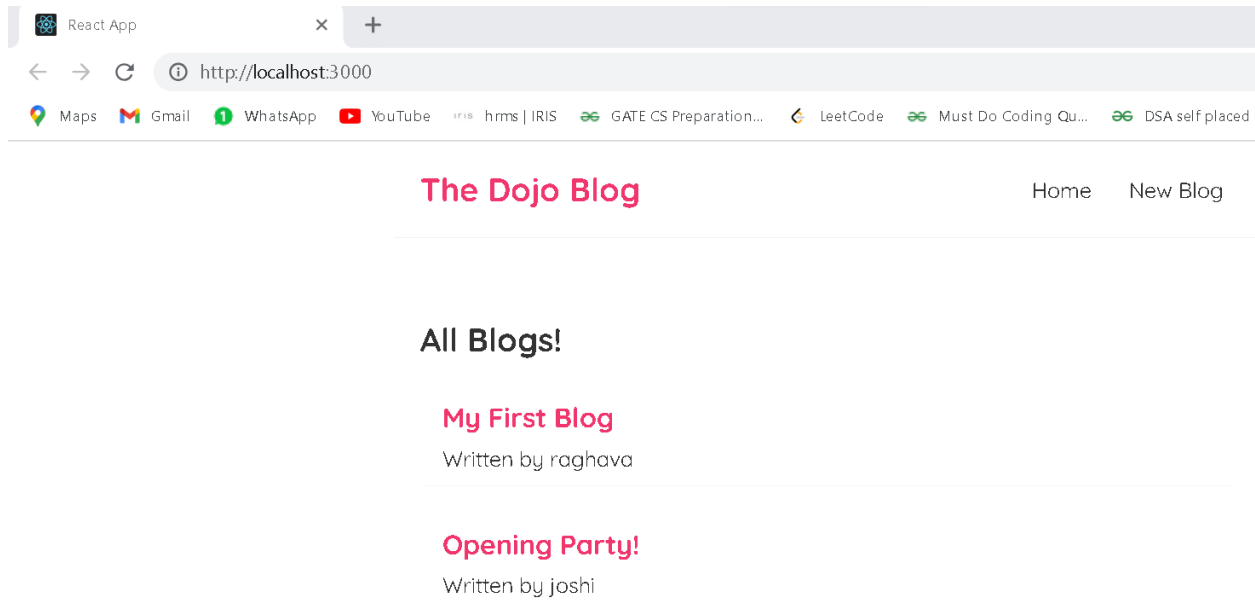
```

const Create = () => {
  return (
    <div className="create">
      <h2>Add a New Blog</h2>
    </div>
  );
}

export default Create;

```

O/P:



23. Router Links

App.js

```
import Navbar from './Navbar';
import Home from './Home';
import { BrowserRouter as Router, Route, Switch } from
'react-router-dom';
import Create from './Create';

function App() {
  return (
    <Router>
      <div className="App">
        <Navbar></Navbar>
        <div className="content">
          <Switch>
            <Route exact path="/">
              <Home></Home>
            </Route>
            <Route path="/create">
              <Create></Create>
            </Route>
          </Switch>
        </div>
      </div>
    </Router>
  );
}

export default App;
```

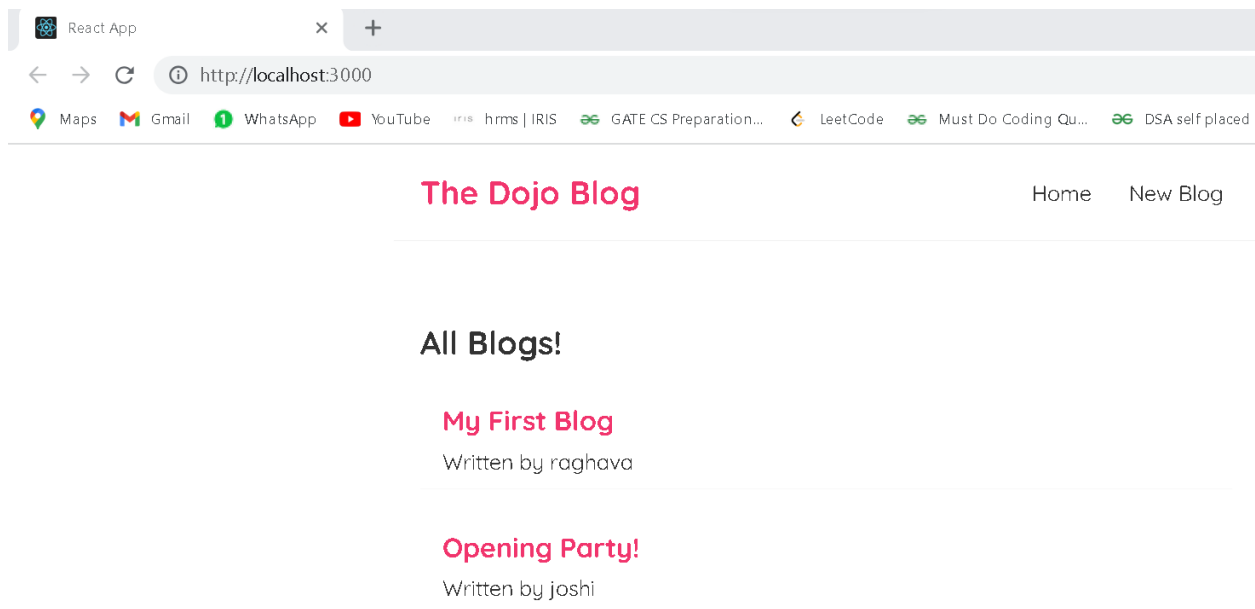
Navbar.js

```
import { Link } from "react-router-dom";

const Navbar = () => {
  return (
    <nav className="navbar">
      <h1>The Dojo Blog</h1>
      <div className="links">
        <Link to="/">Home</Link>
        <Link to="/create">New Blog</Link>
      </div>
    </nav>
  );
}

export default Navbar;
```

O/P:

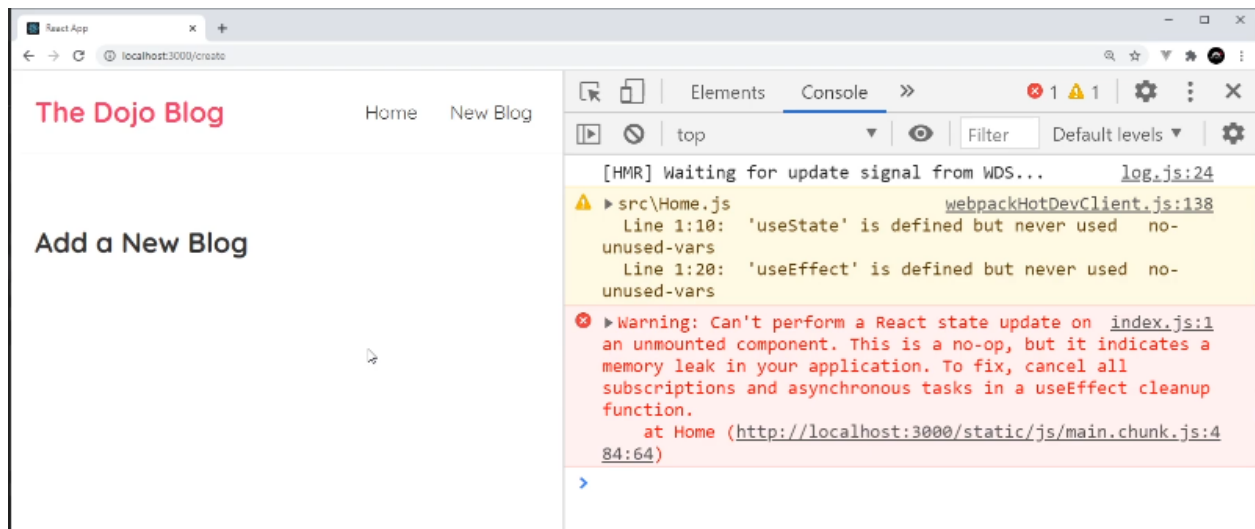




The Dojo Blog

Home New Blog

Add a New Blog



24. useEffect CleanUp

useFetch.js

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

const useFetch = (url) => {
  const [data, setData] = useState(null);
  const [isPending, setIsPending] = useState(true);
  const [error, setError] = useState(null);

  useEffect(() =>{
    const abortCont = new AbortController();

    setTimeout(() => {
      fetch(url, { signal: abortCont.signal })
        .then(res => {
          if(!res.ok){
            throw Error('could not fetch the data
for that resource');
          }
          return res.json();
        })
        .then(data => {
          setData(data);
          setIsPending(false);
          setError(null);
        })
        .catch(err => {
          if (err.name === 'AbortError') {
            console.log('fetch aborted')
          } else{
            setIsPending(false);

```



```

        setError(err.message);
    }
    })
    }, 1000);

    return () => abortCont.abort();
}, [url]);

    return { data, isPending, error}
}

export default useFetch;

```

Home.js

```

import BlogList from "../BlogList";
import useFetch from "../useFetch";

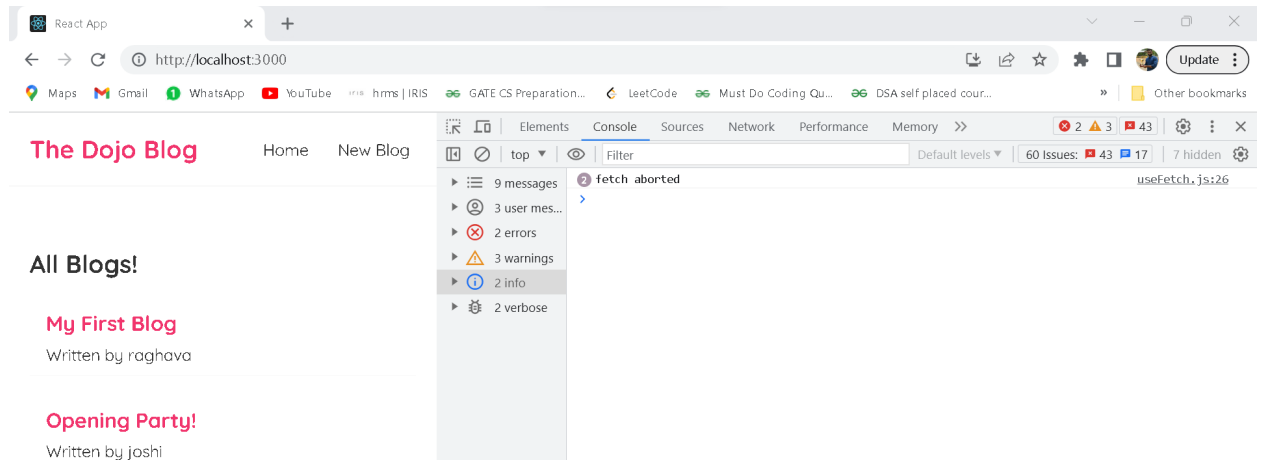
const Home = () => {
    const { data: blogs, isPending, error } =
useFetch('http://localhost:8000/blogs');

    return (
        <div className="home">
            { error && <div>{ error }</div>}
            { isPending && <div>Loading...</div> }
            {blogs && <BlogList blogs= {blogs} title="All
Blogs!"></BlogList> }
        </div>
    );
}

export default Home;

```

O/P:



25. Route Parameters

In React, route parameters are used to pass dynamic values as part of the URL path when defining routes in a React router. They allow you to create dynamic and flexible routes to handle various scenarios in your application.

Route Parameters

/blogs/123

/blogs/456

/blogs/789

App.js

