Version 6

task 73: ab i want ki agar mai review add krna chahta hu to usse add krte hue jab mai login page par firse hit kru to us tym par mujhe rather then showing things from the starting it should show me from the last step itself to uske liye i have an object namely **req.originalUrl**. jiski madad se mai simply use kr sakta hu last route.

```
//middleware.js

const isLoggedIn = (req,res,next)=>{
    console.log(req.originalUrl); //added
    if(!req.isAuthenticated()){
        req.flash('error' , 'you need to login first');
        return res.redirect('/login');
    }
    next();
}
```

ab mai vapas ussi page par jaana chahta hu after login so, hum login button hit waali functionality kaise kar paa rhe hai because of our route i.e jahan hum products ko redirect krva rhe hai vahan simply hum previous route i.e **originalUrl** dedenge. vo simply mai session ke andar store krke le sakta hu.

task 74: ab hum kuch autherisation krenge as we are done with authentication.

—> jaise ki jo user hai kya vo **seller** hai ya **buyer.**

to uske liye aap signup.ejs form mei add kroge and simply

```
//signup.ejs
<% layout('layouts/boilerplate') -%>


<div class="row">
    <div class="col-md-6 mx-auto">
        <%- include('../partials/flash')  %>
        <h1 class="display-6">Sign Up</h1>
        <form action="/register" method="post">
            <div class="mb-3">
                <label class="form-label" for="username">Username</label>
                <input class="form-control" type="text" name="username" id="username" placeholder="Username">
            </div>
            <div class="mb-3">
                <label class="form-label" for="email">Email</label>
                <input class="form-control" type="email" name="email" id="email" placeholder="Email">
            </div>
            <div class="mb-3">
                <p class="mb-2">Want to register as ? </p>
                <div class="mx-2">
                    <label class="form-label" for="seller">Seller</label>
                    <input class="form-check-input mt-0" type="radio" name="role" value="seller" id="seller" aria-label="Radio button for fo
                </div>
                <div class="mx-2">
                    <label class="form-label" for="buyer">Buyer</label>
                    <input class="form-check-input mt-0" type="radio" name="role" value="buyer"  id="buyer" aria-label="Radio button for fo
                </div>

            </div>
            <div class="mb-3">
                <label class="form-label" for="password">Password</label>
                <input class="form-control" type="password" name="password" id="password" placeholder="Password">
            </div>
            <button type="submit" class="btn btn-success btn-small">SignUp</button>
        </form>
    </div>
</div>
```

task 75: ab aapke paas ek naya field add hua hai to aap schema mei bhi change kroge.

```
//models --> user.js

const userSchema = new mongoose.Schema({
    email: {
        type: String,
        trim: true,
        required: true
    },
    role:{
        type:String,
        default:'buyer',

    }

});
```

and ab jab aap ek new user create kr rhe ho i.e auth.js mei, to simply you will add the property.

```
//auth.js
router.post('/register' , async(req,res)=>{
    try{
        let {email,username,password,role} = req.body; //added here
        const user = new User({email,username,role}); //added here
        const newUser = await User.register(user,password);
        // res.send(newUser);
        req.login(newUser, function(err) {
            if (err) { return next(err); }
            req.flash('success' , 'welcome, you are registered successfully')
            return res.redirect('/products');
        });
    }
    catch(e){
        req.flash('error' , e.message);
        return res.redirect('/products');
    }
})
```

task 75: ab is role ke basis par mai kuch hide kr skta hu i.e ki user hai ya buyer us basis par cheeze hide krdunga. i.e agar user hai to product add nhi krne dunga lekin agar buyer hai to user ko review and buy krne dunga.

