Table of Contents

[Day 2](#__RefHeading___Toc478_2080771048)

[1. Make two folder named ‘static’ and ‘templates’ 2](#__RefHeading___Toc478_2080771048)

[2. Go to setting and add static directory(for static work) 2](#__RefHeading___Toc480_2080771048)

[3. Go to Hello/setting and give templates directory(for templates work) 3](#__RefHeading___Toc482_2080771048)

[4. Variables sending to templates 4](#__RefHeading___Toc484_2080771048)

[Day 3](#__RefHeading___Toc486_2080771048)

[1) making website using bootstrap: 5](#__RefHeading___Toc486_2080771048)

[2) Using authentication and creating the username and password 5](#__RefHeading___Toc488_2080771048)

[3) To change the letter and customize admin panel shown in above figure. 6](#__RefHeading___Toc490_2080771048)

[4) templates inheritances: 7](#__RefHeading___Toc492_2080771048)

[Day 4](#__RefHeading___Toc494_2080771048)

[1) templates ma changes garnu parni kura haru 8](#__RefHeading___Toc494_2080771048)

[A) put post and path in contact.html 8](#__RefHeading___Toc496_2080771048)

[B) now making models,register of models: 9](#__RefHeading___Toc498_2080771048)

[step 1: write code in home/models.py (i.e creating models) 9](#__RefHeading___Toc722_2080771048)

[step 2: Go to home/admin.py ( I.e registering your models in apps) 9](#__RefHeading___Toc724_2080771048)

[Step 3: registering models in project. 9](#__RefHeading___Toc726_2080771048)

[Step 4: write code in terminal 10](#__RefHeading___Toc728_2080771048)

[Step 5: making table values through contact file. 11](#__RefHeading___Toc730_2080771048)

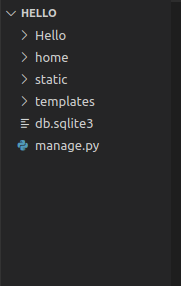
[for defining the table entry name as person on tables: 12](#__RefHeading___Toc732_2080771048)

[home/models.py 12](#__RefHeading___Toc500_2080771048)

Day 2

trying templates.

# 1. Make two folder named ‘static’ and ‘templates’



# **2. Go to setting and add static directory(for static work)**

code:

1. import os #(in the top most of import)
2. #-----------Added manually------------
3. STATICFILES\_DIRS = [
4. os.path.join(BASE\_DIR,"static"),
5. ]

#4 it means when “static” named URL cames then static ma vako folder serve garni

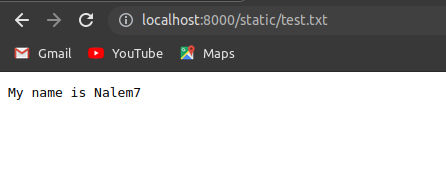
**for testing haii**

go to “static” folder and make file named ‘’test.txt”

write in “test.txt”:

My name is Nalem7

then write “localhost:8000/static/test.txt”



Note: here we put the source code in static not an important data.

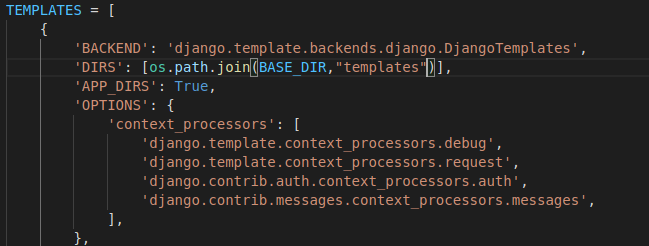
It is seen by public

# 3. Go to Hello/setting and give templates directory(for templates work)

a) search for templates:

make:

‘DIRS’ :[os.path.join(BASE\_DIR,"templates")],



**b) now make “index.html” inside “templates” files then,**

code in index.html file

1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4. <meta charset="UTF-8">
5. <meta http-equiv="X-UA-Compatible" content="IE=edge">
6. <meta name="viewport" content="width=device-width, initial-scale=1.0">
7. <title>NAlem7</title>
8. </head>
9. <body>
10. Welcome to my website guys i am going to make website
11. <p>
12. Lorem ipsum dolor sit.
13. </p>
14. </body>
15. </html>

then save it.

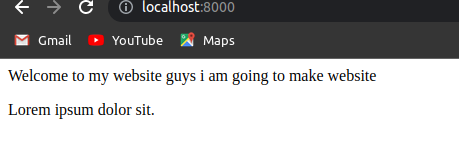
c) Make changes in hello/views.py or write this in terms of return HttpResponse(“hello world!!”).

Code:

1. def index(request):

2. return render(request,’index.html’)

d) After this type “localhost:8080/admin/index.html” in browser.



