user.js

import { request } from "@/utils";  
export function loginAPI(formData) {  
 return request({  
 url: '/user/login',  
 method: 'POST',//请求方式  
 data: formData  
 })  
}  
export function getProfileAPI() {  
 return request({  
 url: '/personal',  
 method: 'GET',//请求方式  
 })  
}  
export function updateProfileAPI(formData) {  
 return request({  
 url: '/user/update',  
 method: 'POST',//请求方式  
 data: formData  
 })  
}  
export function registerAPI() {  
 return request({  
 url: '/user/add',  
 method: 'POST',  
 })  
}  
export function sendVerificationCodeAPI(){  
 return request({  
 url: '/user/send',  
 method: 'POST',  
 })  
}

App.js

import { BrowserRouter, Routes, Route } from 'react-router-dom'  
import { useState } from 'react'  
import Login from "./pages/Login/Login"  
import Lesson from './pages/Lesson/Lesson'  
import CZone from './pages/Lesson/CZone/CZone'  
import PythonZone from './pages/Lesson/PythonZone/PythonZone'  
import FirstContent from './pages/FirstContent/FirstContent'  
import Question from './pages/Question/Question'  
import Contact from './pages/Contact/Contact'  
import News from './pages/Contact/News/News'  
import Attention from './pages/Contact/Attention/Attention'  
import Logon from "./pages/Logon/Logon"  
import Amend from "./pages/Amend/Amend"  
import Letter from "./pages/Letter/Letter"  
import Official from './pages/Letter/Official/Official'  
import Personal from "./pages/Personal/Personal"  
import Reply from "./pages/Letter/Reply/Reply"  
import All from './pages/Letter/Official/All/All'  
import Comment from './pages/Letter/Reply/Comment/Comment'  
import Examine from "./pages/Letter/Official/Examine/Examine"  
import Notice from "./pages/Letter/Official/Notice/Notice"  
import Competition from './pages/Personal/Competition/Competition'  
import Data from "./pages/Personal/Data/Data"  
import Information from "./pages/Personal/Prove/Prove"  
import Project from "./pages/Personal/Project/Project"  
import Publish from "./pages/Publish/Publish"  
import Match from './pages/Contact/Match/Match'  
function App () {  
 const [posts, setPosts] = useState([]);  
 const addPost = (newPost) => {  
 setPosts((prevPosts) => [...prevPosts, newPost]);  
 };  
 return (  
 <BrowserRouter>  
 <Routes>  
 <Route path='/' element={<Login />} />  
 <Route path='/Lesson/\*' element={<Lesson />}></Route>  
 <Route path='/PythonZone' element={<PythonZone />}></Route>  
 <Route path='/CZone' element={<CZone />}></Route>  
 <Route path='/FirstContent' element={<FirstContent />}></Route>  
 <Route path='/Question' element={<Question />}></Route>  
 <Route path='/News' element={<News />}></Route>  
 <Route path='/Attention' element={<Attention />}></Route>  
 <Route path='/Match' element={<Match />}></Route>  
 <Route path="/contact/\*" element={<Contact posts={posts} />} />  
 <Route path='/Logon' element={<Logon />} />  
 <Route path='/Amend' element={<Amend />} />  
 <Route path='/Letter/\*' element={<Letter />} />  
 <Route path='/Personal/\*' element={<Personal />} />  
 <Route path='/Reply' element={<Reply />} />  
 <Route path="/Official" element={<Official/>}></Route>  
 <Route path='/All' element={<All />} />  
 <Route path='/Notice' element={<Notice />} />  
 <Route path='/Examine' element={<Examine />} />  
 <Route path='/Competition' element={<Competition />} />  
 <Route path='/Data' element={<Data />} />  
 <Route path='/Information' element={<Information />} />  
 <Route path='/Project' element={<Project />} />  
 <Route path="/publish" element={<Publish addPost={addPost} />} />  
 <Route path='/Match' element={<Match />} />  
 </Routes>  
 </BrowserRouter>  
 )  
}  
export default App

App.js

import { useEffect, useRef, useState } from 'react'  
import './App.scss'  
import avatar from '../../assets/images/personalheadlike.png'  
import \_ from 'lodash'  
import classNames from 'classnames'  
import { v4 as uuidV4 } from 'uuid'  
import dayjs from 'dayjs'  
import axios from 'axios'  
import { message } from 'antd' // 引入 Ant Design 的 message 组件  
import 'antd/dist/reset.css'; // 引入 Ant Design 的样式  
const user = {  
 uid: '30009257',  
 avatar,  
 uname: '烽火部落',  
}  
const tabs = [  
 { type: 'hot', text: '最热' },  
 { type: 'time', text: '最新' },  
]  
function useGetList () {  
 const [commentList, setCommentList] = useState([])  
 useEffect(() => {  
 async function getList () {  
 const res = await axios.get(' http://localhost:3004/list')  
 setCommentList(res.data)  
 }  
 getList()  
 }, [])  
 return {  
 commentList,  
 setCommentList  
 }  
}  
function Item ({ item, onDel }) {  
 return (  
 <div className="reply-item">  
 {/\* 头像 \*/}  
 <div className="root-reply-avatar">  
 <div className="bili-avatar">  
 <img  
 className="bili-avatar-img"  
 alt=""  
 src={item.user.avatar}  
 />  
 </div>  
 </div>  
 <div className="content-wrap">  
 {/\* 用户名 \*/}  
 <div className="user-info">  
 <div className="user-name">{item.user.uname}</div>  
 </div>  
 {/\* 评论内容 \*/}  
 <div className="root-reply">  
 <span className="reply-content">{item.content}</span>  
 <div className="reply-info">  
 {/\* 评论时间 \*/}  
 <span className="reply-time">{item.ctime}</span>  
 {/\* 评论数量 \*/}  
 <span className="reply-time">点赞数:{item.like}</span>  
 {/\* 条件：user.id === item.user.id \*/}  
 {user.uid === item.user.uid &&  
 <span className="delete-btn" onClick={() => onDel(item.rpid)}>  
 删除  
 </span>}  
 </div>  
 </div>  
 </div>  
 </div>  
 )  
}  
const App = () => {  
 const { commentList, setCommentList } = useGetList()  
 const handleDel = (id) => {  
 console.log(id)  
 setCommentList(commentList.filter(item => item.rpid !== id))  
 }  
 const [type, setType] = useState('hot')  
 const handleTabChange = (type) => {  
 console.log(type)  
 setType(type)  
 if (type === 'hot') {  
 setCommentList(\_.orderBy(commentList, 'like', 'desc'))  
 } else {  
 setCommentList(\_.orderBy(commentList, 'ctime', 'desc'))  
 }  
 }  
 const [content, setContent] = useState('')  
 const inputRef = useRef(null)  
 const handlPublish = () => {  
 if (content.trim() === '') {  
 message.warning('评论内容不能为空') // 显示提示消息  
 return  
 }  
 setCommentList([  
 ...commentList,  
 {  
 rpid: uuidV4(), // 随机id  
 user: {  
 uid: '30009257',  
 avatar,  
 uname: '烽火部落',  
 },  
 content: content,  
 ctime: dayjs(new Date()).format('MM-DD hh:mm'), // 格式化 月-日 时:分  
 like: 66,  
 }  
 ])  
 setContent('')  
 inputRef.current.focus()  
 }  
 return (  
 <div className="appHeader">  
 {/\* 导航 Tab \*/}  
 <div className="reply-navigation">  
 <ul className="nav-bar">  
 <li className="nav-title">  
 <span className="nav-title-text">评论</span>  
 {/\* 评论数量 \*/}  
 <span className="total-reply">{commentList.length}</span>  
 </li>  
 <li className="nav-sort">  
 {/\* 高亮类名： active \*/}  
 {tabs.map(item =>  
 <span  
 key={item.type}  
 onClick={() => handleTabChange(item.type)}  
 className={classNames('nav-item', { active: type === item.type })}>  
 {item.text}  
 </span>)}  
 </li>  
 </ul>  
 </div>  
 <div className="reply-wrap">  
 {/\* 发表评论 \*/}  
 <div className="box-normal">  
 {/\* 当前用户头像 \*/}  
 <div className="reply-box-avatar">  
 <div className="bili-avatar">  
 <img className="bili-avatar-img" src={avatar} alt="用户头像" />  
 </div>  
 </div>  
 <div className="reply-box-wrap">  
 {/\* 评论框 \*/}  
 <textarea  
 className="reply-box-textarea"  
 placeholder="发一条友善的评论"  
 ref={inputRef}  
 value={content}  
 onChange={(e) => setContent(e.target.value)}  
 />  
 {/\* 发布按钮 \*/}  
 <div className="reply-box-send">  
 <div className="send-text" onClick={handlPublish}>发布</div>  
 </div>  
 </div>  
 </div>  
 {/\* 评论列表 \*/}  
 <div className="reply-list">  
 {/\* 评论项 \*/}  
 {commentList.map(item => <Item key={item.rpid} item={item} onDel={handleDel} />)}  
 </div>  
 </div>  
 </div>  
 )  
}  
export default App