> 💡 ab mai poora user ka collection drop krke simply buyer ka ek banda and user ka ek banda bana dunga so that i can see both functionalities.

ab navbar mei name bhi nazar ajaae to mai usse navbar.ejs mei edit krdeta hu.

```
//navbar.ejs

<%if(!currentUser){%>
          <a href="/login" class="nav-link">Login</a>
        <%} else{%>
          <a href="#" class="nav-link text-capitalize">Hello <%=currentUser.username%></i>  //adding this
            <a href="#" class="nav-link"><i class="fas fa-shopping-cart"></i>
            <sup><span class="badge bg-danger">2</span></sup>
          </a>
          <a href="/logout" class="nav-link">Logout</a>
 <%}%>
```

task 76: ab mai yehi chahunga ki jis bande ne us particular product ko banaya tha vahi isse delete krpae.

iske liye aapko product ke schema mei kuch value add krni hai i.e **author** of that product.

```
//models --> product.js
avgRating: {
        type: Number,
        default:0
    },
    author:{  ////added form this to below
        type:mongoose.Schema.Types.ObjectId,
        ref:'User'
    },
```

ab my task is ki jab bhi mai product ko create kr rha hu mai vahan us product ke saath author bhi add kru.

```
//routes --> product.js
router.post('/products',isLoggedIn,validateProduct,async (req, res) => {

    try {
        const { name, img, desc, price } = req.body;  //added below 1 line
        await Product.create({ name, img, price: parseFloat(price), desc,author:req.user._id });
        req.flash('success', 'Successfully added a new product!');
        res.redirect('/products');
    }
    catch (e) {
        res.status(500).render('error', { err: e.message })
    }
});
```

💡 Ab author ki id store hogyi hogi is db mei check krlo.

task 77: ab simply add krchuke hai new product ko and vahi author ab delete kr sakta hai us product ko so simply ek middleware banalo for the same.

```
//middleware.js

//added poora
const isSeller = (req,res,next)=>{
    if(!req.user.role){
        req.flash('error' , 'you donot have the permission to do that');
        return res.redirect('/products');
    }
    else if(req.user.role !== 'seller'){
        req.flash('error' , 'you donot have the permission to do that');
        return res.redirect('/products');
    }
    next();
}
//changed below
module.exports = {validateProduct ,validateReview , isLoggedIn , isSeller} ;
```

Ab tum isse check krlo ki ye seller hai ya nhi to product.js mei jaakr grab that middleware and check it.

```
//product.js         //changed below
const { validateProduct , isLoggedIn  , isSeller} = require('../middleware');

//added middleware
router.post('/products',isLoggedIn,isSeller,validateProduct,async (req, res) => {

    try {
        const { name, img, desc, price } = req.body;
        await Product.create({ name, img, price: parseFloat(price), desc,author:req.user._id });
        req.flash('success', 'Successfully added a new product!');
        res.redirect('/products');
    }
    catch (e) {
        res.status(500).render('error', { err: e.message })
    }
});
```

task 78: ab agar user seller nhi hai to we can remove the new link ka access from the navbar.

```
//navbar.ejs

<%if(currentUser && currentUser.role === 'seller'){%>
        <li class="nav-item">
          <a class="nav-link" href="/products/new" tabindex="-1" aria-disabled="true">New</a>
        </li>
 <%}%>
```

same goes for buy and delete agar vo seller nhi hai to hata dete hai access.

```
//show.ejs
<%if(currentUser && currentUser.role ==='seller'){%>
                    <a href="/products/<%=product._id%>/edit" class="btn btn-info btn-sm">Edit</a>
                    <form class="d-inline-block" action="/products/<%=product._id%>?_method=DELETE" method="POST">
                        <button class="btn btn-sm btn-danger">Delete</button>
                    </form>
<%}%>
```

task 79: abhi bhi we have to make 1 more check ki kya ye current user humara us product ka auther hai agar nhi hai to we will simply not let him delete or edit.

iske liye again we will use a middleware.

```
//middleware.js
//added poora
const isProductAuthor = async(req,res,next)=>{
//to get the id of thatparticular product
    let {id} = req.params;
    const product = await Product.findById(id);
    console.log(product.author);
    console.log(req.user);
    if(!product.author.equals(req.user._id)){
        req.flash('error' , 'you donot have the permission to do that');
        return res.redirect(`/products/${id}`);
    }
    next();
}

//chnaged below
module.exports = {validateProduct ,validateReview , isLoggedIn , isSeller , isProductRouter} ;
```

go to product.js and add the middleware

```
//product.js

//delete mei added middleware
router.delete('/products/:id',isLoggedIn,isProductAuthor, async (req, res) => {

    try {
        const { id } = req.params;
        await Product.findByIdAndDelete(id);
        res.redirect('/products');
    }
    catch (e) {
        res.status(500).render('error',{err:e.message})
    }
});
```