



Django project – Creating a Basic E-commerce Website for Displaying Products

Last Updated : 24 Feb, 2022

Project Title – Basic Ecommerce Website using Django

Django is a powerful framework based on python. Here we will see how to create a basic e-commerce website in Django. This project includes storing products in the database and show them on the website.

Refer to the following articles to check how to create a project and an app in Django.

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

How to Create a Basic Ecommerce Website using Django?

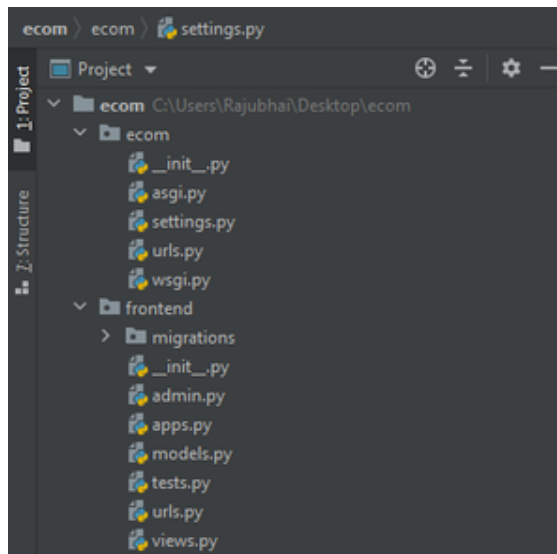
Now when you have successfully installed Django. Create a new project –

```
django-admin startproject ecom
```

Now create a new app called frontend inside the ecom project. Now we have 1 project and 1 app inside of that project.]

```
django-admin startapp frontend
```

Directory structure –



Creating URLs –

In *ecom* > *urls.py* add the following lines. This file handles main project URLs. But we don't want to disturb it, so we will do our work in *frontend* > *URLs.py*. And for that we need to include *frontend* > *URLs* inside the *ecom* > *URLs*.

```
urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('frontend.urls')),
]
```

Creating Models –

Add a product model here.

```
class Product(models.Model):
    productname = models.CharField(max_length=200)
    price = models.DecimalField(max_digits=5, decimal_places=2)
    description = models.TextField()
    image = models.CharField(max_length=5000, null=True, blank=True)
```

To know more about Django models, check out – [Django Models](#)

Register Models into admin –

Just after creating model, we should register that form into *admin.py*.

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

```
from django.contrib import admin
from .models import *
```

```
admin.site.register(Product)
```

Creating Views for Displaying Products –

In *frontend > views.py* we write a function to get and display products from our database.

```
from django.shortcuts import render
from .models import *

def products(request):
    products = Product.objects.all()
    return render(request, 'products.html', {'products':products})
```

Creating Url for products –

Here we will set dynamic URL which can be useful to display our products. In *frontend > urls.py*

```
from django.urls import path
from . import views

urlpatterns = [

    path('products/', views.products, name='products'),

]
```

Creating Template –

Creating template depends upon how you want to display products on website. For displaying products we have shared a simple code. Add it in *frontend > templates > products.html*

```
{% for product in products %}
<div class="card" style="width: 18rem;">
    
    <div class="card-body">
```

```
<h5 class="card-title">{{ product.productname }}</h5>
<p class="card-text">{{ product.description }}</p>
<p class="card-text">Price - {{ product.price }}</p>
<a href="#" class="btn btn-primary">Buy Now</a>
</div>
</div>
{% endfor %}
```

Makemigrations and Migrate –

Now its time to migrate our model into database. First, we have to create migrations. For that type following code into terminal.

```
python manage.py makemigrations
```

After creating migrations type following code to apply those migrations

```
python manage.py migrate
```

To know more about makemigrations and migrate, check out – [Basic App Model – Makemigrations and Migrate](#)

Creating Superuser –

Now, create Django superuser for handling admin stuff. Type following command in terminal

```
django-admin createsuperuser
```

Then it will ask for username, email and password.

Run the app

After creating superuser, in terminal type,

```
python manage.py runserver
```

to start the server and go to admin panel (<http://localhost:8000/admin>) and add some products to the database.

Output –



Thats it. Your basic eCommerce site is ready where you can display your products

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!

Romil...



12

Next Article

Django Introduction | Set 2 (Creating a Project)

Similar Reads

- E-commerce Website using Django

This project deals with developing a Virtual website ‘E-commerce Website’. It provides the user with a list of the various products available for purchase in the...

11 min read
- E-commerce Website using Tailwind and React using Django

Our e-commerce website, "ECOM," aims to offer users a smooth online shopping experience. We use Tailwind CSS for styling, React for interactive user interface...

6 min read

Django - Creating a Multi-Page Website

Django is a python based open-source framework used for web development. Django allows the usage of the M-T-V (Model-Templates-View) architectural...

8 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 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

E-commerce Product Catalog using Django

Django is an excellent choice for building a robust product catalog. Django, a high-level Python web framework, provides a solid foundation for creating...

4 min read

Tableau - Displaying data in Worksheet

After creating the visualisations or while creating the visualisations if you want to look at the data in a worksheet, then, in that case, you need to follow some step...

3 min read

Displaying the coordinates of the points clicked on the image using Python...

OpenCV helps us to control and manage different types of mouse events and gives us the flexibility to operate them. There are many types of mouse events....

3 min read

Python - Displaying real time FPS at which webcam/video file is processed...

We will be displaying the real-time processing FPS of the video file or webcam depending upon our choice.FPS or frame per second or frame rate can be define...

4 min read

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

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

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
- Social Science
- English Grammar

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++
- Web Development
- Data Science

@GeeksforGeeks, Sanchhaya Education Private Limited, All rights reserved