

Adding Categories & propTypes to NewsMonkey React App | Complete React Course in Hindi #30

In the last video, we have enhanced our NewsMonkey Application by adding a loading spinner. In today's tutorial, we would pass categories, such as Sports, General, Entertainment, Business, etc in the navbar of the application. So without further ado let's begin:

Passing "country" as a Prop

Earlier, we have passed India as a country to the filters of the NewsItems. Now, since we are going to change the country for other different NewsCategories so we would be passing 'country' as a Prop. To do so follow the below steps:

1. Firstly, You need to pass the country as a prop from App.js as shown below. For the demo, we are passing the country like India:

```
<News pageSize={5} country= "in" />
```

2. Secondly, we would be using the created country prop in URL to filter the News as shown below:

```
let url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&apiKey=db57b02`
```

Prop types in Class-Based Component

React has a mechanism for props authentication called prop types. As some functions need compulsory arguments, similarly in react components might require a prop to be defined. Otherwise, it won't render properly.

Using PropTypes with static variable

Since we are about to use prop types in our application so firstly we have to import them into the "News.js" as follows:

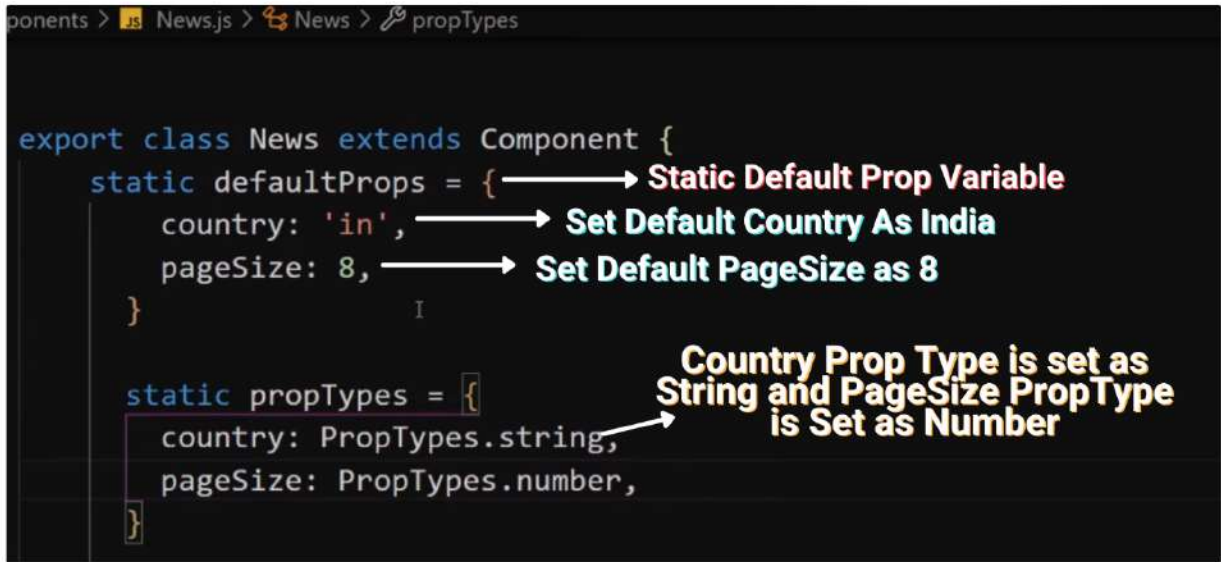
```
import PropTypes from 'prop-types'
```

Static variable

The static variable can be used to refer to the common property of all objects. Static is the property that belongs to the class only. Earlier in React, we used to define props outside of the class. But now we will be defining them inside the class by using the static variable.

Creating Static Variable

Now, we would create a static variable inside the class to write proptype. Remember, we would be using static default props as well.



```
components > .js News.js > News > propTypes

export class News extends Component {
  static defaultProps = {
    country: 'in',
    pageSize: 8,
  }

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

Annotations in the image:

- Static Default Prop Variable (points to `static defaultProps = {`)
- Set Default Country As India (points to `country: 'in',`)
- Set Default PageSize as 8 (points to `pageSize: 8,`)
- Country Prop Type is set as String and PageSize PropType is Set as Number (points to `country: PropTypes.string,` and `pageSize: PropTypes.number,`)

Figure 1.1: Using Prop Types with Static variable

Till now, we have created two props, which are 'country' and 'Page size'. Here, we have passed the default value to the props also we have declared the prop type as string and number respectively for the two values. Later on, we would be adding more proptype in this static variable.

Creating Categories

Moreover, we would like to show the different categories in the Navbar such as Business, Entertainment, General, Health, Science, Sports, and technology. To display these categories in the Navbar, You will have to add all the above categories in the Navbar.js as list items.

