



Django Project MVT Structure

Last Updated : 16 Aug, 2021

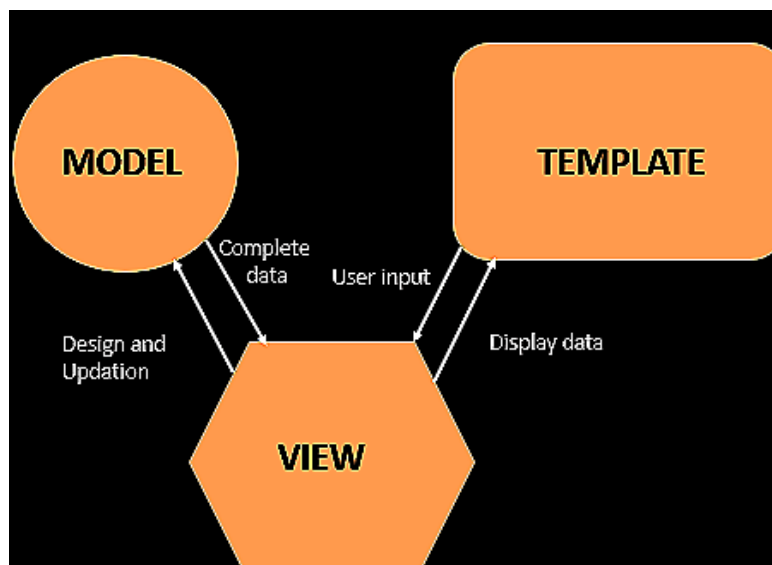
Django is based on **MVT (Model-View-Template)** architecture. MVT is a software design pattern for developing a web application.

MVT Structure has the following three parts –

Model: The model is going to act as the interface of your data. It is responsible for maintaining data. It is the logical data structure behind the entire application and is represented by a database (generally relational databases such as MySQL, Postgres). To check more, visit – [Django Models](#)

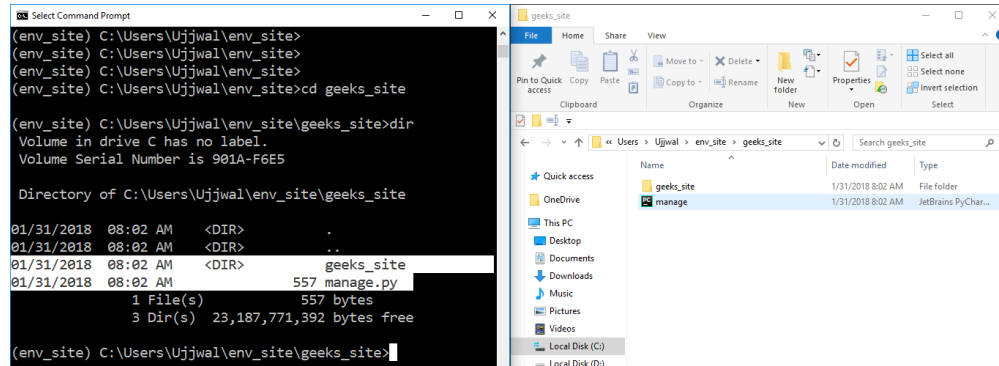
View: The View is the user interface — what you see in your browser when you render a website. It is represented by HTML/CSS/Javascript and Jinja files. To check more, visit – [Django Views](#).

Template: A template consists of static parts of the desired HTML output as well as some special syntax describing how dynamic content will be inserted. To check more, visit – [Django Templates](#)

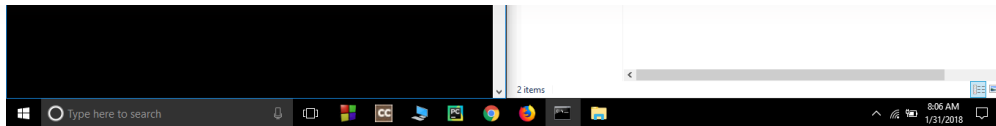


Project Structure :

A Django Project when initialized contains basic files by default such as manage.py, view.py, etc. A simple project structure is enough to create a single-page application. Here are the major files and their explanations. Inside the `geeks_site` folder (project folder) there will be the following files-



