www.anandjn.com

**Author: Anand Jain**

**Email: anandjn505@gmail.com**

PORTFOLIO  
App : /

\_\_\_

# www.anandjn.com/

## Header / navigation

* Footer / social links / made with <3, django
* Photo : author.jpg
* Name : Anand Jain
* Tag (mini description) : Software Engineer @ IBM
* Greeting : Namaste
* Para 1 : This is para 1
* Para 2: This is para 2
* Para 2: This is para 3
* Download Resume : drive link to resume

\_\_\_

# 18 June

**Pushing code to github**

Avoid leaking any confidential information

* DB username and password
* SECRET KEY

Solution : <https://stackoverflow.com/questions/64208678/hiding-secret-key-in-django-project-on-github-after-uploading-project>

\_\_

# 20 June

**Setting up gitHub ssh keys**

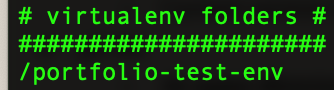
On Mac

Check if ssh key exists

* *ls -al ~/.ssh*
  + Lists all ssh keys we have
* *ssh-keygen -t ed25519 -C* [*anandjn505@gmail.com*](mailto:anandjn505@gmail.com)
  + Generates our private/public key pair after asking our file location and paraphrase
* *eval “$(ssh-agent -s)”*
  + Starts our ssh agent
* *open ~/.ssh/config*
  + Look for config
* *touch ~/.ssh/config*
  + Create config file
* *vim ~/.ssh/config*
  + *Host\**
  + *AddKeychain yes*
  + *UseKeychain yes*
  + *IdentityFIle !/.ssh/id\_ed25519*
    - Name of our private key
    - Save above lines
* *ssh-add -K ~/.ssh/id\_ed25519*
  + Adding our key to keygen after entering paraphrase
* *pbcopy < ~/.ssh/id\_ed25519.pub*
  + Copy our public key to clipboard
  + We can paste this key to our gitHub for further use.

**Setting up Virtual environment**

we create a Virtual environment

* *python -m venv portfolio-env*
  + **cd** to environment and activate
  + *source bin/activate*
* *python -m pip install Django* 
  + Current version 3.2.4
* *git init*
  + **At same directory level as portfolio-test-env**
  + **So that we dont push our environment to gitHub**
* *git remote add origin* [*git@github.com*](mailto:git@github.com)*:anandjn/portfolio-test.git*
* *git pull origin main*
  + Pulling license and readme.md
* Setting up git ignore to ignore our virtual environment
  + In **.gitignore** add
  + 

**Create a project : Portfolio**

* *django-admin startproject portfolio .*
  + “.” help to tell django our project's desired location.
* Hiding our secret key
  + Create a file **.env** at same location as **manage.py** and add
    - SECRET\_KEY = '....your secret key ....' # --- the one indicated in your settings.py, cut and paste it here
  + In **settings.py** add
    - # for hiding secret key #
    - from decouple import config
    - #
    - SECRET\_KEY = config("SECRET\_KEY")
  + Create a file **.gitignore** if not present and add
    - *.env*
  + From terminal install and run
    - *pip install python decouple*
    - *pip freeze>requirements.txt*
      * Creates a requirements.txt file
      * To install from requirement.txt use
        + *pip install -r requirements.txt*

**Create an app: home**

At same location as **manage.py** run

* *python manage.py startapp home*
* In **home/views.py** write
  + from django.shortcuts import render
  + from django.http import HttpResponse
  + # Create your views here.
  + def index(request):
  + return HttpResponse("Namaste!")
* Create a local **urlsCONF** file i.e. **home/urls.py** and add
  + from django.urls import path
  + from . import views
  + urlpatterns = [
  + path('', views.index, name = 'index'),
  + ]
* Connect this local **urlCONF** file to global **urlCONF** i.e in **/portfolio/urls.py** add
  + from django.contrib import admin
  + from django.urls import path, include
  + urlpatterns = [
  + path('', include('home.urls')),
  + path('admin/', admin.site.urls),
  + ]