App.scss

:where(.css-dev-only-do-not-override-zl9ks2).ant-drawer .ant-drawer-title {  
 color: #9499a0;  
}  
.reply-navigation {  
 margin-bottom: 22px;  
 .nav-bar {  
 display: flex;  
 align-items: center;  
 margin: 0;  
 padding: 0;  
 list-style: none;  
 .nav-title {  
 display: flex;  
 align-items: center;  
 width: 114px;  
 font-size: 20px;  
 .nav-title-text {  
 color: #18191c;  
 font-weight: 500;  
 }  
 .total-reply {  
 margin: 0 36px 0 6px;  
 color: #9499a0;  
 font-weight: normal;  
 font-size: 13px;  
 }  
 }  
 .nav-sort {  
 display: flex;  
 align-items: center;  
 color: #9499a0;  
 font-size: 13px;  
 .nav-item {  
 cursor: pointer;  
 &:hover {  
 color: #FDA75F;  
 }  
 &:last-child::after {  
 display: none;  
 }  
 &::after {  
 content: ' ';  
 display: inline-block;  
 height: 10px;  
 width: 1px;  
 margin: -1px 12px;  
 background-color: #9499a0;  
 }  
 }  
 .nav-item.active {  
 color: #18191c;  
 }  
 }  
 }  
}  
.reply-wrap {  
 position: relative;  
}  
.box-normal {  
 display: flex;  
 transition: 0.2s;  
 .reply-box-avatar {  
 display: flex;  
 align-items: center;  
 justify-content: center;  
 width: 80px;  
 height: 50px;  
 }  
 .reply-box-wrap {  
 display: flex;  
 position: relative;  
 flex: 1;  
 .reply-box-textarea {  
 width: 100%;  
 height: 50px;  
 padding: 5px 10px;  
 box-sizing: border-box;  
 color: #181931;  
 font-family: inherit;  
 line-height: 38px;  
 border: 1px solid #f1f2f3;  
 border-radius: 6px;  
 outline: none;  
 resize: none;  
 transition: 0.2s;  
 &::placeholder {  
 color: #9499a0;  
 font-size: 12px;  
 }  
 &:focus {  
 height: 60px;  
 border-color: #c9ccd0;  
 }  
 }  
 }  
 .reply-box-send {  
 position: relative;  
 display: flex;  
 flex-basis: 86px;  
 align-items: center;  
 justify-content: center;  
 margin-left: 10px;  
 border-radius: 4px;  
 cursor: pointer;  
 transition: 0.2s;  
 & .send-text {  
 position: absolute;  
 z-index: 1;  
 color: #fff;  
 font-size: 16px;  
 }  
 &::after {  
 position: absolute;  
 width: 100%;  
 height: 100%;  
 background-color: #FDA75F;  
 border-radius: 4px;  
 opacity: 0.5;  
 content: '';  
 }  
 &:hover::after {  
 opacity: 1;  
 }  
 }  
}  
.bili-avatar {  
 position: relative;  
 display: block;  
 width: 48px;  
 height: 48px;  
 margin: 0;  
 padding: 0;  
 border-radius: 50%;  
}  
.bili-avatar-img {  
 position: absolute;  
 top: 50%;  
 left: 50%;  
 display: block;  
 width: 48px;  
 height: 48px;  
 object-fit: cover;  
 border: none;  
 border-radius: 50%;  
 image-rendering: -webkit-optimize-contrast;  
 transform: translate(-50%, -50%);  
}  
.reply-list {  
 margin-top: 14px;  
}  
.reply-item {  
 padding: 22px 0 0 80px;  
 .root-reply-avatar {  
 position: absolute;  
 left: 0;  
 display: flex;  
 justify-content: center;  
 width: 80px;  
 cursor: pointer;  
 }  
 .content-wrap {  
 position: relative;  
 flex: 1;  
 &::after {  
 content: ' ';  
 display: block;  
 height: 1px;  
 width: 100%;  
 margin-top: 14px;  
 }  
 .user-info {  
 display: flex;  
 align-items: center;  
 margin-bottom: 4px;  
 .user-name {  
 height: 30px;  
 margin-right: 5px;  
 color: #61666d;  
 font-size: 13px;  
 line-height: 30px;  
 cursor: pointer;  
 }  
 }  
 .root-reply {  
 position: relative;  
 padding: 2px 0;  
 color: #181931;  
 font-size: 15px;  
 line-height: 24px;  
 .reply-info {  
 position: relative;  
 display: flex;  
 align-items: center;  
 margin-top: 2px;  
 color: #9499a0;  
 font-size: 13px;  
 .reply-time {  
 width: 86px;  
 margin-right: 20px;  
 }  
 .reply-like {  
 display: flex;  
 align-items: center;  
 margin-right: 19px;  
 .like-icon {  
 width: 14px;  
 height: 14px;  
 margin-right: 5px;  
 color: #9499a0;  
 background-position: -153px -25px;  
 &:hover {  
 background-position: -218px -25px;  
 }  
 }  
 .like-icon.liked {  
 background-position: -154px -89px;  
 }  
 }  
 .reply-dislike {  
 display: flex;  
 align-items: center;  
 margin-right: 19px;  
 .dislike-icon {  
 width: 16px;  
 height: 16px;  
 background-position: -153px -153px;  
 &:hover {  
 background-position: -217px -153px;  
 }  
 }  
 .dislike-icon.disliked {  
 background-position: -154px -217px;  
 }  
 }  
 .delete-btn {  
 cursor: pointer;  
 &:hover {  
 color: #FDA75F;  
 }  
 }  
 }  
 }  
 }  
}  
.reply-none {  
 height: 64px;  
 margin-bottom: 80px;  
 color: #99a2aa;  
 font-size: 13px;  
 line-height: 64px;  
 text-align: center;  
}

First.js

import React, { useState } from 'react'  
import "./First.scss"  
import Header from '../Header/Header'  
const First = () => {  
 const [menuItems, setMenuItems] = useState([  
 { id: 2, label: '热门赛事', path: '/FirstContent' },  
 {  
 id: 3,  
 label: '培训指导',  
 path: '/Lesson/Czone',  
 subMenu: [  
 { id: 31, label: '培训课程', path: '/Lesson/Czone' },  
 { id: 32, label: '训练题库', path: '/Question' },  
 ],  
 },  
 { id: 4, label: '讨论专区', path: '/Contact/attention' },  
 ])  
 const [activeSubMenu, setActiveSubMenu] = useState(null)  
 const handleMouseEnter = (id) => {  
 setActiveSubMenu(id)  
 }  
 const handleMouseLeave = () => {  
 setActiveSubMenu(null)  
 }  
 const handleSubMenuClick = (path) => {  
 }  
 return (  
 <>  
 <Header />  
 <div className='app'>  
 <div className='one'>  
 <ul className='ul-one'>  
 {menuItems.map((item, id) => (  
 <li key={item.id}  
 onClick={() => {  
 handleMouseEnter(id)  
 }}  
 className={activeSubMenu === id ? 'borderActive' : " "}  
 onMouseEnter={() => handleMouseEnter(item.id)}  
 onMouseLeave={handleMouseLeave}  
 >  
 <a href={item.path}>{item.label}</a>  
 {item.subMenu && activeSubMenu === item.id && (  
 <ul className='ul-two'>  
 {item.subMenu.map((subItem) => (  
 <li  
 onClick={() => {  
 handleMouseEnter(id)  
 }}  
 className={activeSubMenu === id ? 'borderActive' : " "}  
 key={subItem.id} >  
 <a href={subItem.path} onClick={() => handleSubMenuClick(subItem.path)}>  
 {subItem.label}  
 </a>  
 </li>  
 ))}  
 </ul>  
 )}  
 </li>  
 ))}  
 </ul>  
 <div className='borderBottom'></div>  
 </div>  
 {/\* <Routes>  
 {/\* 实现进入页面时候重定向 \*/}  
 {/\* <Route path="/FirstContent" element={<FirstContent />} />  
 <Route path="/Lesson" element={<Lesson />}></Route>  
 <Route path='/Question' element={<Question />}></Route>  
 </Routes> \*/}  
 </div>  
 </>  
 )  
}  
export default First

First.scss

.app {  
 display: flex;  
 flex-direction: column;  
 background-color: rgb(242, 242, 242);  
 align-items: center;  
}  
.one {  
 background-color: white;  
 width: 100%;  
 display: flex;  
 align-items: center;  
 height: 75px;  
 justify-content: center;  
}  
.borderBottom{  
 content: '';  
 position: absolute;  
 top: 145px;  
 width: 100%;  
 height: 1px;  
 background-color: rgb(199, 191, 191);  
}  
.ul-one {  
 color: black;  
 position: absolute;  
 margin-left: 8%;  
 justify-content: center;  
 display: flex;  
}  
.ul-one li {  
 display: inline-block;  
 padding: 10px;  
 font-size: 25px;  
}  
.ul-one a {  
 color: #6F6B6B;  
 padding-right: 70px;  
 display: flex;  
}  
.ul-one a:hover {  
 color: #FFAA5E;  
 text-decoration-line: underline;  
 text-underline-offset: 0.8em;  
 transition: all 0.5s;  
}  
.borderActive {  
 color: #FFAA5E !important;  
}  
.ul-two li {  
 margin-left: 125%;  
 display: block !important;  
 position: relative;  
 z-index: 1;  
 padding: 10px;  
 top:10px;  
 font-size: large;  
 width: 100%;  
 transition: all 0.3s;  
}  
.ul-two {  
 position: absolute;  
 left: 0;  
 transition: all 0.3s;  
}

Header.js

import React, { useState } from 'react'  
import "./Header.css"  
import { PlusCircleOutlined } from '@ant-design/icons'  
const Header = () => {  
 const [menuItems, setMenuItems] = useState([  
 { id: 1, label: <img src={require('../../assets/images/发布.png')} alt='发布' />, path: '/Publish' },  
 { id: 2, label: <img src={require('../../assets/images/个人中心-置灰 1.png')} alt="个人中心" />, path: '/Personal' },  
 { id: 3, label: <img src={require('../../assets/images/Vector-1.png')} alt='消息中心' />, path: '/Letter' },  
 ])  
 const [activeSubMenu, setActiveSubMenu] = useState(null)  
 const handleMouseEnter = (id) => {  
 setActiveSubMenu(id)  
 }  
 const handleMouseLeave = () => {  
 setActiveSubMenu(null)  
 }  
 return (  
 <div className='header'>  
 <div className='header-background'>  
 <div className='header-content'>  
 <div className='header-name'>  
 <img src={require("../../assets/images/Component 60.png")}></img>  
 </div>  
 <div className='header-one'>  
 <ul className='header-ul-one'>  
 {menuItems.map((item, id) => (  
 <li key={item.id}  
 onClick={() => {  
 handleMouseEnter(id)  
 }}  
 className={activeSubMenu === id ? 'borderActive' : " "}  
 onMouseEnter={() => handleMouseEnter(item.id)}  
 onMouseLeave={handleMouseLeave}  
 >  
 <a href={item.path}>{item.label}</a>  
 </li>  
 ))}  
 </ul>  
 </div>  
 </div>  
 </div>  
 </div>  
 )  
}  
export default Header

Logo.js

import React, { useEffect, useRef } from "react"  
import './Logo.scss'  
const Logo = () => {  
 const logoRef = useRef(null)  
 useEffect(() => {  
 if (logoRef.current) {  
 const logoHeight = logoRef.current.clientHeight  
 const lineName = document.querySelector('.line')  
 if (lineName) {  
 lineName.style.height = `${logoHeight}px`  
 }  
 }  
 }, [])  
 return (  
 <div className='logo'>  
 <img  
 className='logo-picture'  
 ref={logoRef}  
 alt="logo"  
 />  
 <div className='line'>  
 <div className='black-bg' style={{ height: '50%', margin: '-10px' }}></div>  
 <div className='blue-bg' style={{ height: '50%', backgroundColor: 'rgba(14, 93, 232, 0.45)', display: 'flex', alignItems: 'flex-end' }}>  
 <span className='logo-name'>闲享社群</span>  
 </div>  
 </div>  
 </div>  
 )  
}  
export default Logo

Logo.scss

.line {  
 margin-left: 20px; //左间距  
 .logo-name {  
 margin-bottom: 5px; //底间距  
 font-size: 32px; //字号  
 color: black; //颜色  
 letter-spacing: 12px; //字间距  
 }  
}

index.js

import React from 'react';  
import ReactDOM from 'react-dom/client';  
import './index.css';  
import App from './App';  
import 'antd/dist/reset.css';//引入antd全局样式  
import { RouterProvider } from 'react-router-dom';  
import { Provider } from 'react-redux';  
import store from './store';  
import router from './router';  
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(  
 <Provider store={store}>  
 <React.StrictMode>  
 {/\*路由绑定\*/}  
 {/\* <RouterProvider router={router}></RouterProvider> \*/}  
 <App></App>  
 </React.StrictMode>  
 </Provider>,  
);

