

What is Django?

Features and creating your first project.



Django কি? এর ফিচারস এবং এটা দিয়ে আপনার প্রথম প্রোজেক্ট তৈরি।

Django হলো একটি high-level **Python web framework**, যা ডেভেলপারদের robust, secure এবং scalable web applications দ্রুত তৈরি করতে সাহায্য করে। এটি database handling, user authentication, এবং URL routing-এর মতো অনেক কাজ সহজ করে, যাতে ডেভেলপাররা repetitive tasks নিয়ে চিন্তা না করে তাদের অ্যাপ্লিকেশন তৈরি করতে ফোকাস করতে পারে।

নিচে Django-এর ফিচারস এবং step-by-step প্রথম প্রজেক্ট তৈরি করার প্রক্রিয়া সহজ ভাষায় ব্যাখ্যা করা হলো, সাথে কিছু real-life উদাহরণ।



Django কী?

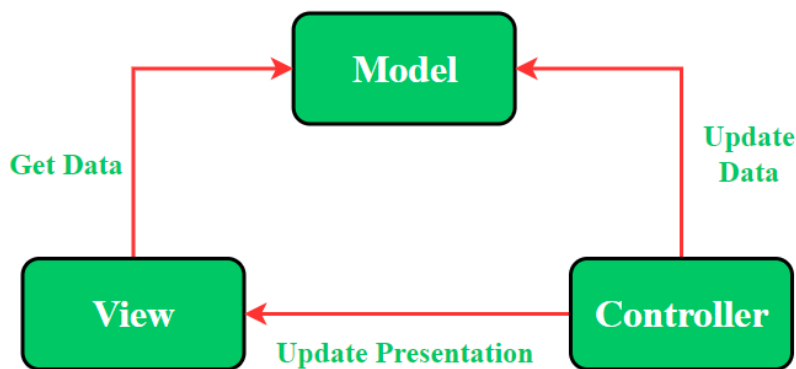
Django হলো pre-built tools-এর একটি সেট, যেটি websites তৈরি করার প্রক্রিয়াকে সহজ এবং দ্রুত করে তোলে। ধরুন, আপনি একটি বাড়ি তৈরি করবেন। আপনি যদি প্রতিটি টুল নিজে বানান, তাহলে সময় বেশি লাগবে। Django সেই tools-গুলো ready-made ভাবে দিয়ে দেয়, যাতে আপনি ফোকাস করতে পারেন আপনার মূল কাজের উপর।

রিয়াল-লাইফ উদাহরণ

একটি অনলাইন স্টোর (যেমন **Amazon**) বা ব্লগ সাইট (যেমন **WordPress**) তৈরি করার কথা ভাবুন। Django আপনার জন্য backend কাজগুলো (যেমন database management, user authentication) সহজ করে দেয়, যাতে আপনি আপনার users-এর জন্য সেরা experience তৈরি করতে পারেন।

Django-এর Key Features

a. MVC Architecture



Django **Model-View-Controller (MVC)** আর্কিটেকচার ফলো করে:

- **Model:** Database manage করে (যেমন user information সংরক্ষণ)।
- **View:** User-দের কাছে data প্রদর্শন করে (যেমন product page)।
- **Controller:** Model এবং View-এর মধ্যে connection তৈরি করে।

রিয়াল-লাইফ উদাহরণ

ধরুন একটি স্কুল management system। এখানে **Model** স্টুডেন্টদের data store করবে, **View** ওয়েবসাইটে student list প্রদর্শন করবে এবং **Controller** database থেকে data এনে তা user-দের দেখাবে।

b. Built-in Admin Interface

Django একটি automatic admin interface প্রদান করে, যেখানে আপনি app-এর data manage করতে পারবেন। যেমন, আপনি একটি ব্লগ তৈরি করলে, admin panel থেকে posts add, edit বা

delete করতে পারবেন।

c. Security

Django built-in security features নিয়ে আসে, যা SQL injection, cross-site scripting (XSS), এবং clickjacking-এর মতো common threats থেকে অ্যাপ্লিকেশনকে রক্ষা করে।

d. Scalability

Django বড় বড় প্রজেক্টেও smooth কাজ করতে পারে। যেমন, **Instagram** এবং **Spotify** Django ব্যবহার করে।

e. Batteries-Included Framework

Django সবকিছু আগে থেকেই নিয়ে আসে, যেমন:

- Database support (MySQL, PostgreSQL, SQLite)।
- URL routing।
- Frontend templates।
- Forms এবং validation।

Django সেটআপ করতে হবে

প্রথমে, আপনার computer-এ Python ইন্সটল আছে কিনা নিশ্চিত করুন।

Step 1: Install Django

নিচের command ব্যবহার করে Django ইন্সটল করুন:

```
bash
pip install django
```

আপনার প্রথম Django প্রজেক্ট তৈরি করে ফেলুন ধাপে ধাপে

Step 1: Start a New Project

নতুন একটি প্রজেক্ট শুরু করতে নিচের command লিখুন:

```
bash
```

```
django-admin startproject myfirstproject
```