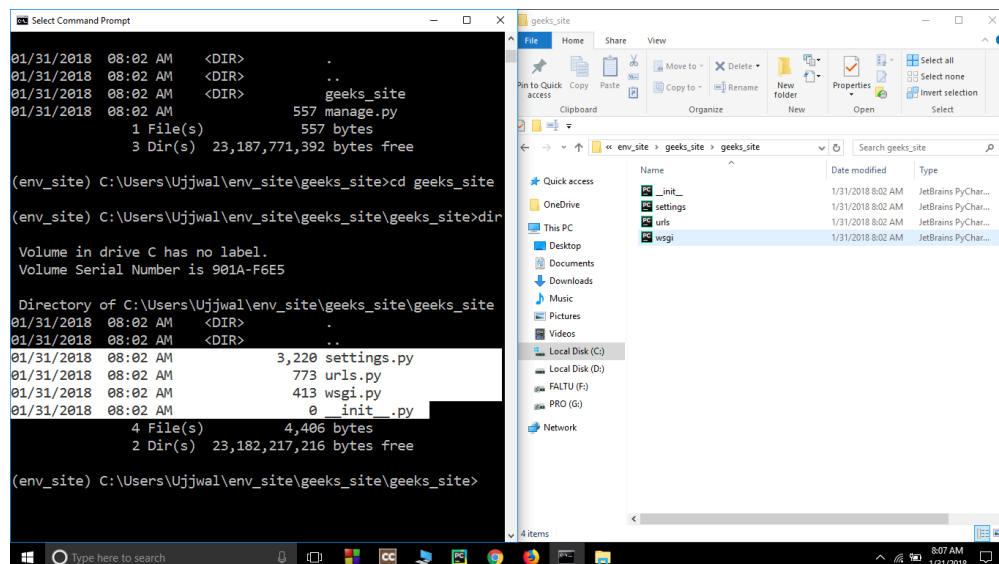
Django Views Model Template Forms Jinja Python SQLite Flask Json Postman Interview Ques



manage.py- This file is used to interact with your project via the command line(start the server, sync the database... etc). For getting the full list of commands that can be executed by manage.py type this code in the command window-

```
$ python manage.py help
```

folder (geeks_site) – This folder contains all the packages of your project. Initially, it contains four files –



- **_init_.py** – It is a python package. It is invoked when the package or a module in the package is imported. We usually use this to execute package initialization code, for example for the initialization of package-level data.
- **settings.py** – As the name indicates it contains all the website settings. In this file, we register any applications we create, the location of our static files, database configuration details, etc.
- **urls.py** – In this file, we store all links of the project and functions to call.
- **wsgi.py** – This file is used in deploying the project in WSGI. It is used to help your Django application communicate with the webserver.

Are you ready to elevate your web development skills from foundational knowledge to advanced expertise? Explore our [Mastering Django Framework - Beginner to Advanced Course](#) on GeeksforGeeks, designed for aspiring developers and experienced programmers. This comprehensive course covers everything you need to know about Django, from the basics to advanced features. Gain practical experience through **hands-on projects** and real-world applications, mastering essential Django principles and techniques. Whether you're just starting or looking to refine your skills, this course will empower you to build sophisticated web applications efficiently. Ready to enhance your web development journey? Enroll now and unlock your potential with Django!

N Nave... + Follow



110

Previous Article

When to Use Django? Comparison with other Development Stacks

Next Article

How to Create a Basic Project using MVT in Django ?

Similar Reads

How to Create a Basic Project using MVT in Django ?

Prerequisite - Django Project MVT Structure Assuming you have gone through the previous article. This article focuses on creating a basic project to render a...

2 min read

Adding Tags Using Django-Taggit in Django Project

Django-Taggit is a Django application which is used to add tags to blogs, articles etc. It makes very easy for us to make adding the tags functionality to our django...

2 min read

Best Practice for Django Project Working Directory Structure

When organizing a Django project, the best practices regarding the structure of the working directory are followed to ensure that the project is maintainable,...

8 min read

How to customize Django forms using Django Widget Tweaks ?