Amend.js

import React, { useState } from 'react'  
import './Amend.css'  
const Amend = () => {  
 const [phoneNumber, setPhoneNumber] = useState('')  
 const [verificationCode, setVerificationCode] = useState('')  
 const [password, setPassword] = useState('')  
 const [confirmPassword, setConfirmPassword] = useState('')  
 const [showPassword] = useState(false)  
 const handlePhoneNumberChange = (event) => {  
 setPhoneNumber(event.target.value)  
 }  
 const handleVerificationCodeChange = (event) => {  
 setVerificationCode(event.target.value)  
 }  
 const handlePasswordChange = (event) => {  
 setPassword(event.target.value)  
 }  
 const handleConfirmPasswordChange = (event) => {  
 setConfirmPassword(event.target.value)  
 }  
 const handleRegister = () => {  
 }  
 return (  
 <div className='amend-all'>  
 <div className='amendbox'>  
 <div className='amend-logo'>  
 <div className='amend-main'>  
 <div className='amend-title'>  
 修改密码  
 </div >  
 {/\* 输入手机号 \*/}  
 <div className='phone'>  
 <input  
 type='text'  
 placeholder='注册手机号'  
 value={phoneNumber}  
 onChange={handlePhoneNumberChange}  
 />  
 </div>  
 {/\* 输入验证码 \*/}  
 <div className='code'>  
 <input  
 type='text'  
 placeholder='验证码'  
 value={verificationCode}  
 onChange={handleVerificationCodeChange}  
 />  
 {/\* 获取验证码 \*/}  
 <button>获取验证码</button>  
 </div>  
 <div className='password-section'>  
 {/\* 输入新密码 \*/}  
 <div className='password'>  
 <input  
 type={showPassword ? 'text' : 'password'}  
 placeholder='新密码'  
 value={password}  
 onChange={handlePasswordChange}  
 />  
 </div>  
 {/\* 输入确认新密码 \*/}  
 <div className='confirm-password'>  
 <input  
 type={showPassword ? 'text' : 'password'}  
 placeholder='确认新密码'  
 value={confirmPassword}  
 onChange={handleConfirmPasswordChange}  
 />  
 </div>  
 {/\* 完成注册按钮 \*/}  
 <div className='complete'>  
 <button onClick={handleRegister}>确定</button>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 )  
}  
export default Amend

Attention.js

import React, { useState } from "react";  
import "./Attention.scss";  
import Icon, { LikeOutlined, FormOutlined, HeartOutlined } from '@ant-design/icons';  
import { Button, Drawer, Image } from 'antd';  
import App from '../../../components/comments/App'; // 确保导入 App 组件  
const Attention = ({ posts = [] }) => {  
 const [open, setOpen] = useState(false);  
 const showDrawer = () => {  
 setOpen(true);  
 };  
 const onClose = () => {  
 setOpen(false);  
 };  
 const [likes, setLikes] = useState(0);  
 const [isLiked, setIsLiked] = useState(false);  
 const handleLikeClick = () => {  
 if (isLiked) {  
 let likes = 100;  
 setLikes(likes - 1);  
 setIsLiked(false);  
 } else {  
 setLikes(likes + 1);  
 setIsLiked(true);  
 }  
 };  
 return (  
 <>   
 <div className="rightMainBox">  
 {posts.map((post, index) => (  
 <div className="reply-wrap" key={index}>  
 <div className="reply-item">  
 {/\* 头像 \*/}  
 <div className="root-reply-avatar">  
 {/\* 当前用户头像 \*/}  
 <div className="reply-box-avatar">  
 <div className="bili-avatar">  
 <img className="bili-avatar-img" src={require("../../../assets/images/Ellipse 23.png")} alt="用户头像" />  
 </div>  
 </div>  
 </div>  
 <div className="content-wrap">  
 {/\* 用户名 \*/}  
 <div className="user-info">  
 <div className="user-name">夜冰</div>  
 </div>  
 {/\* 评论时间 \*/}  
 <div className='reply-time'>  
 <span className="reply-time">{post.date}</span>  
 </div>  
 {/\* 评论内容 \*/}  
 <div className="root-reply">  
 <span className="reply-content">{post.content}<br></br>  
 <span className="reply-HL">{post.selectedTag}</span></span>  
 <div className="reply-img">  
 <Image.PreviewGroup>  
 {post.images && post.images.map((image, idx) => (  
 <Image key={idx} width={300} height={250} src={image} />  
 ))}  
 </Image.PreviewGroup>  
 </div>  
 <div className="reply-info">  
 <span  
 onClick={handleLikeClick}  
 style={{ color: isLiked ? '#FDA75F' : ' ' }}  
 >  
 <LikeOutlined></LikeOutlined>  
 <span className="delete-btn">点赞数:{likes}</span>  
 </span>  
 <span className="delete-btn"><FormOutlined />  
 <Button type="primary" onClick={showDrawer} className="comment-btn">  
 点击评论  
 </Button>  
 <Drawer title="请开始你的评论~" onClose={onClose} open={open} className="comment-drawer">  
 <App></App>  
 </Drawer></span>  
 <span className="delete-btn">  
 <HeartOutlined /> 收藏:{post.favorites}  
 </span>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 ))}  
 </div>  
 <div className="rightMainBox">  
 <div className="reply-wrap">  
 {/\* 评论项 \*/}  
 <div className="reply-item">  
 {/\* 头像 \*/}  
 <div className="root-reply-avatar">  
 {/\* 当前用户头像 \*/}  
 <div className="reply-box-avatar">  
 <div className="bili-avatar">  
 <img className="bili-avatar-img" src={require("../../../assets/images/Ellipse 22.png")} alt="用户头像" />  
 </div>  
 </div>  
 </div>  
 <div className="content-wrap">  
 {/\* 用户名 \*/}  
 <div className="user-info">  
 <div className="user-name">儛踄^O^苟爱</div>  
 </div>  
 {/\* 评论时间 \*/}  
 <div className='reply-time'>  
 <span className="reply-time">{'2023-11-11'}</span>  
 </div>  
 {/\* 评论内容 \*/}  
 <div className="root-reply">  
 <span className="reply-content">求汇编大佬 在线答疑<br></br>  
 <span className="reply-HL">#学习讨论</span></span>  
 <div className="reply-img">  
 <Image.PreviewGroup>  
 <Image width={300} height={250}  
 src={require("../../../assets/images/Rectangle 2027.png")} />  
 <Image  
 width={300} height={250}  
 src={require("../../../assets/images/Rectangle 2028.png")}  
 />  
 </Image.PreviewGroup>  
 </div>  
 <div className="reply-info">  
 {/\* 评论数量 \*/}  
 <span  
 onClick={handleLikeClick}  
 style={{ color: isLiked ? '#FDA75F' : ' ' }} // 根据是否已点赞动态改变颜色  
 >  
 <LikeOutlined></LikeOutlined>  
 <span className="delete-btn">点赞数:{likes}</span>  
 </span>  
 <span className="delete-btn"><FormOutlined />  
 <Button type="primary" onClick={showDrawer} className="comment-btn">  
 点击评论  
 </Button>  
 <Drawer title="请开始你的评论~" onClose={onClose} open={open} className="comment-drawer">  
 <App></App>  
 </Drawer></span>  
 <span className="delete-btn">  
 <HeartOutlined /> 收藏:{100}  
 </span>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </>  
 );  
};  
export default Attention;

Attention.scss

.wrap {  
 height: 100%;  
 width: 70%;  
 margin-right: 15%;  
 margin-left: 15%;  
 margin-top: 20px;  
}  
.leftUlBox {  
 padding: 10px;  
 display: flex;  
 align-items: center;  
 float: left;  
 width: auto;  
 height: auto;  
 background-color: #fff;  
}  
.leftUl {  
 list-style: none;  
}  
.leftUl li:hover {  
 color: #FFAA5E;  
 text-decoration-line: underline;  
 text-underline-offset: 0.5em;  
}  
.borderActive {  
 color: #FFAA5E !important;  
}  
.leftUl li {  
 color: #6F6B6B;  
 letter-spacing: 5px;  
 font-size: 25px;  
 margin-bottom: 5px;  
 padding: 5px;  
}  
.rightMainBox {  
 width: 80%;  
 display: flex;  
 float: right;  
}  
.noStyle {  
 letter-spacing: 0;  
}  
.reply-wrap {  
 position: relative;  
 width: 100%;  
}  
.bili-avatar {  
 position: relative;  
 display: block;  
 width: 48px;  
 height: 48px;  
 margin: 0;  
 padding: 0;  
 border-radius: 50%;  
}  
.bili-avatar-img {  
 position: absolute;  
 top: 50%;  
 left: 50%;  
 display: block;  
 width: 48px;  
 height: 48px;  
 object-fit: cover;  
 border: none;  
 border-radius: 50%;  
 image-rendering: -webkit-optimize-contrast;  
 transform: translate(-50%, -50%);  
}  
.reply-item .content-wrap {  
 position: relative;  
}  
.reply-item {  
 padding: 22px 0 0 80px;  
 .root-reply-avatar {  
 position: absolute;  
 left: 0;  
 display: flex;  
 justify-content: center;  
 width: 80px;  
 cursor: pointer;  
 }  
 .content-wrap {  
 position: relative;  
 flex: 1;  
 &::after {  
 content: ' ';  
 display: block;  
 height: 1px;  
 width: 100%;  
 margin-top: 14px;  
 background-color: #e3e5e7;  
 }  
 .user-info {  
 display: flex;  
 align-items: center;  
 margin-bottom: 4px;  
 .user-name {  
 height: 30px;  
 margin-right: 5px;  
 font-size: 20px;  
 line-height: 30px;  
 cursor: pointer;  
 }  
 }  
 .reply-content {  
 width: 100%;  
 border: 1px solid #f0f4f7;  
 background-color: rgb(242, 242, 242);  
 margin-left: 0;  
 }  
 .reply-HL {  
 color: #FDA75F;  
 }  
 .reply-img {  
 width: 100%;  
 margin-top: 10px;  
 display: flex;  
 flex-wrap: wrap;  
 gap: 10px;  
 }  
 .replyImg {  
 flex: 1 1 45%;  
 height: auto;  
 }  
 .root-reply {  
 position: relative;  
 padding: 2px 0;  
 color: #181931;  
 font-size: 20px;  
 line-height: 24px;  
 margin-top: 15px;  
 .reply-info {  
 position: relative;  
 display: flex;  
 align-items: center;  
 margin-top: 5px;  
 color: #9499a0;  
 font-size: 13px;  
 .reply-time {  
 width: 86px;  
 margin-right: 20px;  
 }  
 .reply-like {  
 display: flex;  
 align-items: center;  
 margin-right: 19px;  
 .like-icon {  
 width: 14px;  
 height: 14px;  
 margin-right: 5px;  
 color: #9499a0;  
 background-position: -153px -25px;  
 &:hover {  
 background-position: -218px -25px;  
 }  
 }  
 .like-icon.liked {  
 background-position: -154px -89px;  
 }  
 }  
 .delete-btn {  
 cursor: pointer;  
 margin-right: 20px;  
 &:hover {  
 color: #FDA75F;  
 }  
 }  
 }  
 }  
 }  
}  
.reply-none {  
 height: 64px;  
 margin-bottom: 80px;  
 color: #99a2aa;  
 font-size: 13px;  
 line-height: 64px;  
 text-align: center;  
}

Competition.js

import React from "react";  
import "./Competition.scss"  
import Icon, { LikeOutlined, FormOutlined, HeartOutlined } from '@ant-design/icons';  
import { Link } from "react-router-dom";  
import { useState } from "react";  
const Competition = () => {  
 const [likes, setLikes] = useState(0);  
 const [isLiked, setIsLiked] = useState(false);  
 const handleLikeClick = () => {  
 if (isLiked) {  
 let likes = 100  
 setLikes(likes - 1);  
 setIsLiked(false);  
 } else {  
 setLikes(likes + 1);  
 setIsLiked(true);  
 }  
 };  
 return (  
 <>  
 {/\* <div className="wrap"> \*/}  
 <div className="rightMainBox">  
 <div className="reply-wrap">  
 {/\* 评论项 \*/}  
 <div className="reply-item">  
 {/\* 头像 \*/}  
 <div className="root-reply-avatar">  
 {/\* 当前用户头像 \*/}  
 <div className="reply-box-avatar">  
 <div className="bili-avatar">  
 <img className="bili-avatar-img" src={require("../../../assets/images/Ellipse 23.png")} alt="用户头像" />  
 </div>  
 </div>  
 </div>  
 <div className="content-wrap">  
 {/\* 用户名 \*/}  
 <div className="user-info">  
 <div className="user-name">夜冰</div>  
 </div>  
 {/\* 评论时间 \*/}  
 <div className='reply-time'>  
 <span className="reply-time">{'2023-11-11'}</span>  
 </div>  
 {/\* 评论内容 \*/}  
 <div className="root-reply">  
 <span className="reply-content">零基础如何参加互联网＋大赛？<br></br>  
 <span className="reply-HL">#竞赛分享</span></span>  
 <div className="reply-img">  
 <img className='replyImg'  
 src={require("../../../assets/images/Rectangle 2031.png")}></img>  
 <img className='replyImg'  
 src={require("../../../assets/images/Rectangle 2032.png")}></img>  
 </div>  
 <div className="reply-info">  
 {/\* 评论数量 \*/}  
 <span  
 onClick={handleLikeClick}  
 style={{ color: isLiked ? '#FDA75F' : ' ' }} // 根据是否已点赞动态改变颜色  
 >  
 <LikeOutlined></LikeOutlined>  
 <span className="delete-btn">点赞数:{likes}</span>  
 </span>  
 <span className="delete-btn"><FormOutlined />评论数:{100}</span>  
 <span className="delete-btn">  
 <HeartOutlined /> 收藏:{100}  
 </span>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 {/\* <div className="wrap"> \*/}  
 <div className="rightMainBox">  
 <div className="reply-wrap">  
 {/\* 评论项 \*/}  
 <div className="reply-item">  
 {/\* 头像 \*/}  
 <div className="root-reply-avatar">  
 {/\* 当前用户头像 \*/}  
 <div className="reply-box-avatar">  
 <div className="bili-avatar">  
 <img className="bili-avatar-img" src={require("../../../assets/images/Ellipse 23.png")} alt="用户头像" />  
 </div>  
 </div>  
 </div>  
 <div className="content-wrap">  
 {/\* 用户名 \*/}  
 <div className="user-info">  
 <div className="user-name">夜冰</div>  
 </div>  
 {/\* 评论时间 \*/}  
 <div className='reply-time'>  
 <span className="reply-time">{'2023-11-11'}</span>  
 </div>  
 {/\* 评论内容 \*/}  
 <div className="root-reply">  
 <span className="reply-content">零基础如何参加互联网＋大赛？  
 <br></br>  
 <span className="reply-HL">#竞赛分享</span></span>  
 <div className="reply-img">  
 <img className='replyImg'  
 src={require("../../../assets/images/Rectangle 2031.png")}></img>  
 <img className='replyImg'  
 src={require("../../../assets/images/Rectangle 2032.png")}></img>  
 </div>  
 <div className="reply-info">  
 {/\* 评论数量 \*/}  
 <span  
 onClick={handleLikeClick}  
 style={{ color: isLiked ? '#FDA75F' : ' ' }} // 根据是否已点赞动态改变颜色  
 >  
 <LikeOutlined></LikeOutlined>  
 <span className="delete-btn">点赞数:{likes}</span>  
 </span>  
 <span className="delete-btn"><FormOutlined />评论数:{100}</span>  
 <span className="delete-btn">  
 <HeartOutlined /> 收藏:{100}  
 </span>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 {/\* </div> \*/}  
 </div>  
 </>  
 )  
}  
export default Competition

Competition.scss

.wrap {  
 height: 100%;  
 width: 70%;  
 margin-right: 15%;  
 margin-left: 15%;  
 margin-top: 20px;  
}  
.leftUlBox{  
 padding: 10px;   
 display: flex;  
 align-items: center;  
 float: left;  
 width: auto;  
 height: auto;  
 background-color: #fff;  
}  
.leftUl{  
 list-style: none;  
}  
.leftUl li:hover {  
 color: #FFAA5E;  
 text-decoration-line: underline;  
 text-underline-offset: 0.5em;  
}  
.borderActive {  
 color: #FFAA5E !important;  
}  
.leftUl li{  
 color: #6F6B6B;  
 letter-spacing:5px;  
 font-size: 25px;  
 margin-bottom: 5px;  
 padding: 5px;  
}  
.rightMainBox{  
 width: 80%;  
 display: flex;  
 float: right;  
}  
.noStyle{  
 letter-spacing: 0;  
}  
.reply-wrap {  
 position: relative;  
 width: 100%;  
}  
.bili-avatar {  
 position: relative;  
 display: block;  
 width: 48px;  
 height: 48px;  
 margin: 0;  
 padding: 0;  
 border-radius: 50%;  
}  
.bili-avatar-img {  
 position: absolute;  
 top: 50%;  
 left: 50%;  
 display: block;  
 width: 48px;  
 height: 48px;  
 object-fit: cover;  
 border: none;  
 border-radius: 50%;  
 image-rendering: -webkit-optimize-contrast;  
 transform: translate(-50%, -50%);  
}  
.reply-item .content-wrap {  
 position: relative;  
}  
.reply-item {  
 padding: 22px 0 0 80px;  
 .root-reply-avatar {  
 position: absolute;  
 left: 0;  
 display: flex;  
 justify-content: center;  
 width: 80px;  
 cursor: pointer;  
 }  
 .content-wrap {  
 position: relative;  
 flex: 1;  
 &::after {  
 content: ' ';  
 display: block;  
 height: 1px;  
 width: 100%;  
 margin-top: 14px;  
 background-color: #e3e5e7;  
 }  
 .user-info {  
 display: flex;  
 align-items: center;  
 margin-bottom: 4px;  
 .user-name {  
 height: 30px;  
 margin-right: 5px;  
 font-size: 20px;  
 line-height: 30px;  
 cursor: pointer;  
 }  
 }   
 .reply-HL{  
 color: #FDA75F;  
 }  
 .reply-img {  
 width: 100%;  
 height: 240px;  
 margin-top: 10px;  
 }  
 .replyImg {  
 width: 45%;  
 height: 100%;  
 margin-right: 5px;   
 }  
 .root-reply {  
 position: relative;  
 padding: 2px 0;  
 color: #181931;  
 font-size: 20px;  
 line-height: 24px;  
 margin-top: 15px;  
 .reply-info {  
 position: relative;  
 display: flex;  
 align-items: center;  
 margin-top: 5px;  
 color: #9499a0;  
 font-size: 13px;  
 .reply-time {  
 width: 86px;  
 margin-right: 20px;  
 }  
 .reply-like {  
 display: flex;  
 align-items: center;  
 margin-right: 19px;  
 .like-icon {  
 width: 14px;  
 height: 14px;  
 margin-right: 5px;  
 color: #9499a0;  
 background-position: -153px -25px;  
 &:hover {  
 background-position: -218px -25px;  
 }  
 }  
 .like-icon.liked {  
 background-position: -154px -89px;  
 }  
 }  
 .delete-btn {  
 cursor: pointer;  
 margin-right: 20px;  
 &:hover {  
 color: #FDA75F;  
 }  
 }  
 }  
 }  
 }  
}  
.reply-none {  
 height: 64px;  
 margin-bottom: 80px;  
 color: #99a2aa;  
 font-size: 13px;  
 line-height: 64px;  
 text-align: center;  
}

Contact.js

import React, { useState } from 'react';  
import './Contact.scss';  
import First from '../../components/First/First';  
import { Link, Route, Routes } from 'react-router-dom';  
import News from './News/News';  
import Attention from './Attention/Attention';  
import Match from './Match/Match';  
const Contact = ({ posts = [] }) => {  
 const [open, setOpen] = useState(false);  
 const [isShow, setIsShow] = useState(0);  
 const showDrawer = () => {  
 setOpen(true);  
 };  
 const leftUl = [  
 {  
 index: 1,  
 name: '最新资讯',  
 path: 'Attention'  
 },  
 {  
 index: 2,  
 name: '我的关注',  
 path: 'Attention'  
 },  
 {  
 index: 3,  
 name: '学习分享'  
 },  
 {  
 index: 4,  
 name: '竞赛分享',  
 path: 'Match'  
 }  
 ];  
 const listStatus = (item, index) => {  
 setIsShow(index);  
 };  
 return (  
 <>  
 <First />  
 <div className="backgroundTwo">  
 <div className="wrap">  
 <div className="leftUlBox">  
 <ul className="leftUl">  
 {leftUl.map((post, index) => (  
 <Link to={post.path} key={index}>  
 <li  
 onClick={() => {  
 listStatus(post, index);  
 }}  
 className={isShow === index ? 'borderActive' : ''}  
 >  
 {post.name}  
 </li>  
 </Link>  
 ))}  
 </ul>  
 </div>  
 <div className="rightContent">  
 <Routes>  
 <Route path="/" element={<News />} />  
 <Route path="/Attention" element={<Attention posts={posts} />} />  
 <Route path="/Match" element={<Match />} />  
 </Routes>  
 </div>  
 </div>  
 </div>  
 </>  
 );  
};  
export default Contact;

Contact.scss

.backgroundTwo {  
 width: 100%;  
 min-height: 100vh;  
 display: flex;  
 flex-direction: column;  
 background-color: rgb(242, 242, 242);  
}  
.wrap {  
 height: 100%;  
 width: 70%;  
 margin-right: 15%;  
 margin-left: 15%;  
 margin-top: 20px;  
}  
.leftUlBox {  
 padding: 10px;  
 display: flex;  
 align-items: center;  
 float: left;  
 width: auto;  
 height: auto;  
 background-color: #fff;  
}  
.leftUl {  
 list-style: none;  
}  
.leftUl li:hover {  
 color: #ffaa5e;  
 text-decoration-line: underline;  
 text-underline-offset: 0.5em;  
}  
.borderActive {  
 color: #ffaa5e !important;  
}  
.leftUl li {  
 color: #6f6b6b;  
 letter-spacing: 5px;  
 font-size: 25px;  
 margin-bottom: 5px;  
 padding: 10px !important;  
}  
.rightMainBox {  
 width: 80%;  
 display: flex;  
 float: right;  
}  
.noStyle {  
 letter-spacing: 0;  
}  
.reply-wrap {  
 position: relative;  
 width: 100%;  
}  
.bili-avatar {  
 position: relative;  
 display: block;  
 width: 48px;  
 height: 48px;  
 margin: 0;  
 padding: 0;  
 border-radius: 50%;  
}  
.bili-avatar-img {  
 position: absolute;  
 top: 50%;  
 left: 50%;  
 display: block;  
 width: 48px;  
 height: 48px;  
 object-fit: cover;  
 border: none;  
 border-radius: 50%;  
 image-rendering: -webkit-optimize-contrast;  
 transform: translate(-50%, -50%);  
}  
.reply-item .content-wrap {  
 position: relative;  
}  
.reply-item {  
 padding: 22px 0 0 80px;  
 .root-reply-avatar {  
 position: absolute;  
 left: 0;  
 display: flex;  
 justify-content: center;  
 width: 80px;  
 cursor: pointer;  
 }  
 .content-wrap {  
 position: relative;  
 flex: 1;  
 &::after {  
 content: ' ';  
 display: block;  
 height: 1px;  
 width: 100%;  
 margin-top: 14px;  
 background-color: #e3e5e7;  
 }  
 .user-info {  
 display: flex;  
 align-items: center;  
 margin-bottom: 4px;  
 margin-left: 0 !important;  
 margin-top: 1%;  
 .user-name {  
 height: 30px;  
 margin-right: 5px;  
 font-size: 20px;  
 line-height: 30px;  
 cursor: pointer;  
 }  
 }  
 .reply-HL {  
 color: #fda75f;  
 }  
 .reply-img {  
 width: 100%;  
 height: 240px;  
 margin-top: 10px;  
 }  
 .replyImg {  
 width: 45%;  
 height: 100%;  
 margin-right: 5px;  
 }  
 .root-reply {  
 position: relative;  
 padding: 2px 0;  
 color: #181931;  
 font-size: 20px;  
 line-height: 24px;  
 margin-top: 15px;  
 .reply-info {  
 position: relative;  
 display: flex;  
 align-items: center;  
 margin-top: 20px !important;  
 color: #9499a0;  
 font-size: 13px;  
 .reply-time {  
 width: 86px;  
 margin-right: 20px;  
 }  
 .reply-like {  
 display: flex;  
 align-items: center;  
 margin-right: 19px;  
 .like-icon {  
 width: 14px;  
 height: 14px;  
 margin-right: 5px;  
 color: #9499a0;  
 background-position: -153px -25px;  
 &:hover {  
 background-position: -218px -25px;  
 }  
 }  
 .like-icon.liked {  
 background-position: -154px -89px;  
 }  
 }  
 .delete-btn {  
 cursor: pointer;  
 margin-right: 20px;  
 &:hover {  
 color: #fda75f;  
 }  
 }  
 }  
 }  
 }  
}  
.reply-none {  
 height: 64px;  
 margin-bottom: 80px;  
 color: #99a2aa;  
 font-size: 13px;  
 line-height: 64px;  
 text-align: center;  
}  
:where(.css-dev-only-do-not-override-zl9ks2).ant-float-btn-primary  
 .ant-float-btn-body {  
 background-color: #fda75f;  
 transition: background-color 0.2s;  
}