**Database setup**

Using postgreSQL

* Install **psql binding** 
  + *pip install -U pip*
  + *pip install psycopg2*
* Create SQL database
  + *CREATE DATABASE portfolioAj;*
* Configure Django to use our Database
  + In **settings.py** add
    - DATABASES = {
    - 'default': {
    - 'ENGINE' : 'django.db.backends.postgresql',
    - 'NAME' : config("NAME"),
    - 'USER' : config("USERNAME"),
    - 'PASSWORD' : config("PASSWORD"),
    - 'HOST' : config("HOST"),
    - 'PORT' : config("PORT"),
    - }
    - }
  + Update the **.env** file to pass those environment variables.
    - Dont use “USER” as environment variable
* Also update the TIME\_ZONE
  + TIME\_ZONE = 'Asia/Kolkata'
* Appy basic migrations
  + python mange.py migrate

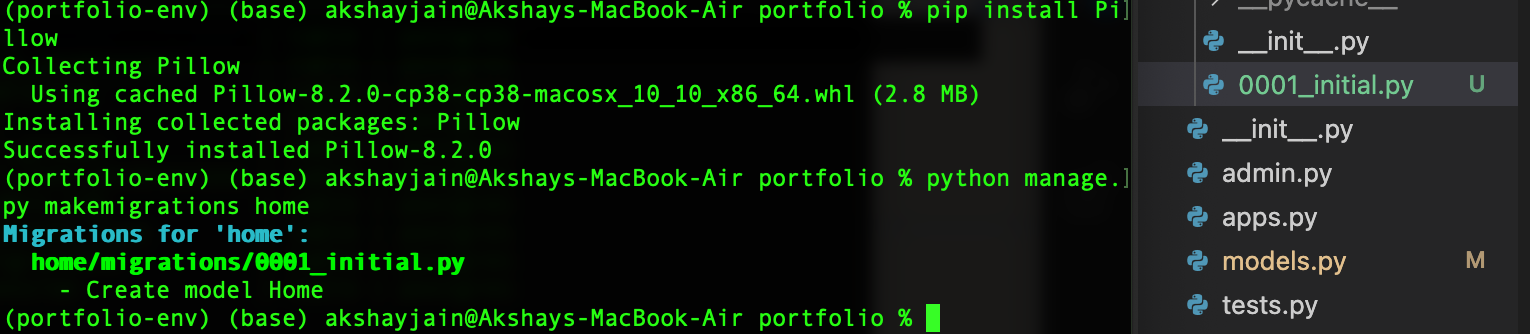
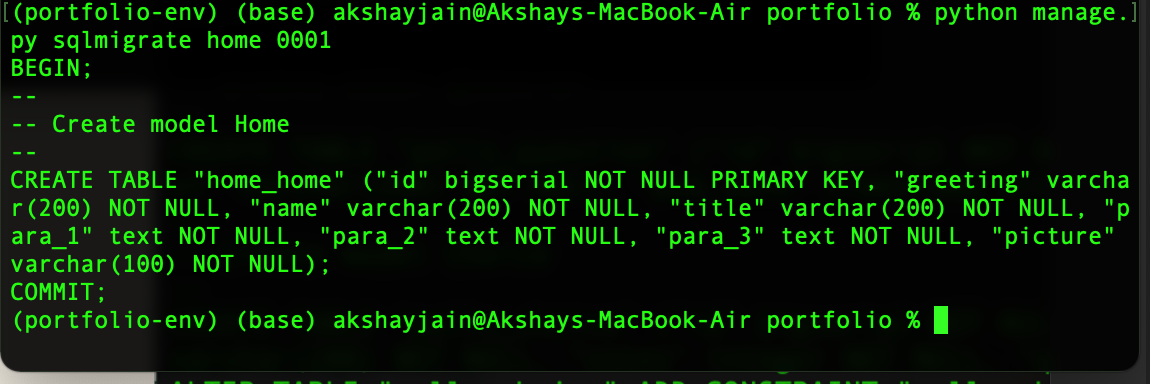
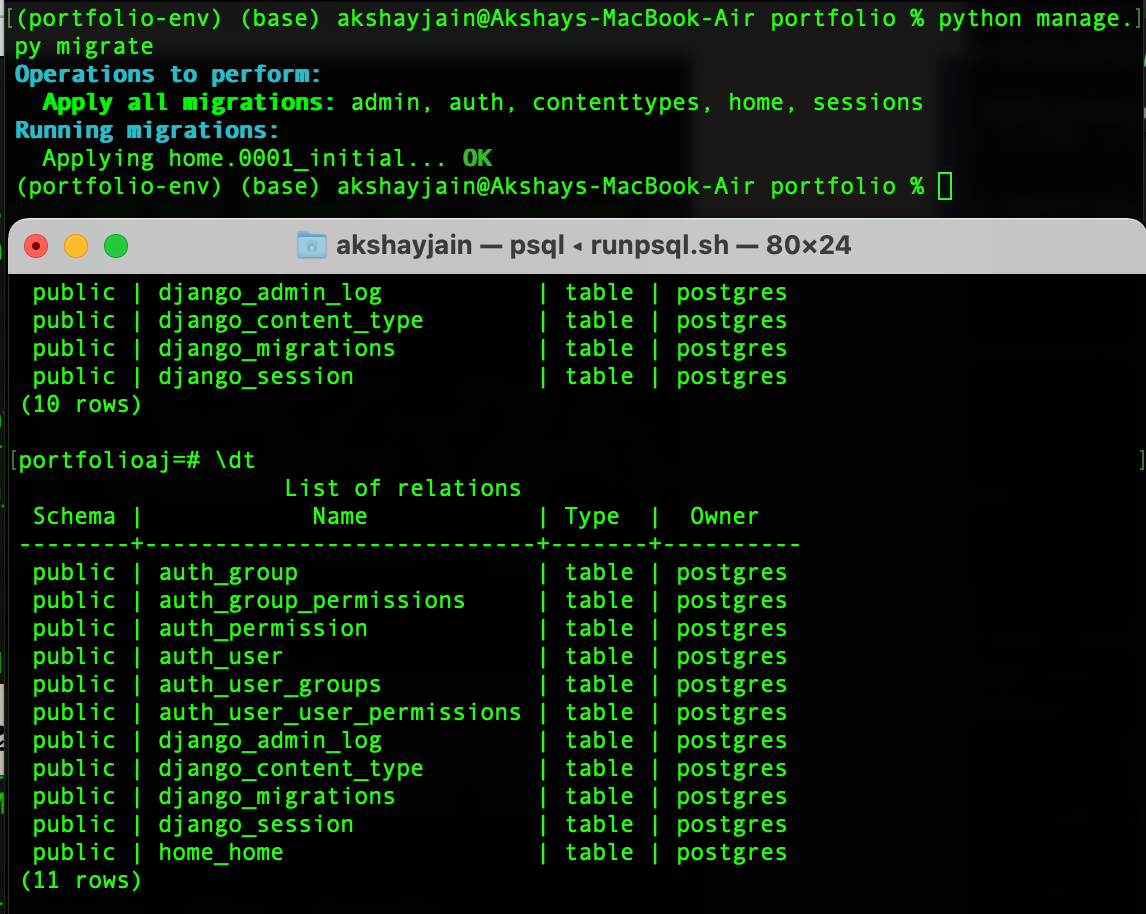
**Model for Home:**

**Home**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Index** | **Greeting** | **Name** | **Title/Tag** | **Para 1** | **Para 2** | **Para 3** | **Picture** |
|  |  |  |  |  |  |  |  |

