Adding more Logic to TextUtils | Complete React Course in Hindi #8

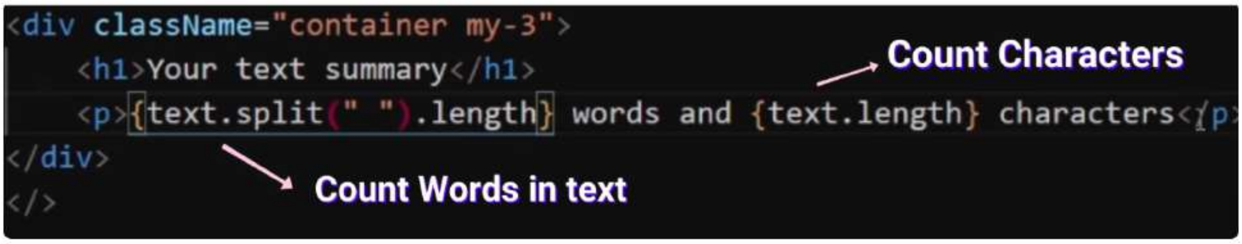
In the previous tutorial, we have added the feature of converting our text to uppercase. In this tutorial, we will be adding some more features to our app. So without any further ado let's begin:

Editing textform.js:

Earlier, we have created a textbox in textform.js and have rendered it in our application with the help of "app.js". Create a container that will have a summary of our text. Let's analyze our text by creating some more features:

1. Counting Words and Characters

Now, let's add a feature of counting the number of words and characters in our text. Inside the new div tag in Transform.js we would add our code.



```
<div className="container my-3">
    <h1>Your text summary</h1>
    <p>{text.split(" ").length} words and {text.length} characters</p>
</div>
</>
```

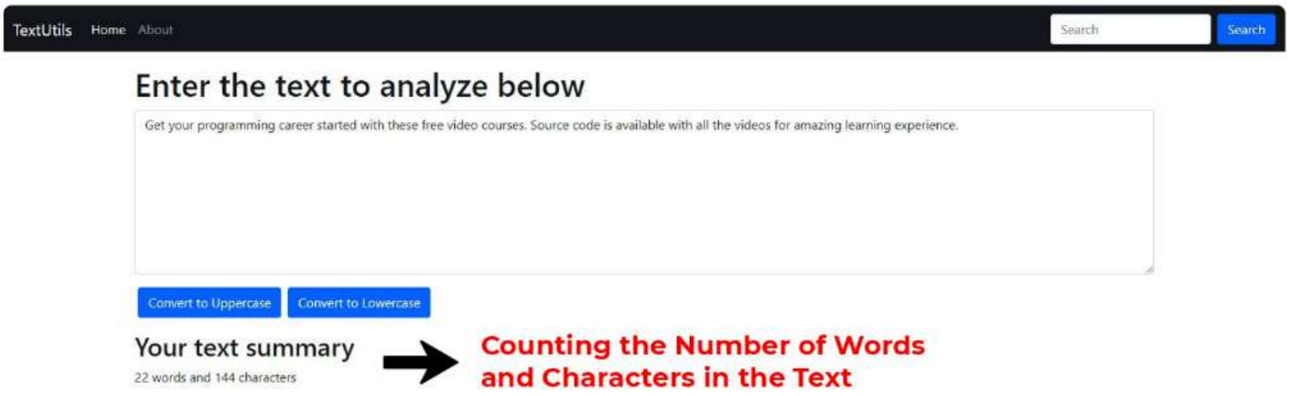
Count Characters

Count Words in text

Figure 1.1: Counting Words and Character

In the above code, *{text.length}* counts the length of text, that is the number of characters. *Javascript split()* is used to split the given string in an array. Here, we have used the split function, *{text.split(" ").length}*, to count the number of words in the given text.

Result: We have successfully added the word and Character counter for our app.



Enter the text to analyze below

Get your programming career started with these free video courses. Source code is available with all the videos for amazing learning experience.