token.js

const TokenKey = 'token\_key'  
function setToken(token){  
 localStorage.setItem(TokenKey,token)  
}  
function getToken(){  
 return localStorage.getItem(TokenKey)  
}  
function removeToken(){  
 localStorage.removeItem(TokenKey)  
}  
export {  
 setToken,  
 getToken,  
 removeToken  
}

request.js

import axios from "axios";  
import {getToken} from "@/utils/token";  
const request = axios.create({  
 baseURL: 'https://mock.apipark.cn/m1/4425505-0-default', //请求根地址:Apifox 云端mock  
 timeout: 2000, //超时时间2s  
 headers: {'X-Custom-Header': 'foobar'}  
});  
request.interceptors.request.use(function (config) {  
 return config;  
}, function (error) {  
 return Promise.reject(error);  
});  
request.interceptors.response.use(function (response) {  
 return response;  
}, function (error) {  
 return Promise.reject(error);  
});  
export {request}

index.js

import {request} from "./request";  
import {setToken,getToken,removeToken} from './token'  
export {  
 request,  
 setToken,  
 getToken,  
 removeToken,  
}

user.js

import { createSlice } from '@reduxjs/toolkit'  
import { request } from '@/utils'  
import { setToken as \_setToken, getToken } from "@/utils";  
import { loginAPI, getProfileAPI } from "../../api/user";  
const userStore = createSlice({  
 name: 'user',  
 initialState: {  
 token: getToken() || '',  
 userInfo: {}  
 },  
 reducers: {  
 setToken(state, action) {  
 state.token = action.payload  
 \_setToken(action.payload)  
 },  
 setuserInfo(state, action) {  
 state.userInfo = action.payload  
 }  
 }  
})  
const { setToken, setuserInfo } = userStore.actions  
const fetchLogin = (loginForm) => {  
 return async (dispatch) => {  
 const res = await loginAPI(loginForm);  
 dispatch(setToken(res.data.token));  
 return res.data;  
 };  
};  
const fetchUserInfo = () => {  
 return async (dispatch) => {  
 const res = await request.get('/login')  
 dispatch(setuserInfo(res.data.data))  
 }  
}  
export { fetchLogin, fetchUserInfo, setToken }  
const userReducer = userStore.reducer  
export default userReducer

index.js

import {configureStore} from '@reduxjs/toolkit'  
import userReducer from "@/store/modules/user";  
const store = configureStore({  
 reducer:{  
 user:userReducer  
 }  
})  
export default store

index.js

Register.scss

.background-container {  
 background-size: cover;  
 background-position: center;  
 background-repeat: no-repeat;  
 height: 100vh;  
 display: flex;  
 flex-direction: column;  
}  
.wrap-contain{  
 height: 85%;  
 display: flex;  
 align-items: center;  
 align-self: flex-end; /\* 在交叉轴底部对齐 \*/  
 width: 50%;  
 margin-top: auto; /\* 将顶部边距设置为 auto，使内容靠近页面底部 \*/  
 margin-right: 28%;  
 flex-direction: column;  
}  
.welcome{  
 width: 70%;  
 height: 15%;  
 margin-left: 13%;  
 margin-top: 5px;  
}  
.center-contain {  
 display: flex;  
 align-items: center;  
 align-self: flex-end; /\* 在交叉轴底部对齐 \*/  
 width: 90%;  
 height: 80%;  
 background-color: rgb(243,252,255);  
 margin-top: auto; /\* 将顶部边距设置为 auto，使内容靠近页面底部 \*/  
 box-shadow: 20px 3px 10px -5px rgba(167,207,236, 0.75);  
}  
.shareCommunity{  
 display: flex; /\* 设置为弹性容器 \*/  
 flex-direction: column;  
 position: absolute;  
 top: 31%;  
 padding: 10px;  
 width: 45%;  
 height: 25%;  
}  
.shareCommunity img{  
 padding: 10px;  
 width: 45%;   
 height: auto; /\* 保持图片宽高比例 \*/  
}  
.shareCommunity a{  
 width: 30%;  
 float: right;  
}  
.shareCommunity a p{  
 text-align: right;  
 font-size: 20%;  
 height: auto;  
 width: 50%;  
}

Register.js

import React, { useEffect } from 'react';  
import './Register.scss'  
import { Button, Form, Input ,Space,message} from 'antd';  
import { LockOutlined, UserOutlined,MehOutlined,SafetyOutlined} from '@ant-design/icons';  
import { Link, useActionData } from 'react-router-dom';  
const Register = () => {  
 const [form] = Form.useForm();  
 const onFinish = (values) => {  
 console.log('Received values of form: ', values);  
 };  
 const [messageApi, contextHolder] = message.useMessage();  
 const success = useEffect(  
 () => {  
 messageApi.open({  
 type: 'success',  
 content: '获取成功',  
 });  
 },[]);  
 return (  
 <div className='background-container'>  
 <div className='wrap-contain'>  
 {/\* <img src={require('../../assets/images/welcome.png')}  
 className='welcome'  
 ></img> \*/}  
 <div className='center-contain'>  
 <div className='shareCommunity'>  
 {/\* <img src={require("../../assets/images/Share Community.png")}></img> \*/}  
 <div className='content'>  
 <Form  
 form={form}  
 name="normal\_login"  
 className="username-password"  
 validateTrigger='onBlur'//失焦验证  
 initialValues={{  
 remember: true,  
 username: '',  
 name:'',  
 password: '',  
 safePassworf:''  
 }}  
 onFinish={onFinish}  
 >  
 <Form.Item>  
 </Form.Item>  
 <Form.Item  
 name="username"//和后端接口名保持一致  
 rules={[  
 {  
 required: true,  
 message: '请输入您的用户名！',  
 },  
 {  
 pattern: /^1[3-9]\d{9}$/,//匹配中国大陆地区的手机号  
 message: '请输入正确的手机号！'  
 }  
 ]}  
 >  
 <Input prefix={<UserOutlined className="site-form-item-icon" />} placeholder="Username" />  
 </Form.Item>  
 <Form.Item  
 name='name'  
 rules={[  
 {  
 require:true,  
 message:"请输入您的昵称"  
 }  
 ]}  
 >  
 <Input  
 prefix={<MehOutlined className="site-form-item-icon"/>}  
 type='name'  
 placeholder='Name'  
 ></Input>  
 </Form.Item>  
 <Form.Item  
 name="password"  
 rules={[  
 {  
 required: true,  
 message: '请输入您的密码！',  
 },  
 ]}  
 >  
 <Input  
 prefix={<LockOutlined className="site-form-item-icon" />}  
 type="password"  
 placeholder="Password"  
 />  
 </Form.Item>  
 <Form.Item  
 name="password"  
 rules={[  
 {  
 required: true,  
 message: '请输入您验证码！',  
 },  
 ]}  
 >  
 <Input  
 prefix={<SafetyOutlined className="site-form-item-icon" />}  
 type="safePassword"  
 placeholder="safePassword"  
 />  
 <br></br>  
 <br></br>  
 <Button onClick={success}>获取验证码</Button>  
 </Form.Item>  
 </Form>  
 <Form className='remember-forgot-container'>  
 <Link to="/Login" className="login-form-forgot" >  
 返回登录>>  
 </Link>  
 </Form>  
 <Form className='button'>  
 <Form.Item >  
 <Button type="primary" htmlType="submit" className="login-form-button ">  
 注册  
 </Button>  
 </Form.Item>  
 </Form>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 );  
};  
export default Register;

Question.scss

.Lesson{  
 background-color: rgb(242,242,242);  
 width: 100%;  
 height: 100%;  
 display: flex;  
 flex-direction: column;  
}  
.background{  
 width: 100%;  
 height: 100vh;  
 display: flex;  
 flex-direction: column;  
 background-color: rgb(242,242,242);  
}  
.Inside{  
 display: flex;  
 margin-top: 3.2% !important;  
 width: 100%;  
 height: auto;  
 background-color: #fff;  
 border-radius: 20px;  
}  
.InsideLeft {  
 width: 200px;  
 height: 120px;  
 border-radius: 20px;  
 overflow: hidden;  
 float: left;  
 position: relative;  
 margin-left: 10%;  
 margin-top: 2%;  
 margin-bottom: 2%;  
}  
a{  
 color:black  
}  
.InsideLeft img {  
 width: 100%;  
 height: 100%;  
 object-fit: cover;  
 border-radius: 20px;  
 position: relative;  
 top: 0;  
 left: 0;  
}  
.InsideRight {  
 float: right;  
 width: 65%;  
 display: flex;  
 flex-direction: column;  
 display: flex;  
 line-height: 100px;  
}  
.TitleTwo {  
 padding-top: 25px;  
 padding-left: 50px;  
 font-size: 25px;  
}

Question.js

import React from "react";  
import "./Question.scss"  
import First from "../../components/First/First";  
import { Link } from "react-router-dom";  
const Question = () => {  
 return (  
 <div>  
 <First></First>  
 <div className="top">  
 <span>训练题库</span>  
 </div>  
 <div className="background">  
 <div className="wrap">  
 <Link to={"https://www.hello-algo.com/chapter\_preface/"} target="\_blank">  
 <div className="Inside">  
 <div className="InsideLeft">  
 <img className="image-container"  
 src={require("../../assets/images/Rectangle 2022.png")}></img>  
 </div>  
 <div className="InsideRight">  
 <span className="TitleTwo">  
 hello算法,小白的学习天堂，简单又易懂的算法题库  
 </span>  
 </div>  
 </div>  
 </Link>  
 <Link to={"https://www.luogu.com.cn/"} target="\_blank">  
 <div className="Inside">  
 <div className="InsideLeft">  
 <img className="image-container"  
 src={require("../../assets/images/Rectangle 2019.png")}></img>  
 </div>  
 <div className="InsideRight">  
 <span className="TitleTwo">  
 洛谷，学习算法必不可少的算法一站式全面刷题网站  
 </span>  
 </div>  
 </div>  
 </Link>  
 <Link to={"https://loj.ac/"} target="\_blank">  
 <div className="Inside">  
 <div className="InsideLeft">  
 <img className="image-container"  
 src={require("../../assets/images/Rectangle 2021.png")}></img>  
 </div>  
 <div className="InsideRight">  
 <span className="TitleTwo">  
 Libreoj小众算法刷题网站，学习更多解题思路查漏补缺  
 </span>  
 </div>  
 </div>  
 </Link>  
 </div>  
 </div>  
 </div>  
 )  
}  
export default Question

