Refactoring News component to use the same function | Complete React Course in Hindi #33

In the previous video, we have enhanced the NewsItems of the application. In this tutorial, we would be refactoring our code as it is unorganized and complex to understand with too many functions.

Creating update Function

At this time, we are using the same function(updateNews) in componentDidMount, handlePrevClick, and handleNextClick function, just with a slight update in the page number. This time, we would create a function that would be used in place of writing this same code again and again. So, let's create a 'updateNews' Function as shown below:

```
async updateNews(pageNo){
    const url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=
    this.setState({ loading: true });
    let data = await fetch(url);
    let parsedData = await data.json()
    this.setState({
        articles: parsedData.articles,
        totalResults: parsedData.totalResults,
        loading: false
    })
}
```

Here, we are passing the page number in the function as it is different at different places of the function. Now, we would be using this function in place of calling different functions each time. This allows us to make any changes to our application very swiftly.

Using the Update News Function

Below, we have used the "updateNews" Function while invoking the componentDidMount, handlePrevClick, and handleNextClick function.

For componentDidMount

Inside componentDidMount, we would like to render the newsItems by using the Update News Function as shown below:

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

For HandlePrevClick

In the handlePrevClick function, we would like to run the update news function by decreasing the state of page number by 1 as shown below:

```
handlePrevClick = async () => {
    this.setState({ page: this.state.page - 1 });
    this.updateNews();
}
```

For HandleNextClick Function

Similarly, In the handlePrevClick function, we would like to run the update news function by increasing the state of page number by 1 as shown below:

```
handleNextClick = async () => {
    this.setState({ page: this.state.page + 1 });
    this.updateNews();
}
```

NewsMonkey Application: Hence, after removing the unnecessary logic from our NewsMonkey application the app is looking more concise and neat. There is no big change in the application but the same task is being performed by a more precise and optimized code.
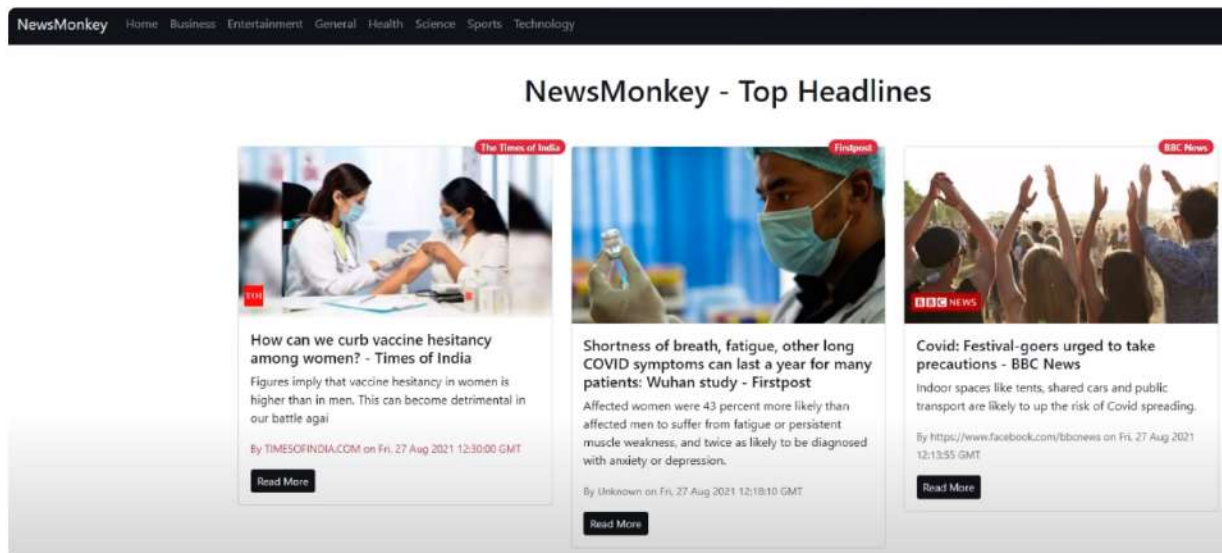

Figure 1.1: The NewsMonkey Application