Django forms are a great feature to create usable forms with just few lines of code. But Django doesn't easily let us edit the form for good designs. Here, we...

3 min read

Styling Django Forms with django-crispy-forms

Django by default doesn't provide any Django form styling method due to which it takes a lot of effort and precious time to beautifully style a form. django-crispy...

1 min read

Integrating Django with Reactjs using Django REST Framework

In this article, we will learn the process of communicating between the Django Backend and React js frontend using the Django REST Framework. For the sake...

15+ min read

How to Fix Django "ImportError: Cannot Import Name 'six' from..."

After Django 3.0, django.utils.six was removed. The "ImportError: Cannot Import Name 'six' from 'django.utils'" error occurs because of the dependency issue. Thi...

3 min read

Django Introduction | Set 2 (Creating a Project)

Note- This article is in continuation of Django introduction. Popularity of Django Django is used in many popular sites like as: Disqus, Instagram, Knight...

3 min read

How to add RSS Feed and Sitemap to Django project?

This article is in continuation of Blog CMS Project in Django. Check this out here - Building Blog CMS (Content Management System) with Django RSS (Really...

3 min read

Top 10 Reasons to Choose Django Framework For Your Project

When it comes to choosing a new language or framework for a project what matters to most of the developers? Simplicity? Reliability? Built-in packages to...

9 min read

Article Tags : [Python](#) [Python Django](#)

Practice Tags : [python](#)



Corporate & Communications Address:-
A-143, 9th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305) | Registered Address:- K 061,
Tower K, Gulshan Vivante Apartment,
Sector 137, Noida, Gautam Buddh
Nagar, Uttar Pradesh, 201305



Company

[About Us](#)
[Legal](#)
[In Media](#)
[Contact Us](#)
[Advertise with us](#)
[GFG Corporate Solution](#)
[Placement Training Program](#)

Languages

[Python](#)
[Java](#)
[C++](#)
[PHP](#)
[GoLang](#)
[SQL](#)
[R Language](#)

[GeeksforGeeks Community](#)[Android Tutorial](#)[Tutorials Archive](#)

DSA

[Data Structures](#)[Algorithms](#)[DSA for Beginners](#)[Basic DSA Problems](#)[DSA Roadmap](#)[Top 100 DSA Interview Problems](#)[DSA Roadmap by Sandeep Jain](#)[All Cheat Sheets](#)

Web Technologies

[HTML](#)[CSS](#)[JavaScript](#)[TypeScript](#)[ReactJS](#)[NextJS](#)[Bootstrap](#)[Web Design](#)

Computer Science

[Operating Systems](#)[Computer Network](#)[Database Management System](#)[Software Engineering](#)[Digital Logic Design](#)[Engineering Maths](#)[Software Development](#)[Software Testing](#)

System Design

[High Level Design](#)[Low Level Design](#)[UML Diagrams](#)[Interview Guide](#)[Design Patterns](#)[OOAD](#)[System Design Bootcamp](#)[Interview Questions](#)

School Subjects

[Mathematics](#)[Physics](#)[Chemistry](#)[Biology](#)

Data Science & ML

[Data Science With Python](#)[Data Science For Beginner](#)[Machine Learning](#)[ML Maths](#)[Data Visualisation](#)[Pandas](#)[NumPy](#)[NLP](#)[Deep Learning](#)

Python Tutorial

[Python Programming Examples](#)[Python Projects](#)[Python Tkinter](#)[Web Scraping](#)[OpenCV Tutorial](#)[Python Interview Question](#)[Django](#)

DevOps

[Git](#)[Linux](#)[AWS](#)[Docker](#)[Kubernetes](#)[Azure](#)[GCP](#)[DevOps Roadmap](#)

Interview Preparation

[Competitive Programming](#)[Top DS or Algo for CP](#)[Company-Wise Recruitment Process](#)[Company-Wise Preparation](#)[Aptitude Preparation](#)[Puzzles](#)

GeeksforGeeks Videos

[DSA](#)[Python](#)[Java](#)[C++](#)

| | |
|-----------------|-----------------|
| Social Science | Web Development |
| English Grammar | Data Science |
| Commerce | CS Subjects |
| World GK | |

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved