A.Kiểm tra bằng chuỗi thông thường

Nên kiểm tra bằng biểu thức chính quy, phần này chỉ là phần nền tảng

1.LOGIN FORM

Khi thao tác với 1 form có 4 thứ thiết yếu nhất phải kiểm soát được là name, value, onChange và onBlur.

1.1.Tạo object để lưu trữ

    var formData = {

        userName : '',

        passWord : '',

    }

1.2.Lấy dữ liệu người dùng nhập vào (onChange)

const submitBtn = document.querySelector('form button.submit');

    const userName = document.querySelector('form input[type="email"]');

    const passWord = document.querySelector('form input[type="password"]');

    const uErrorText = document.querySelector('form p.user\_hidden\_text');

    const pErorText = document.querySelector('form p.pass\_hidden\_text');

    const authErrorText = document.querySelector('form p.auth\_hidden\_text');

Xử lí khi người dùng nhập thông tin vào

Chú ý: sự kiện phải là keyup chứ ko phải là keydown

    userName.addEventListener('keyup', () => {

        // đảm bảo là ô không bị đổi màu

        userName.style.border = '1px solid rgba(255, 255, 255, 0.3)';

        // dữ liệu người dùng nhập sẽ được lưu vào form

        formData.userName = userName.value;

        // nếu đã có dữ liệu trong input rồi thì ko thông báo gì cả

        if (formData.userName !== ''){

            uErrorText.innerHTML = ' ';

        }

    })

    passWord.addEventListener('keyup', () => {

        // đảm bảo là ô không bị đổi màu

        passWord.style.border = '1px solid rgba(255, 255, 255, 0.3)';

        // dữ liệu người dùng nhập sẽ được lưu vào form

        formData.passWord = passWord.value;

        // nếu đã có dữ liệu trong input rồi thì ko thông báo gì cả

        if (formData.passWord !== ''){

            pErorText.innerHTML = ' ';

        }

    })

1.3.Xử lí khi người dùng trỏ vào ô nhập rồi lại trỏ ra ngoài (onBlur)

    userName.addEventListener('blur', () => {

        // nếu khi người dùng blur mà chưa có dữ liệu trong ô input thì báo đỏ lên

        // thông báo cho người dùng biết là ko được để trống

        if (formData.userName === ''){

            userName.style.border = '1px solid red';

            uErrorText.innerHTML = 'This field must not be empty';

        }

    })

    passWord.addEventListener('blur', () => {

        // nếu khi người dùng blur mà chưa có dữ liệu trong ô input thì báo đỏ lên

        // thông báo cho người dùng biết là ko được để trống

        if (formData.passWord === ''){

            passWord.style.border = '1px solid red';

            pErorText.innerHTML = 'This field must not be empty';

        }

    })

1.4.Xử lí cho nút submit

submitBtn.addEventListener('click', (*e*) => {

    // tạo mảng lưu trữ thông báo

    let message = [];

    // nếu cả 2 ô input chưa có dữ liệu thì báo lỗi

    // nếu có rồi thì báo đăng nhập thành công

    if (formData.userName === '' || formData.passWord === ''){

        message.push("Username or password aren't nullable! <br/> Please try again");

    }

    else {

        message.push('Login successfully');

    }

    authErrorText.innerHTML = message;

    // chặn reset trang mặc định

*e*.preventDefault();

})

còn tiếp

2.REGISTER FORM

2.1.Tạo object để lưu trữ

    var registerFormData  = {

        username : '',

        password : '',

        cfPassword: '',

        phonenumber : ''

    }

2.2.Giả file json để lưu trữ các message

    var message = {

        mess1: { value : "empty values are not accepted"},

        mess2: { value : 'username has been existed' },

        mess3: {  value : 'password so weak'  },

        mess4: {  value : 'incorrect password' },

        mess5: {  value : 'form of phone number incorrect'  }

    }

2.3.Lấy DOM

    // select input

    const userName = document.querySelector('form input.reg-user');

    const passWord = document.querySelector('form input.reg-pass');

    const confirmPassWord = document.querySelector('form input.cf-reg-pass');

    const phoneNumber = document.querySelector('form input.reg-phone');

    const submitBtn = document.querySelector('button.submit');