Your text summary

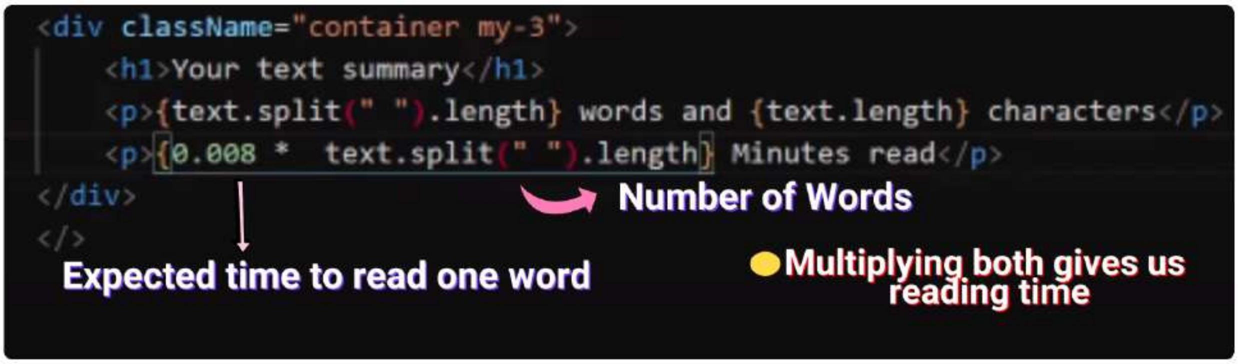
22 words and 144 characters

Counting the Number of Words and Characters in the Text

Figure 1.2: Result of the New feature

2. Calculating Reading Time

Now let's calculate the expected reading time required to read the given set of text in our textbox. For humans, On average to read a single word takes 0.008 minutes, which means if we multiply the number of words by 0.008 minutes. It will show us the time required to read the given text in minutes.



```
<div className="container my-3">
    <h1>Your text summary</h1>
    <p>{text.split(" ").length} words and {text.length} characters</p>
    <p>{0.008 *  text.split(" ").length} Minutes read</p>
</div>
</>
```

Number of Words
Expected time to read one word
Multiplying both gives us reading time

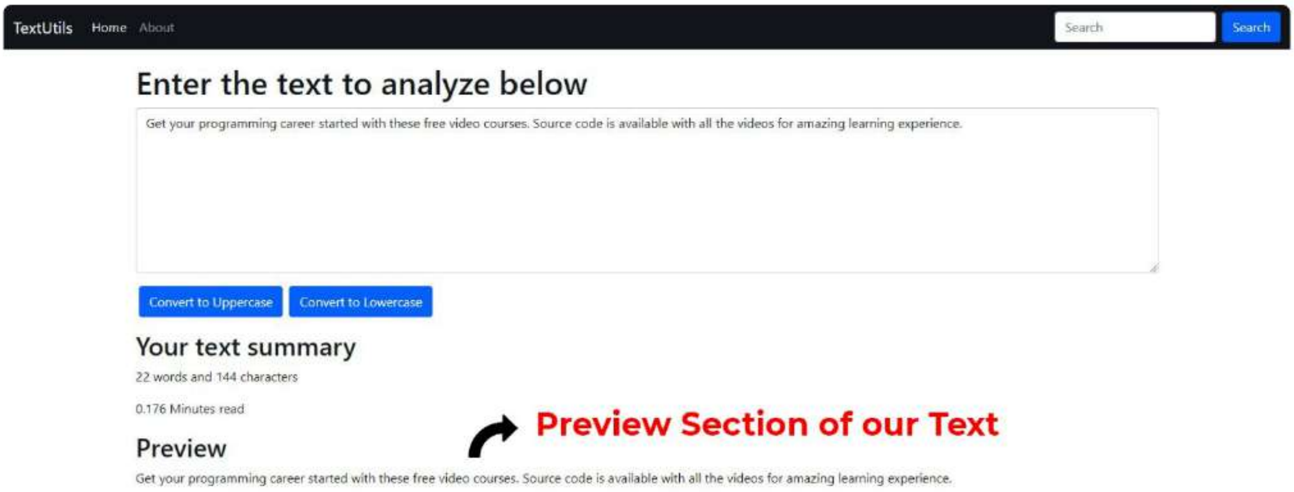
Figure 1.3: Calculating Reading time

The above function will display the expected time it will take for a person to read the given text.

3. Preview Section of text

We would like to add a preview section to read our text thoroughly so we would simply use, *{text}*, to display our text.

Result: Our Preview Section is being Added.



Enter the text to analyze below

Get your programming career started with these free video courses. Source code is available with all the videos for amazing learning experience.

Your text summary

22 words and 144 characters

0.176 Minutes read

Preview

Get your programming career started with these free video courses. Source code is available with all the videos for amazing learning experience.

Preview Section of our Text

Figure1.4: Added Preview section of our text

4. Convert text to lowercase

Let’s create a convert to Lowercase Button for our app and assign a function to it. So that it can convert text to Lowercase.

Adding Button:

```
<button className= "btn btn-primary" onClick={handleloClick}> Convert to Lowercase</button>
```

Adding a function:

We are creating a function named "handleLoClick". It is similar to the function we have created for changing text to uppercase.
Code of *handleLoClick* function:

```
const handleLoClick = () => {
    console.log("lowercase was clicked");
    let newText = text.toLowerCase();
    setText("new text")
}
```

Result: On clicking the button our text will change to Lowercase.