```
import Navbar from './Navbar';
import Home from './Home';
import { BrowserRouter as Router, Route, Switch } from
'react-router-dom';
import Create from './Create';
import BlogDetails from './BlogDetails';

function App() {
```

```

return (
  <Router>
    <div className="App">
      <Navbar></Navbar>
      <div className="content">
        <Switch>
          <Route exact path="/">
            <Home></Home>
          </Route>
          <Route path="/create">
            <Create></Create>
          </Route>
          <Route path="/blogs/:id">
            <BlogDetails></BlogDetails>
          </Route>
        </Switch>
      </div>
    </div>
  </Router>
);
}

export default App;

```

BlogList.js

```

import { Link } from 'react-router-dom';

const BlogList = ({ blogs }) => {
  return (
    <div className="blog-list">
      {blogs.map(blog => (

```

```

        <div className="blog-preview" key={blog.id} >
          <Link to={` /blogs/${blog.id}`} >
            <h2>{ blog.title }</h2>
            <p>Written by { blog.author }</p>
          </Link>
        </div>
      )}
    </div>
  );
}

export default BlogList;

```

BlogDetails.js

```

import { useParams } from "react-router-dom";

const BlogDetails = () => {
  const { id } = useParams();

  return (
    <div className="blog-details">
      <h2>Blog details - { id }</h2>
    </div>
  );
}

export default BlogDetails;

```

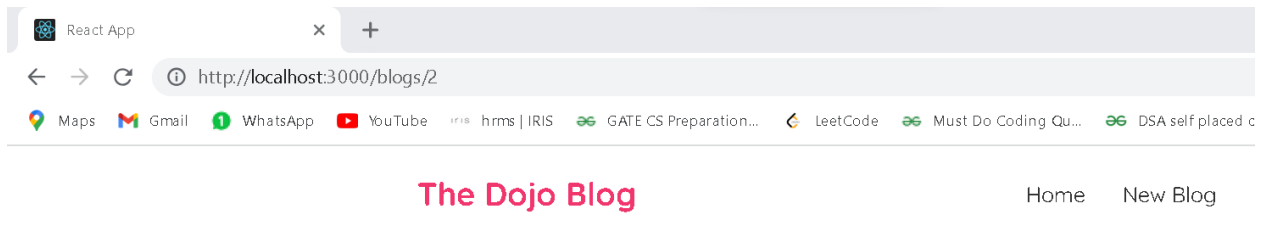
index.css

```
.blog-preview a{  
  text-decoration: none;  
}
```

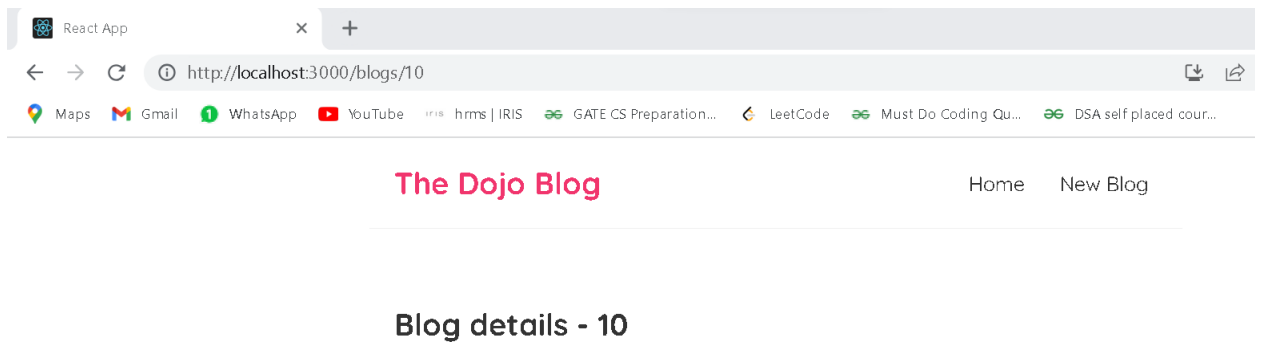
O/P:



Blog details - 1



→Route Parameter (Dynamic path)



26. Reusing Custom Hooks

BlogDetails.js

```
import { useParams } from "react-router-dom";
import useFetch from './useFetch';

const BlogDetails = () => {
  const { id } = useParams();
  const { data: blog, isPending, error } =
useFetch('http://localhost:8000/blogs/' + id);
  return (
    <div className="blog-details">
      { isPending && <div>Loading...</div>}
      { error && <div>{ error }</div>}
      { blog && (
        <article>
          <h2>{ blog.title }</h2>
          <p>Written by { blog.author }</p>
          <div>{ blog.body }</div>
        </article>
      )}
    </div>
  );
}

export default BlogDetails;
```

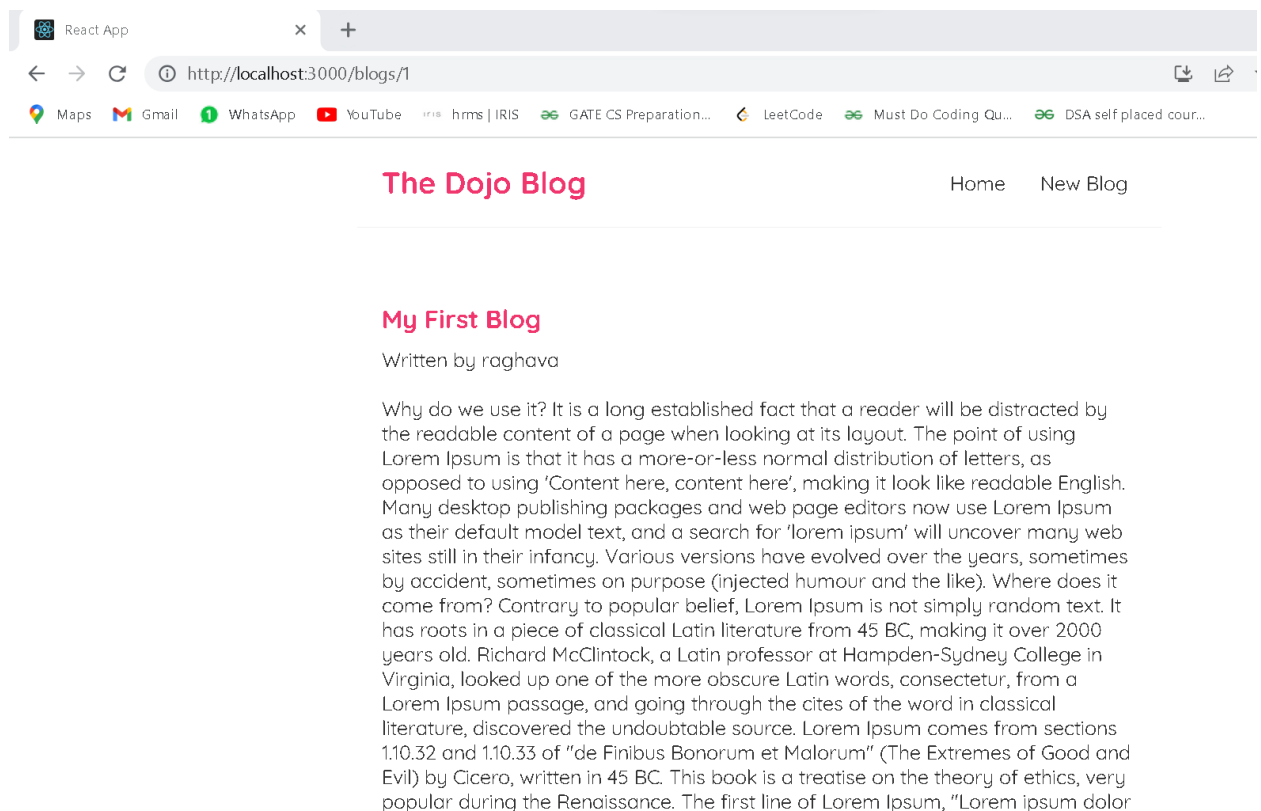
index.css

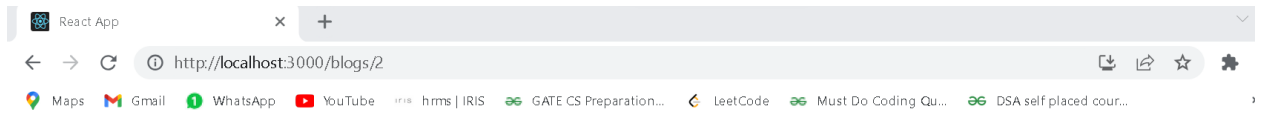
```
/* blog details page */
.blog-details h2 {
  font-size: 20px;
  color: #f1356d;
```



```
margin-bottom: 10px;
}
.blog-details div {
  margin: 20px 0;
}
```

O/P:





The Dojo Blog

[Home](#) [New Blog](#)

Opening Party!

Written by joshi

Why do we use it? It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like). Where does it come from? Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum et Malorum" (The Extremes of Good and

27. Controlled Inputs(forms)

Create.js

```
import { useState } from "react";

const Create = () => {
  const [title, setTitle] = useState('');
  const [body, setBody] = useState('');
  const [author, setAuthor] = useState('raghava');

  return (
    <div className="create">
      <h2>Add a New Blog</h2>
      <form>
        <label>Blog title:</label>
        <input
```

```

        type="text"
        required
        value={title}
        onChange={ (e) => setTitle(e.target.value) }
    </input>
    <label>Blog body:</label>
    <textarea
        required
        value={body}
        onChange={ (e) => setBody(e.target.value) }
    </textarea>
    <label>Blog author:</label>
    <select
        value={author}
        onChange={ (e) => setAuthor(e.target.value) }
    >
        <option value="raghava">raghava</option>
        <option value="joshi">joshi</option>
    </select>
    <button>Add Blog</button>
</form>
</div>
);
}

export default Create;

```

index.css

```

/* create new blog form */
.create {
    max-width: 400px;
    margin: 0 auto;
}

```

```
    text-align: center;
}
.create label {
    text-align: left;
    display: block;
}
.create h2 {
    font-size: 20px;
    color: #f1356d;
    margin-bottom: 30px;
}
.create input, .create textarea, .create select {
    width: 100%;
    padding: 6px 10px;
    margin: 10px 0;
    border: 1px solid #ddd;
    box-sizing: border-box;
    display: block;
}
.create button {
    background: #f1356d;
    color: #fff;
    border: 0;
    padding: 8px;
    border-radius: 8px;
    cursor: pointer;
}
```

O/P:

React App

http://localhost:3000/create

Maps Gmail WhatsApp YouTube hrms | IRIS GATE CS Preparation... LeetCode Must Do Coding Qu... DSA self placed cour...

The Dojo Blog

Home New Blog

Add a New Blog

Blog title:

Blog body:

Blog author:

raghava

Add Blog

28. Submit Events

Create.js

```
import { useState } from "react";

const Create = () => {
  const [title, setTitle] = useState('');
  const [body, setBody] = useState('');
  const [author, setAuthor] = useState('raghava');

  const handleSubmit = (e) => {
```

```
e.preventDefault();  
const blog = { title, body, author };  
  
console.log(blog);  
}  
  
return (  
  <div className="create">  
    <h2>Add a New Blog</h2>  
    <form onSubmit={handleSubmit}>  
      <label>Blog title:</label>  
      <input  
        type="text"  
        required  
        value={title}  
        onChange={ (e) => setTitle(e.target.value) }  
      ></input>  
      <label>Blog body:</label>  
      <textarea  
        required  
        value={body}  
        onChange={ (e) => setBody(e.target.value) }  
      ></textarea>  
      <label>Blog author:</label>  
      <select  
        value={author}  
        onChange={ (e) => setAuthor(e.target.value) }  
      >  
        <option value="raghava">raghava</option>  
        <option value="joshi">joshi</option>  
      </select>  
      <button>Add Blog</button>  
    </form>  
  )  
)
```

```

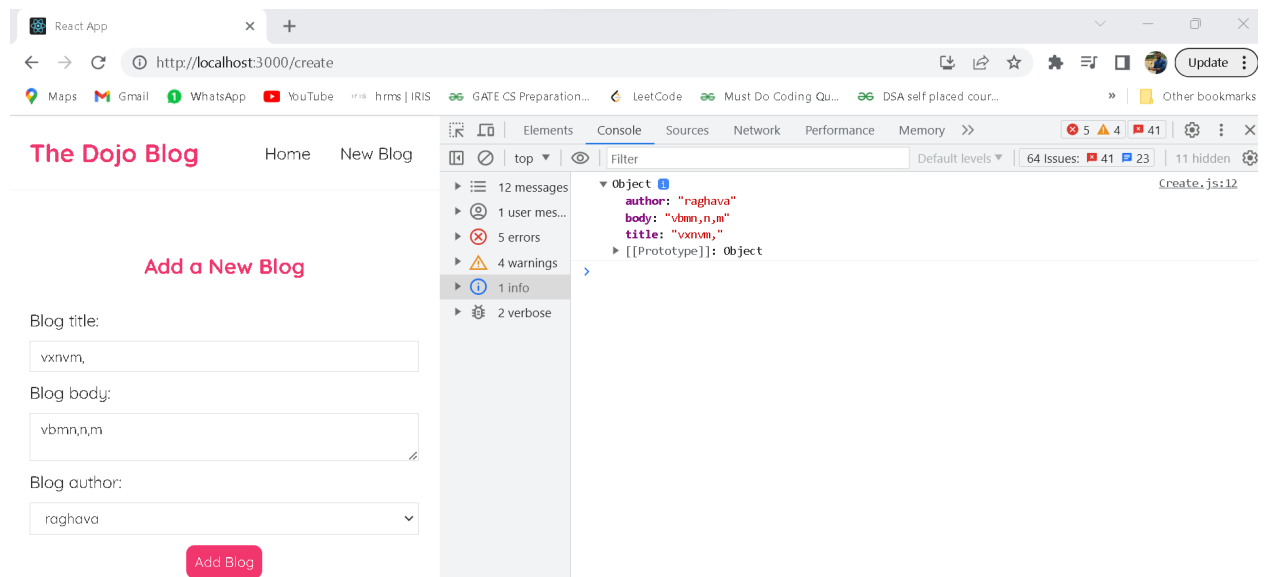
    </div>

  );
}

export default Create;

```

O/P:



29. Making a Post Request

Create.js

```

import { useState } from "react";

const Create = () => {
  const [title, setTitle] = useState('');
  const [body, setBody] = useState('');
  const [author, setAuthor] = useState('raghava');
  const [isPending, setIsPending] = useState(false);

```

```

const handleSubmit = (e) => {
  e.preventDefault();
  const blog = { title, body, author };

  setIsPending(true);

  fetch('http://localhost:8000/blogs', {
    method: 'POST',
    headers: { "Content-Type": "application/json" },
    body: JSON.stringify(blog)
  }).then(() => {
    console.log('new blog added');
    setIsPending(false);
  })
}

return (
  <div className="create">
    <h2>Add a New Blog</h2>
    <form onSubmit={handleSubmit}>
      <label>Blog title:</label>
      <input
        type="text"
        required
        value={title}
        onChange={ (e) => setTitle(e.target.value) }
      ></input>
      <label>Blog body:</label>
      <textarea
        required
        value={body}
        onChange={ (e) => setBody(e.target.value) }

```



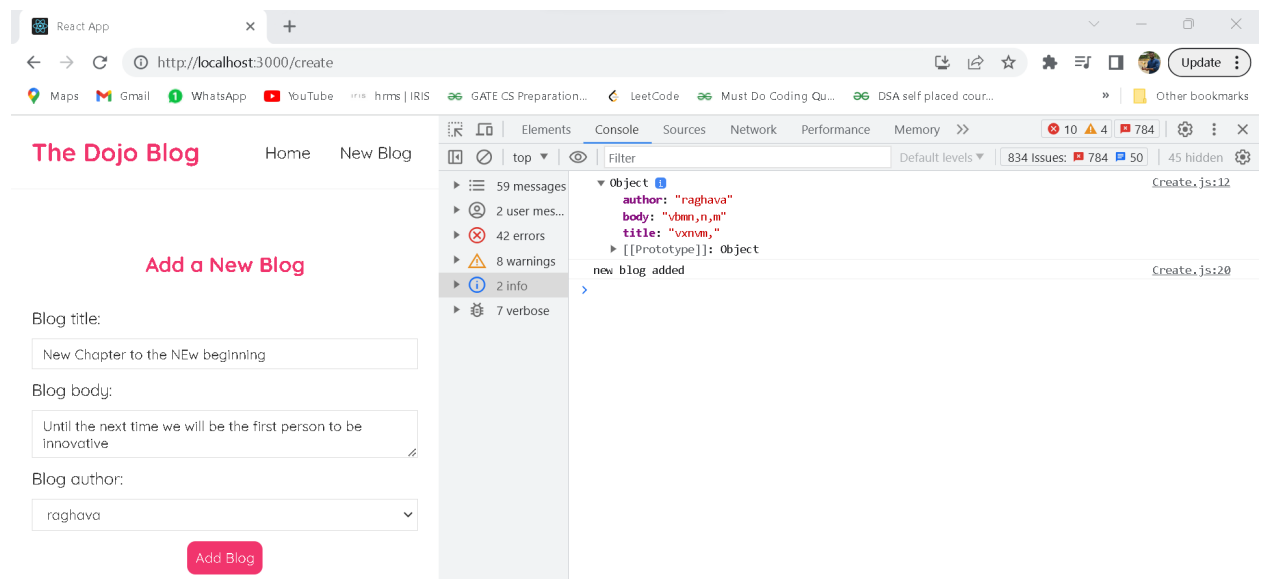
```

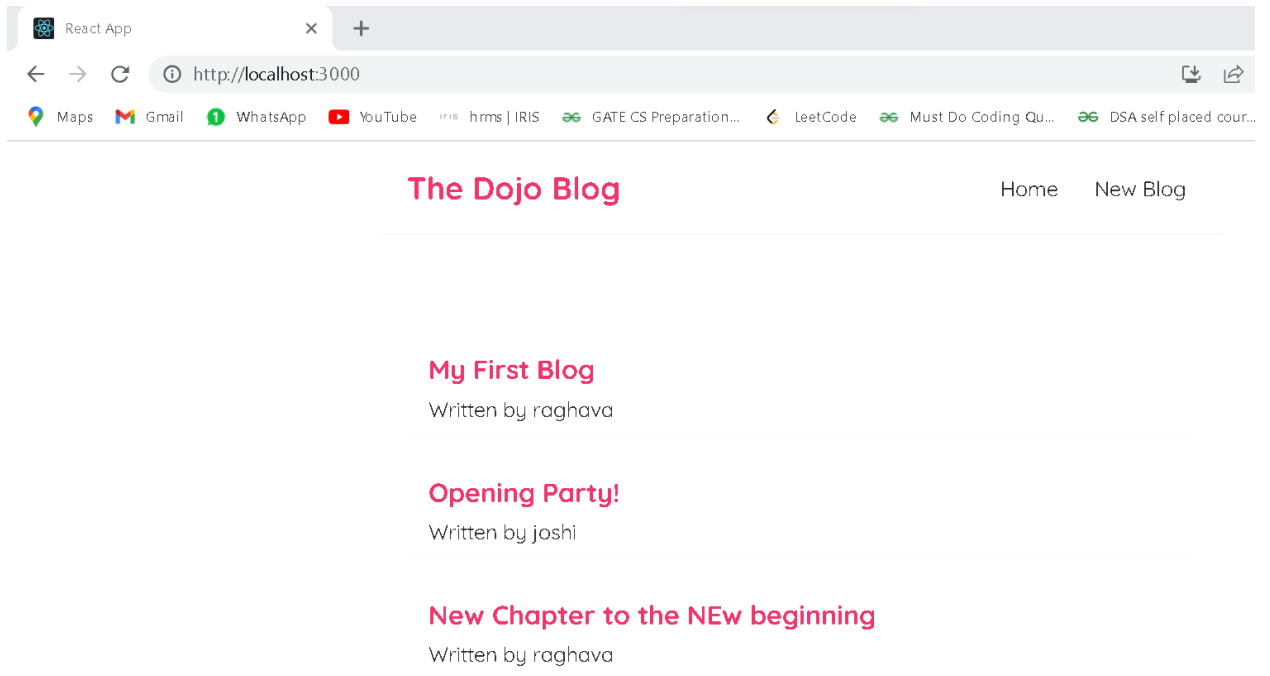
    </textarea>
    <label>Blog author:</label>
    <select
      value={author}
      onChange={ (e) => setAuthor(e.target.value) }
    >
      <option value="raghava">raghava</option>
      <option value="joshi">joshi</option>
    </select>
    { !isPending && <button>Add Blog</button> }
    { isPending && <button>Adding blog...</button> }
  </form>
</div>
);
}

export default Create;

```

O/P:





30. Programmatic Redirects

Create.js

```
import { useState } from "react";
import { useHistory } from "react-router-dom";

const Create = () => {
  const [title, setTitle] = useState('');
  const [body, setBody] = useState('');
  const [author, setAuthor] = useState('raghava');
  const [isPending, setIsPending] = useState(false);
  const history = useHistory();

  const handleSubmit = (e) => {
    e.preventDefault();
```

```

const blog = { title, body, author };

setIsPending(true);

fetch('http://localhost:8000/blogs', {
  method: 'POST',
  headers: {"Content-Type": "application/json" },
  body: JSON.stringify(blog)
}).then(() => {
  console.log('new blog added');
  setIsPending(false);
  //history.go(-1);
  history.push('/');
});

}

return (
  <div className="create">
    <h2>Add a New Blog</h2>
    <form onSubmit={handleSubmit}>
      <label>Blog title:</label>
      <input
        type="text"
        required
        value={title}
        onChange={ (e) => setTitle(e.target.value) }
      ></input>
      <label>Blog body:</label>
      <textarea
        required
        value={body}
        onChange={ (e) => setBody(e.target.value) }

```

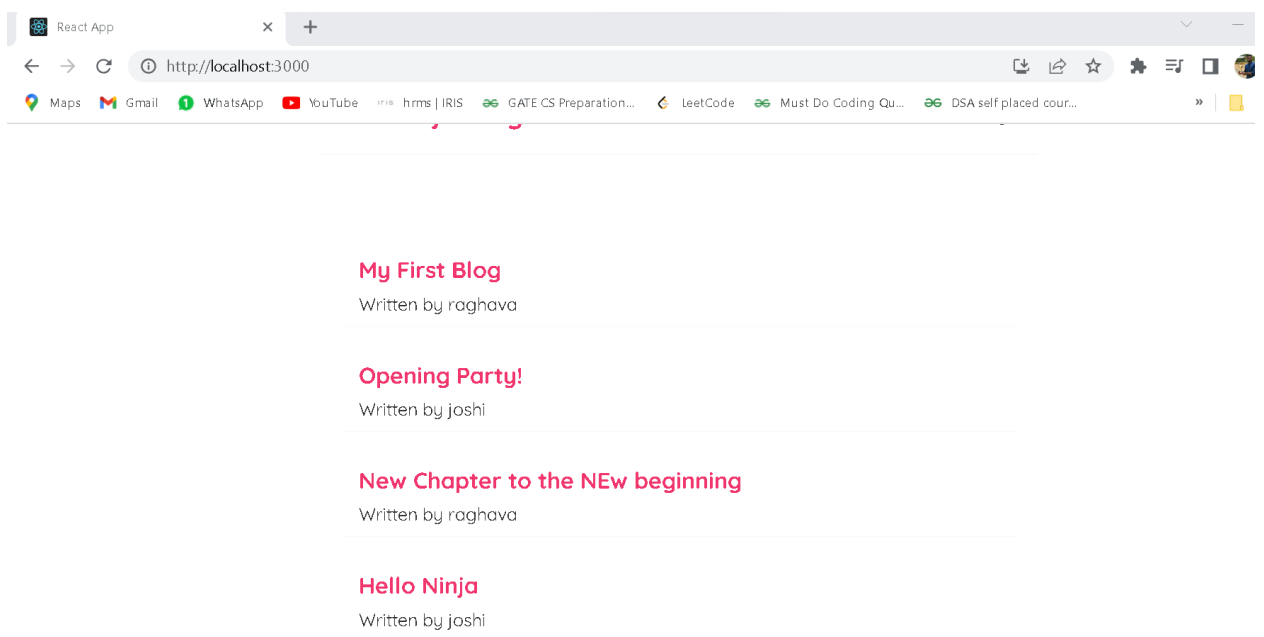
```

