Tops Technology

Module 16)
Python DB and Framework

Presented By : Nandni Vala

Project and App Creation

1.Steps to create a Django project and individual apps within the project.

- **Set Up a Virtual Environment**
- **Create**:
- python -m venv myenv
- **Activate**:
- myenv\Scripts\activate
- Install Django
- pip install django
- Create a Django Project
- django-admin startproject project_name
- Navigate into the project directory:
- cd project_name

- **Run the Development Server**
- Verify the setup:
- python manage.py runserver
- **Create an Individual App**
- Use the startapp command:
- python manage.py startapp app_name
- **Register the App**
- Add the app to INSTALLED_APPS in settings.py:
- INSTALLED_APPS = [
- 'app_name',
- ...
-]
- **Create Models and Migrations**
- Define models in models.py of the app.
- Run migrations:
- python manage.py makemigrations
- python manage.py migrate

- **Create Views and URLs**
- Define views in views.py.
- Configure URLs in urls.py of the app and include them in the project's urls.py.
- **Test the App**
- Run the server again and test the app:
- python manage.py runserver
- **Optional: Create Admin Panel**
- Register models in admin.py to manage them via Django Admin:
- from .models import ModelName
- admin.site.register(ModelName)

2.Understanding the role of manage.py, urls.py, and views.py.

- **manage.py**
- **Role**: A command-line utility for managing your Django project.
- **Uses**:
  - Start the development server: python manage.py runserver
  - Make and apply database migrations: python manage.py makemigrations and python manage.py migrate
  - Create a superuser for the admin panel: python manage.py createsuperuser
  - Run custom management commands.
- **Importance**: Acts as an interface to manage the project environment and its settings.

- **urls.py**
- **Role**: Maps URLs to their corresponding views.
- **Uses**:
    - Define URL patterns using Django's path() or re_path() functions.
    - Route requests to the appropriate views.py function or class-based view.
    - Include app-specific URLs into the project’s main urls.py.
- **Example**:
- from django.urls import path
- from . import views
- urlpatterns = [
- path('home/', views.home, name='home'),
-]
- **Importance**: Acts as the entry point for HTTP requests, linking them to views.

- **views.py**
- **Role**: Handles the logic for processing requests and returning responses.
- **Uses**:
    - Define functions or class-based views to process data and return an HTTP response.
    - Interact with models and templates to render dynamic content.
- **Example**:
- from django.http import HttpResponse
- def home(request):
-     return HttpResponse("Welcome to the Home Page!")