



When to Use Django? Comparison with other Development Stacks

Last Updated: 17 Jan 2022

[Django](#) [Views](#) [Model](#) [Template](#) [Forms](#) [Jinja](#) [Python SQLite](#) [Flask](#) [Json](#) [Postman](#) [Interview Ques](#)

Prerequisite – [Django Introduction and Installation](#)

When to Use Django and Why?

After getting to know the basics of [Python](#), one should know when to use Django and why?

Django is a high-level python based web framework which allows you to quickly create web applications without all of the installation or dependency problems that you normally will find with other frameworks.

One should be using Django for web development in the following cases:

- For developing a Web Application or API Backend.
- For Rapid Development of some web application.
- Deploying the application Fast and Scaling it according to your needs
- A Perfect ORM for working with databases instead of database queries
- To develop a secure single-page application for either retrieving data or posting data.

Django	MEAN/MERN	Spring
Python	JavaScript	Java
Open Source	NoSQL Databases	Open Source
Great Community	Modularity	Powerful
Easy to Learn	Mongoose	Dependency Injection

Django	MEAN/MERN	Spring
MVT Based	Organized	Stability
Batteries Included Framework	Dynamic Front End	MVC

Django is a good choice when:

- 1.You need to build a robust, scalable web application with a lot of data.
Django’s built-in ORM (Object-Relational Mapping) and support for database migrations make it easy to work with large amounts of data, and its built-in admin interface makes it easy to manage that data.
- 2.You need to build an application with a lot of user-facing functionality.
Django’s built-in forms, authentication, and authorization make it easy to implement user registration, login, and other common functionality.
- 3.You need to get an application up and running quickly. Django’s “batteries-included” philosophy means that it comes with a lot of built-in functionality, so you can get an application up and running quickly without having to write a lot of boilerplate code.
- 4.You want to use Python for your backend. Since Django is built on Python, it’s a natural choice if you’re already familiar with the language or want to use it for your backend.

When compared to other development stacks, Django may not be the best choice for certain types of projects. For example:

- 1.If you’re building a small, simple website or a single-page application (SPA), a JavaScript-based framework like React or Angular might be a better choice.
- 2.If you’re building a real-time, highly-concurrent application, a framework like Node.js might be a better choice since it is built on an event-driven, non-blocking I/O model.
- 3.If you’re building a mobile application, you’ll probably want to use a mobile-specific framework like React Native or Xamarin.

Ultimately, the choice of web development stack will depend on the specific requirements of your project and your team’s expertise. Django is a great choice for many types of web applications, but it may not be the best choice for every project.

Here is an example of a simple Django view function that handles a GET request and returns a template:

```
from django.shortcuts import render

def my_view(request):
    if request.method == 'GET':
        return render(request, 'mytemplate.html')
```

In this example, the my_view function is handling a GET request and using the render shortcut function to return the mytemplate.html template. The template will be rendered with the context data supplied by the view.

Here is an example of a view that handles a POST request and performs some action with the data:

```
from django.shortcuts import render, redirect
from .models import MyModel

def my_view(request):
    if request.method == 'POST':
        form = MyForm(request.POST)
        if form.is_valid():
            data = form.cleaned_data
            MyModel.objects.create(**data)
            return redirect('success_page')
    else:
        form = MyForm()
    return render(request, 'mytemplate.html', {'form': form})
```

In this example, the view is handling a POST request, it creates an instance of MyForm with the data from the request, it validates the form and if it's valid, it creates an object with MyModel with the cleaned data. After that, it redirects the user to a success page.

If the request method is GET, it creates an empty form and render the template passing the form as context data.

Please note that you need to have the necessary imports and correctly set up models and forms for the above examples to work.

Also, Django views can also be written as class-based views which are more powerful and flexible.

Also check –

- [Django Project MVT Structure](#)
- [How to Create a Basic Project using MVT in Django ?](#)

Companies using Django:

- Instagram
- Disqus
- Pinterest
- Mozilla Firefox
- Spotify
- YouTube

Conclusion :

Django is a rapid web development framework and if you want to get your application built fast within a few days, there is no better framework than Django Web Framework. Django gives all the features included, also called “Batteries Included Framework”. It has a built-in admin interface which makes it easy to work with it.

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



46

Previous Article

[Django Installation and Setup](#)

Next Article

[Django Project MVT Structure](#)

Similar Reads

Comparison of Java with other programming languages

Java is one of the most popular and widely used programming language and platform. A platform is an environment that helps to develop and run programs...

[4 min read](#)

Comparison of Python with Other Programming Languages

Python is an easily adaptable programming language that offers a lot of features. Its concise syntax and open-source nature promote readability and...

[3 min read](#)

Comparison of FastAPI with Django and Flask

For starters, as you may know, Python is an interpreted programming language that is becoming more and more popular for building web applications. Howeve...

[4 min read](#)

Choosing the Right IDE for Python Development: A Comparison

Python, being one of the most popular programming languages, has a rich ecosystem of development environments. Selecting the right Integrated...

[5 min read](#)

Using Stacks to solve Desert Crossing Problem in Python

A man is using a compass to cross a desert, and he has been given specific directions for his task. Following directions is only permitted: "NORTH", "EAST",...

[5 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 djang...

2 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

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
GeeksforGeeks Community

Languages

Python
Java
C++
PHP
GoLang
SQL
R Language
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

Data Science & ML

Data Science With Python
Data Science For Beginner
Machine Learning
ML Maths
Data Visualisation
Pandas
NumPy
NLP
Deep Learning

Web Technologies

HTML
CSS
JavaScript
TypeScript
ReactJS
NextJS
Bootstrap
Web Design

Python Tutorial

Python Programming Examples
Python Projects
Python Tkinter
Web Scraping
OpenCV Tutorial
Python Interview Question
Django

Computer Science

Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design

DevOps

Git
Linux
AWS
Docker
Kubernetes

Engineering Maths
Software Development
Software Testing

Azure
GCP
DevOps Roadmap

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD
System Design Bootcamp
Interview Questions

Inteview Preparation

Competitive Programming
Top DS or Algo for CP
Company-Wise Recruitment Process
Company-Wise Preparation
Aptitude Preparation
Puzzles

School Subjects

Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar
Commerce
World GK

GeeksforGeeks Videos

DSA
Python
Java
C++
Web Development
Data Science
CS Subjects

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved