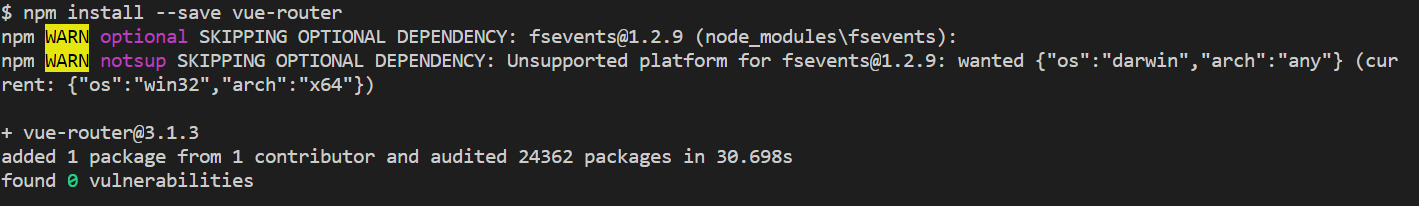
**VUE JS**obal

* **Npm install –g @vue/cli-service-global**
* **For vuex spread operator we have to use bellow commend**
* **npm install --save-dev @babel/preset-stage-2**
* 
* **ROUTER**
* **npm install --save vue-router**
* **create separate file for router to make easy**
* 
* **Main.js**
* import VueRouter from 'vue-router'
* //vue router
* Vue.use(VueRouter);

**Routes.js**

import Home from './components/Home.vue'

import Portfolio from './components/portfolio/Portfolio.vue'

import Stocks from './components/stockts/Stocks.vue'

export const routes = [

    {path:'/',component:Home},

    {path:'/portfolio',component:Portfolio},

    {path:'/stocks',component:Stocks},

    {path:'/stocks',component:Stocks}

]

**Main.js**

import VueRouter from 'vue-router'

//router

import {routes} from './routes.js';

//vue router

Vue.use(VueRouter);

const router = new VueRouter({

  mode:'history',

  routes

});

Vue.config.productionTip = false

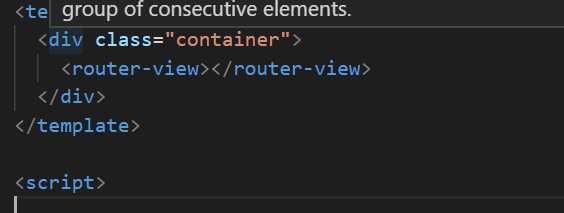
new Vue({

  router,

  render: h => h(App),

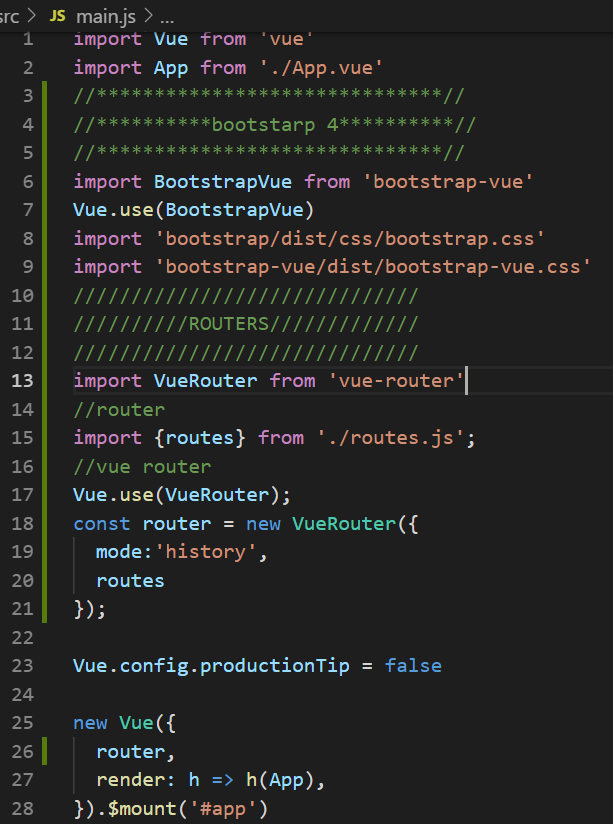
}).$mount('#app')

**App.Vue**

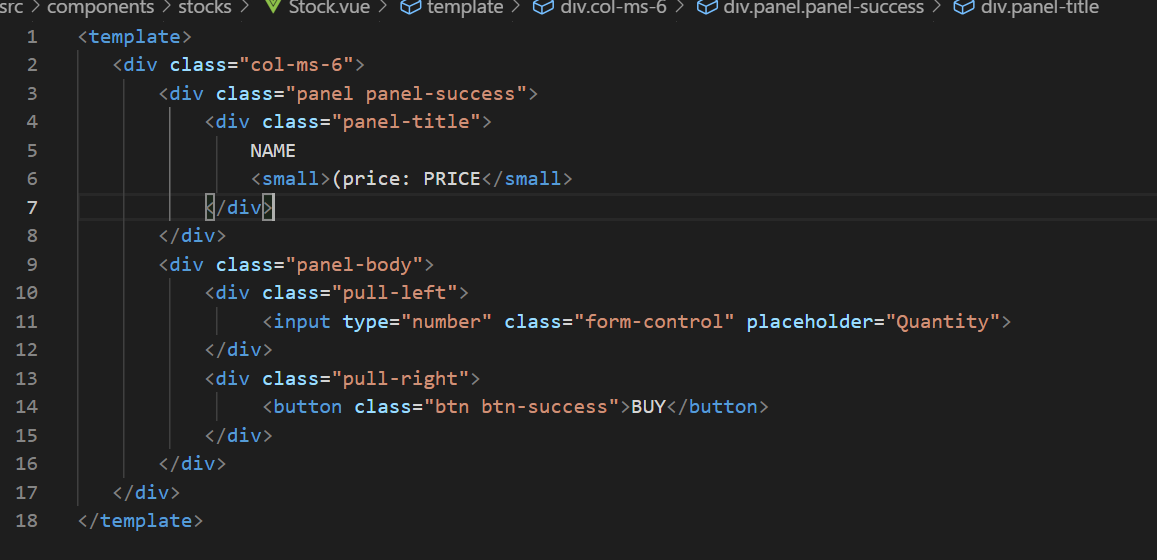


**BOOTSTRAP INSTALL**

npm install vue bootstrap-vue bootstrap



**Stock.vue**



**Stocks.vue**



**Header.js**

<template>

<div>

  <b-navbar toggleable="lg" type="dark" variant="info">

    <b-navbar-brand ><router-link class="white" to="/">Stock Render</router-link></b-navbar-brand>

    <b-navbar-toggle target="nav-collapse"></b-navbar-toggle>

    <b-collapse id="nav-collapse" is-nav>

      <b-navbar-nav>

        <b-nav-item ><router-link class="white" to="/portfolio">Portfolio </router-link></b-nav-item>

        <b-nav-item ><router-link class="white" to="/stocks">Stocks</router-link></b-nav-item>

      </b-navbar-nav>

      <!-- Right aligned nav items -->

      <b-navbar-nav class="ml-auto">

        <!-- <b-nav-form>

          <b-form-input size="sm" class="mr-sm-2" placeholder="Search"></b-form-input>

          <b-button size="sm" class="my-2 my-sm-0" type="submit">Search</b-button>

        </b-nav-form> -->

        <b-navbar-nav>

        <b-nav-item ><router-link class="white" to="/stocks">End Day</router-link></b-nav-item>

        </b-navbar-nav>

        <b-nav-item-dropdown right>

          <!-- Using 'button-content' slot -->

          <template v-slot:button-content>

            <em>Save & Load</em>

          </template>

          <b-dropdown-item href="#">Save</b-dropdown-item>

          <b-dropdown-item href="#">Load</b-dropdown-item>

        </b-nav-item-dropdown>

      </b-navbar-nav>

    </b-collapse>

  </b-navbar>

</div>

</template>

<script>

</script>

<style scoped>

body {

  font-family: Menlo, Consolas, monospace;

  color: #444;

}

.item {

  cursor: pointer;

}

.bold {

  font-weight: bold;

}

ul {

  padding-left: 1em;

  line-height: 1.5em;

  list-style-type: dot;

}

a:hover{

  text-decoration: none

}

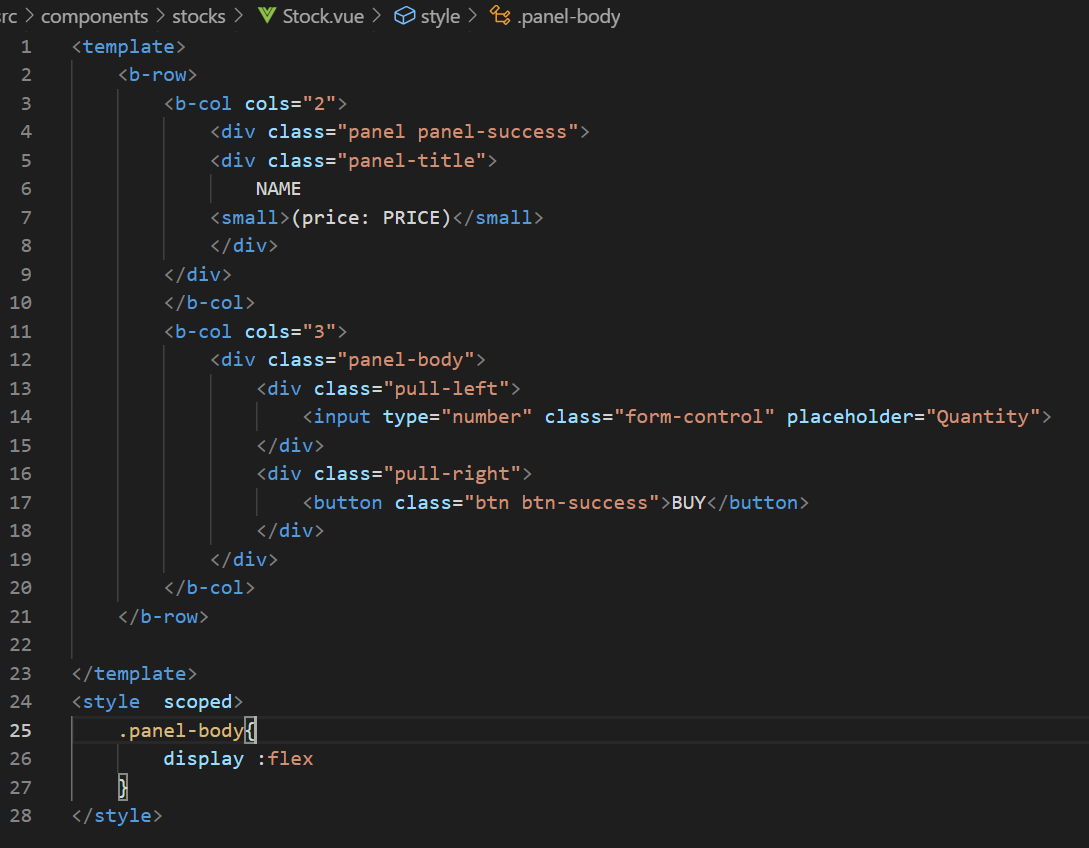
.white{

  color: white!important

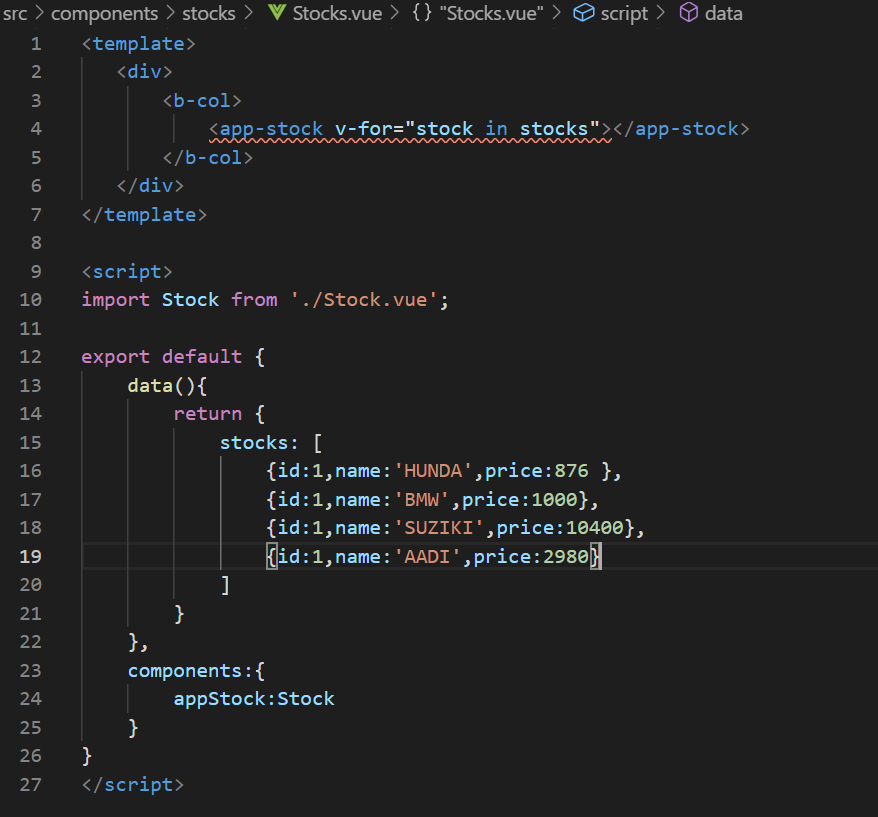
}

</style>

**Stock.vue**



**Stock.vue**



**VUEX-STORE MANAGMENAT**



**Create store folder**

1. import Vue from 'vue'
2. import Vuex from 'vuex'
3. //stocks modules
4. import stocks from './modules/stocks';
5. Vue.use(Vuex)
6. export default new Vuex.Store({
7. modules: {
8. stocks
9. }
10. });

**Create modules folder 🡪stocks.js**

import stocks from '../../data/stocks';

const state = {

    stocks:[]

};

const mutations = {

    'SET\_STOCKS'(state,stocks){

        state.stocks = stocks

    },

    'RND\_STOCKS'(state){

    }

}

const actions = {

    buyStock:({commit},order)=>{

        commit();

    },

    initStocks:({commit})=>{

        commit('SET\_STOCKS', stocks)

    },

    randomizeStocks:({commit})=>{

        commit('RND\_STOCKS')

    }

}

const getters = {

    stocks: state => {

        return state.stocks;

    }

}

export default {

    state,

    mutations,

    actions,

    getters

}

**Create dammi data folder in stocks.js**

export default [

    { id:1, name:'HUNDA', price:876 },

    { id:1, name:'BMW', price:1000 },

    { id:1, name:'SUZIKI', price:10400 },

    { id:1, name:'AADI', price:2980 }

]

**Import store in main.js**

//store

import store from './store';

new Vue({

  router,

  store,

  render: h => h(App),

}).$mount('#app')

**App.vue init the dispatch**

<template>

  <div>

    <Header />

    <main class="container">

      <router-view></router-view>

    </main>

  </div>

</template>

<script>

import Header from './components/Header.vue';

export default {

  components: {

    Header

  },

  created(){

    this.$store.dispatch('initStocks');

    console.log( this.$store.dispatch('initStocks'))

  }

}

</script>

<style>

</style>

**Stocks.vue**

**{**

computed:{

        stocks(){

            console.log(this.$store.getters.stocks)

            return this.$store.getters.stocks;

        }

    }

**}**

**Linke**

<template>

   <div>

       <b-col>

           <app-stock v-for="stock in stocks" :stock="stock"></app-stock>

       </b-col>

   </div>

</template>

<script>

import Stock from './Stock.vue';

export default {

    components:{

        appStock:Stock

    },

    computed:{

        stocks(){

            console.log(this.$store.getters.stocks)

            return this.$store.getters.stocks;

        }

    }

}

</script>

**Stock.js**

<template>

    <b-row>

        <b-col cols="2">

            <div class="panel panel-success">

            <div class="panel-title">

                {{stock.name}}

            <small>(price: {{stock.price}})</small>

            </div>

        </div>

        </b-col>

        <b-col cols="3">

            <div class="panel-body">

                <div class="pull-left">

                    <input type="number" class="form-control" placeholder="Quantity" v-model="quantity">

                </div>

                <div class="pull-right">

                    <button class="btn btn-success"

                        @click="buyStock"

                         >BUY</button>

                </div>

            </div>

        </b-col>

    </b-row>

</template>

<script>

export default {

    props:['stock'],

    data(){

        return {

            quantity:0

        }

    },

    methods:{

        buyStock(){

            const order = {

                stockId : this.stock.id,

                stockPrice:this.stock.price,

                quantity:this.quantity,

                price:this.stock.price\*this.quantity

            };

            console.log(order);

            this.quantity = 0;

        }

    }

}

</script>

<style  scoped>

    .panel-body{

        display :flex

    }

</style>

