

Npx create-react-app bulletinblaze
This command inside our folder

Include **bootstrap css, js links** inside **index.html** file inside **public folder**

Changing title inside index.html to bulletinblaze...

Now changing content inside meta tags inside index.html

Remove index.css delete all code of it

Create components folder inside src, then create Navbar.js inside components folder

Write rcep (react class based component) in Navbar.js and import react class based template

Now go to bootstrap and copy navbar code and paste it in Navbar.js

Replace class with className, and put all closing tags

CONCEPT- Render is a life cycle method- jab react ek component ko run karti hai toh kuch series of methods run hote hai
jsx ko compile krna and screen par html ko run karna

```
<a className="nav-link" href="/about">About</a>
```

Yaha par href mai /about likha hai inside navbar.js

```
<a className="navbar-brand" href="/">BulletinBlaze</a>
```

Yaha pr href mai / likha hai inside navbar.js

HREF MAI “#” nahi use karte “/” use krte hai

News.js, newsitem.js files banao

Ab News mai newsitem.js ko import karo

```
<News/>
```

-> newsitem news ke andar hai

Ab mai newsitem mai kuch changes krta hu toh vo automatically news ke andar changes show ho rhe hai kyuki news ke andar newsitem hai aur news ne import kara hai newsitem ko thats the **POWER OF REACT**

endpoint is a specific URL (Uniform Resource Locator) or URI (Uniform Resource Identifier) that a client can access to interact with a web service or an application programming interface (API).

Each endpoint typically corresponds to a specific operation or resource. For example, you might have endpoints like `/api/books/{id}` to retrieve a specific book by its ID, or `/api/books/{id}/reviews` to retrieve the reviews for a particular book.

Sabse bahar ek file banao **sampleOutput.json**

Go to newsapi website

Ab login pr click karke ek account banao

Go to top headlines section and copy this-

```
https://newsapi.org/v2/top-headlines?country=us&apiKey=d60dc46da182403294520833a18e8a0f
```

Type your url on the browser and copy the json from there

Koi bhi server json bhej raha hota hai toh vo ek compress form mai bhejta hai

Go to component section in bootstrap for newsitem and paste the code in the file

In the class based components we use JS concept which is destructuring in arrays render ke andar isko likhna

let {title, description} = this.props;

Uske baad direct jaha title likhoge vha props pas ho jayega

We cannot use inline style in **react style="18 rem"** likhna galat hai

```
style={{width: "18 rem"}}
```

Ese likh sakte hai

Js ke andar ek object banaya `{ }` ese karke bahar wala bracket signifies ki vo ek `{ }` js script hai

News file mai jaake newsitem ko div ke andar daalo

```
export class NewsItem extends Component {  
  
  constructor() {  
    super();  
  }  
}
```

constructor tab tab call hoga jab ek component call hoga, matlab jab us class ka object banega

News.js file se apan apni http calls marenge aur waha se apni saari news leke aayenge toh apan **News.js** mai constructor banayenge!!!! **Newsitem** mai nahi upar bas ek example dikhaya hai

Articles ek array hai kuch is type se
Articles = [json yaha pe]

```
constructor(){
  //articles ek array hai
  super();
  console.log('Hello i am a constructor from news file or news component');
  this.state = {
    articles = this.articles
  }
}
```

Yaha state ek array hai!!!!!!!!!!!!

- State tab use karte hai jab apan ko kuch cheez baar baar change hoti huye dikhe page par bina load hote huye
- Props ko change nahi kar sakte apan, **props are read only**

Map ek higher order array method hai

```
{this.state.articles.map((element) => { console.log(element) })}
```

Map jab use karte toh ek unique key honi chahiye aur elements ko ek unique key deni padegi

```
{this.state.articles.map((element) => {
  return <div className="col-md-4" key={element.url}>
    <NewsItem props_walatile={element.title} props_waladescription =
{element.description} props_walaimageUrl={element.urlToImage}
    props_walanewsUrl = {element.url}
  />
  </div>
})}
```

Ye upar wale code se iteration kar sakte hai map wala method use karke using state concept

{js ka ek variable ke liye curly bahar lagate hai}

```
{this.state.articles.map((element) => {
  return <div className="col-md-4" key={element.url}>
```

```

        <NewItem props_wal=title={element.title.slice(0, 45)}
props_wal=description = {element.description.slice(0,88)}
props_wal=imageUrl={element.urlToImage}
        props_wal=newsUrl = {element.url}
      />
    </div>
  )}
}