    </textarea>
    <label>Blog author:</label>
    <select
      value={author}
      onChange={ (e) => setAuthor(e.target.value) }
    >
      <option value="raghava">raghava</option>
      <option value="joshi">joshi</option>
    </select>
    { !isPending && <button>Add Blog</button> }
    { isPending && <button>Adding blog...</button> }
  </form>
</div>
);
}

export default Create;

```

O/P:



31. Deleting Blogs

BlogDetails.js

```
import { useHistory, useParams } from "react-router-dom";
import useFetch from './useFetch';

const BlogDetails = () => {
  const { id } = useParams();
  const { data: blog, isPending, error } =
useFetch('http://localhost:8000/blogs/' + id);
  const history = useHistory();

  const handleClick= () =>{
    fetch('http://localhost:8000/blogs/' + blog.id, {
      method: 'DELETE',
    }).then(() => {
      history.push('/');
    })
  }

  return (
    <div className="blog-details">
      { isPending && <div>Loading...</div>}
      { error && <div>{ error }</div>}
      { blog && (
        <article>
          <h2>{ blog.title }</h2>
          <p>Written by { blog.author }</p>
          <div>{ blog.body }</div>
        </article>
      )}
    </div>
  )
}
```

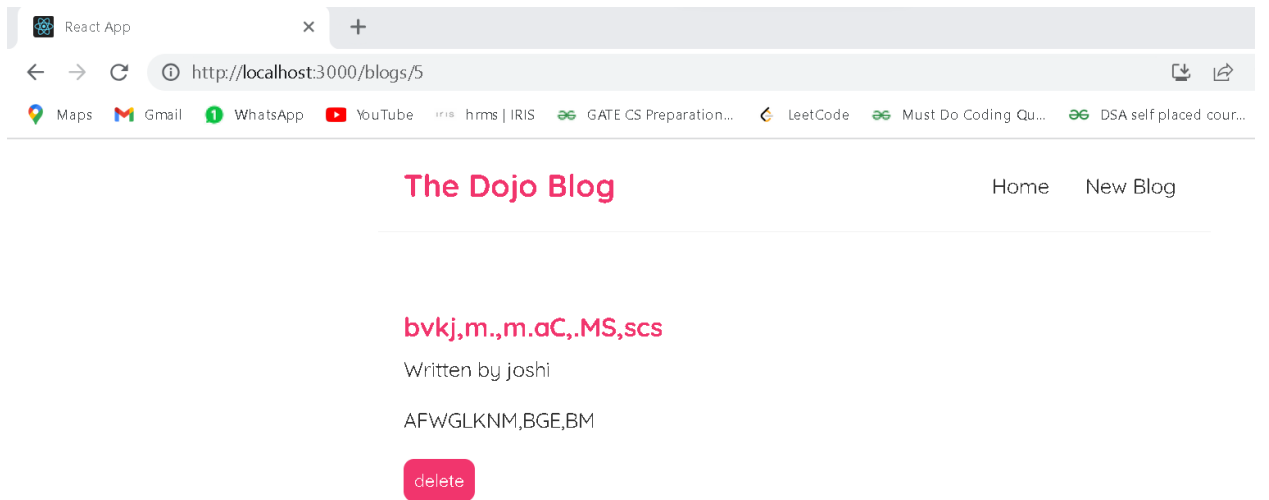
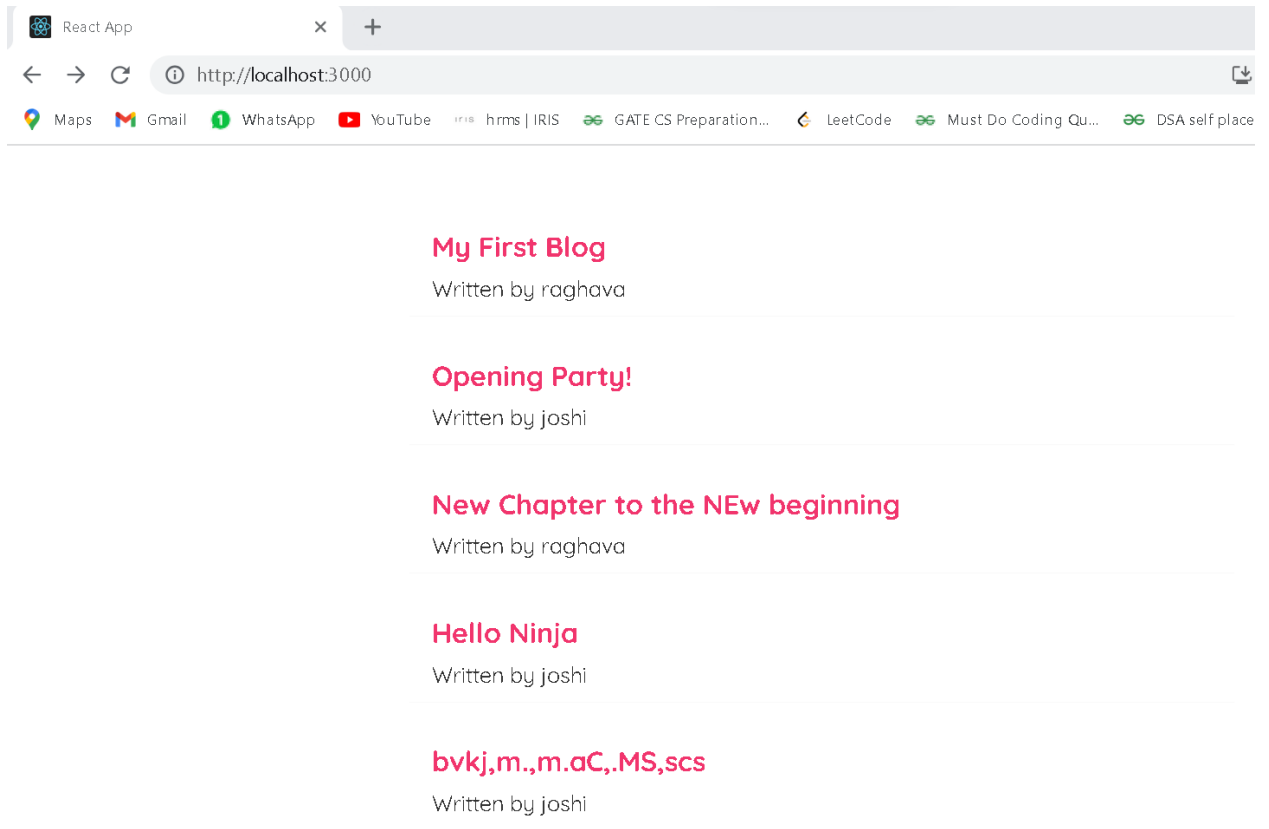
```
        <button
onClick={handleClick}>delete</button>
      </article>
    )}
  </div>
);
}