***(****only 1 row as All the details are for 1 user. i.e. me****)***

# 25 June

* In **models.py** add
* from django.db import models
* # Create your models here.
* def user\_directory\_path(instance, filename):
* #file will be uploaded to MEDIA\_ROOT/images//user\_<id>\_<filename>
* return "images/profile/user\_{0}\_{1}".format(instance.id, filename)
* class Home(models.Model):
* greeting = models.CharField(max\_length=200)
* name = models.CharField(max\_length=200)
* title = models.CharField(max\_length=200)
* para\_1 = models.TextField()
* para\_2 = models.TextField()
* para\_3 = models.TextField()
* picture = models.ImageField(upload\_to = user\_directory\_path)
* In **settings.py** add
* #for geting current path
* import os
* # ROOT Directory
* MEDIA\_URL = '/media/'
* MEDIA\_ROOT = os.path.join(BASE\_DIR, 'media')
* INSTALLED\_APPS = [
* 'home.apps.HomeConfig',
* 'django.contrib.admin',
* 'django.contrib.auth',
* 'django.contrib.contenttypes',
* 'django.contrib.sessions',
* 'django.contrib.messages',
* 'django.contrib.staticfiles',
* ]
  + Connecting our app to main project / installing our app.
* Installing Pillow library to handle images for python
  + *pip install Pillow*
* Pushing these changes to our database
  + *python manage.py makemigrations home*
  + **
  + *python manage.py sqlmigrate home 0001*
  + **
    - View the corresponding query
  + *python manage.py migrate*
  + **
    - Running migrations

# 27 June

* Adding custom method for our **Home Object**
  + **def \_\_str\_\_(self):**
  + **return self.name + "'s Homepage."**

**Creating a superuser**

* *python manage.py create superuser*
  + **Username** : *testadmin*
  + **Useremail** *:* [*testadmin@example.com*](mailto:testadmin@example.com)
  + **Userpassword** *: testpassword*

**Register our model with admin**

To view our models in admin panel

* In **admin.py** add
* from django.contrib import admin
* # Register your models here.
* from .models import Home
* admin.site.register(Home)

**Writing our views**

* **Index view :** this view will take an **id,** fetch a record from model **Home** and send them to the browser.
* In **views.py** write
* from django.shortcuts import render, get\_object\_or\_404
* from django.http import HttpResponse
* from .models import Home
* # Create your views here.
* def index(request, user\_id):
* user\_home = get\_object\_or\_404(Home, pk = user\_id)
* return HttpResponse(f"Namaste! your are looking at Home page of user {user\_home.name}.")
* In **urls.py** write
* urlpatterns = [
* path('<int:user\_id>/', views.index, name = 'index'),
* ]
  + He we are fetching id from browser and passing it our views

**Using Template**

* Create a folder **home/templates/home**
* Create a file **home/templates/home/index.html**
  + **<!DOCTYPE html>**
  + **<html>**
  + **<head>**
  + **<meta charset = "utf-8">**
  + **<title>Home</title>**
  + **</head>**
  + **<body>**
  + **<h1>{{user.greeting}}</h1>**
  + **<h2>I'm {{user.name}}</h2>**
  + **<h3>{{user.title}}</h3>**
  + **<img src = "{{user.picture.url}}", title = "Profile picture">**
  + **<p>{{user.para\_1}}</p>**
  + **<p>{{user.para\_2}}</p>**
  + **<p>{{user.para\_3}}</p>**
  + **</body>**
  + **</html>**
* In **portfolio/urls.py** add
* #for images
* from django.conf import settings
* from django.conf.urls.static import static
* urlpatterns += static(settings.MEDIA\_URL, document\_root = settings.MEDIA\_ROOT)
  + This way our app will be able to locate our media files.
* In **home/views.py** modify **index view**
  + **from django.shortcuts import render, get\_object\_or\_404**
  + **from django.http import HttpResponse**
  + **from .models import Home**
  + **# Create your views here.**
  + **def index(request, user\_id):**
  + **user\_home = get\_object\_or\_404(Home, pk = user\_id)**
  + **context = {**
  + **'user' : user\_home**
  + **}**
  + **return render(request, 'home/index.html', context)**