```

slice(0,88) matlab 88 characters le lega text ke

componentDidMount(){
 //function tab run hoga jab render function ya method run ho jayega
 componentDidMount render ke baad run hoga

```

async componentDidMount() {
  console.log('componentDidMount');
  let url =
    'https://newsapi.org/v2/top-headlines?country=us&apiKey=d60dc46da182403294520833a18e8a0f';
  let data = await fetch(url); //using fetch api concept in JS and
  returns a promise
  let parsedData = await data.json();
  console.log(parsedData);
}

```

Ek async function await kar sakta hai kuch promises ke resolve hone ka

fetch function ek promise return karta hai

Ye jo data aaya hai tum isko text mai convert kar sakte ho ya tum usko json mai parse karna chahte ho vo kar sakte ho

Baar baar reload mat karna usse api ki limit khatam hoti hai

```

async componentDidMount() {
  console.log('componentDidMount');
  let url =
    'https://newsapi.org/v2/top-headlines?country=us&apiKey=d60dc46da182403294520833a18e8a0f';
  let data = await fetch(url); //using fetch api concept in JS and
  returns a promise
  let parsedData = await data.json();
  console.log(parsedData);
  this.setState({articles: parsedData.articles})
}

```

Main concepts studied till now are in this code:-

```
export class News extends Component {

  constructor() {
    //articles ek array hai
    super();
    console.log('Hello i am a constructor from news file or news component');
    this.state = {
      articles : [],
      loading  : false
    }
  }

  async componentDidMount() {
    console.log('componentDidMount');
    let url =
    'https://newsapi.org/v2/top-headlines?country=us&apiKey=d60dc46da182403294520833a18e8a0f';
    let data = await fetch(url); //using fetch api concept in JS and returns a promise
    let parsedData = await data.json();
    console.log(parsedData);
    this.setState({articles: parsedData.articles})
  }

  render() {
    return (
      <div className='container my-3'>
        <h2> BulletinBlaze - Top Headlines</h2>

        <div className="row">
          {/* {this.state.articles.map((element) => {
            console.log(element)
          })} */}
          {this.state.articles.map((element) => {
            return <div className="col-md-4" key={element.url}>
              <NewsItem props_wal=title={element.title ?
element.title.slice(0, 45) : ""} props_wal=description = {element.title
? element.description.slice(0,88) : ""}
props_wal=imageUrl={element.urlToImage}
              props_wal=newsUrl = {element.url}
            </div>
          )}
        </div>
      </div>
    )
  }
}
```

```

        />
      </div>
    ))}

  </div>
</div>
)
}
}

```

&page=2 karke api ki link mai jaake 2nd page par articles ko display kar sakte ho

<https://newsapi.org/v2/top-headlines?country=us&apiKey=d60dc46da182403294520833a18e8a0f&page=2>

Now implementing previous and next button kyuki json mai 36 objects the and website par sirf 20 show ho rhe the baaki ko show karwane ke liye next button banaya and pehele ko show karwane ke liye previous button banaya

```

<button type="button" class="btn btn-primary"
onClick={this.handlePrevClick}>&larr; Previous</button>
<button type="button" class="btn btn-primary"
onClick={this.handleNextClick}>Next &rarr;</button>

```

this.handlePrevClick kyuki apan class ke andar implement kar rhe hai

```

<button disabled={this.state.page<=1} type="button" class="btn
btn-primary" onClick={this.handlePrevClick}>&larr; Previous</button>

```

Button ko disabled kardo agar **this.state.page<=1** hai toh

```
handlePrevClick = () => {
```

```
  console.log('previous');
```

```
}
```

State mai **page** ko add kiya kyuki vo previous and next mai kaam aayega

```

this.state = {
  articles : [],
  loading  : false,
  page     : 1
}