export default BlogDetails;
```

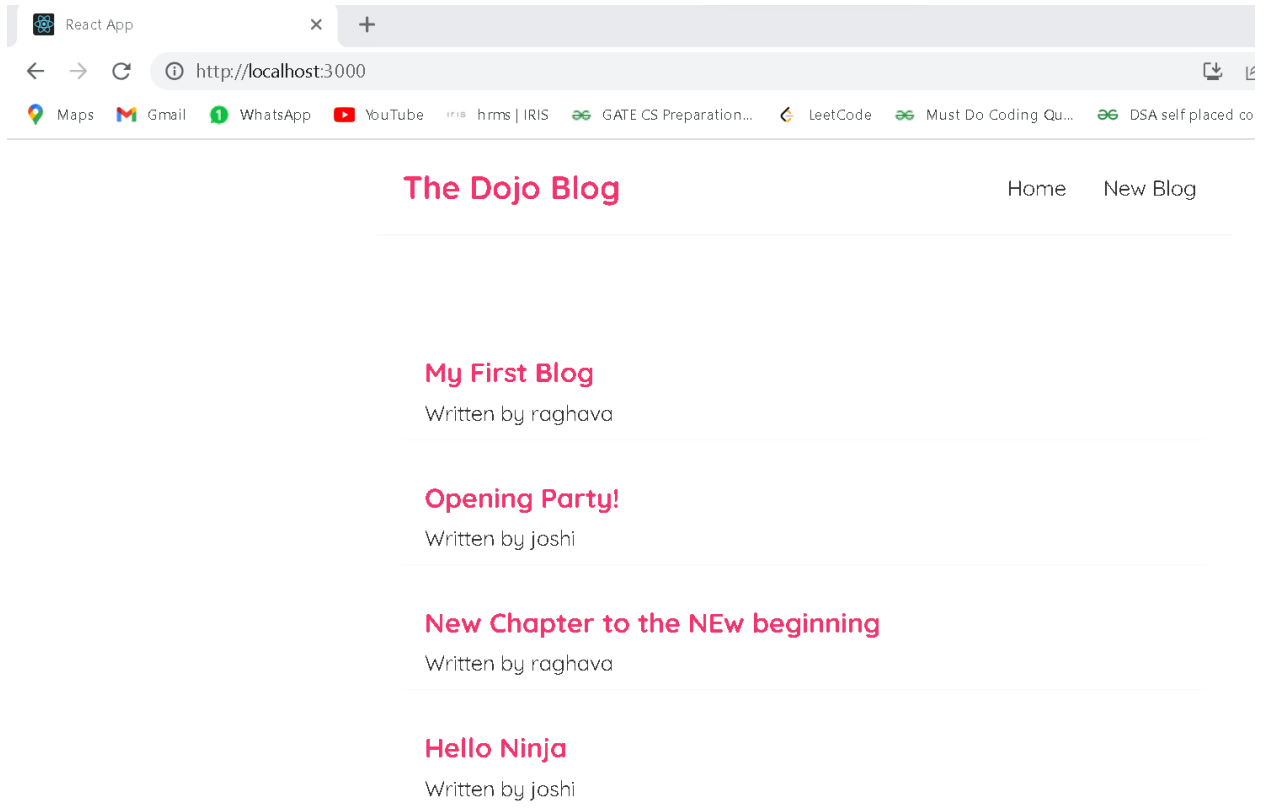
index.css

```
.blog-details button {
  background: #f1356d;
  color: #fff;
  border: 0;
  padding: 8px;
  border-radius: 8px;
  cursor: pointer;
}
```

O/P:



→After Deletion



32. 404 Pages & Next Page

App.js

```
import Navbar from './Navbar';
import Home from './Home';
import { BrowserRouter as Router, Route, Switch } from
'react-router-dom';
import Create from './Create';
import BlogDetails from './BlogDetails';
import NotFound from './NotFound';
```



```
function App() {  
  return (  
    <Router>  
      <div className="App">  
        <Navbar></Navbar>  
        <div className="content">  
          <Switch>  
            <Route exact path="/">  
              <Home></Home>  
            </Route>  
            <Route path="/create">  
              <Create></Create>  
            </Route>  
            <Route path="/blogs/:id">  
              <BlogDetails></BlogDetails>  
            </Route>  
            <Route path="*">  
              <NotFound></NotFound>  
            </Route>  
          </Switch>  
        </div>  
      </div>  
    </Router>  
  );  
}  
  
export default App;
```

NotFound.js

```
import { Link } from "react-router-dom";

const NotFound = () => {
  return (
    <div className="not-found">
      <h2>Sorry</h2>
      <p>That page cannot be found</p>
      <Link to="/">Back to the homepage...</Link>
    </div>
  );
}

export default NotFound;
```

O/P:

