from django.contrib.auth.models import User  
from django.db import models  
  
class Product(models.Model):  
 name = models.CharField(max\_length=100)  
 price = models.DecimalField(max\_digits=10, decimal\_places=2)  
 description = models.TextField(blank=True)  
  
 def \_\_str\_\_(self):  
 return self.name  
  
class Cart(models.Model):  
 user = models.ForeignKey(User, on\_delete=models.CASCADE)  
 product = models.ForeignKey(Product, on\_delete=models.CASCADE)  
 quantity = models.PositiveIntegerField(default=1)  
  
 def \_\_str\_\_(self):  
 return **f"**{self.user.username} **-** {self.product.name}**"**

forms.py

from django import forms  
from django.contrib.auth.models import User  
  
  
class RegisterForm(forms.ModelForm):  
 # password = forms.CharField(widget=forms.PasswordInput)  
  
 class Meta:  
 model = User  
 fields = [**'username'**, **'email'**, **'password'**]  
  
class LoginForm(forms.Form):  
 username = forms.CharField()  
 password = forms.CharField(widget=forms.PasswordInput)

views

from django.shortcuts import render, redirect  
from django.contrib.auth import authenticate, login  
from .models import Product, Cart  
from .forms import RegisterForm, LoginForm  
from django.contrib.auth.decorators import login\_required  
  
def register\_view(request):  
 if request.method == **"POST"**:  
 form = RegisterForm(request.POST)  
 if form.is\_valid():  
 user = form.save()  
 user.set\_password(form.cleaned\_data[**'password'**])  
 user.save()  
 return redirect(**'login'**)  
 else:  
 form = RegisterForm()  
 return render(request, **'register.html'**, {**'form'**: form})  
  
def login\_view(request):  
 if request.method == **"POST"**:  
 form = LoginForm(request.POST)  
 if form.is\_valid():  
 username = form.cleaned\_data[**'username'**]  
 password = form.cleaned\_data[**'password'**]  
 user = authenticate(request, username=username, password=password)  
 if user:  
 login(request, user)  
 return redirect(**'product'**)  
 else:  
 form = LoginForm()  
 return render(request, **'login.html'**, {**'form'**: form})  
  
def product(request):  
 products = Product.objects.all()  
 return render(request, **'products.html'**, {**'products'**: products})  
  
  
**@login\_required**def addToCart(request, product\_id):  
 # product = Product.objects.get(id=product\_id) # selected product with id  
 # Check if the product is already in the user's cart  
 # cart\_item, created = Cart.objects.get\_or\_create(user=request.user, product=product)  
 # if not created: # If the product already exists in the cart, increase the quantity  
 # cart\_item.quantity += 1  
 # cart\_item.save()  
 # return redirect('view\_cart')  
 product = Product.objects.get(id=product\_id)  
 cart\_item = Cart.objects.filter(user=request.user, product=product).first()  
  
 if cart\_item:  
 cart\_item.quantity += 1  
 cart\_item.save()  
 else:  
 Cart.objects.create(user=request.user, product=product, quantity=1)  
  
 return redirect(**'view\_cart'**)  
  
**@login\_required**def view\_cart(request):  
 cart\_items = Cart.objects.filter(user=request.user) # Get all cart items for the logged-in user  
 return render(request, **'cart.html'**, {**'cart\_items'**: cart\_items})  
  
**@login\_required**def remove(request, item\_id):  
 # try:  
 # # Get the cart item to be removed  
 cart\_item = Cart.objects.get(id=item\_id, user=request.user)  
 cart\_item.delete() # Remove the item from the cart  
 return redirect(**'view\_cart'**)  
 # except Cart.DoesNotExist:  
 # pass # If the item doesn't exist, do nothing  
  
 # return redirect('view\_cart')

pro urls

from django.contrib import admin  
from django.urls import path, include  
  
urlpatterns = [  
 path(**'admin/'**, admin.site.urls),  
 path(**''**, include(**'app1.urls'**)),  
]

app urls

from django.urls import path  
from . import views  
  
urlpatterns = [  
 path(**''**, views.register\_view, name=**'register'**),  
 path(**'login/'**, views.login\_view, name=**'login'**),  
 path(**'products/'**, views.product, name=**'product'**),  
 path(**'addToCart/<int:product\_id>/'**, views.addToCart, name=**'addToCart'**),  
 path(**'view\_cart'**,views.view\_cart,name=**'view\_cart'**),  
 path(**'remove/<int:item\_id>/'**, views.remove, name=**'remove'**),  
]

regisration.html

<h2>Register</h2>  
<form method="post">  
 {% csrf\_token %}  
 {{ form.as\_p }}  
 <button type="submit">Register</button>  
 <a href="{% url 'login' %}">Login</a>  
</form>

Login.html

<h2>Login</h2>  
<form method="post">  
 {% csrf\_token %}  
 {{ form.as\_p }}  
 <button type="submit">Login</button>  
</form>

Products.html

<h2>Products</h2>  
<div>  
 {% for product in products %}  
<!-- <li>{{ product.name }}-->  
<!-- - Rs.{{ product.price }}-->  
<!-- {{ product.description}}-->  
<!-- <a href="{% url 'addToCart' product.id %}">Add to Cart</a>-->  
  
 <div class="product-card " style=" border: 2px solid; width: 200px;">  
 <h3>{{ product.name }}</h3>  
 <p>Price: Rs.{{ product.price }}</p>  
 <p>{{ product.description }}</p>  
 <a href="{% url 'addToCart' product.id %}" >Add to Cart</a>  
 </div>  
  
 {% endfor %}  
</div>

Cart.html

<h2>Your Cart</h2>  
<ol>  
 {% for item in cart\_items %}  
 <li>  
 {{ item.product.name }} - Quantity: {{ item.quantity }}  
 <a href="{% url 'remove' item.id %}">Remove</a>  
 </li>  
 {% empty %}  
 <p>Your cart is empty.</p>  
 {% endfor %}  
</ol>  
  
<!-- Link to continue shopping -->  
<a href="{% url 'product' %}">Go back to products</a>