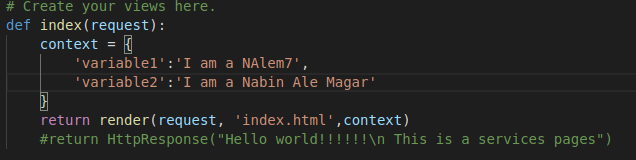
Here templates is seen.

# 4. Variables sending to templates

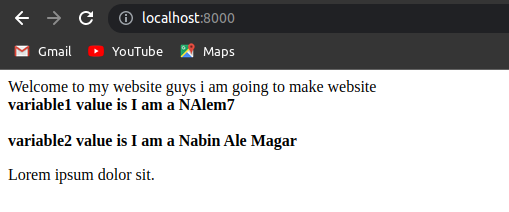
a) code in templates/index.html

1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4. <meta charset="UTF-8">
5. <meta http-equiv="X-UA-Compatible" content="IE=edge">
6. <meta name="viewport" content="width=device-width, initial-scale=1.0">
7. <title>NAlem7</title>
8. </head>
9. <body>
10. Welcome to my website guys i am going to make website
11. <br><b>variable1 value is {{variable1}}</b></br>
12. <br><b>variable2 value is {{variable2}}</b></br>
13. <p>Lorem ipsum dolor sit.</p>
14. </body>
15. </html>

b) code changes in home/views.py

Note: context means the set of variables what we sent to the templates.

c) then reload the browser you will see following output:



**Day 4:**

# 1) making website using bootstrap:

bootstarp is collection of templates files of website.

Write “/admin” in href for admin page loaded.

write “/contact” in href for contact page loaded.

# 2) Using authentication and creating the username and password

commands:

a) will see the databases schema is changes or not

$ python3 manage.py makemigrations

b) make default tables of authentication

$ python3 manage.py migrate

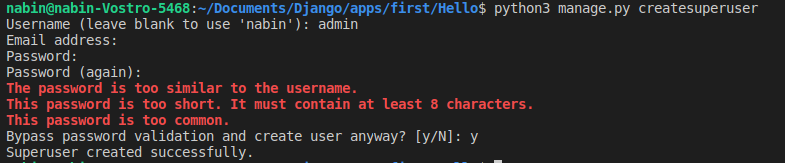
c) Making Superuser

$ python3 manage.py createsuperuser

type username: admin

email: blank

password : admin



d) Go to browser “locslhost:8000” and type

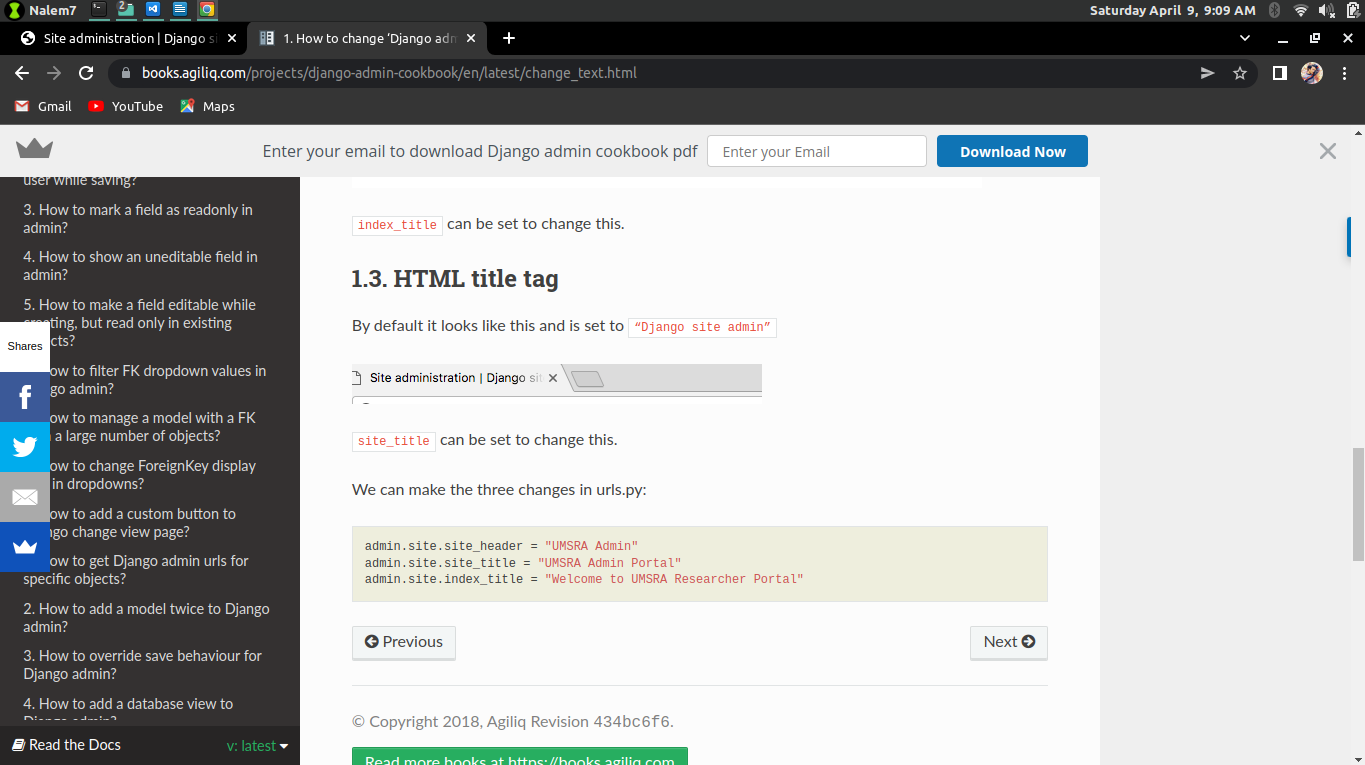
username: admin

password: admin

and enter ‘login’.

# 3) To change the letter and customize admin panel shown in above figure.

a) Go to google and type ”how to change django admin text”

b) copy the text and paste in project “hello/ urls.py”

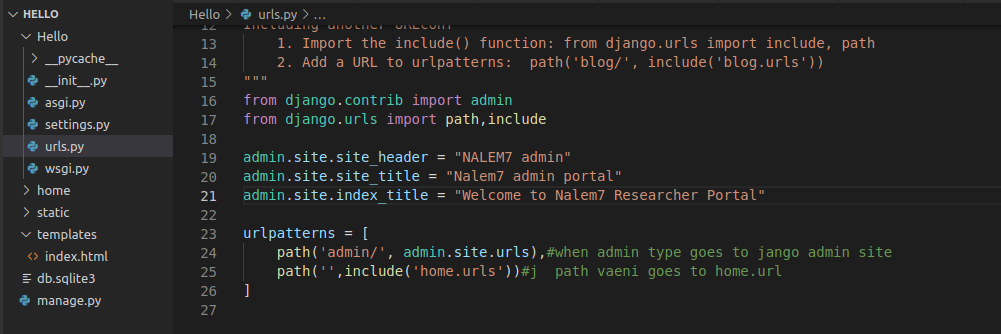
code:

admin.site.site\_header = "UMSRA Admin"

admin.site.site\_title = "UMSRA Admin Portal"

admin.site.index\_title = "Welcome to UMSRA Researcher Portal"

like this:



it will changes the names.

# 4) templates inheritances:

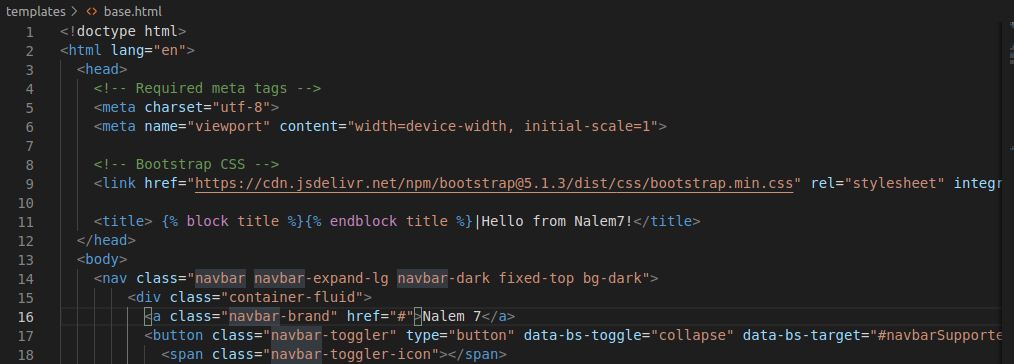
makes an file base.html



database connections:

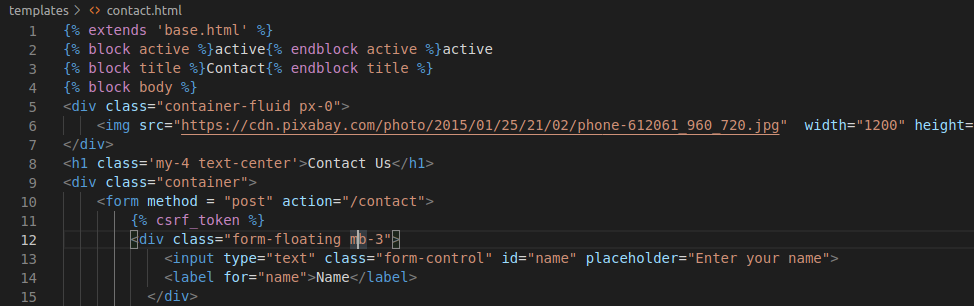
# 1) templates ma changes garnu parni kura haru

**index.html**

#11 will denotes that block pattern of django.(shortcut: type”block” and enter)

Kura haru tyo block vako thauma rakhdinxa arko file bata

# A) put post and path in contact.html

#1 it means base.html content is passed in it which is inheritance

#3 base.html vanni file ma rakh dinxaa block vako thau ma

#10 is needed “**post**” and path “**/contact**”.

