Конспект 812023

*Сегодня будем делать* SocialNetwork

*сначала создаем проект и приложение.*

*После того как создали проект и перешли в проект пишем:*

django-admin startapp registrationapp

*и регистрируем его с помощью* settings.py *и* INSTALLED\_APPS

Теперь запускаем сервер чтобы создалась база данных

python manage.py migrate (если вы ничего не написали в models.py то он пример все миграции от django)

Далее создаем папки в registrationapp/templates/registrationapp и в нем создаем папки reg\_form.html и reg\_seccess.html

в reg\_form.html в body пишем

<h1>Реєстрація</h1>

в reg\_success.html в body пишем

<h1>Реєстрація успішна</h1>

в views.py пишем:

def show\_reg\_form(request):

return render(request,”registrationapp/reg\_form.html”

def show\_reg\_success(request):

return render(request,”registrationapp/reg\_success.html”

в urls.py подключаем функции из views.py: show\_reg\_form и show\_reg\_success но в path также пишем „reg\_form/“ и „reg\_success/”

Ну и далее остается только запустить локальный сервер «python manage.py runserver»

теперь в reg\_form пишем

<form method=”post”>

<p><input type=”text”></p>

<p><input type=”submit” value=”Реєстрація”></p>

</form>

method в теге form отправляет данные на сервер

есть 2 вида method это get и post

get – возращает ответ

post – просто отправляет данные на сервер

в начало тега form пишем

{% csrf\_token %} - это подтверждение оригинальный формы

подключаем модуль from django.conftib.auth.models impotr User и в django.shortcuts подключаем не только render ну и redirect

<p><input type=”text” placeholder=”Логін” name=”username” required></p>

<p><input type=”password” placeholder=”Пароль” name=”password” required></p>

<p><input type=”password” placeholder=”Підтвердження паролю” name=”password” required></p>

в views.py в show\_reg\_form пишем

if request.method == “POST”:  
 username = request.POST.get(‘username’)

password = request.POST.get(‘password’)

password\_confirm = request.POST.get(‘password\_confirm’)

if password == password\_confirm:

User.objects.create\_user(username=username,password=password)

return redirect(“reg\_success”)

также в самом начале функции пишем

context={}

в urls.py пишем

path(“reg\_success”,show\_reg\_success,name = )

если пароль не равен подтверждение пароля то

else:

context[“error\_text”] = „Паролі не співпадають!“

подключаем модуль from django.db.utils import IntegrityError

если пароль равен подтверждение пароля то

try:

User.objects.create\_user(username = username, password = password)

return redirect(“reg\_success”)

expect IntegrityError:

context[‘error\_text’] = “Такий користувач вже існує!”

в context пишимо

context = {

‘password’:password,

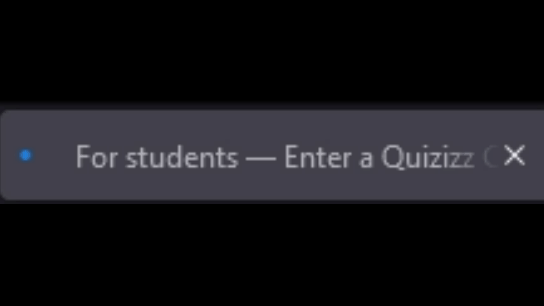
‘password\_confirm’:password\_confirm,

‘username’:username,

}

далее если хотите, можем написать конспект

И.. на этом конспект заканчивается, потому что к вечеру, люди в квартирах пользуется интернетом, каждую полминуты выкидало из конференции на 3-5 минут и на тест я не смог зайти.

Поэтому до свидание.

Этап Тест в конспекте не будет.