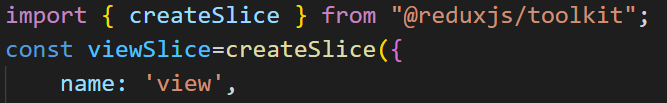
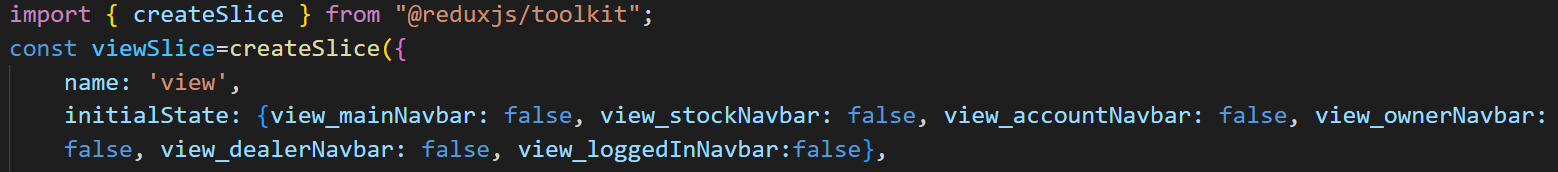
**System Of Changing Navbar As Required In The Different Pages**

1. First we have to make a folder named **store** in the **src** folder of the project
2. Than in it we have to make the different slice as per our requirement, Example: **view-slice.js,** And we have made this slice for the navbar, We can also make this type of slice for the other components also
3. Now before using this system you have to install the following dependencies:
   1. @reduxjs/toolkit
   2. react-router-dom
   3. react-redux
4. Now first go to the view-slice.js, And will make the code in it:
   1. First import the createSlice from the @reduxjs/toolkit
   2. Than make the function named viewSlice and in that use the hook named createSlice which we had imported, As shown in the figure below:



* 1. And than give an name for the slice, In this example we will give the name as the view, As shown in the upper image
  2. And than write the initialState and there declare the variables that are to be used, As shown below:



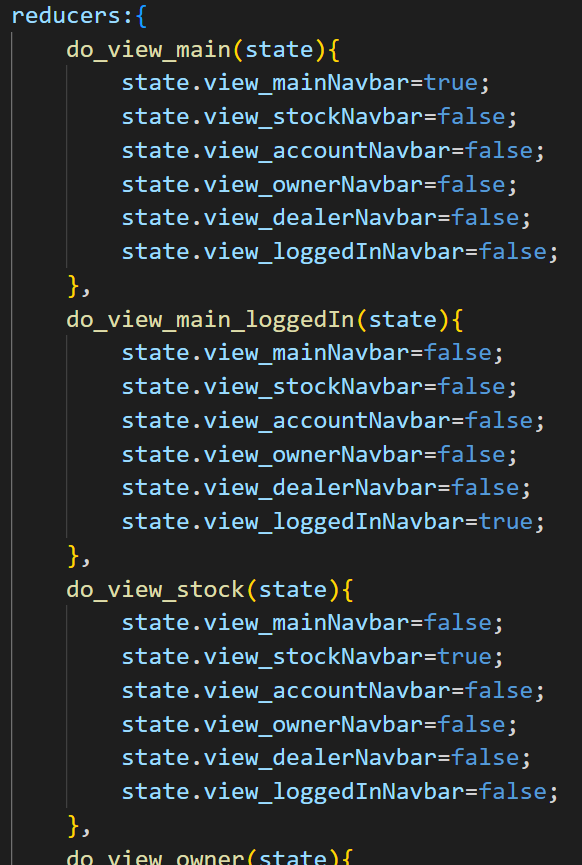
* 1. And than now comes the final part of the in which we will declare the different methods under the reduces’s name

🡪In which we will give the value of the each of the variable that is declared above, And this value is applicable when that particular function is called, We will talk about the calling of this function in the later points

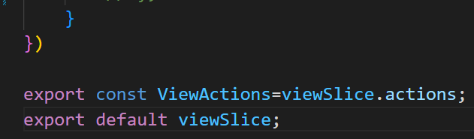
🡪We will make the method for each of the cases, In the cases where we want to change the availability of the navbar