#11 **crf tokenization** for security purposes (shortcut: type”crf” and enter)

**Note: you have to give attribute name=”email” for data access**

# B) now making models,register of models:

pahila models banainxa ani migration ma run garera migrate garinxa

models means define the databases.

## step 1: write code in home/models.py (i.e creating models)

code :

1. from django.db import models
2. # Create your models here.
3. class Contact(models.Model):
4. name = models.CharField(max\_length=122)
5. email = models.CharField(max\_length=122)
6. phone = models.CharField(max\_length=122)
7. desc = models.TextField()
8. date = models.DateField()

**Note: to know more(google: ’ modelfield in references ’ )**

## step 2: Go to home/admin.py ( I.e registering your models in apps)

**write code:**

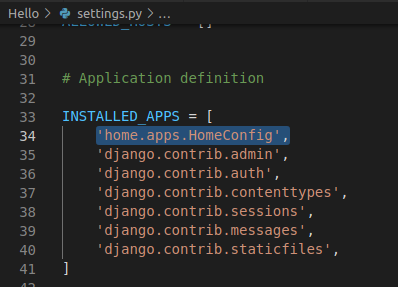
1. from django.contrib import admin
2. from home.models import Contact
3. # Register your models here.
4. admin.site.register(Contact)

## **Step 3: registering models in project.**

a) copy the file name as “HomeConfig“ from home/apps.py



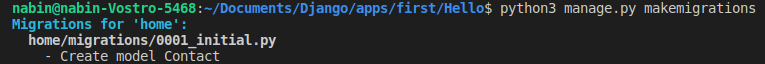
b) then paste on hello/setting.py in installed sections as lined in figure.



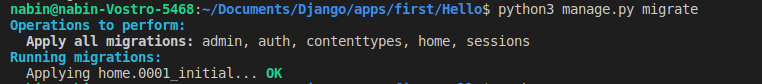
## Step 4: write code in terminal

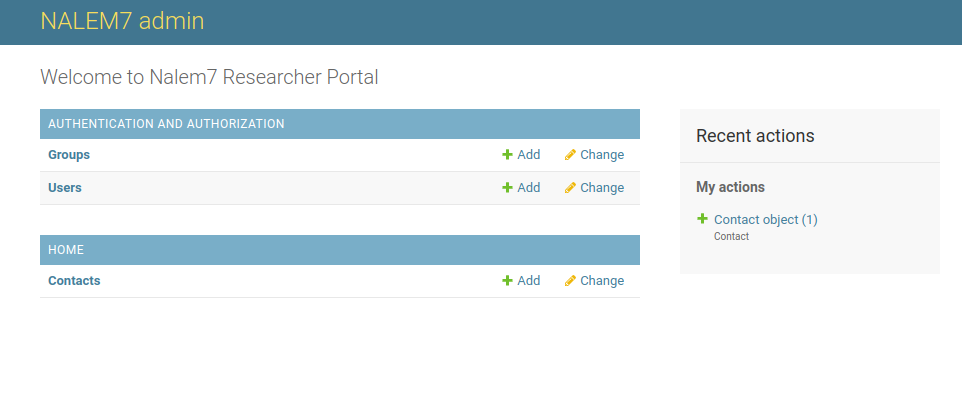
$ python3 manage.py makemigrations

(it will generate the migration files)

$ python3 manage.py migrate

(it will makes the table)



then run server from code and in browser you will see this in localhost:

## Step 5: making table values through contact file.

go to home/views.py and write code:

import:

1. from datetime import datetime
2. from home.models import Contact

code:

1. def contact(request):
2. if request.method == "POST":
3. name = request.POST.get('name')
4. email = request.POST.get('email')
5. phone = request.POST.get('phone')
6. desc = request.POST.get('desc')
7. contact = Contact(name = name, email= email, phone = phone , desc = desc, date = datetime.today())
8. contact.save()
9. return render(request,'contact.html')
10. #return HttpResponse("This is a contact page")

# for defining the table entry name as person on tables:

# home/models.py

code:

def \_\_str\_\_(self):

 return self.name