Publish.js

import React, { useState } from 'react';  
import { useNavigate } from 'react-router-dom';  
import First from "../../components/First/First";  
import './Publish.css';  
import { PlusOutlined } from '@ant-design/icons';  
import { Image, Upload, Input, message } from 'antd';  
const getBase64 = (file) =>  
 new Promise((resolve, reject) => {  
 const reader = new FileReader();  
 reader.readAsDataURL(file);  
 reader.onload = () => resolve(reader.result);  
 reader.onerror = (error) => reject(error);  
 });  
const Publish = ({ addPost }) => {  
 const [topic, setTopic] = useState('');  
 const [selectedTag, setSelectedTag] = useState('#学习讨论');  
 const [content, setContent] = useState('');  
 const [fileList, setFileList] = useState([]);  
 const [previewOpen, setPreviewOpen] = useState(false);  
 const [previewImage, setPreviewImage] = useState('');  
 const navigate = useNavigate();  
 const handleContentChange = (event) => {  
 setContent(event.target.value);  
 };  
 const handlePreview = async (file) => {  
 if (!file.url && !file.preview) {  
 file.preview = await getBase64(file.originFileObj);  
 }  
 setPreviewImage(file.url || file.preview);  
 setPreviewOpen(true);  
 };  
 const handleChange = ({ fileList: newFileList }) => setFileList(newFileList);  
 const handleTagClick = (tag) => {  
 setSelectedTag(tag);  
 };  
 const handlePublish = () => {  
 if (!topic || !content) {  
 message.error('标题和内容不能为空');  
 return;  
 }  
 const newPost = { topic, content, selectedTag, images: fileList.map(file => file.url || file.preview) };  
 addPost(newPost);  
 setTopic('');  
 setContent('');  
 setFileList([]);  
 setSelectedTag('#学习讨论');  
 message.success('发帖成功');  
 navigate('/contact/attention');  
 };  
 const uploadButton = (  
 <button  
 style={{  
 border: 0,  
 background: 'none',  
 }}  
 type="button"  
 >  
 <PlusOutlined />  
 <div style={{ marginTop: 8 }}>  
 点击上传  
 </div>  
 </button>  
 );  
 return (  
 <div className='publish-all'>  
 <First />  
 <div className='publish-header'>  
 <button className='publish-button' onClick={handlePublish}>  
 一键发帖  
 </button>  
 </div>  
 <div className='publish-main'>  
 <div className='publish-container'>  
 <div className='publish-header'>  
 <span className='publish-title'>主题：</span>  
 <Input value={topic} onChange={(e) => setTopic(e.target.value)} placeholder="请输入标题" />  
 </div>  
 <div className='publish-content'>  
 <textarea  
 placeholder='分享新鲜事...'  
 value={content}  
 onChange={handleContentChange}  
 ></textarea>  
 </div>  
 <div className='file-upload-container'>  
 <Upload  
 listType="picture-card"  
 fileList={fileList}  
 onPreview={handlePreview}  
 onChange={handleChange}  
 >  
 {fileList.length >= 8 ? null : uploadButton}  
 </Upload>  
 {previewImage && (  
 <Image  
 wrapperStyle={{  
 display: 'none',  
 }}  
 preview={{  
 visible: previewOpen,  
 onVisibleChange: (visible) => setPreviewOpen(visible),  
 afterOpenChange: (visible) => !visible && setPreviewImage(''),  
 }}  
 src={previewImage}  
 />  
 )}  
 </div>  
 <div className="publish-tags">  
 <span>添加标签：</span>  
 <span  
 className={`publish-tag ${selectedTag === '#学习讨论' ? 'selected' : ''}`}  
 onClick={() => handleTagClick('#学习讨论')}  
 >  
 </span>  
 <span  
 className={`publish-tag ${selectedTag === '#竞赛分享' ? 'selected' : ''}`}  
 onClick={() => handleTagClick('#竞赛分享')}  
 >  
 </span>  
 </div>  
 </div>  
 </div>  
 </div>  
 );  
};  
export default Publish;

Prove.js

import React, { useState } from 'react'  
import './Prove.css'  
const Information = () => {  
 const [realName, setRealName] = useState('')  
 const [realSchool, setRealSchool] = useState('')  
 const [idCardNumber, setIdCardNumber] = useState('')  
 const handleSubmit = () => {  
 const chineseNameRegex = /^[\u4e00-\u9fa5]+$/  
 if (!chineseNameRegex.test(realName)) {  
 alert('请输入有效的真实姓名（仅限汉字）')  
 return  
 }  
 const idCardRegex = /^\d{18}$/  
 if (!idCardRegex.test(idCardNumber)) {  
 alert('请输入有效的18位身份证号码')  
 return  
 }  
 console.log('Real Name:', realName)  
 console.log('ID Card Number:', idCardNumber)  
 }  
 return (  
 <div className='prove-all'>  
 <div className='prove-down'>  
 <div className='prove-content'>  
 <div className='realbox'>  
 <p>信息认证</p>  
 <div className='provebox'>  
 <input  
 type='text'  
 placeholder='请输入真实姓名'  
 value={realName}  
 onChange={(e) => setRealName(e.target.value)}  
 />  
 <input  
 type='text'  
 placeholder='请输入所在学校'  
 value={realSchool}  
 onChange={(e) => setRealSchool(e.target.value)}  
 />  
 <input  
 type='text'  
 placeholder='请输入身份证号'  
 value={idCardNumber}  
 onChange={(e) => setIdCardNumber(e.target.value)}  
 />  
 <button onClick={handleSubmit}>提交</button>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 )  
}  
export default Information

Wait.js

import React from 'react'  
import './Wait.css'  
const Wait = () => {  
 return (  
 <div className='wait-all'>  
 <div className='wait-up'>  
 <div className='wait-up1'>  
 申请项目  
 </div>  
 <div className='wait-up1'>  
 申请人  
 </div>  
 <div className='wait-up1'>  
 申请时间  
 </div>  
 <div className='wait-up1'>  
 状态  
 </div>  
 <div className='wait-up1'>  
 操作  
 </div>  
 </div>  
 <div className='wait-content'>  
 <div className='wait-content1'>  
 ZJSRTP项目  
 <div>  
 立项申请  
 </div>  
 </div>  
 <div className='wait-content1'>  
 周芹宇  
 </div>  
 <div className='wait-content1'>  
 2023.10.01  
 </div>  
 <div className='wait-content1' style={{ color: '#FDA75F' }}>  
 待审核  
 </div>  
 <div className='wait-content1' style={{ color: '#FDA75F' }}>  
 查看  
 </div>  
 </div>  
 <div className='wait-down'>  
 <div className='wait-down-main'>  
 共1条记录  
 </div>  
 </div>  
 </div >  
 )  
}  
export default Wait

Project.js

import React, { useState } from "react"  
import { Link, Routes, Route, Navigate } from "react-router-dom"  
import './Project.css'  
import Pass from './Pass/Pass'  
import Fail from './Fail/Fail'  
import Wait from './Wait/Wait'  
import Apply from './Apply/Apply'  
const Project = () => {  
 const projectList = [  
 { index: 0, label: "已通过", path: "/Personal/project/Pass" },  
 { index: 1, label: "待通过", path: "/Personal/project/Wait" },  
 { index: 2, label: "未通过", path: "/Personal/project/Fail" },  
 ]  
 const [isShow, setIsShow] = useState(0)  
 const listStatus = (item, index) => {  
 setIsShow(index)  
 }  
 const resetNav = () => {  
 setIsShow(-1) // 设置为负数，使得导航栏其他项恢复到未点击状态  
 }  
 return (  
 <div className='project-all'>  
 <div className="project-main">  
 <ul className="projectList">  
 {projectList.map((post, index) => (  
 <Link to={post.path || "/"} key={index}>  
 <li  
 onClick={() => listStatus(post, index)}  
 className={isShow === index ? 'projectActive' : ""}  
 >  
 {post.label}  
 </li>  
 </Link>  
 ))}  
 </ul>  
 <div className="right-button">  
 <Link to="/Personal/Project/Apply" onClick={resetNav}>  
 <button className="right-button-style" >申请+</button>  
 </Link>  
 </div>  
 </div>  
 <div className="project-container">  
 <div className="project-content">  
 <Routes>  
 {/\* 实现进入页面时候重定向 \*/}  
 <Route path="/" element={<Navigate to="/Personal/Project/Pass" />} />  
 <Route path="/Pass" element={<Pass />} />  
 <Route path="/Fail" element={<Fail />} />  
 <Route path="/Wait" element={<Wait />} />  
 <Route path="/Apply" element={<Apply />} />  
 </Routes>  
 </div>  
 </div>  
 </div>  
 )  
}  
export default Project

Pass.js

import React from 'react'  
import './Pass.css'  
const Pass = () => {  
 return (  
 <div className='pass-all'>  
 <div className='pass-up'>  
 <div className='pass-up1'>  
 申请项目  
 </div>  
 <div className='pass-up1'>  
 申请人  
 </div>  
 <div className='pass-up1'>  
 申请时间  
 </div>  
 <div className='pass-up1'>  
 状态  
 </div>  
 <div className='pass-up1'>  
 操作  
 </div>  
 </div>  
 <div className='pass-content'>  
 <img src={require('../../../../assets/images/暂无消息.png')} alt='暂无消息' />  
 </div>  
 <div className='pass-down'>  
 <div className='pass-down-main'>  
 共0条记录  
 </div>  
 </div>  
 </div>  
 )  
}  
export default Pass

Fail.js

import React from 'react'  
import './Fail.css'  
const Fail = () => {  
 return (  
 <div className='fail-all'>  
 <div className='fail-up'>  
 <div className='fail-up1'>  
 申请项目  
 </div>  
 <div className='fail-up1'>  
 申请人  
 </div>  
 <div className='fail-up1'>  
 申请时间  
 </div>  
 <div className='fail-up1'>  
 状态  
 </div>  
 <div className='fail-up1'>  
 操作  
 </div>  
 </div>  
 <div className='fail-content'>  
 <img src={require('../../../../assets/images/暂无消息.png')} alt='暂无消息' />  
 </div>  
 <div className='fail-down'>  
 <div className='fail-down-main'>  
 共0条记录  
 </div>  
 </div>  
 </div>  
 )  
}  
export default Fail

Apply.js

import React, { useState } from 'react';  
import 'react-datepicker/dist/react-datepicker.css';  
import './Apply.css';  
import { CloudUploadOutlined } from '@ant-design/icons';  
import { Image, Upload, Input, Button, message, DatePicker, Space } from 'antd';  
const onChange = (date, dateString) => {  
 console.log(date, dateString);  
};  
const getBase64 = (file) =>  
 new Promise((resolve, reject) => {  
 const reader = new FileReader();  
 reader.readAsDataURL(file);  
 reader.onload = () => resolve(reader.result);  
 reader.onerror = (error) => reject(error);  
 });  
const Apply = () => {  
 const [startDate, setStartDate] = useState(new Date());  
 const [previewOpen, setPreviewOpen] = useState(false);  
 const [previewImage, setPreviewImage] = useState('');  
 const [fileList, setFileList] = useState([]);  
 const [formData, setFormData] = useState({  
 projectName: '',  
 applicantName: '',  
 members: '',  
 introduction: '',  
 teacher: '',  
 speciality: ''  
 });  
 const [isSubmitting, setIsSubmitting] = useState(false);  
 const [lastSubmitTime, setLastSubmitTime] = useState(0);  
 const handlePreview = async (file) => {  
 if (!file.url && !file.preview) {  
 file.preview = await getBase64(file.originFileObj);  
 }  
 setPreviewImage(file.url || file.preview);  
 setPreviewOpen(true);  
 };  
 const handleChange = ({ fileList: newFileList }) => setFileList(newFileList);  
 const handleInputChange = (e) => {  
 const { name, value } = e.target;  
 setFormData({ ...formData, [name]: value });  
 };  
 const uploadButton = (  
 <button style={{ border: 'none', background: 'none', color: '#fda75f' }} type="button">  
 <CloudUploadOutlined />  
 <div style={{ marginTop: 8 }}>上传材料</div>  
 </button>  
 );  
 const handleSubmit = (e) => {  
 e.preventDefault();  
 const currentTime = Date.now();  
 if (currentTime - lastSubmitTime < 3000) {  
 message.warning('请勿重复提交');  
 return;  
 }  
 if (isSubmitting) {  
 message.warning('请勿重复提交');  
 return;  
 }  
 for (let key in formData) {  
 if (!formData[key]) {  
 message.error(`${key}不能为空`);  
 return;  
 }  
 }  
 if (fileList.length === 0) {  
 message.error('请上传申请材料');  
 return;  
 }  
 setIsSubmitting(true);  
 setLastSubmitTime(currentTime);  
 const submitData = {  
 ...formData,  
 startDate,  
 files: fileList.map(file => file.originFileObj)  
 };  
 console.log('提交的数据:', submitData);  
 setTimeout(() => {  
 message.success('申请已提交');  
 setIsSubmitting(false);  
 }, 2000); // 模拟网络延迟  
 };  
 return (  
 <div className='apply-all'>  
 <form onSubmit={handleSubmit}>  
 <label>  
 申请项目:  
 <Input  
 name="projectName"  
 value={formData.projectName}  
 onChange={handleInputChange}  
 placeholder="请输入申请项目"  
 />  
 </label>  
 <label>  
 申请人姓名:  
 <Input  
 name="applicantName"  
 value={formData.applicantName}  
 onChange={handleInputChange}  
 placeholder="请输入申请人姓名"  
 />  
 </label>  
 <label>  
 项目组成员:  
 <Input  
 name="members"  
 value={formData.members}  
 onChange={handleInputChange}  
 placeholder="请输入项目组成员"  
 />  
 </label>  
 <label>  
 项目简介:  
 <Input  
 name="introduction"  
 value={formData.introduction}  
 onChange={handleInputChange}  
 placeholder="请输入项目简介"  
 />  
 </label>  
 <label>  
 指导老师:  
 <Input  
 name="teacher"  
 value={formData.teacher}  
 onChange={handleInputChange}  
 placeholder="请输入指导老师"  
 />  
 </label>  
 <label>  
 所学专业:  
 <Input  
 name="speciality"  
 value={formData.speciality}  
 onChange={handleInputChange}  
 placeholder="请输入所学专业"  
 />  
 </label>  
 <label >  
 申请日期:  
 <DatePicker selected={startDate} onChange={(date) => setStartDate(date)} className='apply-data'/>  
 </label>  
 <label className='apply-input'>  
 申请材料:  
 <div className='file-upload-container1'>  
 <Upload  
 listType="picture-card"  
 fileList={fileList}  
 onPreview={handlePreview}  
 onChange={handleChange}  
 accept=".jpg,.jpeg,.png,.pdf,.doc,.docx"  
 >  
 {fileList.length >= 8 ? null : uploadButton}  
 </Upload>  
 {previewImage && (  
 <Image  
 wrapperStyle={{ display: 'none' }}  
 preview={{  
 visible: previewOpen,  
 onVisibleChange: (visible) => setPreviewOpen(visible),  
 afterOpenChange: (visible) => !visible && setPreviewImage(''),  
 }}  
 src={previewImage}  
 />  
 )}  
 </div>  
 </label>  
 <Button type="primary" htmlType="submit" disabled={isSubmitting}>  
 {isSubmitting ? '提交中...' : '提交'}  
 </Button>  
 </form>  
 </div>  
 );  
};  
export default Apply;

Personalheader.js

import React, { useEffect, useState } from 'react';  
import './Personalheader.css';  
import { useNavigate } from 'react-router-dom';  
import { getProfileAPI } from '../../../api/user';  
const Personalheader = () => {  
 const [userInfo, setUserInfo] = useState({  
 name: '',  
 location: '',  
 avatar: ''  
 });  
 const navigate = useNavigate();  
 const handleEditClick = () => {  
 navigate('/');  
 };  
 useEffect(() => {  
 async function fetchData() {  
 try {  
 const response = await getProfileAPI();  
 const data = response.data;  
 setUserInfo({  
 name: data.nickname,  
 location: `${data.city} | ${data.school}`,  
 avatar: data.avatar.startsWith('http') ? data.avatar : require('../../../assets/images/personalheadlike.png')  
 });  
 } catch (error) {  
 console.error('Failed to fetch user info:', error);  
 }  
 }  
 fetchData();  
 }, []);  
 return (  
 <div className='personalheader-all'>  
 <div className='personalheader-content'>  
 <div className='personalheader-main'>  
 <div className='user-profile'>  
 <div className='personalheader-left'>  
 {/\* 左侧头像 \*/}  
 <div className='avatar'>  
 <img src={userInfo.avatar} alt='用户头像' />  
 </div>  
 {/\* 中间用户信息 \*/}  
 <div className='user-info'>  
 <h2>{userInfo.name}</h2>  
 <p>{userInfo.location}</p>  
 </div>  
 </div>  
 {/\* 右侧修改信息按钮 \*/}  
 <div className='edit-button'>  
 {/\* 点击按钮直接执行跳转逻辑 \*/}  
 <button onClick={handleEditClick}>退出登录</button>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 );  
};  
export default Personalheader;

Personal.js

import React from "react"  
import "./Personal.css"  
import First from "../../components/First/First"  
import { Link, Route, Routes, Navigate } from "react-router-dom"  
import { useState } from "react"  
import Information from "./Information/Information"  
import Competition from "./Competition/Competition"  
import Project from "./Project/Project"  
import Data from "./Data/Data"  
import Prove from "./Prove/Prove"  
import Personalheader from "./Personalheader/Personalheader"  
import personalImage from "../../assets/images/personal.png"  
import competitionImage from "../../assets/images/competition.png"  
import projectImage from "../../assets/images/project.png"  
import dataImage from "../../assets/images/data2.png"  
import proveImage from "../../assets/images/information.png"  
const Contact = () => {  
 const leftUl = [  
 {  
 index: 0,  
 name: "个人信息",  
 path: "Information ",  
 imageSrc: personalImage  
 },  
 {  
 index: 1,  
 name: "竞赛管理",  
 path: "Competition",  
 imageSrc: competitionImage  
 },  
 {  
 index: 2,  
 name: "项目申请",  
 path: "Project",  
 imageSrc: projectImage  
 },  
 {  
 index: 3,  
 name: "数据看板",  
 path: "Data",  
 imageSrc: dataImage  
 },  
 {  
 index: 4,  
 name: "信息认证",  
 path: "Prove",  
 imageSrc: proveImage  
 }  
 ]  
 const [isShow, setIsShow] = useState(0)  
 const listStatus = (item, index) => {  
 setIsShow(index)  
 }  
 return (  
 <>  
 <First></First>  
 <Personalheader></Personalheader>  
 <div className="personal-all">  
 <div className="personal-main">  
 <div className="personal-tab">  
 <ul className="personallist">  
 {  
 leftUl.map((post, index) => {  
 return (  
 <Link to={post.path}>  
 <li key={index} onClick={() => {  
 listStatus(post, index)  
 }}  
 className={isShow === index ? 'personalActive' : ''}>  
 {/\* 此处用post方法表示遍历到当前层的name，而不是用leftUl.name \*/}  
 <img src={post.imageSrc} alt={post.name} />  
 {post.name}  
 </li>  
 </Link>  
 )  
 })}  
 </ul>  
 </div>  
 <Routes>  
 {/\* 实现进入页面时候重定向 \*/}  
 <Route path="/" element={<Navigate to="/Personal/Information" />} />  
 <Route path="/Information" element={<Information />} />  
 <Route path="/Competition" element={<Competition />} />  
 <Route path="/Project/\*" element={<Project />} />  
 <Route path="/Data" element={<Data />} />  
 <Route path="/Prove" element={<Prove />} />  
 </Routes>  
 </div>  
 </div>  
 </>  
 )  
}  
export default Contact