**Adding some CSS**

* In **setting.py** add
* STATIC\_URL = '/static/'
* STATIC\_ROOT = os.path.join(BASE\_DIR, 'static')
* Create CSS file at **/home/static/home/style.css**
* Include CSS in html using
* <!DOCTYPE html>
* {% load static %}
* <html>
* <head>
* <meta charset = "utf-8" name = "viewport" content = "width = device-width, initial-scale=1.0">
* <link rel = "stylesheet", type = "text/css", href = {% static "home/style.css"%}>
* <title>Home</title>
* </head>

# 2 July

* In CSS add
* /\* General formatting \*/
* body {
* background-color: rgb(236, 236, 236);
* font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
* }
* h1,h2,h3,h4,h5,h6 {
* letter-spacing: 2px;
* }
* a{
* text-decoration: none;
* }
* .social a {
* margin-right: 3px;
* }
* .social::before{
* content: "";
* display: block;
* width: 20%;
* background-color: #e1e1e1;
* height: 8px;
* margin-left: auto;
* margin-right: auto;
* margin-bottom: 15px;
* }
* /\* Section specefic formatting \*/
* In **index.html** add
* <!DOCTYPE html>
* {% load static %}
* <html lang = "en">
* <head>
* <meta charset = "utf-8">
* <meta name = "viewport" content = "width = device-width, initial-scale=1.0">
* <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
* <link rel = "stylesheet", type = "text/css", href = {% static "home/style.css"%}>
* <title>Home</title>
* </head>
* <body>
* <header>
* <!-- Navigation (Sits on top) -->
* <div class="w3-top w3-bar w3-padding w3-card w3-wide w3-light-grey w3-animate-top">
* <div class = "w3-container">
* <a href="#" class = "w3-bar-item w3-opacity" > {{user.greeting}}, <br> I'm <strong>{{user.name}}</strong> </a>
* </div>
* <!-- Right-sided navbar links. Hide them on small screens -->
* <div class = "w3-right w3-display-right w3-hide-small w3-container">
* <a href="#" class="w3-bar-item w3-button">Resume</a>
* <a href="#" class="w3-bar-item w3-button">Blog</a>
* <a href="#" class="w3-bar-item w3-button">Projects</a>
* </div>
* </div>
* <!-- NAVIGATION ENDS-->
* </header>
* <!-- ABOUT -->
* <section class = "w3-container w3-cell-row w3 w3-padding-top-32 w3-section" id ="about">
* <!-- INTRO CARD -->
* <div class ="w3-container w3-cell w3-third w3-padding-large w3-animate-left w3-display-container">
* <center>
* <img src = "{{user.picture.url}}", alt = "Profile picture" class = "w3-circle w3-image" style="width:35%">
* </center>
* <!-- Card text. Hide them on small screens -->
* <div class ="w3-container w3-twothird w3-right w3-border-right w3-hide-small">
* <h4>{{user.greeting}}</h4>
* <h3>I'm {{user.name}}</h3>
* <h6>{{user.title}}</h6>
* </div>
* </div>
* <!-- INTRO CARD ENDS-->
* <!-- CONTENT -->
* <div class ="w3-container w3-cell w3-twothird w3-padding-large">
* <h1 class="w3-center">About me</h1><br>
* <p>{{user.para\_1}}</p>
* <p>{{user.para\_2}}</p>
* <p>{{user.para\_3}}</p>
* </div>
* <!-- CONTENT ENDS-->
* </section>
* <!-- ABOUT ENDS -->
* <!-- SOCIAL -->
* <section class = "w3-container w3-center w3-margin-top social" id = "social">
* <a href = "#"> YT </a>
* <a href = "#"> FB </a>
* <a href = "#"> TW </a>
* <a href = "#"> IN </a>
* <a href = "#"> IG </a>
* </section>
* <!-- SOCIAL ENDS-->
* </body>
* </html>

After adding some users the web-page will look like this

* 