```

Url mai &page=1 karo previous ke liye page=page-1
and next ke liye page=page+1

Iske baad is function mai ye add kiya

```
handlePrevClick = async () => {  
    console.log('componentDidMount');  
    let url =  
    `https://newsapi.org/v2/top-headlines?country=us&apiKey=d60dc46da182403294520833a18e8a0f&page=${this.state.page-1}`;  
    let data = await fetch(url); //using fetch api concept in JS  
    and returns a promise  
    let parsedData = await data.json();  
    console.log(parsedData);  
    this.setState({articles: parsedData.articles})  
    console.log('previous');  
    this.setState({  
        page: this.state.page-1,  
        articles: parsedData.articles  
    })  
}
```

```
handleNextClick =() => {  
  
    console.log('next');  
  
}
```

Iske baad is function mai ye add kiya

```
handleNextClick = async () => {  
    console.log('componentDidMount');  
    let url =  
    `https://newsapi.org/v2/top-headlines?country=us&apiKey=d60dc46da182403294520833a18e8a0f&page=${this.state.page+1}`;  
    let data = await fetch(url); //using fetch api concept in JS  
    and returns a promise  
    let parsedData = await data.json();  
    console.log(parsedData);  
    console.log('next');  
    this.setState({  
        page: this.state.page+1,  
    })  
}
```

```

        articles: parsedData.articles
      })
    }
  }
}

```

Kab tak next button valid hoga apan ko empty data nahi dikhana hai page par uske liye logic use kiya hai totalResults jo parsed data se number of results mile hai unko 20 se divide karke ek logic banaya hai
Usko state objects mai daaldo
And condition check se display karo

```

handleNextClick = async () => {
  console.log(this.state.totalResults);
  if (this.state.page + 1 >
Math.ceil(this.state.totalResults / 20)) {

  }
  else {
    let url =
`https://newsapi.org/v2/top-headlines?country=us&apiKey=d60dc46da182403294520833a18e8a0f&page=${this.state.page + 1}&pageSize=20`;
    let data = await fetch(url); //using fetch api concept
in JS and returns a promise
    let parsedData = await data.json();
    console.log(parsedData);
    this.setState({
      page: this.state.page + 1,
      articles: parsedData.articles,
      totalResults: parsedData.totalResults
    })
  }
}
}

```

And next button ko bhi disabled kardo jaise hi vo condition true hoti hai

```

<button disabled={this.state.page + 1 >
Math.ceil(this.state.totalResults / 20)} type="button" class="btn
btn-primary" onClick={this.handleNextClick}>Next &rarr;</button>

```

Page size ko props ke jaise pass karwa skte hai taaki number of news per page hamare upar rahe decide karne ko ye changes url mai karenge

```

pageSize=${this.props.pageSize}`;

```


Importing loading gif and placing it in spinner.js

This code is from News.js

```
return (
  <div className='container my-3'>
    <h2> BulletinBlaze - Top Headlines</h2>
    { /* agar loading == true hua toh hi spinner ko
dikhayenge else nahi dikhayenge */ }
    {this.state.loading && <Spinner/> }
  </div>
)
```

Bas loading wale comments ko padho dhyaan se jo green mai likhe hai

```
handleNextClick = async () => {
  console.log(this.state.totalResults);
  if (!(this.state.page + 1 >
Math.ceil(this.state.totalResults / this.props.pageSize))) {
    let url =
`https://newsapi.org/v2/top-headlines?country=us&apiKey=d60dc46da1
82403294520833a18e8a0f&page=${this.state.page +
1}&pageSize=${this.props.pageSize}`;

    this.setState({loading:
true});

    // jab tak fetch nahi ho rha tab
tak loading wala gif dikhao isleye await
wali line ke upar likha hai

    let data = await fetch(url); //using fetch api concept
in JS and returns a promise
    let parsedData = await data.json();

    //jaise hi parsedData mila
wese hi apan ne loading ko false kar
diya

    this.setState({loading: false});

    console.log(parsedData);
  }
}
```

```

        this.setState({
          page: this.state.page + 1,
          articles: parsedData.articles,
          totalResults: parsedData.totalResults
        })
      }
    }
  }
}

```

Point to note:-

Jab hamari app start ho rhi hai toh vo componentDidMount function se start ho rhi hai toh waha bhi loading wala lagayenge

News.js mai jaake url mai location par props pass karo

```

let url =
`https://newsapi.org/v2/top-headlines?country=${this.props.country}
&apiKey=d60dc46da182403294520833a18e8a0f&page=${this.state.page +
1}&pageSize=${this.props.pageSize}`;

```

And navbar par jao type of news set karo naam likho jaise ki business, general, sports, health, etc..

Now we will add react routes in our website taaki alag alag pages par jaa sake

npm install react-router-dom

```

import {
  BrowserRouter as Router,
  Routes,
  Route,
  Link
} from 'react-router-dom';

```

App.js file mai kuch is type se code read karo

```

render() {
  return (
    <Router>
      <div>
        <Navbar />

```

```

    <Routes>
      <Route exact path="/" element={<News pageSize={5}
country="in" category="general"/>}></Route>
      <Route exact path="/business" element={<News
pageSize={5} country="in" category="business"/>}></Route>
      <Route exact path="/entertainment" element={<News
pageSize={5} country="in" category="entertainment"/>}></Route>
      <Route exact path="/health" element={<News
pageSize={5} country="in" category="health"/>}></Route>
      <Route exact path="/science" element={<News
pageSize={5} country="in" category="science"/>}></Route>
      <Route exact path="/sports" element={<News
pageSize={5} country="in" category="sports"/>}></Route>
      <Route exact path="/technology" element={<News
pageSize={5} country="in" category="technology"/>}></Route>
    </Routes>
  </div>
</Router>
)
}

```

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

Now go to Navbar.js and replace **“a”** with **“link”** and **“href”** with **“to”** like this
One example of this is:-

```

<li className="nav-item">
  <Link className="nav-link active" aria-current="page"
to="/">Home</Link>
</li>

```

Abhi sirf website ke end points change ho rhe hai uske component wapis se **mount** nahi ho rhe ab apan chahte hai ki uske component wapis se mount ho

Directly refresh ke baad jispr click karoge vo khulega but jaise hi dusra heading click karoge to nahi khulega usi time matlab **re-mount** nahi ho rha hai

Matlab re-mount matlab force mount denge toh naya request ki tarah samjhega react uske liye apan key add karte hai key kuch bhi ho sakta hai

```

<Route exact path="/" element={<News key="in"
pageSize={5} country="in" />}></Route>

<Route exact path="/us" element={<News
key="us" pageSize={5} country="us" />}></Route>

```

Now change the title according to the country wise news shown
 Constructor ke andar props pass kardo
 Aur document.title = this.props.country; kardo

```

constructor(props) {
  //articles ek array hai
  super(props);
  console.log('Hello i am a constructor from news file or
news component');
  this.state = {
    articles: [],
    loading: false,
    page: 1
  }
  document.title = `${this.props.country} - BulletinBlaze`;
}

```

Now we want infinite scroll so we import infinite scroll in News.js

```

import InfiniteScroll from "react-infinite-scroll-component";

```

And we place our map wala iterator div inside container div in News.js file

```

<InfiniteScroll
  dataLength={this.state.articles.length}
  next={this.fetchMoreData}
  hasMore={this.state.articles.length !==
this.state.totalResults}
  loader={<Spinner />}
>
  <div className="container">
    <div className="row">
      { /* {this.state.articles.map((element) => {
        console.log(element)
      })} */ }
      {this.state.articles.map((element) => {

```

```

                                return <div className="col-md-4"
key={element.url}>
                                <NewItem
props_waltitle={element.title ? element.title : ""}
props_waladescription={element.title ? element.description : ""}
props_walaimageUrl={element.urlToImage}
                                props_walanewsUrl={element.url}
                                />
                                </div>
                                }}}
                                </div>

                                </div>
                                </InfiniteScroll>

```

Now adding top loader bar in our website ->

<https://www.npmjs.com/package/react-top-loading-bar>

App.js mai loading bar ko import kara hai

```
import LoadingBar from 'react-top-loading-bar'
```

Inside app.js we do this-

```

export default class App extends Component {

  state = {
    progress: 0
  }

  setProgress = (progress) => {
    this.setState({
      progress: progress
    })
  }

  render() {
    return (

      <div>
        <Router>

```

```

        <Navbar />
        <LoadingBar
          height={3}
          color='#f11946'
          progress={this.state.progress}
        />
        <Routes>
          <Route exact path="/" element={<News
setProgress={this.setProgress}    key="in"
pageSize={5} country="in" />}></Route>
          <Route exact path="/us" element={<News
setProgress={this.setProgress}    key="us"
pageSize={5} country="us" />}></Route>
        </Routes>
      </Router>
    </div>

  )
}
}