```
<div className="collapse navbar-collapse" id="navbarSupportedContent">
  <ul className="navbar-nav me-auto mb-2 mb-lg-0">
    <li className="nav-item">
      <a className="nav-link" aria-current="page" href="/">Home</a>
    </li>
    <li className="nav-item"><a className="nav-link" href="/about">About</a> </li>
    <li className="nav-item"><a className="nav-link" href="/about">business</a></li>
    <li className="nav-item"><a className="nav-link" href="/about">entertainment</a></li>
    <li className="nav-item"><a className="nav-link" href="/about">general</a></li>
    <li className="nav-item"><a className="nav-link" href="/about">health</a></li>
    <li className="nav-item"><a className="nav-link" href="/about">science</a></li>
    <li className="nav-item"><a className="nav-link" href="/about">sports</a></li>
    <li className="nav-item"><a className="nav-link" href="/about">technology</a></li>
  </ul>
</div>
```

All categories as List Items in Navbar.js

We Will changing the href with 'Link to', later on

Figure 1.2: Adding Categories in Navbar.js

Result: We have successfully added all the categories in the Navbar. At this moment all the href links are /about so clicking on any of the categories will still render the same news.

Note: We would be adding the proper link to these categories later on.

Passing Categories as Props

Till now, we have created the different categories, but now we would like to pass the category from App.js. For the demo, we are passing the category as science in the following manner:

```
<News pageSize={5} country= "in" category="science" />
```

Secondly, we have to embed this category in the URL. To do so we have to first add the category in the static default prop and static prop type variable.

```
export class News extends Component {
  static defaultProps = {
    country: 'in',
    pageSize: 8,
    category: 'general',
  }

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

In Static Default Prop Variable

Set Default Category As General

In Static Proptype Variable

Set the proptypes of Category As String

Figure 1.3: Set Prop Type of Category

Now, We can easily pass the Category to the URL of componentDidMount, handlePrevClick, and handleNextClick function as follows:

```
let url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this
```

Result: Here, we have passed the category as 'science' and as a result, we are getting all the NewsItems related to Science.

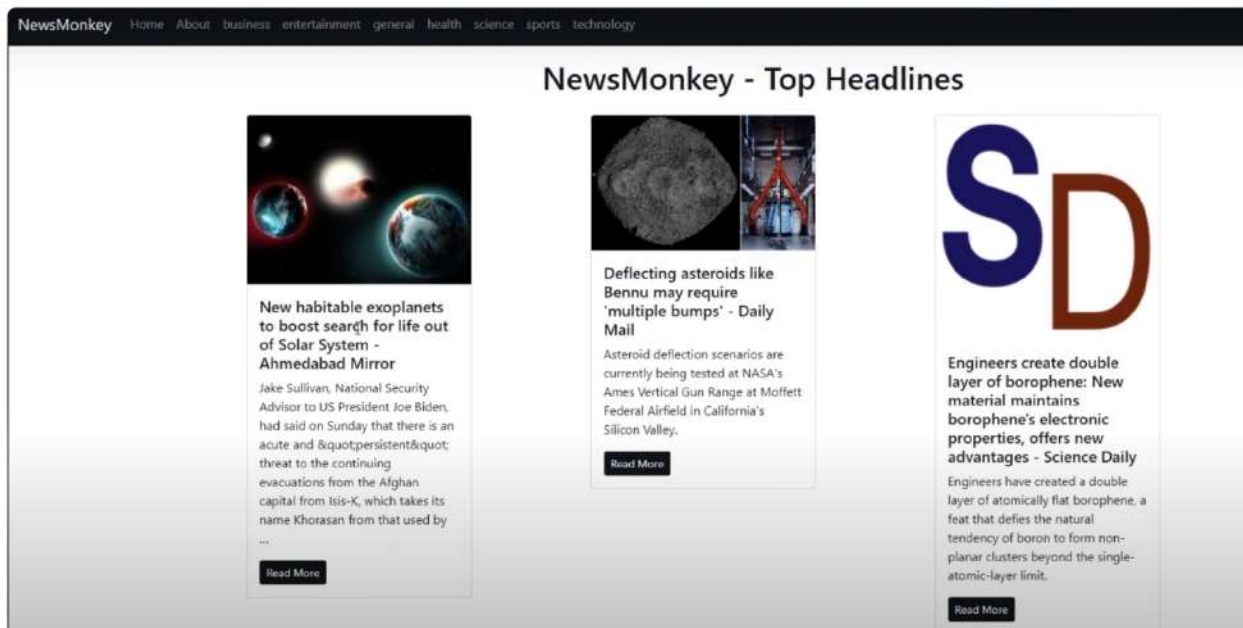


Figure 1.4: NewsMonkey Application