    // select tooltip

    const user = document.querySelector('.user-tooltip');

    const pass = document.querySelector('.pass-tooltip');

    const cfPass = document.querySelector('.cf-pass-tooltip');

    const phone = document.querySelector('.phone-tooltip');

    const tooltip = document.querySelectorAll('.tooltip');

2.4.Xử lí blur cho ô input

Khi người dùng blur ô input thì kiểm tra, nếu ko có dữ liệu thì thông báo cho người dùng đây là trường bắt buộc.

    function handleBlur(*input*){

        switch (*input*){

            case 'userName':

                if (registerFormData.username === ''){

                    user.classList.add('appear');

                    user.innerHTML = message.mess1.value;

                }

                break;

            case 'password':

                if (registerFormData.password === ''){

                    pass.classList.add('appear');

                    pass.innerHTML = message.mess1.value;

                }

                break;

            case 'confirm':

                if (registerFormData.cfPassword === ''){

                    cfPass.classList.add('appear');

                    cfPass.innerHTML = message.mess1.value;

                }

                break;

            case 'phoneNumber':

                if (registerFormData.phonenumber === ''){

                    phone.classList.add('appear');

                    phone.innerHTML = message.mess1.value;

                }

                break;

            default:

                console.log("error");

        }

    }

    userName.addEventListener('blur',() => { handleBlur('userName')});

    passWord.addEventListener('blur',() => { handleBlur('password')});

    confirmPassWord.addEventListener('blur',() => { handleBlur('confirm')});

    phoneNumber.addEventListener('blur',() => { handleBlur('phoneNumber')});

2.5.Xử lí nhập dữ liệu cho ô input

Khi người dùng nhập vào ô input, bỏ tooltip thông báo đi, và lưu dữ liệu người dùng nhập vào trong form

    // function handleClick(){

        userName.addEventListener('keyup', () => {

            user.classList.remove('appear');

            registerFormData.username = userName.value;

            console.log(registerFormData);

        })

        passWord.addEventListener('keyup', () => {

            pass.classList.remove('appear');

            registerFormData.password = passWord.value;

            console.log(registerFormData);

        })

        confirmPassWord.addEventListener('keyup', () => {

            cfPass.classList.remove('appear');

            registerFormData.cfPassword = confirmPassWord.value;

            console.log(registerFormData);

        })

        phoneNumber.addEventListener('keyup', () => {

            phone.classList.remove('appear');

            registerFormData.phonenumber = phoneNumber.value;

            console.log(registerFormData);

        })

2.6.Xử lí cho nút submit

2.6.1.Kiểm tra 2 mật khẩu xem có giống nhau không

submitBtn.addEventListener('click', (*e*) => {

    // check password

    // sử dụng hàm tự gọi chính nó -> iife -> Immediately-invoked Function Expression

    var checkPoint1  = false;

    (function checkPassword () {

        switch (registerFormData.password){

        case '':

            // nếu mk == '' thì thông báo mess1

            pass.classList.add('appear');

            pass.innerHTML = message.mess1.value;

            break;

        case (registerFormData.cfPassword):

            // nếu mk == confirm mk thì đánh dấu checkpoint

            console.log("password ok");;

            checkPoint1 = true;

            break;

        default:

            // mặc định là thông báo mk bị sai

            cfPass.classList.add('appear');

            cfPass.innerHTML = message.mess4.value;

            break;

        }

// phần kiểm tra các trường có null không

//Bỏ mặc định reset trang đi

*e*.preventDefault();

    })();

2.6.2.Kiểm tra xem có trường nào bị empty/null hay không

submitBtn.addEventListener('click', (*e*) => {

// phần kiểm tra 2 mk

// check null field

            // nếu thỏa điều kiện cả 3 trường đều khác rỗng và 2 mk phải giống nhau

            //  -> đăng ký thành công

            var checkPoint = 0;

            if (registerFormData.username === '' ){

                user.classList.add('appear');

                user.innerHTML = message.mess1.value;

            } else { checkPoint ++}

            if (registerFormData.phonenumber === '' ){

                phone.classList.add('appear');

                phone.innerHTML = message.mess1.value;

            } else {checkPoint ++}

            if (registerFormData.cfPassword === '' ){

                cfPass.classList.add('appear');

                cfPass.innerHTML = message.mess1.value;

            } else {checkPoint ++}

            if (checkPoint >= 3 && checkPoint1 === true){

                alert('register successfully');

            }

*e*.preventDefault();

        })