🡪If we are using this method for any other case than that of the navbar than we have to think about different way

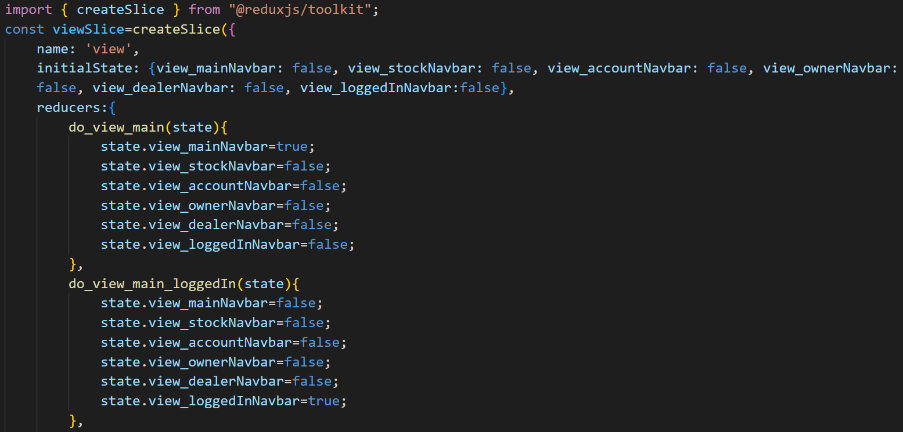
🡪Example of the reducers is as follow:

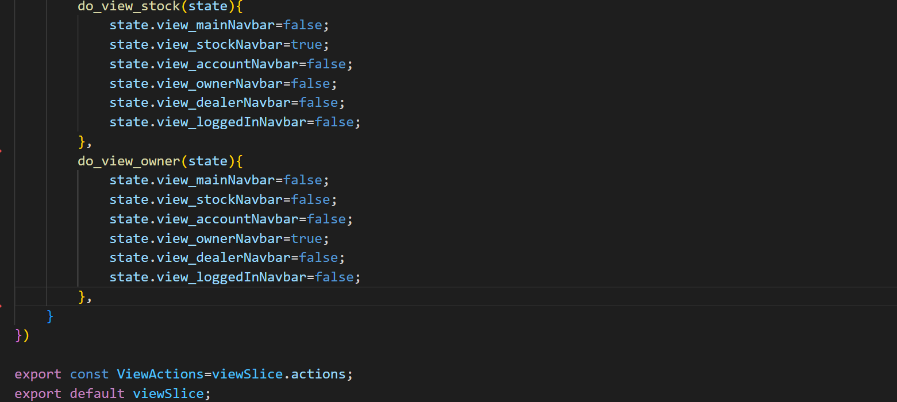


* 1. And than at the last of the slice we will give the name of the slice in the terms of the actions, and will call the action method of this slice and than at the last we will export the viewSlice, As show belo:

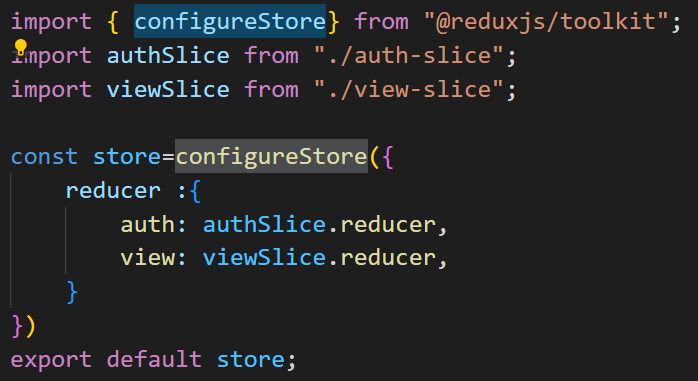


* 1. The Whole slice which we have made is as shown below in the one picture:





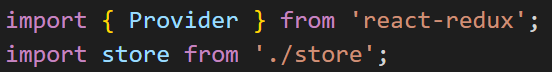
1. And than in the store folder we will make an index.js file, Steps for it is as shown below:
   1. First we will import the configureStore from the @reduxjs/toolkit
   2. Than we will import the viewSlice that we have made in the above point in the index.js
   3. Than we will make an method by the name of the store and than we will use the configureStore method and than we will use the reducer method in it and than we will write the name that we have declared in the slice in createSlice method and than will write the imported name of that slice and than will use the reducer method of that, Example is as shown below:



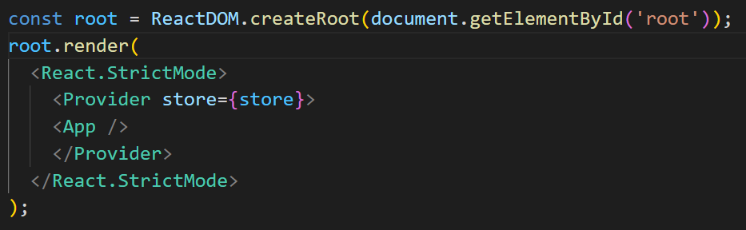
🡪And do not forget to add the export default statement at the last

🡪And in this example we have made the two slices but for now as we discussed we have made only one slice, The image is from other project

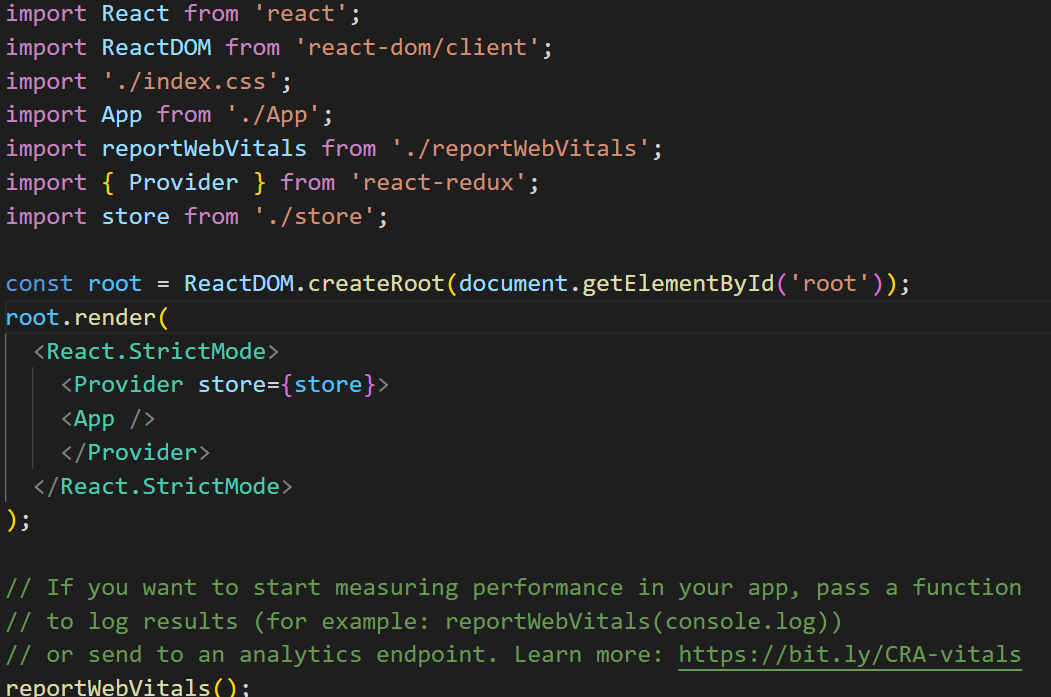
1. Than we will go to the index.js file of the src folder not of the store folder, And in it we will follow the following steps:
   1. First we will import the following two statements:



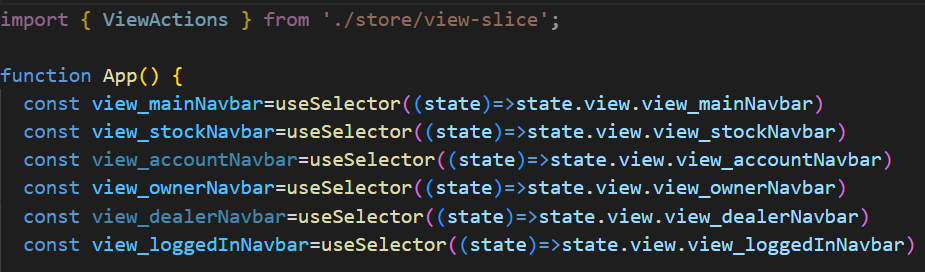
* 1. Than we will cover the app tag in the index.js by the Provider tags and in it we will pass the store as the parameter, As shown in the figure below:



* 1. The whole index.js file of the src folder looks like:

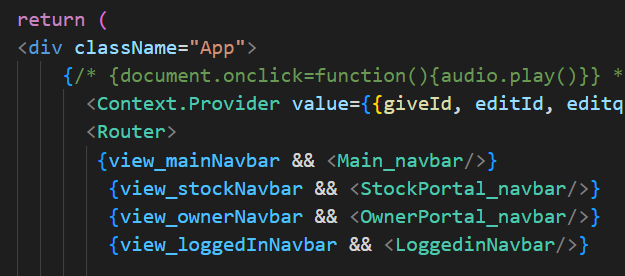


1. Now in the App.js we will declare all the variables that we want to use in the starting by help of the useSelector hook which is coming from the react-redux, Example:



1. Now in App.js we will import viewActions from the view-slice which is in the store folder, And will also import the required navbars on which this system is to be applied

🡪And than in the JSX part and in the router tags we will put the condition while calling the specific navbar, Sample of this is as shown below:

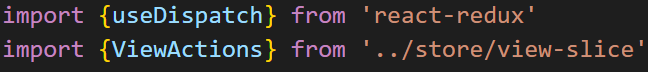


🡪For example let we take the first condition, Which will be executed when the view\_mainNavbar will be true and if it is true than only the Main\_navbar will be called, And the occurring of the true and false of the view\_mainNavbar is due to the functions which are runned on the starting of the particular file when that url is called

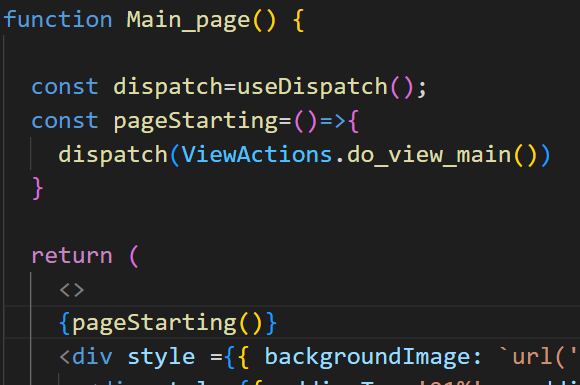
1. Now let see what we have to put on the starting of the each of the file, And this code will be runned when the url is called:

🡪For sample this is the case of the main page, There would be some change for the each of the pages, And that change is mentioned below in the points

* 1. For applying this method we have to import the following two statement, In which the second one is of the slice, Which is changed as per the slice name and its address, Sample import code is as follow:



* 1. And than comes the master code:

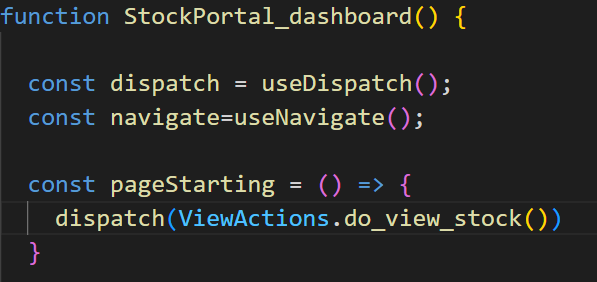


🡪In this we have to make the mainNavbar to be visible, So that we have used the .do\_view\_main method of the ViewActions, If we want the another navbar than we would have used the different method, As per our requirment

🡪And we have also declared the dispatch by help of the useDispatch() hook

🡪And than at the last on the starting of the code in the JSX we will call the pageStarting function that we have made and our work is completed

1. Now If we want to have the navbar of the stockPortal than we have used the following one method as shown below:



🡪And in this method the only change is that we have called the .do\_view\_stock() method of the ViewActions

🡪And by this it will run this method that is present in the view-slice and will change the variables according to it

1. Now the method will be remained same only the variable and slice name would be changed for any other project or the components