এটি myfirstproject নামে একটি ফোল্ডার তৈরি করবে, যেখানে প্রজেক্টের প্রয়োজনীয় সব ফাইল থাকবে।

রিয়েল-লাইফ উদাহরণ

ধরুন, আপনি একটি বেকারি খুলতে চান। এই প্রজেক্টটি হবে সেই বেকারি, যেখানে সব operations (যেমন কেক ম্যানেজ করা, অর্ডার নেওয়া, পেমেন্ট প্রসেস করা) চলবে।

Step 2: Run the Development Server

প্রজেক্ট ফোল্ডারে যান এবং এই command লিখুন:

```
bash
```

```
python manage.py runserver
```

এরপর, আপনার browser-এ **http://127.0.0.1:8000/** লিখুন। Django-এর welcome page দেখতে পাবেন, যা প্রমাণ করে আপনার প্রজেক্ট ঠিকমতো চলছে!

Step 3: Create an App

Django প্রজেক্ট বিভিন্ন ছোট components বা “apps”-এ বিভক্ত। প্রতিটি app একটি নির্দিষ্ট functionality manage করে।

নিচের command লিখে একটি blog নামে নতুন app তৈরি করুন:

```
bash
```

```
python manage.py startapp blog
```

রিয়েল-লাইফ উদাহরণ

যদি আপনার প্রজেক্ট একটি অনলাইন স্টোর হয়, তাহলে আপনার apps হতে পারে:

- **Products:** যেখানে items ম্যানেজ করা হবে।
- **Orders:** যা customer-এর অর্ডারগুলো ম্যানেজ করবে।

Step 4: Configure the App

নতুন তৈরি app (blog) প্রজেক্টের INSTALLED_APPS লিস্টে যুক্ত করুন। **settings.py** ফাইল খুলে নিচের মতো যোগ করুন:

```
python

INSTALLED_APPS = [
    ...
    'blog',
]
```

Step 5: Create a View

blog অ্যাপের **views.py** ফাইল খুলে একটি simple function যোগ করুন:

```
python

from django.http import HttpResponse

def home(request):
    return HttpResponse("Welcome to my first Django app!")
```

Step 6: Add a URL

blog অ্যাপের ফোল্ডারে একটি **urls.py** ফাইল তৈরি করুন এবং এতে এই code লিখুন:

```
python

from django.urls import path
from . import views

urlpatterns = [
    path('', views.home, name='home'),
]
```

এরপর, মূল **urls.py** ফাইলে এই নতুন **urls.py** ফাইলটি include করুন:

```
python
```

```
from django.contrib import admin
from django.urls import path, include

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

Step 7: Run the Server Again

সার্ভারটি পুনরায় চালু করুন:

```
bash
```

```
python manage.py runserver
```

Browser-এ <http://127.0.0.1:8000/> খুলুন। সেখানে দেখতে পাবেন:

"Welcome to my first Django app!"

Django-এর Real-Life Applications

- **E-commerce Sites:** Django ব্যবহার করে **eBay**-এর মতো সাইটগুলি complex transaction পরিচালনা করে।
- **Social Media Platforms:** Instagram Django ব্যবহার করে photos, videos, এবং millions of users ম্যানেজ করে।
- **Learning Platforms:** Online education platforms যেমন **Udemy** Django ব্যবহার করে courses এবং instructors ম্যানেজ করতে।

Django শেখার উপকারিতা

- **Time-Saving:** Pre-built tools development দ্রুত করে।
- **Scalable:** অ্যাপ্লিকেশন বাড়ার সাথে সাথে grow করতে পারে।
- **Community Support:** Django-র বিশাল community সমস্যা সমাধানে সহায়ক।
- **Versatile:** Blogs থেকে বড়-scale applications, Django সব প্রকারের প্রজেক্টের জন্য উপযুক্ত।

Django শেখার মাধ্যমে ক্যারিয়ার গড়ার সুযোগ

Django শেখার মাধ্যমে আপনি নিম্নলিখিত ক্যারিয়ারগুলিতে প্রবেশ করতে পারেন:

- Web Developer
- Backend Developer
- Full-Stack Developer

বড় বড় টেক কোম্পানি Django-skilled ডেভেলপারদের খুঁজে, তাই এটি একটি গুরুত্বপূর্ণ স্কিল।

উপসংহার

Django একটি powerful এবং beginner-friendly framework, যা ওয়েব ডেভেলপমেন্ট সহজ করে। এর tools এবং features ব্যবহার করে functional এবং engaging web applications তৈরি করুন। এখনই প্র্যাকটিস শুরু করুন এবং নিজের প্রজেক্ট তৈরি করে hands-on অভিজ্ঞতা অর্জন করুন!