Information.js

import React, { useEffect, useState } from 'react';  
import './Information.css';  
import { getProfileAPI, updateProfileAPI } from '@/api/user';  
import { message, Input, Button } from 'antd';  
import defaultAvatar from '../../../assets/images/personalheadlike.png'; // 确保路径正确  
const Information = () => {  
 const [userInfo, setUserInfo] = useState(null);  
 const [editableField, setEditableField] = useState(null);  
 const [formData, setFormData] = useState({  
 nickname: '',  
 phone: '',  
 email: '',  
 userType: '',  
 school: '',  
 verification: '已认证', // 设置默认值  
 password: '',  
 avatar: '../../../assets/images/personalheadlike.png'  
 });  
 useEffect(() => {  
 async function fetchData() {  
 try {  
 const response = await getProfileAPI();  
 setUserInfo(response.data);  
 setFormData(response.data);  
 } catch (error) {  
 console.error('获取用户信息失败:', error);  
 }  
 }  
 fetchData().then(r => console.log(r));  
 }, []);  
 const handleInputChange = (e) => {  
 const { name, value } = e.target;  
 setFormData({ ...formData, [name]: value });  
 };  
 const handleEditClick = (field) => {  
 setEditableField(field);  
 };  
 const handleSaveClick = async (field) => {  
 if (field === 'email') {  
 const emailPattern = /^[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/;  
 if (!emailPattern.test(formData.email)) {  
 message.error('请输入有效的邮箱地址');  
 return;  
 }  
 }  
 if (field === 'phone') {  
 const phonePattern = /^1[3-9]\d{9}$/;  
 if (!phonePattern.test(formData.phone)) {  
 message.error('请输入有效的电话号码');  
 return;  
 }  
 }  
 try {  
 await updateProfileAPI({ [field]: formData[field] });  
 setUserInfo({ ...userInfo, [field]: formData[field] });  
 setEditableField(null);  
 message.success('更新成功');  
 } catch (error) {  
 console.error('更新用户信息失败:', error);  
 message.error('更新失败');  
 }  
 };  
 const handleAvatarChange = (e) => {  
 const file = e.target.files[0];  
 const isJpgOrPng = file.type === 'image/jpeg' || file.type === 'image/png';  
 if (!isJpgOrPng) {  
 message.error('头像图片格式必须为 JPG 或 PNG');  
 return;  
 }  
 const newAvatarUrl = URL.createObjectURL(file);  
 setFormData({ ...formData, avatar: newAvatarUrl });  
 message.success('头像上传成功');  
 };  
 if (!userInfo) {  
 return <div>加载中...</div>;  
 }  
 return (  
 <div className='information-all'>  
 <div className='information-down'>  
 <div className='page-wrap-user'>  
 <div className='opt-item'>  
 <div className='opt-label'>头像：</div>  
 <div className='opt-content'>  
 <div className='opt-value'>  
 <img className='avatarTwo' src={formData.avatar} />  
 </div>  
 <div className='opt-icon'>  
 <label htmlFor="avatar-upload">  
 <img src={require('../../../assets/images/arrow-r.png')} alt='修改头像' />  
 </label>  
 <input  
 id="avatar-upload"  
 type="file"  
 style={{ display: 'none' }}  
 accept=".jpg,.jpeg,.png"  
 onChange={handleAvatarChange}  
 />  
 </div>  
 </div>  
 </div>  
 {['nickname', 'phone', 'email', 'school'].map((field) => (  
 <div className='opt-item' key={field}>  
 <div className='opt-label'>{field === 'nickname' ? '昵称' : field === 'phone' ? '手机号' : field === 'email' ? '用户邮箱' : '所在学校'}：</div>  
 <div className='opt-content'>  
 {editableField === field ? (  
 <>  
 <Input  
 type='text'  
 name={field}  
 value={formData[field]}  
 onChange={handleInputChange}  
 style={{ width: '200px', marginRight: '10px' }}  
 />  
 <Button type="primary" onClick={() => handleSaveClick(field)}>保存</Button>  
 </>  
 ) : (  
 <>  
 <div className='opt-value'>{userInfo[field]}</div>  
 <div className='opt-icon'>  
 <img  
 src={require('../../../assets/images/edit.png')}  
 alt='修改'  
 onClick={() => handleEditClick(field)}  
 />  
 </div>  
 </>  
 )}  
 </div>  
 </div>  
 ))}  
 <div className='opt-item'>  
 <div className='opt-label'>用户类型：</div>  
 <div className='opt-content pad'>  
 <div className='opt-value'>{userInfo.userType}</div>  
 </div>  
 </div>  
 <div className='opt-item'>  
 <div className='opt-label'>认证信息：</div>  
 <div className='opt-content pad'>  
 <div className='opt-value'>{userInfo.verification || '已认证'}</div>  
 </div>  
 </div>  
 <div className='opt-item'>  
 <div className='opt-label'>账户密码：</div>  
 <div className='opt-content'>  
 {editableField === 'password' ? (  
 <>  
 <Input.Password  
 name='password'  
 value={formData.password}  
 onChange={handleInputChange}  
 style={{ width: '200px', marginRight: '10px' }}  
 />  
 <Button type="primary" onClick={() => handleSaveClick('password')}>保存</Button>  
 </>  
 ) : (  
 <>  
 <div className='opt-value'>\*\*\*\*\*\*\*\*</div>  
 <div className='opt-icon'>  
 <img  
 src={require('../../../assets/images/edit.png')}  
 alt='修改'  
 onClick={() => handleEditClick('password')}  
 />  
 </div>  
 </>  
 )}  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 );  
};  
export default Information;

Data.js

import React, { useState } from 'react';  
import './Data.css'  
import { Card, Row, Col } from 'antd';  
import ReactEcharts from 'echarts-for-react';  
const Data = (props) => {  
 const [sales, setSales] = useState([5, 20, 36, 10, 10, 20]);  
 const [stores, setStores] = useState([15, 120, 36, 110, 110, 20]);  
 const getOption = (sal, sto) => {  
 return {  
 tooltip: {  
 trigger: 'item',  
 formatter: "{a} <br/>{b} : {c} ({d}%)"  
 },  
 legend: {  
 orient: 'vertical',  
 left: 'left',  
 textStyle: {  
 fontSize: 13,  
 color: '#FFFFFF',  
 },  
 data: ['大一', '大二', '大三', '大四']  
 },  
 series: [  
 {  
 name: '竞赛组成',  
 type: 'pie',  
 radius: '55%',  
 center: ['50%', '60%'],  
 data: [  
 { value: 500, name: '大一' },  
 { value: 1210, name: '大二' },  
 { value: 834, name: '大三' },  
 { value: 280, name: '大四' }  
 ],  
 itemStyle: {  
 emphasis: {  
 shadowBlur: 10,  
 shadowOffsetX: 0,  
 shadowColor: 'rgba(0, 0, 0, 0.5)'  
 }  
 }  
 }  
 ]  
 }  
 };  
 const [salesTwo, setSalesTwo] = useState([5, 20, 36, 10, 10, 20,158]);  
 const [storesTwo, setStoresTwo] = useState([15, 120, 36, 110, 110, 20,50]);  
 const [ccc ,setccc] = useState([150,105,30,64,85,154,36])  
 const [Visual ,setVisual] = useState([130,148,60,64,59,160,180])  
 const [cc3 ,setcc3] = useState([140,50,80,64,25,184,180])  
 const [HTML ,setHTML] = useState([160,60,180,64,45,134,180])  
 const [Linux ,setLinux] = useState([170,80,160,94,75,144,80])  
 const [Python ,setPython] = useState([170,80,160,94,75,144,80])  
 const getOptionTwo = (sal, sto) => {  
 return {  
 tooltip: {},  
 legend: {  
 data: ['Java', "C++","Python","机器学习","HTML","Linux","MYSQL"]  
 },  
 xAxis: {  
 data: ["6.17", "6.18", "6.19", "6.20", "6.21", "6.22","6.23"],  
 },  
 yAxis: {},  
 series: [{  
 name: 'Java',  
 type: 'line',  
 data: sales,  
 },  
 {  
 name: 'C++',  
 type: 'line',  
 data: ccc  
 },  
 {  
 name: 'Python',  
 type: 'line',  
 data: Python  
 },  
 {  
 name: 'HTML',  
 type: 'line',  
 data: HTML  
 },  
 {  
 name: 'Linux',  
 type: 'line',  
 data: Linux  
 },  
 {  
 name: 'MYSQL',  
 type: 'line',  
 data: stores  
 },  
 {  
 name: '机器学习',  
 type: 'line',  
 data: stores  
 },  
 ],  
 legend:{  
 textStyle: {  
 fontSize: 13,  
 color: '#6d7071',  
 }  
 }  
 }  
 };  
 return (  
 <div className='data-all'>  
 <div className='data-down'>  
 <div className='data-content'>  
 <div className='zhexian'>  
 <Row gutter={14} justify="center">  
 <Col span={24}>  
 <ReactEcharts option={getOptionTwo(salesTwo, storesTwo, ccc, Visual)} style={{ width: "850px", height: "450px" }} />  
 <div className='data-title'>近七日各课程点击量趋势图</div>  
 </Col>  
 </Row>  
 </div>  
 <Row gutter={24} justify="center">  
 <Col span={24}>  
 <ReactEcharts option={getOption(sales, stores)} style={{ width: "100%", height: "450px" }} />  
 <div className='data-title'>竞赛组成人员统计图</div>  
 </Col>  
 </Row>  
 </div>  
 </div>  
 </div>  
 )  
}  
export default Data;

Competition.js

import React from 'react'  
import './Competition.css'  
const Competition = () => {  
 return (  
 <div className='competition-all'>  
 <div className='competition-title'>  
 参加的竞赛与项目  
 </div>  
 <div className='competition-container'>  
 <div className='competition-main'>  
 1.蓝桥杯  
 </div>  
 <div className='competition-main'>  
 2.校大创·ZJSRTP项目立项  
 </div>  
 </div>  
 </div>  
 )  
}  
export default Competition

Logon.js

import React, {useState, useEffect} from 'react';  
import './Logon.css';  
import {message} from 'antd';  
import axios from 'axios';  
import {useNavigate} from 'react-router-dom';  
const Logon = () => {  
 const [phoneNumber, setPhoneNumber] = useState('');  
 const [verificationCode, setVerificationCode] = useState('');  
 const [password, setPassword] = useState('');  
 const [showPassword, setShowPassword] = useState(false);  
 const [timer, setTimer] = useState(0);  
 const navigate = useNavigate();  
 useEffect(() => {  
 let interval;  
 if (timer > 0) {  
 interval = setInterval(() => {  
 setTimer(timer - 1);  
 }, 1000);  
 } else if (timer === 0) {  
 clearInterval(interval);  
 }  
 return () => clearInterval(interval);  
 }, [timer]);  
 const handlePhoneNumberChange = (e) => {  
 setPhoneNumber(e.target.value);  
 };  
 const handleVerificationCodeChange = (e) => {  
 setVerificationCode(e.target.value);  
 };  
 const handlePasswordChange = (e) => {  
 setPassword(e.target.value);  
 };  
 const togglePasswordVisibility = () => {  
 setShowPassword(!showPassword);  
 };  
 const handleRegister = async () => {  
 try {  
 const response = await axios.post('https://mock.apipark.cn/m1/4425505-4070744-default/user/add', {  
 phoneNumber,  
 verificationCode,  
 password,  
 });  
 if (response.data.message) {  
 message.success('注册成功！');  
 setPhoneNumber('');  
 setVerificationCode('');  
 setPassword('');  
 navigate('/');  
 } else {  
 message.error('注册失败，请重试！');  
 }  
 } catch (error) {  
 console.error('注册请求失败:', error);  
 message.error('注册失败，请重试！');  
 }  
 };  
 const handleGetCaptcha = async () => {  
 try {  
 if (!phoneNumber) {  
 message.warning('请输入手机号后再获取验证码！');  
 return;  
 }  
 const response = await axios.post('https://mock.apipark.cn/m1/4425505-4070744-default/user/send', {  
 phoneNumber,  
 });  
 if (response.data.code) {  
 message.success('验证码已发送！');  
 console.log(response.data.code);  
 setTimer(60);  
 } else {  
 message.error('获取验证码失败，请重试！');  
 }  
 } catch (error) {  
 console.error('获取验证码请求失败:', error);  
 message.error('获取验证码失败，请重试！');  
 }  
 console.log("已发送请求");  
 };  
 return (  
 <div className='logon-all'>  
 <div className='logonbox'>  
 <div className='logon-logo'>  
 <div className='logon-main'>  
 <div className='logon-title'>  
 账号注册  
 </div>  
 <div className='phone'>  
 <input  
 type='text'  
 placeholder='手机号'  
 value={phoneNumber}  
 onChange={handlePhoneNumberChange}  
 />  
 </div>  
 <div className='code'>  
 <input  
 type='text'  
 placeholder='验证码'  
 value={verificationCode}  
 onChange={handleVerificationCodeChange}  
 />  
 <button onClick={handleGetCaptcha} disabled={timer > 0}>  
 {timer > 0 ? `${timer}秒` : '获取验证码'}  
 </button>  
 </div>  
 <div className='password1'>  
 <input  
 type={showPassword ? 'text' : 'password'}  
 placeholder='密码'  
 value={password}  
 onChange={handlePasswordChange}  
 />  
 <button onClick={togglePasswordVisibility}>  
 {showPassword ? '??️' : '隐藏'}  
 </button>  
 </div>  
 <div className='warn'>  
 注册表示您已认真阅读并同意<span>烽火部落</span>的<span>《使用协议》</span>  
 </div>  
 <div className='complete'>  
 <button onClick={handleRegister}>完成注册</button>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 );  
};  
export default Logon;