```

```

<LoadingBar
  height={3}
  color='#f11946'
  progress={this.state.progress}
/>

<Routes>
  <Route exact path="/" element={<News
setProgress={this.setProgress}    key="in" pageSize={5} country="in"
/>}></Route>
  <Route exact path="/us" element={<News
setProgress={this.setProgress}    key="us" pageSize={5} country="us"
/>}></Route>
</Routes>
</Router>
</div>

```

```

async componentDidMount() {
  this.props.setProgress(10);
}

```

```

        console.log('componentDidMount');
        let url =
`https://newsapi.org/v2/top-headlines?country=${this.props.country}&api
Key=d60dc46da182403294520833a18e8a0f&page=1&pageSize=${this.props.pageS
ize}`;

        // jab tak fetch nahi ho rha tab tak loading wala gif dikhao
        isleye await wali line ke upar likha hai

        this.setState({ loading: true });
        let data = await fetch(url); //using fetch api concept in JS
        and returns a promise

        this.props.setProgress(30);

        let parsedData = await data.json();
        this.props.setProgress(50);

        //jaise hi parsedData mila wese hi apan ne loading ko false kar
        diya

        this.setState({ loading: false });
        console.log(parsedData);
        this.setState({
            articles: parsedData.articles,
            loading: false
        })

        this.props.setProgress(100);
    }

```

Now hiding api key:-

.env.local file banao main folder ke andar

```
REACT_APP_NEWS_API = 'd60dc46da182403294520833a18e8a0f';
```

Ye line likho

Ab App.js mai isko add karo-

```

export default class App extends Component {
    apiKey = process.env.REACT_APP_NEWS_API;

```

Props pass kardo App.js mai

```

<Route exact path="/" element={<News setProgress={this.setProgress}
propswalaapiKey = {this.apiKey} key="in" pageSize={5}
country="in" />}> </Route>
    <Route exact path="/us" element={<News
setProgress={this.setProgress} propswalaapiKey =
{this.apiKey} key="us" pageSize={5} country="us" />}></Route>

```

News.js ke url mai props pass kardo

```

let url =
`https://newsapi.org/v2/top-headlines?country=${this.props.country}&api
Key=${this.props.propswalaapiKey
}&page=1&pageSize=${this.props.pageSize}`;

```

Now converting class based components to function based components:-

This is a class based component so we will convert it to function based component

```

c > components > JS Spinner.js > Spinner > render
1  import React, { Component } from 'react'
2  import loading from './loading.gif'
3
4  export class Spinner extends Component {
5      render() {
6          return (
7              <div className="text-center">
8                  <img className="my-3" src={loading} alt="loading spinner" />
9              </div>
10         )
11     }
12 }
13
14 export default Spinner

```

converted code:-


```

components > Spinner.js > default
1 import React, { Component } from 'react'
2 import loading from './loading.gif'
3
4 const Spinner = () => {
5
6     return (
7         <div className="text-center">
8             <img className="my-3" src={loading} alt="loading" />
9         </div>
10     )
11 }
12
13
14 export default Spinner
15

```

2)

```

static defaultProps = {
    country: 'in',
    pageSize: 8,
    category: 'general',
}

static propTypes = {
    country: PropTypes.string,
    pageSize: PropTypes.number,
    category: PropTypes.string,
}

export default News

```

```

News.defaultProps = {
  country: 'in',
  pageSize: 8,
  category: 'general',
}

News.propTypes = {
  country: PropTypes.string,
  pageSize: PropTypes.number,
  category: PropTypes.string,
}

export default News

```

3)

```

constructor(props) {
  super(props);
  this.state = {
    page: 1,
    totalResults: 0
  }
}

```

```

const [articles, setArticles] = useState([])
const [loading, setLoading] = useState(true)
const [page, setPage] = useState(1)
const [totalResults, setTotalResults] = useState(0)

```

4)

```

this.setState({
  articles: parsedData.articles,
  totalResults: parsedData.totalResults,
  loading: false,
})

```

```

setArticles(parsedData.articles)
setTotalResults(parsedData.totalResults)
setLoading(false)

```

5)

```
    async componentDidMount() {  
        |   this.updateNews();  
    }  
}
```

```
    useEffect(() => {  
        |   this.updateNews();  
    }, [])
```