**User Registration**

Go to the registration view

Delete the message or comment

**View.py**

from django.shortcuts import render,redirect

from django.contrib import messages,auth

from django.contrib.auth.models import User

def register(request):

if request.method == 'POST':

firstname = request.POST['firstname']

lastname = request.POST['lastname']

username = request.POST['username']

email = request.POST['email']

password = request.POST['password']

confirm\_password = request.POST['confirm\_password']

if password == confirm\_password:

if User.objects.filter(username=username).exists():

messages.error(request,'UserName already exist.')

return redirect('register')

else:

if User.objects.filter(email=email).exists():

messages.error(request,'Email already exist.')

return redirct('register')

else:

user = User.objects.create\_user(first\_name=firstname, last\_name=lastname, username=username, email=email, password=password)

auth.login(request,user)

messages.success(request,'You are logged in successfully.')

return redirect('dashboard')

user.save()

messages.success(request,'You are register successfully.')

return redirct('login')

else:

messages.error(request,'Password do not match.')

return redirect('register')

else:

return render(request,'accounts/register.html')

1. First check that request is POST or not
2. Then we are storing the data in the variables which is passes from web using HTML name keyword.
3. Then we are checking password is matching with confirm password or not. If yes then we will check email and also username if both are exist then we simply print error message.

User

user' is **the model of the auth application** which is part of Django

**We have to import the User model and also auth.**

1. If all the data is correct then we simply pass in the User model so it will contain in user variable. And then
2. We refer this data to the login with **auth.login(request,user)** so it will check and user will login.
3. And we will give the success message and redirect to the dashboard page.
4. And then we will save this user.

Register.html

We have to include the message.html file to show the error message. Before the form

{% include 'includes/messages.html' %}

<form action="{% url 'register' %}" method="POST">

**Login Authentication**

First change in the login.html

We are checking with email do it username.

div class="form-group form-box">

<input type="text" name="username" class="input-text" placeholder="UserName" required>

<i class="flaticon-mail"></i>

</div>

**Login View**

def login(request):

if request.method == 'POST':

username = request.POST['username']

password = request.POST['password']

user = auth.authenticate(username=username, password=password)

if user is not None:

auth.login(request,user)

messages.success(request,'You are logged in.')

return redirect('dashboard')

else:

messages.error(request,'Invalid Username and Password')

return render(request,'accounts/login.html')

**Include the messages.html**

{% include 'includes/messages.html' %}

<form action="{% url 'login' %}" method="POST">

**Logout functionality**

When we login to topbar should be change

It should show dashboard or logout option

And

When we logged Out then it should redirect to home and option should be login – register

Here we used javascripts and we are generating the POST request and making a javascripts form.

We can do it without java scripts but this is not good.

1. we use the **user.is\_authenticated** here we not need to be import user we can use user in all over the project.

**Topbar.html**

<div class="col-lg-4 col-md-4 col-sm-5">

<ul class="top-social-media pull-right">

{% if user.is\_authenticated %}

<li>

<a href="{% url 'dashboard' %}" class="sign-in"><i class="fa fa-techometer"></i> dashboard</a>

</li>

<li>

<a href="javascript:{document.getElementById('logout').submit()}" class="sign-out"><i class="fa fa-sign-out"></i>Logout</a>

<form action="{% url 'logout' %}" id="logout" method="POST">

{% csrf\_token %}

<input type="hidden">

</form>

</li>

{% else %}

<li>

<a href="{% url 'login' %}" class="sign-in"><i class="fa fa-sign-in"></i> Login</a>

</li>

<li>

<a href="{% url 'register' %}" class="sign-in"><i class="fa fa-user"></i> Register</a>

</li>

{% endif %}

</ul>

</div>

Now go to the logout view

**Logout view**

def logout(request):

if request.method == 'POST':

auth.logout(request)

return redirect('home')

return redirect('home')

Now I want to tab Title should be Dynamic

**Base.html**

Go to title tag

<head>

<title>

Car Zone {% block title %} {% endblock %}

</title>

Do it for every page

**Home.html**

{% block title %} | Home {% endblock %}

When we click on any car so it should show the car name.

**Car\_details.html**

{% block title %} | {{ single\_car.car\_title }} {% endblock %}